Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финуниверситет)

Самарский финансово-экономический колледж
(Самарский филиал Финуниверситета)

УТВЕРЖДАЮ
Заместитель директора по учебно-методической работе
_____Л.А Косенкова
«__» _____20__г.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ «ПМ.07 СОАДМИНИСТРИРОВАНИЕ БАЗ ДАННЫХ И СЕРВЕРОВ»

СПЕЦИАЛЬНОСТЬ: 09.02.07 ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ПРОГРАММИРОВАНИЕ

Самара – 2022

Методические указания по организации и выполнению практических занятий разработаны на основе рабочей программы по профессиональному модулю «Соадминистрирование баз данных и серверов», в соответствии с федеральным государственным образовательным стандартом среднего профессионального образования по специальности 38.02.06 Финансы, утвержденного приказом Министерства образования науки Российской Федерации от 09.12.2016 года № 1547, с учетом Профессионального стандарта, утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 11 февраля 2014 г. № 647н «Об утверждении профессионального стандарта 06.011 Администратор баз данных» (зарегистрирован Министерством юстиции Российской Федерации 24 ноября 2014 г., регистрационный № 34846)
Присваиваемая квалификация: администратор баз данных

Разработчики:

Платковская Е.А. _____ Преподаватель Самарского филиала Финуниверситета

Методические указания по организации и выполнению практических занятий рассмотрены и рекомендованы к утверждению на заседании предметной (цикловой) комиссии естественно-математических дисциплин

Протокол от «24» января 20 22 г. № 5

Председатель ПЦК _____ М.В. Писцова

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические указания по выполнению практических работ по профессиональному модулю ПМ.07 Соадминистрирование баз данных и серверов разработаны с целью оказания помощи студентам специальности 09.02.07 Информационные системы и программирование и преподавателям по организации практических занятий по изучаемой дисциплине, в соответствии с требованиями федерального государственного стандарта среднего профессионального образования.

Методические разработка включает в себя краткие теоретические сведения, указания по выполнению практических работ, контрольные вопросы, формы контроля.

В соответствии с учебным планом на практические работы для студентов отводится 136 часов.

В результате изучения профессионального модуля студент должен освоить основной вид деятельности Осуществление интеграции программных модулей и соответствующие ему общие компетенции и профессиональные компетенции:

Перечень общих компетенций

Код	Наименование общих компетенций
ОК 1.	Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам
ОК 2.	Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности.
ОК 3	Планировать и реализовывать собственное профессиональное и личностное развитие.
ОК 4	Планировать и реализовывать собственное профессиональное и личностное развитие.
ОК 5	Планировать и реализовывать собственное профессиональное и личностное развитие.
ОК 6	Проявлять гражданскопатриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей
ОК 7	Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях.
ОК 8	Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности
ОК 9	Использовать информационные технологии в профессиональной деятельности.
ОК 10	Пользоваться профессиональной документацией на государственном и иностранном языке
ОК 11	Планировать предпринимательскую деятельность в профессиональной сфере

Перечень профессиональных компетенций

Код	Наименование видов деятельности и профессиональных компетенций
ВД 7	***Соадминистрирование баз данных и серверов***
ПК 7.1	Выявлять технические проблемы, возникающие в процессе эксплуатации баз данных и серверов
ПК 7.2	Осуществлять администрирование отдельных компонент серверов
ПК 7.3	Формировать требования к конфигурации локальных компьютерных сетей и серверного оборудования, необходимые для работы баз данных и серверов
ПК 7.4	Осуществлять администрирование баз данных в рамках своей компетенции
ПК 7.5	Проводить аудит систем безопасности баз данных и серверов, с использованием регламентов по защите информации.

1.1.3. В результате освоения профессионального модуля студент должен:

Иметь практический опыт	В участии в соадминистрировании серверов; разработке политики безопасности SQL сервера, базы данных и отдельных объектов базы данных; применении законодательства Российской Федерации в области сертификации программных средств информационных технологий
уметь	проектировать и создавать базы данных; выполнять запросы по обработке данных на языке SQL; осуществлять основные функции по администрированию баз данных; разрабатывать политику безопасности SQL сервера, базы данных и отдельных объектов базы данных; владеть технологиями проведения сертификации программного средства
знать	модели данных, основные операции и ограничения; технологию установки и настройки сервера баз данных; требования к безопасности сервера базы данных; государственные стандарты и требования к обслуживанию баз данных

Количество практических работ по разделам

Раздел 1. Технологии администрирования серверов и баз данных	98
Раздел 2. Обеспечение качества и сертификация информационных систем	38

Перечень практических занятий

№п.п.	НАЗВАНИЕ ПРАКТИЧЕСКИХ ЗАНЯТИЙ
1	Практическая работа №1 Создание объектов баз данных (таблиц). Создание объектов баз данных (форм, отчётов).
2	Практическая работа №2 Установка атрибутов и ключей Установка и нормализация отношений в базе данных (различные нормальные формы).
3	Практическая работа №3 Построение схем баз данных (различного уровня сложности).
4	Практическая работа №4 Манипулирование данными (хранение, добавление, редактирование данных). Манипулирование данными (удаление данных, навигация по набору данных).
5	Практическая работа №5 Сортировка, поиск и фильтрация данных. Построение запросов к СУБД (различного уровня сложности).
6	Практическая работа №6 Построение концептуальной модели базы данных.
7	Практическая работа №7 Создание логической модели данных с помощью утилиты автоматизированного проектирования базы данных.
8	Практическая работа №8 Создание физической модели данных с помощью утилиты автоматизированного проектирования базы данных.
9	Практическая работа №9 Разработка серверной части базы данных в инструментальной оболочке. Разработка клиентской части базы данных в инструментальной оболочке.
10	Практическая работа №10 Разработка технических требований к серверу баз данных
11	Практическая работа №11 Модель сервера баз данных
12	Практическая работа №12 Компоненты SQL server
13	Практическая работа №13 Модели клиентсервер
14	Практическая работа №14 Системные базы данных
15	Практическая работа №15 Оптимизация запросов, управляемых правилами
16	Практическая работа №16 Объектноориентированные модели данных
17	Практическая работа №17 Разработка требований к корпоративной сети
18	Практическая работа №18 Cache и WWWтехнологии

19	Лабораторная работа№1 Конфигурирование сети
20	Практическая работа №19 Формирование аппаратных требований и схемы банка данных
21	Лабораторная работа№2 Установка и настройка сервера MySQL
22	Лабораторная работа№3 Конфигурирование SQL Server Agent и SQL Server Enterprise Manager
23	Лабораторная работа№4 Управление файлами базы данных
24	Лабораторная работа№5 Команды Transact_sqL
25	Лабораторная работа№6 Обеспечение безопасности в SQL SERVER
26	Лабораторная работа№7 Установка и настройка сервера под UNIX
27	Лабораторная работа№8 Выполнение запросов к базе данных
28	Лабораторная работа№9 Выполнение изменений в базе данных, создание триггеров
29	Лабораторная работа№10 Создание запросов и процедур на изменение структуры базы данных
30	Лабораторная работа№11 Работа с журналом аудита базы данных
31	Лабораторная работа№12 Резервное копирование баз данных
32	Лабораторная работа№13 Мониторинг нагрузки сервера
33	Лабораторная работа№14 Автоматизация административных задач
34	Лабораторная работа№15 Настройка политики безопасности
35	Лабораторная работа№16 Создание резервных копий базы данных
36	Лабораторная работа№17 Восстановление базы данных
37	Лабораторная работа№18 Восстановление носителей информации
38	Лабораторная работа№19 Восстановление удаленных файлов
39	Лабораторная работа№20 Мониторинг активности портов
40	Лабораторная работа№21 Блокирование портов
41	Лабораторная работа№22 Проверка наличия и сроков действия сертификатов
42	Лабораторная работа№23 Разработка политики безопасности корпоративной сети
43	Лабораторная работа№24 Получение сертификата

МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ СТУДЕНТОВ ПО ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ ЗАНЯТИЙ

МДК. 07.01 Управление и автоматизация баз данных

Практическая работа №1 Создание объектов баз данных (таблиц). Создание объектов баз данных (форм, отчётов).

Цель занятия: Получить навыки разработки баз данных в среде MS SQL SERVER Management Studio 2008 (2012).

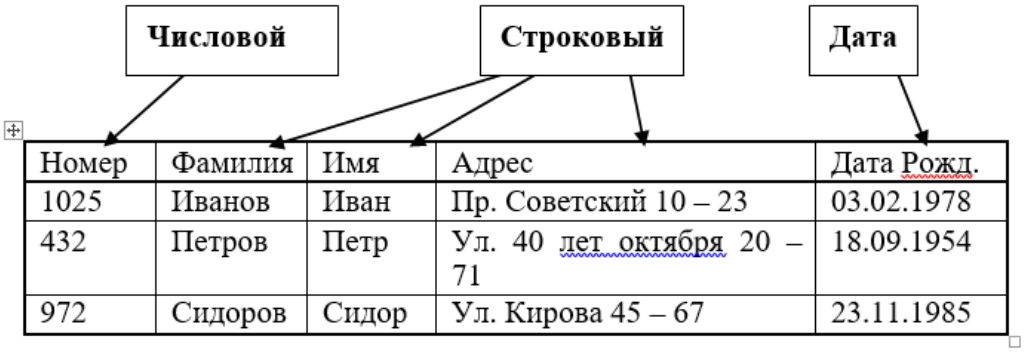
Краткие теоретические сведения:

Базовым элементом баз данных, построенных на основе реля- ционной модели, является отношение. Отношение реализуется в среде различный СУБД как таблица.

Таким образом, таблица это объект, предназначенный для хранения информации в реляционной БД. Информация об единич- ном экземпляре данных представляется как **запись (кортеж)** или строка в таблице. **Поля (атрибуты)** объекта представляются как – столбцы в табличном виде.

Поля в реляционных базах данных характеризуются следующими свойствами:

1. *Имя поля* – идентификатор поля, по которому организуется программный доступ к нему.
2. *Тип поля* – тип данных, находящихся в этом поле.Примеры типов представлен на рис. 1.1.



Номер	Фамилия	Имя	Адрес	Дата Рожд.
1025	Иванов	Иван	Пр. Советский 10 – 23	03.02.1978
432	Петров	Петр	Ул. 40 лет октября 20 – 71	18.09.1954
972	Сидоров	Сидор	Ул. Кирова 45 – 67	23.11.1985

Рис.1.1 Таблица – основной элемент базы данных

3. *Размер поля* – величина в байтах, выделяемая для хранения данных в поле. Например: если тип поля СТРОКОВЫЙ, а размер будет равен 10-ти, то это значит, что в ячейку такого поля нельзя будет записать строку более 10 символов. Если задать ЦЕЛЫЙ ЧИ- СЛОВОЙ тип и установить размер в 4 байта, то числа в ячейке бу- дут принимать значения от 0 до 65535
4. *Инкриментность (счетчик)* – автозаполнение поля в добавленной записи неким значением (как правило числового целого типа).
5. *Ключ* – уникальный идентификатор, характеризующий запись
6. *Необходимость заполнения* – если поле не обязательно для заполнения, то при добавлении записи (в случае отсутствия данных в поле) оно автоматически заполняется значением по умолчанию, если таковое имеется. Если значения по умолчанию нет, записывается псевдопустое значение «NULL», которое определено в системе специальным идентификатором.

Системы управления базами данных(СУБД). СУБД MS SQL SERVER 20XX

Система управления базами данных (СУБД). СУБД – вспомо гательная система, обеспечивающая работу базы данных.

СУБД обеспечивает:

- логически согласованную работу файлов хранящих данные;
- язык манипулирования данными;
- восстановление информации после сбоев;

- возможность совместной (параллельной работы) нескольких пользователей с данными.

Существуют различные СУБД от разных разработчиков ORACLE, Microsoft SQL Server, MYSQL, PostgreSQL и другие. Каждая СУБД имеет несколько версий. Обычно версия соответствует развитию технологии на некоторый момент времени. Например MS SQL Server 2017.

Microsoft SQL Server (MS SQL Server), – это масштабируемая высокопроизводительная система управления реляционными база- ми данных для платформ на базе MS Windows. Она разработана с учетом требований к современным распределенным клиент- серверным вычислениям и тесно интегрирована с серверными про- дуктами семейства Microsoft Office.

Включает в себя библиотеки и службы ядра сервера СУБД. При установке MS SQL SERVER система представляется в виде системной службы MSSQLSERVER. Данная служба все запросы, приходящие на сервер.

Отображение службы MSSQLSERVER в диспетчере задач операционной системы показано на рисунке 1.2. В данном случае Server EXPRESS с именем сервера EXPRESS208R2.

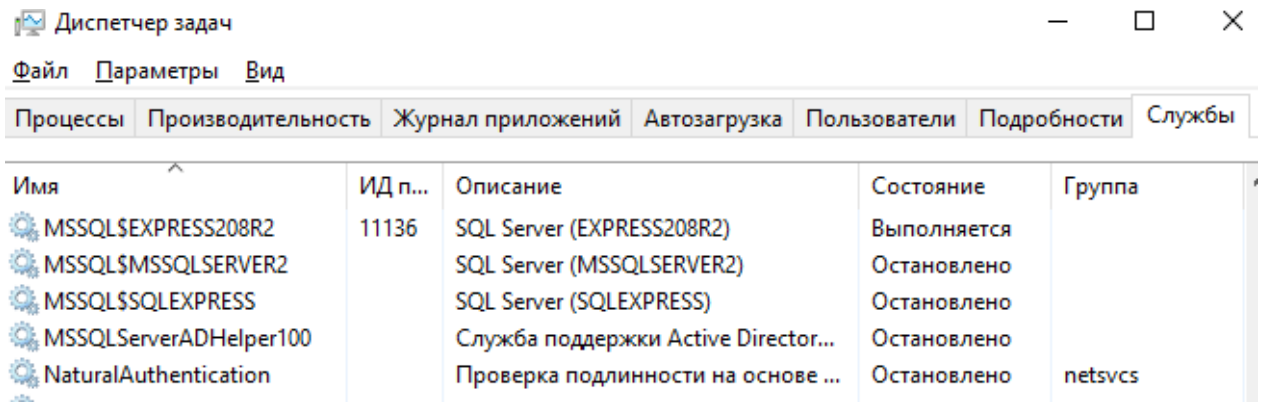


Рис.1.2 Служба MSSQLSERVER. В данном случае SQL Server EXPRESS с именем сервера EXPRESS208R2

В стандартный пакет Microsoft SQL Server входят несколько приложений, служащих для администрирования и разработки клиент-серверных приложений.

Для разработки таблиц и серверных механизмов используется приложение MS SQL SERVER Management Studio (также можетбыть различных версий).

При запуске приложения открывается окно соединения приложения с сервером. Приложение можно использовать для работы серверами, установленными не зависимо от MS SQL SERVER Management Studio.

CIT-208_01

Рис.1.3 Окно соединения с сервером

Для соединения с сервером необходимо знать его имя, имя за писи, зарегистрированной на сервере и пароль для этой записи. Если используется авторизация на основе учетной записи Windows, данная учетная запись должна быть зарегистрирована на сервере БДПосле соединения с сервером открывается окно приложения MS SQL SERVER Management Studio.

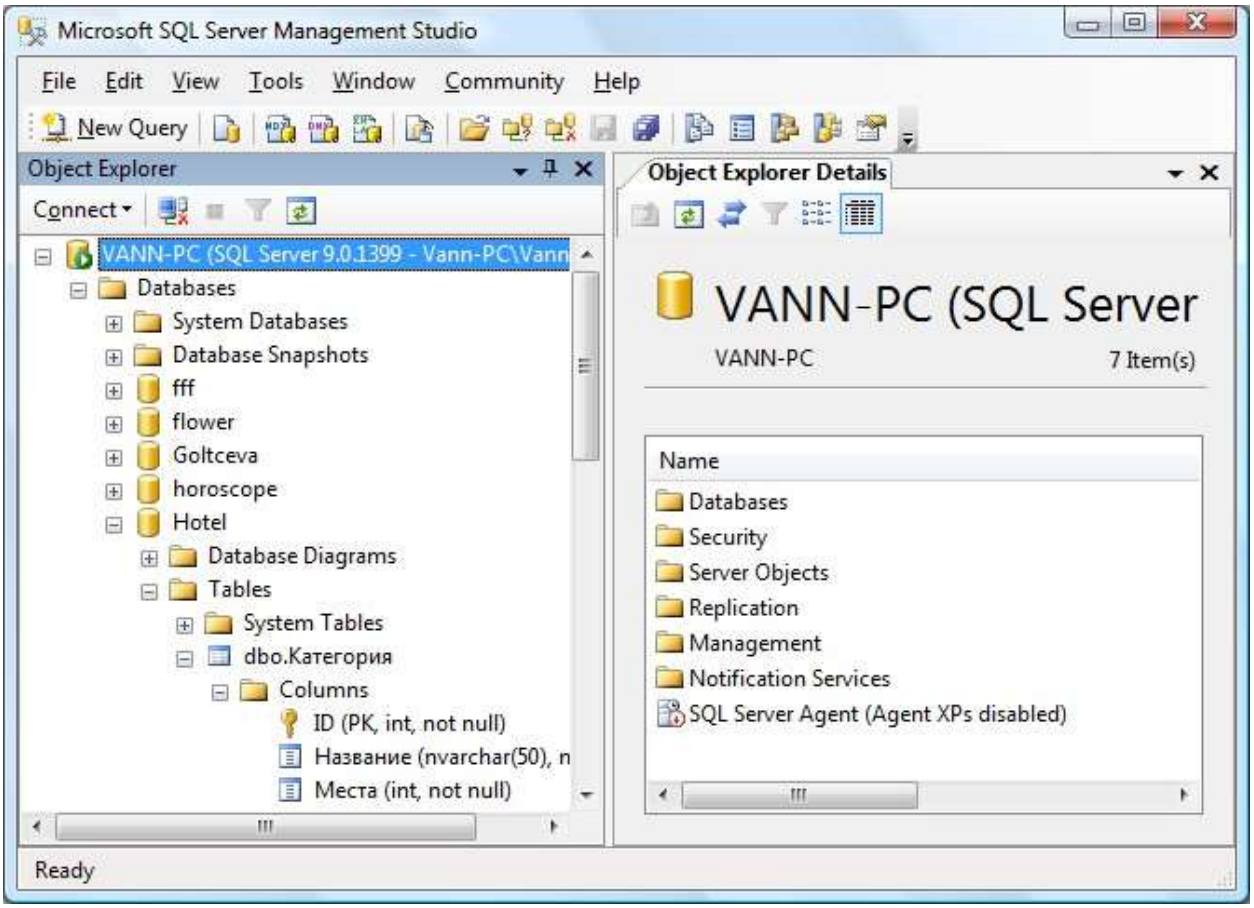


Рис.1.4 Рабочее окно MS SQL SERVER Management Studio

Левую часть окна занимает рабочее окно обозревателя объектов сервера. Объекты сервера представлены в виде древовидной структуры. Корнем дерева является соединение. Management Studio может быть одновременно соединено с несколькими серверами. Работа с любыми объектами сервера может осуществляться через кон текстное меню на

соответствующем узле дерева. База данных отображается в виде узла Databases. В среде MS SQL Server база данных содержит в себе различные типы объектов (рис. 1.5).

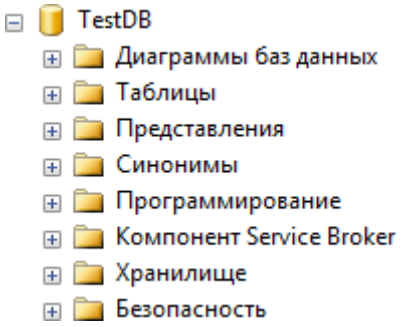


Рис.1.5 Объекты базы данных в среде MS SQL Server

Объекты базы данных в обозревателе объектов сервера сгруппированы в функциональные узлы. Выделяются следующие типы объектов:

- Таблицы – узел «таблицы».
- Представления – узел «Представления».
- Программные объекты (механизмы сервера) – узел «Программирование».
- Объекты обеспечения безопасности – узел «Безопасность».
- Диаграммы баз данных – узел «Диаграммы баз данных».

Все узлы создаются автоматически при создании базы данных. Согласно рабатам основоположника теории реляционных баз, данных Дейту [1] в базе данных выделяются структурная часть, манипуляционная и целостная.

Структурная часть базы данных – таблицы базы данных или реляционные отношения содержится в узле «Таблицы». Создать, но вую таблицу можно через контекстное меню на данном узле.

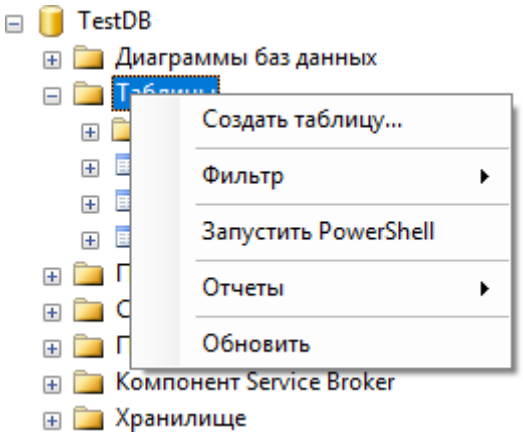


Рис.1.6 Создание новой таблицы в средеMS Management Studio

Создание таблицы подразумевает создание еѐ атрибутов (столбцов) и присвоение имени таблицы.

После вызова команды создания таблицы в левой (рабочей об- ласти) Management Studio открывается табличная форма для созда- ния и корректировки атрибутов таблицы.

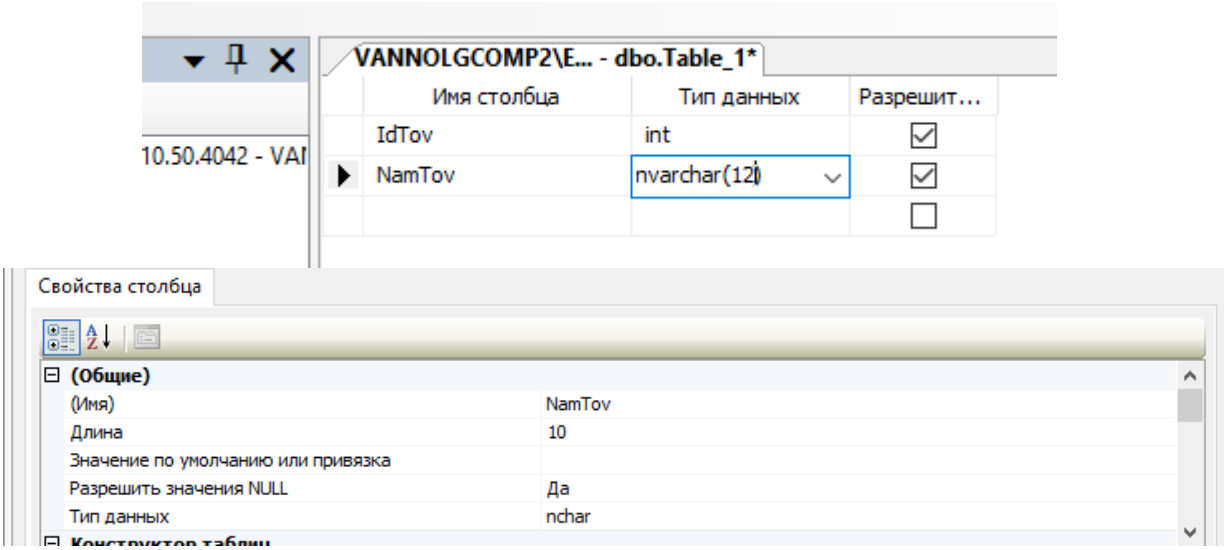


Рис.1.7 Работа с таблицей в режиме еѐ модификации

При создании и модификации таблицы еѐ атрибуты представляются в виде строк таблицы. Каждая строка соответствует отдельному столбцу (атрибуту).

Для каждого столбца необходимо указать его имя и тип данных. Имя можно выбрать любое, но для обеспечения простоты формирования запросов целесообразно для задания имѐн атрибутов использовать латинский шрифт и не использовать внутри имени пробелы и другие служебные символы. Пример хорошего имени столбца «NameStud» – то есть смысловые части разделяются заглавной буквой. Пример не рекомендуемого имени столбца – «Имя Студента». При использовании такого типа имени при написании запросов их придѐтся заключать в квадратные скобки. Например

«*Select [Имя Студента] from [Студенты]*». Гораздо проще будет выглядеть запись той же команды при использовании рекомендованных именований – *«Select NameStud from Studs»*

Обычно таблицы имеют некоторые идентифицирующий ключевой атрибут и некоторую совокупность описательных атрибутов.

При задании столбцов (атрибутов) таблицы (отношения) могут использоваться различные типы данных, предусмотренных средой конкретного СУБД, в которой производится работа, используются следующие типы данных.

Используемые типы данных представлены на рис. 1.8.

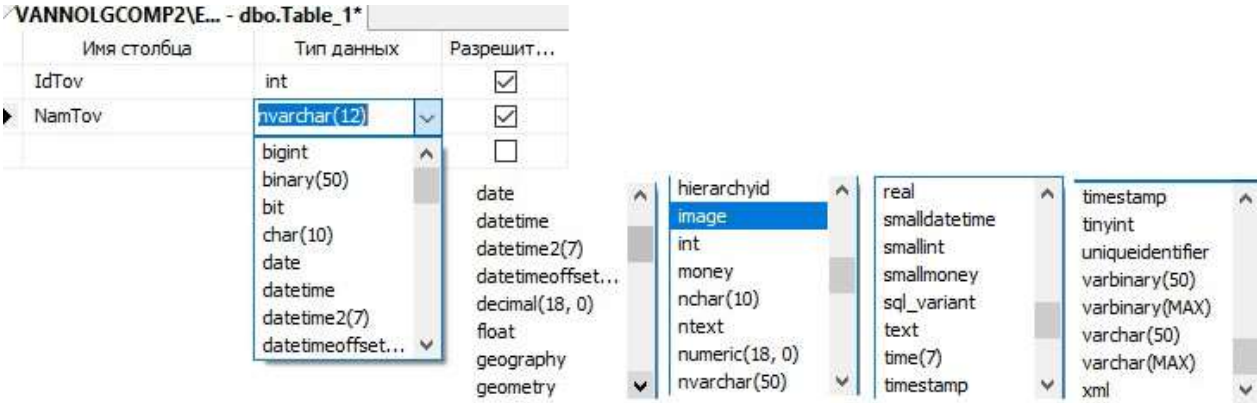


Рис.1.8 Задание типа данных для столбца (атрибута таблицы)

Тип данных выбирается с помощью соответствующего выпадающего списка.
В MS SQL Server 2008R2 объединены в следующие категории:

- Точные числа.

- Приблизительные числа.
- Символьные строки.
- Символьные строки в Юникоде.
- Дата и время.
- Двоичные данные.
- Прочие типы данных.

Точные числа:

- int – целые.
- tinyint – малые целые.
- smallint – малые целые.
- bigint – большие целые.
- numeric, decimal – числа с фиксированной точностью.
- bit – битовые числа.
- smallmoney, money – для работы с денежными величинами.
- float real – приблизительные числа

Типы данных для работы с датой и временем представлены следующими: *date, datetimeoffset, datetime2, smalldatetime, datetime, time.*

Символьные строки:

- char
- varchar
- text
- char [(n)]

- nchar
- nvarchar
- ntext
- nchar [(n)]

Двоичные данные:

- binary
- varbinary
- image

Прочие типы данных:

cursor, timestamp, hierarchyid, uniqueidentifier,sql_variant, xml, table

Можно также определять собственные типы данных в Tran- sact-SQL или Microsoft.NET Framework. Псевдонимы типов данных основываются на системных типах. Дополнительные сведения о псевдонимах типов данных см. в разделе

Внесение, изменить данных в таблице можно в среде Manage- ment Studio через команду «*Изменить первые 200 строк*», вызы- ваемую через контекстное меню на редактируемой таблице.

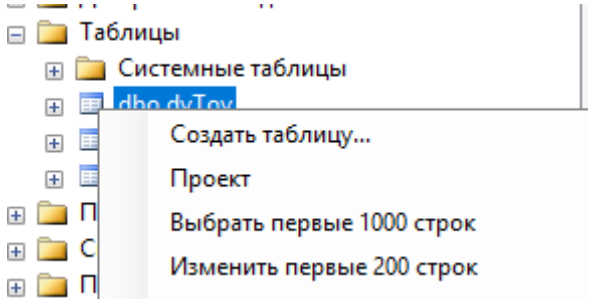
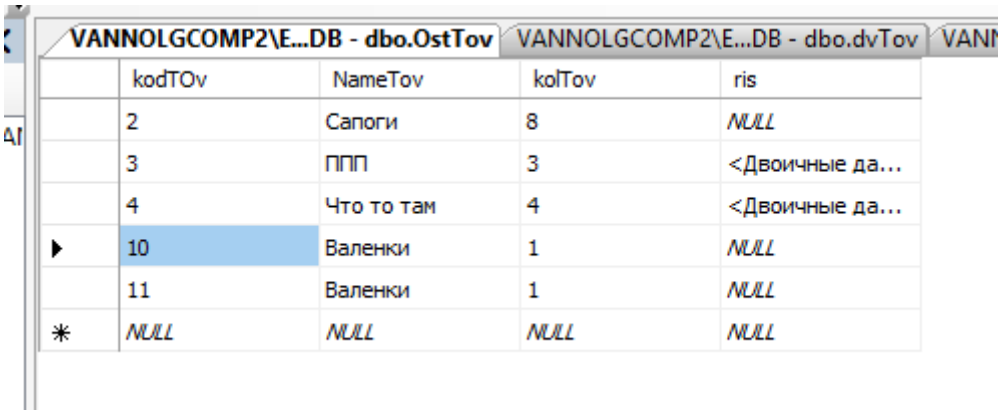


Рис.1.9 Вызов таблицы для изменения и внесения данных



	kodTOv	NameTov	kolTov	ris
	2	Сапоги	8	NULL
	3	ППП	3	<Двоичные да...
	4	Что то там	4	<Двоичные да...
▶	10	Валенки	1	NULL
	11	Валенки	1	NULL
*	NULL	NULL	NULL	NULL

Рис.1.10 Вид таблицы вызванной для внесения данныхи редактирования

Практические задания:

Создать и заполнить данными таблицы в соответствие с вариантом задания.

Вариант 1.

- Студенты (Номер зачѐтки, Фамилия студента, Имя студента).
- Состав учебных группы (Наименование группы, Номер за-чѐтки студента).

Вариант 2.

- Товары (наименование товара, код товара).
- Состав покупки (номер покупки, код товара, количество).

Вариант 3.

- Учебные предметы (наименование предмета, код предмета).
- Расписание (наименование группы, код предмета, дата начала).
- Кафедры университета (наименование кафедры, код кафедры).
- Учебные аудитории (номер аудитории, код кафедры). Таблицы заполнить данными в среде MS Management Studio.

Контрольные вопросы:

1. Что такое база данных?
2. Базовые свойства реляционных отношений.
3. Что такое ключ реляционного отношения?
4. Как задаются связи между реляционными отношениями?

Практическая работа №2 Установка атрибутов и ключей Установка и нормализация отношений в базе данных (различные нормальные формы).

Цель занятия: Целью работы является получение практических навыков раз работки схемы базы данных.

Краткие теоретические сведения

Нормализация – метод создания набора отношений с заданными свойствами на основе некоторых требований к данным. Процесс нормализации – формальный метод для оптимизации столбцов отношений и устранения аномалий.

Избыточность данных и аномалии обновления

Основная цель проектирования реляционной БД – группирование атрибутов в отношениях таким образом, чтобы минимизировать избыточность данных (сокращение объема вторичной памяти для хранения БД) и повышение надежности при работе с данными.

Обычно процесс проектирования отношений реляционной БД ведется на основе разработанной ER-диаграммы или на основе просто здравого смысла разработчика. В общем случае при таком подходе расположение атрибутов в отношениях *неоптимальное*. При работе с отношениями, содержащими избыточные данные, могут возникать проблемы – *аномалии обновления*.

Аномалии обновления делят на три вида:

- *аномалии вставки* – возникают при добавлении новых несогласованных данных (нарушающих целостность данных в отношении);
- *аномалии изменения* – возникают при изменении части ранее введенных данных; частичное обновление сведений приведет к нарушению целостности данных отношения;
- *аномалии удаления* – возникают при удалении строк из отношений.

Обычно для решения проблем избыточности и аномалий выполняется деление отношения на такие отношения, в которых избыточности не будет. Для выполнения такого процесса необходимо выявить все зависимости между атрибутами отношения (потеря одной такой зависимости меняет модель внешнего мира).

Функциональные зависимости

Выявление смысловой зависимости между данными – один из способов формализации смысловой информации о данных.

Функциональная зависимость описывает связь типа «многие-к-одному» между атрибутами отношения, где «много» – детерминант функциональной зависимости. Функциональная зависимость является семантическим свойством атрибутов отношения.

Если в отношении *R*, содержащем атрибуты *A* и *B*, атрибут *B* функционально зависит от атрибута *A* (*A* является детерминантом атрибута *B*) $A \to B$, то в каждом кортеже этого отношения каждое конкретное значение атрибута *A* всегда связано только с одним значением атрибута *B*.

Особенности функциональных зависимостей, лежащие в основе процесса нормализации:

- функциональная зависимость является специализированным правилом целостности – она накладывает ограничения на допустимые значения атрибутов отношений; эту особенность можно использовать при обновлении БД, т.к. зная, какие функциональные зависимости есть в отношении, можно понять, нарушат ли новые данные целостность данных отношения;
- функциональная зависимость является обобщением понятия потенциального ключа; функциональные зависимости позволяют определить все потенциальные ключи отношения (и соответственно – первичный ключ): все атрибуты отношения, которые не являются частью первичного (или потенциального) ключа, должны функционально зависеть от этого ключа; если не все остальные атрибуты отношения зависят от некоторого детерминанта, то этот детерминант не является потенциальным ключом этого

отношения.

Нормальные формы и нормализация методом декомпозиции

Нормализация – это формальный метод анализа отношений на основе их первичного ключа и существующих функциональных зависимостей.

Суть процесса нормализации:

- в нормализованных отношениях не разрешаются никакие функциональные зависимости, кроме функциональных зависимостей вида $K \to A$, где K – потенциальный ключ отношения R, а A – неключевой атрибут;
- если же отношение R имеет функциональные зависимости $B \to A$, где B не является потенциальным ключом, то в отношении R будет наблюдаться избыточность данных.

В процессе нормализации реляционных отношений применяются концепции *нормальных форм*. Говорят, что отношение находится в определенной нормальной форме, если оно удовлетворяет правилам этой нормальной формы. В настоящее время используется шесть нормальных форм, которые зависят друг от друга путем усложнения (вложенности) набора правил:

$1НФ \to 2НФ \to 3НФ \to НФБК \to 4НФ \to 5НФ$.

Каждая нормальная форма, таким образом, **удовлетворяет всем предыдущим нормальным формам**. Более высокая нормальная форма приводит к более строгому формату отношения (меньшее число аномалий обновления).

Примечание. БД можно построить и на отношениях, находящихся в первой нормальной форме, но такая БД будет сильно подвержена аномалиям и избыточности данных.

На практике желательно использовать, как минимум, *3НФ*, чтобы устранить большинство аномалий обновления.

1) *1НФ*. Отношение находится в *1НФ* тогда и только тогда, когда в любом допустимом значении этого отношения каждый кортеж содержит только одно значение для каждого из атрибутов, т.е. это значение не имеет внутренней структуры (множество, таблица и т.п.). Отношения в *1НФ* имеют большое количество аномалий обновления.

2) *2НФ*. Отношение находится в *2НФ* тогда и только тогда, когда оно находится в *1НФ*, и каждый атрибут отношения, не входящий в состав первичного ключа, характеризуется полной функциональной зависимостью от этого первичного ключа.

Полной функциональной зависимостью называется такая зависимость $A \to B$, когда B функционально зависит от A и не зависит ни от какого подмножества A (т.е. удаление какого-либо атрибута из A приведет к утрате этой функциональной зависимости). *2НФ* устраняет в отношении частичные функциональные зависимости неключевых атрибутов от первичного ключа, которые выносятся в отдельное отношение вместе с копиями своих детерминантов (частей первичного ключа, от которого они зависят).

3) *3НФ*. Отношение находится в *3НФ* тогда и только тогда, когда оно находится в *2НФ* и не имеет не входящих в первичный ключ атрибутов, которые находились бы в транзитивной функциональной зависимости от этого первичного ключа.

Транзитивной функциональной зависимостью называется зависимость $A \to C$, если существуют зависимости $A \to B$ и $B \to C$ (говорят, что атрибут C транзитивно зависит от A через атрибут B), при условии, что атрибут A функционально не зависит ни от атрибута B, ни от атрибута C.

3НФ устраняет в отношении транзитивные функциональные зависимости неключевых атрибутов от первичного ключа, которые выносятся в отдельное отношение вместе с копиями своих детерминантов. В *3НФ* устранено большинство аномалий от

первичного ключа, но отношение в этой форме имеет аномалии в случае наличия более чем одного потенциального ключа.

Декомпозиция – формирование отношений БД путем разделения их на более мелкие, если эти отношения не выполняют правила необходимой нормальной формы.

Рекомендации по выполнению работы

Этап 1. Выделить функциональные зависимости для каждого отношения исходной реляционной схемы. Проверить практический смысл выделенных функциональных зависимостей.

Этап 2. Для каждого отношения (включая и вновь создаваемые) последовательно применить правила нормальных форм. При несоблюдении текущего правила в отношении выполнить его декомпозицию (удалить проблемный атрибут из отношения с образованием нового отношения, первичным ключом которого будет детерминант рассматриваемой функциональной зависимости (этот атрибут только копируется в новое отношение)). Нормализованное отношение должно удовлетворять как минимум *3НФ*.

Этап 3. Для полученной нормализованной реляционной схемы проверить смысл ссылок.

Этап 4. Реализовать полученные реляционные отношения в виде таблиц в среде целевой СУБД.

Этап 5. Оформить отчет по работе.

Пример приведения отношения к 3НФ

Рассмотрим отношение «Экзаменационная ведомость»

Код студента	Фамилия	Код экзамена	Предмет и дата	Оценка
1	Иванов	1	Математика, 05.06.2019	4
2	Петров	1	Математика, 05.06.2019	5
1	Иванов	2	Физика, 10.06.2019	5
2	Петров	2	Физика, 10.06.2019	5

Первичный ключ таблицы состоит из атрибутов: Код студента, Код экзамена

Отношение находится в первой нормальной форме (1НФ), если все атрибуты отношения принимают простые значения (атомарные или неделимые), не являющиеся множеством или кортежем из более элементарных составляющих.

Наше отношение не находится в 1НФ.

Приведем отношение к 1НФ:

Код студента	Фамилия	Код экзамена	Предмет	Дата	Оценка
1	Иванов	1	Математика	05.06.2019	4
2	Петров	1	Математика	05.06.2019	5
1	Иванов	2	Физика	10.06.2019	5
2	Петров	2	Физика	10.06.2019	5

Для исследования наличия 2НФ следует проанализировать функциональные зависимости между атрибутами отношения.

Единственный способ определить функциональные зависимости – внимательно проанализировать семантику (смысл) атрибутов.

Примеры функциональных зависимостей для отношения ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ:

Код студента → Фамилия

Код студента, Код экзамена → Оценка

Код экзамена → Дата

Код экзамена → Предмет

Отношение находится в 2НФ, если оно находится в 1НФ и каждый неключевой атрибут зависит от всего первичного ключа (не зависит от части ключа).

Отношение находится в 3НФ, если оно находится в 2НФ и каждый ключевой атрибут нетранзитивно зависит от первичного ключа. Отношение находится в 3НФ в том и только том случае, если все неключевые атрибуты отношения взаимно независимы и полностью зависят от первичного ключа.

Продолжим рассмотрение примера с отношением ЭКЗАМЕНАЦИОННАЯ ВЕДОМОСТЬ. Для более краткой записи процесса нормализации введем следующие обозначения: КС – код студента, КЭ – код экзамена, Ф – фамилия, П – предмет, Д – дата, О – оценка.

Наше отношение примет вид: R=(**КС, КЭ**, Ф, П, Д, О)

Выпишем функциональные зависимости:

КС, КЭ → Ф, П, Д, О (КС, КЭ - первичный ключ отношения, все неключевые атрибуты зависят от первичного ключа)

При этом некоторые атрибуты зависят не от всего ключа в целом:

КЭ → П

КЭ → Д (предмет и дата зависят только от кода экзамена)

КС → Ф (фамилия студента зависит тольео от кода студента)

В соответствии с определением, **отношение находится во второй нормальной форме (2НФ), если оно находится в 1НФ и каждый неключевой атрибут зависит от первичного ключа и не зависит от части ключа**. Здесь атрибуты П, Д, Ф зависят от части ключа. Чтобы избавиться от этих зависимостей необходимо произвести декомпозицию отношения.

Выделим неполные зависимости в отдельные отношения. Если какие-то атрибуты зависят от одной части ключа, объединяем их в одну таблицу.

Получим отношение R1(**КС**, Ф) - это отношение находится в 2 НФ, так как ключ отношения простой и частичной зависимости быть не может. Так как в этом отношении нет транзитивных зависимостей, отношение R1(КС, Ф) находится в 3НФ.

Второе отношение R2(**КЭ**, П, Д) - зависимости неключевых атрибутов от части ключа нет, следовательно отношение находится в 2НФ. Транзитивных зависимостей в этом отношении также нет, следовательно отношение находится в 3НФ.

Исходное отношение приведено к виду: R(**КС, КЭ**, О). Из него выведены неключевые атрибуты, зависящие от части ключа. Неключевой атрибут О зависит от ключа КС, КЭ в целом, а не от его части. Значит, это отношение находится в 2НФ. Транзитивные зависимости отсутствуют, то есть отношение находится в 3НФ.

Таким образом все полученные отношения находятся в 3НФ.

Между таблицами установлены связи, как показано на рис. 1.

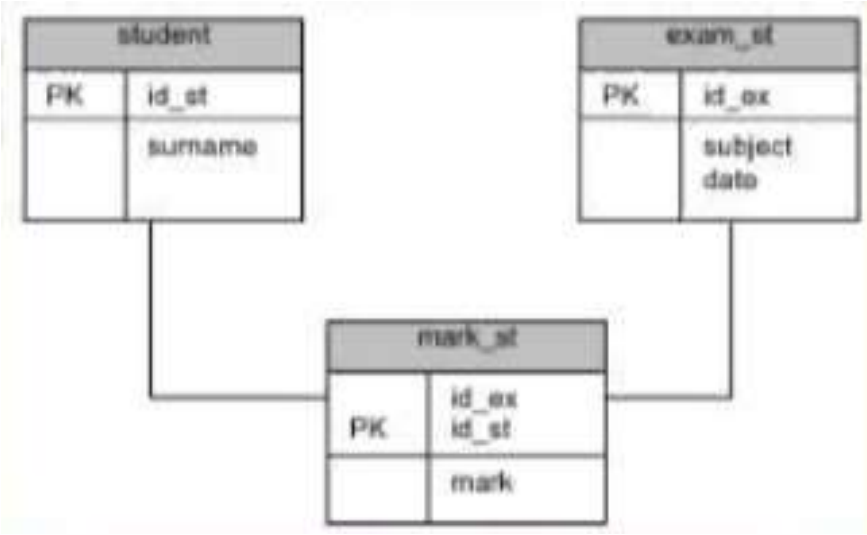


Рис. 1. Полученная реляционная модель (нормализованная)

Использованы следующие обозначения:

id_st – код студента;

surname – фамилия;

id_ex – код экзамена;

subject – предмет;

date – дата;

mark – оценка.

Практические задания

1. Выделить отношения согласно заданию, описать отношения, и их атрибуты.
2. Описать связи между отношениями с точки зрения их множественности с одной и другой стороны, обязательности, соответствия бизнес правилу предметной области.
3. Построить в среде SQL server Manadgment Sydudio таблицы в соответствии с заданием.
4. Построить диаграмму отношений в среде SQL server Ma- nadgment Sydudio.
5. Произвести заполнение отношений тестовыми данными.

4. ЗАДАНИЯ

Варианты заданий

Вариант 1 – отношение «Морские перевозки»

Номер судна	Название	Номер рейса	Дата погрузки	Порт погрузки	Дата прибытия	Порт прибытия	Ф.И.О. капитана	Вид судна	Грузоподъемность, тонны
526	Japan Bear	9201 W	5/31/92	SFO	6/6/92	HNL	Емелин А.О.	Сухогруз	500
603	Korea Bear	9202 W	5/05/92	OAK	6/19/92	OSA	Крылов О.Б.	Ролкер	1000
531	China Bear	9203 W	6/20/92	LAX	7/10/92	PAP	Мухин Е.А.	Универсал	1500
526	Japan Bear	9204 W	8/20/92	SFO	8/27/92	HNL	Емелин А.О.	Сухогруз	500

Вариант 2 – отношение «Контрагенты»

Наименование контрагента	Город	Адрес	Вид контрагента	Должность контактного лица	Ф.И.О. контактного лица	Код города	Телефон
Поршневой завод	Владимир	ул. Кольцевая, 17	Поставщик	зам. дир.	Иванов И.И.	3254	76-15-95
Поршневой завод	Владимир	ул. Кольцевая, 17	Поставщик	нач. отд. сбыта	Петров П.П.	3254	76-15-35
ООО «Вымпел»	Курск	ул. Гоголя, 25	Клиент, Поставщик	директор	Сидоров С.С.	7634	66-65-38
ИП «Альфа»	Владимир	ул.Пушкинская, 37	Клиент, Поставщик	директор	Васильев В.В.	3254	74-57-45

Вариант 3 – отношение «Отдел кадров»

Код сотрудника	ФИО	Должность	Номер отдела	Наименование отдела	Квалификация
7513	Иванов И.И	Программист	120	Отдел проектирования	C, Java
9842	Петров А.А.	Администратор БД	30	Финансовый отдел	MS SQL Server
6651	Сорокин А.П.	Прогрсммист	120	Отдел проектирования	VB, Java
9006	Ворнов Г.Р.	Системный администраторо	120	Отдел проектирования	Windows, Linux

Вариант 4 – отношение «Ведомость расходов»

Номер проекта	Наименование проекта	Номер работника	Ф.И.О. работника	Должность	руб/час	трудозатраты в часах	Общие расходы
15	Alpha Edit	101	Семен Иванов	Программист	200	120	24000
		102	Андрей Петров	Программист	200	100	20000
		110	Антон Сидоров*	Сист. аналитик	300	40	12000
18	Beta Base	103	Федот Антонов	Прграммист	200	250	50000
		102	Андрей Петров	Программист	200	280	56000
		111	Петр Семенов*	Проектировщик БД	250	80	20000
22	Delta CAD	104	Сидор Федотов	Программист	200	180	36000
		105	Иван Андреев	Программист	200	150	30000
		110	Антон Сидоров*	Системный аналитик	300	60	18000

Контрольные вопросы:

1. Что подразумевается под схемой базы данных?
2. Каким образом выявляются отношения базы данных?
3. Чему соответствует таблица базы данных в предметнойобласти?

Содержание отчета

1. Титульный лист.
2. Реляционная таблица для заданного варианта.
3. Описание первичных ключей и функциональных зависимостей для заданного отношения (таблицы).
4. Нормализованная реляционная модель до уровня 3НФ с описанием всех этапов проведения анализа исходной таблицы и выполняемых декомпозиций. Для каждой полученной в результате декомпозиции таблицы должны быть описаны все

функциональные зависимости.

5. Вывод по результатам работы.

Практическая работа №3 Построение схем баз данных (различного уровня сложности).

Цель занятия: получение практических навыков разработки схемы базы данных.

Краткие теоретические сведения

Схема базы данных (от англ. Database schema) – еѐ структура, описанная на формальном языке, поддерживаемом СУБД. В реля- ционных базах данных схема определяет таблицы, поля в каждой таблице (обычно с указанием их названия, типа, обязательности), и ограничения целостности (первичный, потенциальные и внешние ключи и другие ограничения).

Схемы в общем случае хранятся в словаре данных. Хотя схема определена на языке базы данных в виде текста, термин часто ис- пользуется для обозначения графического представления структурыбазы данных.

Основными объектами графического представления схемы яв- ляются таблицы и связи, определяемые внешними ключами.

Выявление отношений в базе данных. В качестве отношений реляционной базы данных отображаются объекты предметной области, обеспечивающие получение информации, определенной в требованиях к системе.

Для выявления сущностей предметной области необходимо еѐ проанализировать и выявить объекты, обладающие свойствами, на основе которых может быть получена информация, определѐнная в требованиях для базы данных. Состав объектов должен быть достаточным, но не избыточным. Обычно выделяются объекты оперативные и справочные.

Оперативные объекты содержат некоторую текущую информацию, они часто обновляются. Это могут быть данные о единичной покупке, например:

таблицаПокупка(покупатель, товар, количествоТовара).

Справочные объекты содержат информацию, которая может использоваться в качестве значений атрибутов для оперативных объектов. Например, данные о товаре:

таблицаТовар(Наименование, цена, производитель).

Атрибуты справочных таблиц могут определяться значениями, других справочных таблиц, например атрибут производитель в таб- лице *таблицаТовар* может определятся значениями таблицы:

таблицаПроизводитель(Наименование, номерСчёта, юрАдрес).

Для выявленных отношений устанавливаются атрибуты и требования к ним.

Для каждого отношения необходимо сформулировать бизнес – правила соответствующей предметной области.

Бизнес – правила характеризуют поведение объекта в предметной области, значение его атрибутов.

Необходимо проанализировать атрибуты, выявленные для отношений, на предмет их атомарности. Не атомарный атрибут подразумевает некоторое множество составных атрибутов, а, следовательно, его можно представить в виде другого отношения.

Например:

Студент – объект, выполняющий обучение на предметах.Характеризуется:

фамилией, именем, отчеством (отдельные атрибуты типастрока);

номером зачетной книжки (атрибут целого типа).

Студент обучается на ***учебном курсе*** *(****учебный курс*** *– это отдельное отношение, так как может иметь свои характеристики).*

Для выявленных объектов и их атрибутов необходимо выявить бизнес правила, определяющие требования целостности сущности, то есть обязательность значения данного атрибута, уникальностьзначения данного атрибута, его допустимые значения.

Например:

Фамилия студента состоит из символов, это обязательныйатрибут.

Номер зачетной книжки – число, минимальное значение -10000,

Максимальное 99999.

Для выявленных отношений необходимо определить бизнес- правила их функционирования в предметной области определяющие их с другими отношениями

В бизнес-правилах, характеризующих связи должна быть дана следующая информация:

- содержания связи;
- множественность связи с одной и другой стороны;
- обязательности и дополнительных ограничений, ограничений накладываемых на связь.

Например, отношение Покупка связано с отношением товар, так как покупка должна всегда содержать товар. Данная связь имеет множественность «один к многим», так как одна покупка может содержать много товаров.

Каждое выявленное бизнес-правило реализуются в виде фрагмента ER диаграммы.

Практические задания

1. Выделить отношения согласно заданию, описать отношения, и их атрибуты.
2. Описать связи между отношениями с точки зрения их множественности с одной и другой стороны, обязательности, соответствия бизнес правилу предметной области.
3. Построить в среде SQL server Manadgment Sydudio таблицы в соответствии с заданием.
4. Построить диаграмму отношений в среде SQL server Ma- nadgment Sydudio.
5. Произвести заполнение отношений тестовыми данными.

ЗАДАНИЯ:

В качестве заданий выдается примерная формулировка темы для курса лабораторных работ, в результате которых должна быть создана база данных архитектуры «сервер баз данных», то есть база данных должна быть дополнена серверными механизмами для работы с ней. База данных, созданная на курсе лабораторных работ должна содержать не менее 4 таблиц.

Примеры вариантов заданий.

1. Разработка информационной системы обеспечения хранения, накопления и выборки данных о рейсах междугородних автобусов автовокзала.
2. Разработка информационной системы обеспечения хранения, накопления и выборки данных об охотничьих угодьях Кемеровской области, их ресурсах и выдаче лицензий на охоту.
3. Разработка информационной системы обеспечения хранения, накопления и выборки данных о садовых участках кемеровско го района, расположении, владельцах, данные об участке, наименование кооператива, председатель кооператива.
4. Разработка информационной системы обеспечения хранения, накопления и выборки данных об аппаратном обеспечении персональных компьютеров и его поставщиков.
5. Разработка информационной системы обеспечения хранения, накопления и выборки данных о цветах, букетах цветочного магазина.
6. Разработка информационной системы обеспечения хранения, накопления и выборки данных о студентах, учебных группах, успеваемости (база деканат).
7. Разработка информационной системы обеспечения хранения, накопления и выборки данных о состоянии сданной в ремонт компьютерной техники.

8. Разработка информационной системы обеспечения хранения, накопления и выборки данных о тарифах и услугах сотовых операторов.

9. Разработка информационной системы обеспечения хранения, накопления и выборки данных о зоологических особенностях животных.

10. Разработка информационной системы обеспечения хранения, накопления и выборки данных о деятельности гостиницы. Кли енты, номера, проживание клиентов в номерах.

11. Разработка информационной системы обеспечения хранения накопление выборки данных материального обеспечения учебного процесса кафедры «Прикладная механика» КузГТУ.

12. Разработка информационной системы обеспечения хранения накопление выборку данных об индивидуальных прогнозах личностей (гороскоп).

13. Разработка информационной системы обеспечения хранения накопление выборку данных о музыкальных направлениях, и произведениях

14. Разработка информационной системы обеспечения хранения накопление выборку данных о содержании учебного процесса, учебном плане, программах курсов, расписании и их выполнении

15. Разработка информационной системы обеспечения хранения накопление выборки данных об Интернет-провайдерах, их услугах и пользователях.

16. Разработка информационной системы обеспечения хранения накопление выборки данных маршрутах средств общественноготранспорта.

17. Разработка информационной системы обеспечения хранения накопление выборки данных о поставке, лицах, осуществляющих поставку и затратах на разгрузку товара в торговый комплекс

«Палата».

18. Разработка информационной системы обеспечения хранения накопление выборки данных о соревнованиях Формула 1.

19. Разработка информационной системы обеспечения хранения накопление выборки данных об анкетировании студентов.

20. Разработка информационной системы обеспечения хранения накопление выборки данных о выполнении графика подготовки спортсмена-лыжника к соревнованиям.

21. Разработка информационной системы обеспечения хранения, накопления и выборки, данных обеспечивающих работу агентства недвижимости.

22. Разработка информационной системы обеспечения хранения накопление выборки данных об игроках в футбол команд высшей и первой лиги.

23. Разработка информационной системы обеспечения хранения накопление выборки данных о лекарственных средствах,имеющихся в наличии в аптеках города.

24. Разработка информационной системы обеспечения хранения накопление выборки данных обеспечивающих работу автомагазина.

25. Разработка информационной системы обеспечения хранения накопление выборки данных об иероглифах и их сочетаниях китайского языка (китайский словарь)

26. Разработка информационной системы обеспечения хранения накопление выборки данных о музыкальных магазинах г. Кемерово, наличии в них аудио, видео дисках, их содержании и исполнителях.

27. Разработка информационной системы обеспечения хранения накопление выборки данных о странах мира, их основных характеристиках, граничащих странах.

28. Разработка информационной системы обеспечения хранения, накопление и выборки данных о результатах игр сезона по футболу.

29. Разработка информационной системы обеспечения хранения, накопление и выборки данных об игровом компьютерном клубе: игроки, игры, результаты.

30. Разработка информационной системы обеспечения хранения накопление

выборки данных о делах, ведомых в ГУВД, фигурантах дел.

31. Разработка информационной системы обеспечения хранения накопление выборки данных о авто-аксессуарах, продаваемы в магазине.

32. Разработка информационной системы обеспечения хранения накопление выборки данных обеспечивающих работу автопредприятия, тип транспортного средства, грузоподъемность, состояние.

33. Разработка информационной системы обеспечения хранения, накопление и выборки данных о начислении зарплаты работникам предприятия. Работник. Дата. Начислено. Необходимо данные об отделах, в которых работают работники.

34. Разработка информационной системы обеспечения хранения, накопление и выборки данных о кулинарных рецептах.

35. Разработка информационной системы обеспечения хранения накопление выборки данных высаженных культурах, исполнителях, проведенных работах, истории посадок на садовом участке.

36. Разработка информационной системы обеспечения хранения накопление выборки данных о нотных записях и текстах музыкальных произведений.

37. Разработка информационной системы обеспечения хранения накопление выборки данных о чемпионате России по баскетболу.

38. Разработка информационной системы обеспечения хранения, накопление и выборки данных о репертуаре театра на сезон.

39. Разработка информационной системы обеспечения хранения, накопление и выборки данных о данных соревнованиях по велоспорту.

40. Разработка информационной системы обеспечения хранения, накопление и выборки данных о товарах ружейного магазина (характеристики оружия, боеприпасы, аксессуары).

Контрольные вопросы:

1. Что подразумевается под схемой базы данных?
2. Каким образом выявляются отношения базы данных?
3. Чему соответствует таблица базы данных в предметнойобласти?

Практическая работа №4 Манипулирование данными (хранение, добавление, редактирование данных). Манипулирование данными (удаление данных, навигация по набору данных).

Цель занятия: освоить процессы процесс создания, редактирования и удаления данных.

Краткие теоретические сведения:

Характеристики СУБД.

В общем случае под СУБД можно понимать любой программный продукт,

поддерживающий процессы создания, ведения и использования БД. Рассмотрим какие из имеющихся на рынке программ имеют отношение к БД и в какой мере они связаны с базами данных. К СУБД относятся следующие основные виды программ:

полнофункциональные СУБД;

серверы БД;

клиенты БД;

Полнофункциональные СУБД (ПФСУБД) представляют собой традиционные СУБД, которые сначала появились для больших машин, затем для мини-машин и для ПЭВМ. Из числа всех СУБД современные ПФСУБД являются наиболее многочисленными и мощными по своим возможностям. К ПФСУБД относятся, например,

такие пакеты как: ClarionDatabaseDeveloper, DataEase, DataFlex, dBase IV, MicrosoftAccess, MicrosoftFoxPro и Paradox R:BASE. Обычно ПФСУБД имеют развитый интерфейс, позволяющий с помощью команд меню выполнять основные действия с БД: создавать и модифицировать структуры таблиц, вводить данные, формировать запросы, разрабатывать отчеты, выводить их на печать и т. п. Для создания запросов и отчетов не обязательно программирование, а удобно пользоваться языком QBE (QueryByExample – формулировки запросов по образцу). Многие ПФСУБД включают средства программирования для профессиональных разработчиков.

Некоторые системы имеют в качестве вспомогательных и дополнительные средства проектирования схем БД или CASE-подсистемы. Серверы БД предназначены для организации центров обработки данных в сетях ЭВМ. Эта группа БД в настоящее время менее многочисленна, но их количество постепенно растет. Серверы БД реализуют функции управления базами данных, запрашиваемые другими (клиентскими) программами обычно с помощью операторов SQL.

Примерами серверов БД являются следующие программы: NetWare SQL (Novell), MS SQL Server (Microsoft), InterBase (Borland), SQLBaseServer (Gupta), IntelligentDatabase (Ingress).

В роли клиентских программ для серверов БД в общем случае, могут использоваться различные программы: ПФСУБД, электронные таблицы, текстовые процессоры, программы электронной почты и т. д. При этом элементы пары “клиент-сервер” могут принадлежать одному или разным производителям программного обеспечения.

В случае, когда клиентская и серверная части выполнены одной фирмой, естественно ожидать, что распределение функций между ними выполнено рационально.

В остальных случаях обычно преследуется цель обеспечения доступа к данным “любой ценой”. Примером такого соединения является случай, когда одна из полнофункциональных СУБД играет роль сервера, а вторая СУБД (другого производителя) – роль клиента. Так, для сервера БД SQL Server (Microsoft) в роли клиентских (фронтальных) программ могут выступать многие СУБД, такие как: dBASE IV, BlythSoftware, Paradox, DataEase, Focus, 1-2-3, MDBS III, Revelation и другие.

Практические задания

Задание:

Задание 1. Создадим запрос на выборку из двух таблиц с помощью мастера. Создайте запрос «Исполнитель», в котором представлены фамилии сотрудников и сокращенное название отдела, в котором они работают.

Запрос − это операция, которая объединяет в себе основные режимы обработки данных: сортировку, фильтрация, объединение данных из разных источников, преобразование данных. Под преобразованием данных понимается возможность создания вычисляемых полей, в которых по формулам на основании имеющейся информации получается новая.

Последовательность работы:

1. В главном окне базы данных щелкните по карточке *Создание* и перейдите на блок команд *Другие* (рис.1).

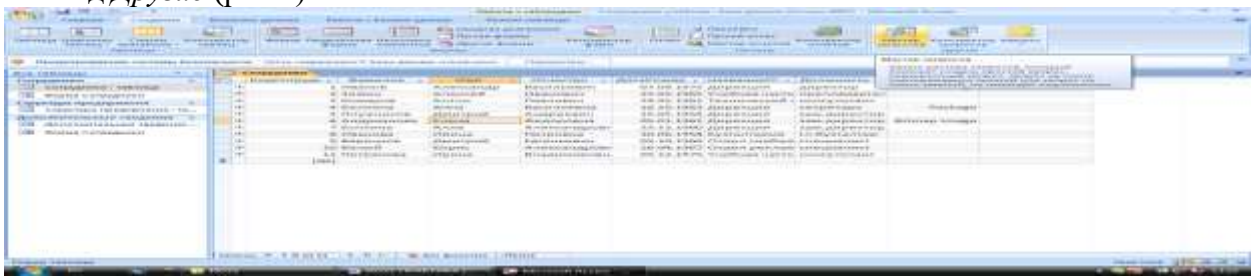


Рис.1

2. Запустите *Мастер запросов*. Появится окно, позволяющее выбрать тип запроса – простой, перекрестный, повторяющиеся записи, записи без подчиненных. Выбираем простой запрос.
3. Далее выбираем нужные нам поля. Интересующая нас информация находится в двух таблицах, поэтому поля будем выбирать последовательно из обеих таблиц. Для этого в списке Таблицы и запросы сначала выбираем таблицу Структура предприятия и в ней – поле Сокращение (переносим его из левой части окна в правую). Затем выбираем следующую таблицу – Сотрудники и в ней – поле Фамилия.
4. На следующем шаге задаем имя запроса - Исполнитель и открываем запрос для просмотра.
5. Просмотрите результаты запроса в режиме таблицы (рис.2):

Рис. 2

3. Перейдите в режим *Конструктор* запросов. Откроется бланк запроса (рис.3).

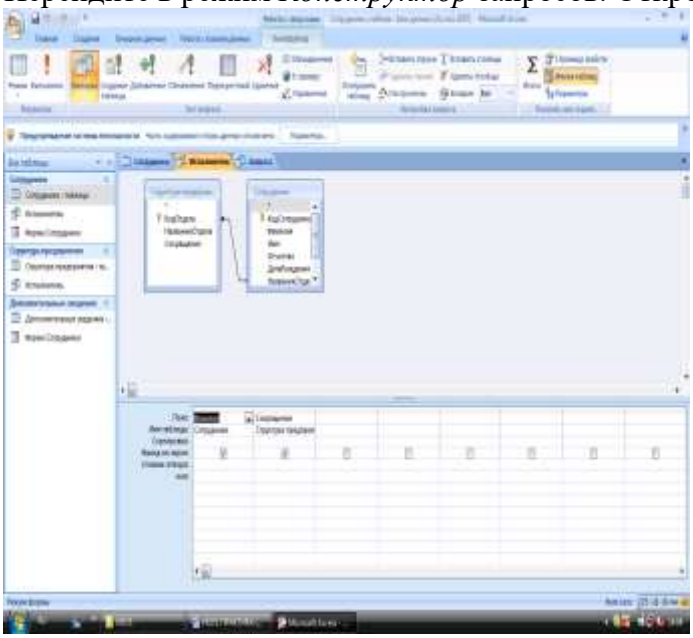


Рис.3

3. В верхней части бланка расположена схема связи таблиц, используемых в запросе. В нижней части расположена таблица описания полей запроса. В первой строке перечислены поля запроса. Во второй строке указано имя таблицы, из которой взято поле. В третьей строке можно задать сортировку полей.

Задание 2. Создание запросов с вычисляемыми полями.

Создадим запрос с именем «Вычисляемые поля», в котором по данным таблицы Сотрудники будут получены новые данные со следующими назначениями (см. таблицу 6):

Таблица 6

Имя поля запроса	**Назначение**
КодСотрудника	Устанавливает связь получаемых в других полях данных с конкретным сотрудником по ключевому полю
ФИО	Содержит фамилию, имя и отчество как одну строку
Возраст	Вычисляет количество полных лет по дате рождения
Месяц	Определяет номер месяца рождения по дате
День	Определяет порядковый день месяца рождения по дате

Созданные в запросе поля Месяц и День позволяют по-другому провести сортировку сотрудников по месяцам и дням даты рождения и составить список, в котором сотрудники будут указаны в порядке дат рождения от начала года.

1. Создайте новый запрос в режиме *Конструктор*. Откроется окно *Добавление таблицы*.
2. В окне *Добавление таблицы* выделите таблицу Сотрудники и щелкните на кнопке *Добавить*. Откроется бланк запроса. В верхней части бланка представлен список полей таблицы Сотрудники. Закройте окно *Добавление таблицы*.
3. В первом столбце бланка запроса введите имя Поля КодСотрудника, выбрав его из списка, который раскроется при щелчке на первой строке. Имя таблицы появится во второй строке автоматически.
4. В следующем столбце напишите имя поля самостоятельно - ФИО, в нем фамилия, имя и отчество сотрудника будут представлены в виде единой текстовой строки. Воспользуемся для этого *Построителем выражений* (см. теорию). Для этого правой кнопкой мыши щелкните на первой строке *Поле* второго столбца и в контекстном меню выберите команду *Построить*: откроется окно *Построитель выражений* (рис. 4).

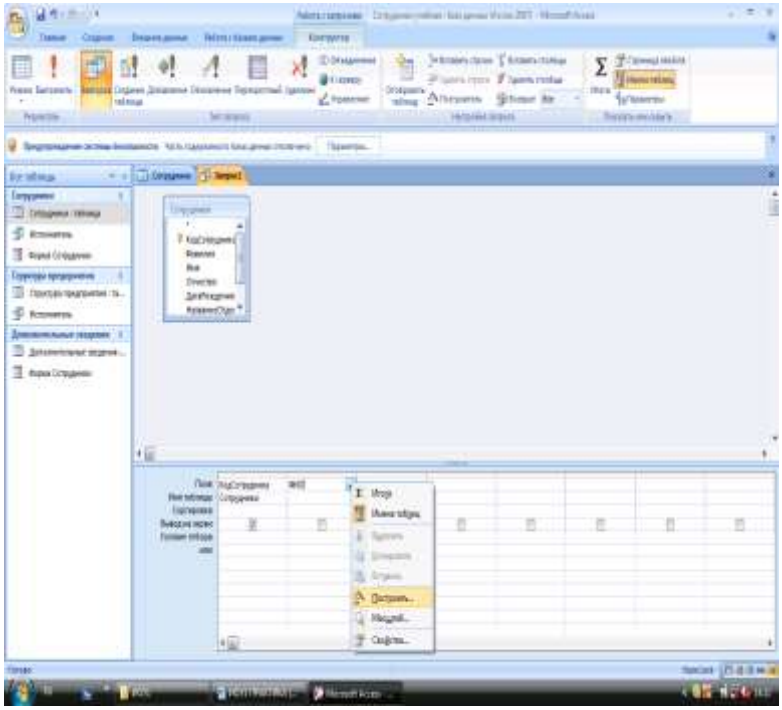


Рис. 4

5. В левом окне обзора раскройте папку Таблицы и в ней вложенную папку Сотрудники: поля таблицы Сотрудники будут представлены в среднем окне построителя запросов (рис.5).

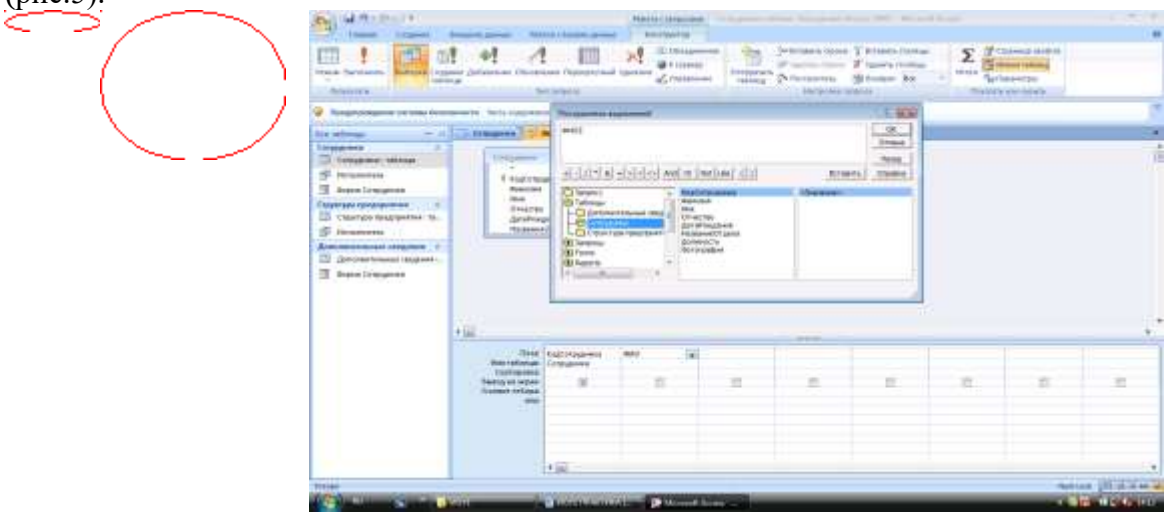


Рис.5

6. Введите формулу для вычисляемого поля ФИО согласно схеме на рис. 7. Часть значков можно набирать с клавиатуры в самом окне *Построителя.*
7. Завершите ввод формулы, нажав *ОК*.
8. Убедитесь, что формула появилась в бланке запроса. Так как ширина столбца не очень большая, то вся формула не будет видна. Либо увеличьте ширину столбца, либо просмотрите формулу, перемещая по ней курсор.
9. В третьем столбце постройте выражение для поля Возраст, в котором производится вычисление количества полных лет по дате рождения:
   Возраст: Year(Now())-Year([Сотрудники]![ДатаРождения])
   Эта формула содержит встроенные функции *Year (),* которая вычисляет год по дате, и *Now(),* которая вычисляет текущую дату. Встроенные функции можно найти, открыв в построителе выражений в окне обзора папки *Функции* папку *Встроенные функции*. Возраст получается как разность между годом, отсчитанным от текущей даты, и годом, отсчитанным от даты рождения (рис.6).

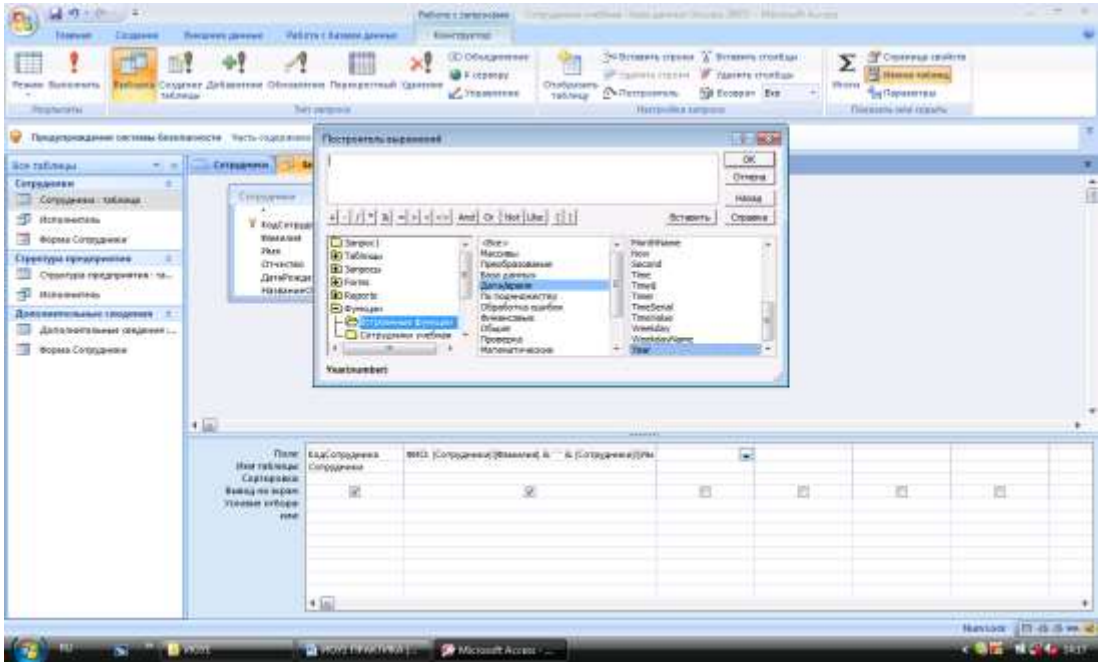


Рис.6

Схема формулы вычисляемого поля ФИО

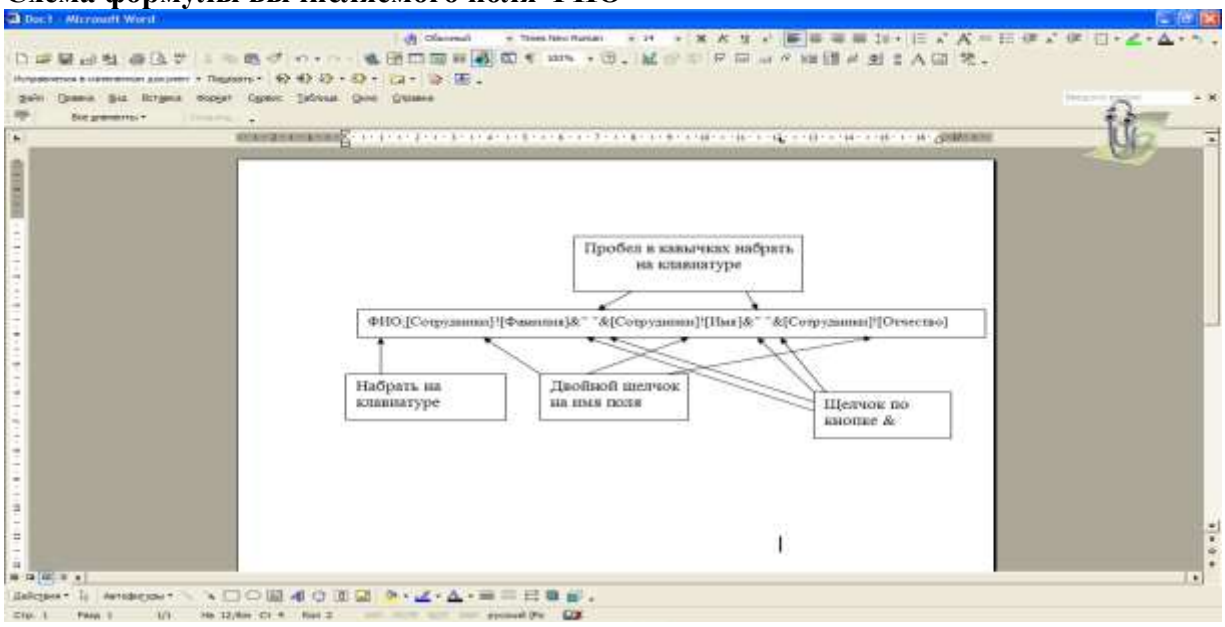


Рис. 7

В следующем столбце постройте выражение для поля Месяц, в котором производится вычисление по дате рождения порядкового номера месяца. В Формуле используется встроенная функция *Month():*

Месяц:Month([Сотрудники]![ДатаРождения])

10. В следующем столбце постройте выражение для поля День, в котором производится вычисление по дате рождения порядкового дня месяца. В формуле используется встроенная функция *Day():*

День: Day([Сотрудники]![ДатаРождения])

Запросы в *Конструкторе* нужно запускать на выполнение. Для этого нажмите кнопку *Выполнить* на панели инструментов  . Просмотрите результаты и сравните их с рис. 8.

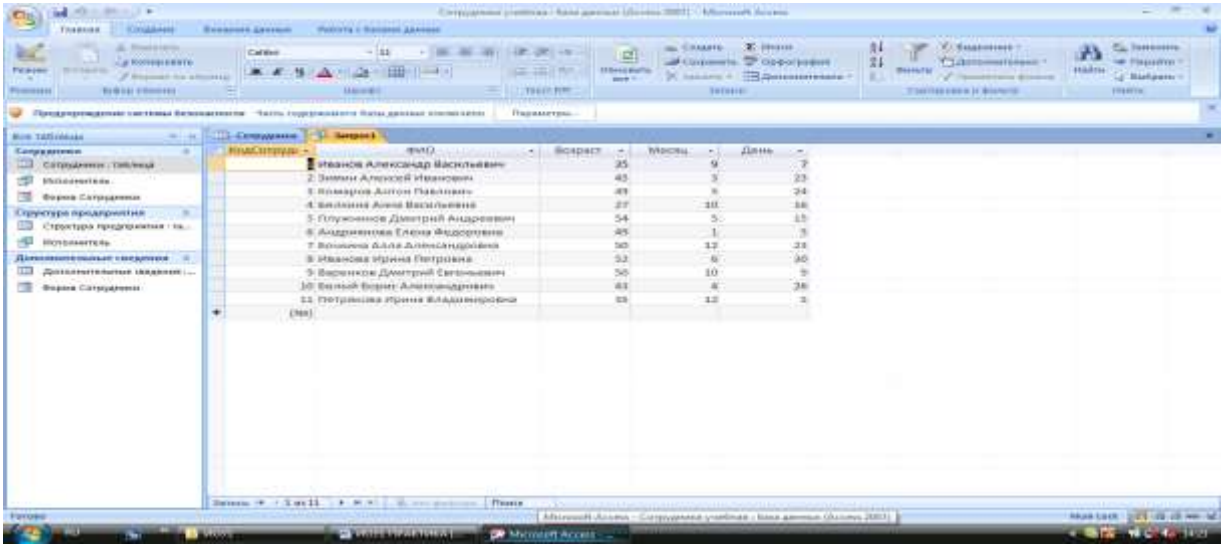


Рис. 8

Задания для самостоятельной работы:

1. Создайте в том же бланке запроса поле Адрес, в котором по названию улицы номеру дома и квартиры формируется адрес в виде одной строки.
2. Измените выражение так, чтобы в адресе автоматически прописывались в нужных местах слова улица, дом и квартира. Например, в результате должно получиться так: улица Иванова дом 5 квартира 75.
3. Введите в бланк запроса условие, по которому отбираются все сотрудники в возрасте от 25 до 40 лет (Вспомните, как задаются логические условия или воспользуйтесь справкой программы в разделе «Логические функции», Использование условий отбора для поиска определенных записей).

Задание 3. Создание параметрических запросов (запросов с параметрами).

Задание: создать запрос, результаты которого зависят от введенного параметра. *Запрос с параметром* – это запрос, при выполнении которого пользователю предлагается ввести значение какого-либо параметра. Это удобно, так как не требуется для изменения какого-либо параметра переходить в режим *Конструктора* запроса. Например, нам нужно отобразить список сотрудников того или иного отдела. В качестве параметра будем использовать название отдела.

Последовательность работы:

1.Создайте новый запрос в режиме *Конструктора.*
2. Укажите тип запроса – *Выборка*.
3. Поскольку нам нужны сведения из двух таблиц – Сотрудники и Структура предприятия – добавьте их в верхнюю часть запроса.
4. Имя первого поля - Фамилия, второго – имя, третьего – отчество. В четвертом поле напишите имя Название отдела. В строке *Условие отбора* четвертого столбца напишите в квадратных скобках фразу: [Введите название отдела].
5. Запустите запрос на выполнение. В результате у вас появится диалоговое окно (рис. 9).

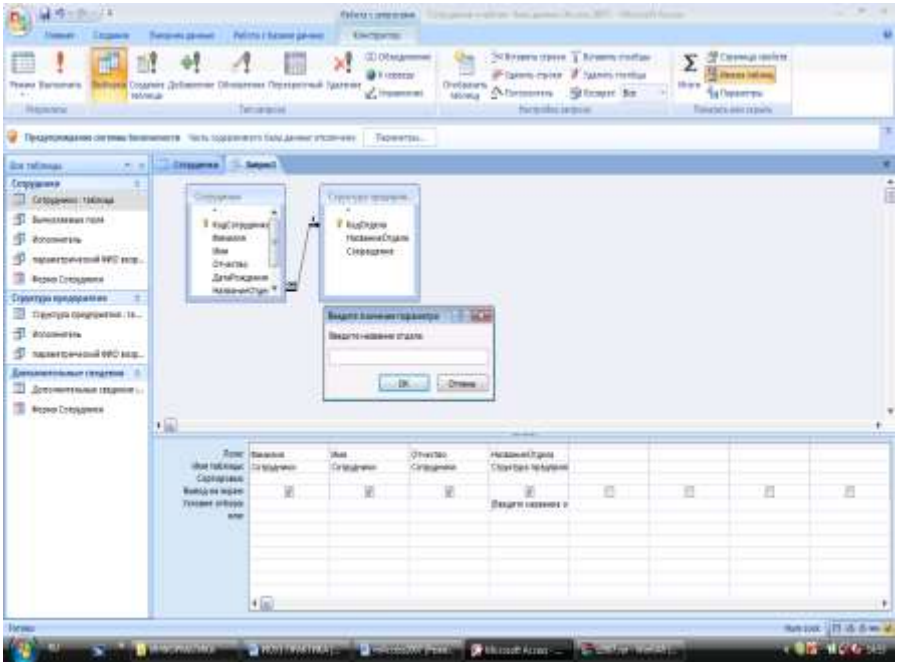


Рис.9

6. После указания нужного отдела, например, Дирекция, вы получите список работающих в нем сотрудников (рис.10). Запрос сохраните с именем «Параметрический отделы».

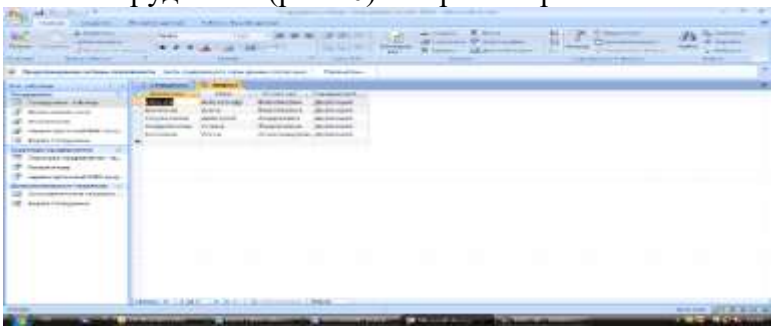


Рис.10

Задание для самостоятельной работы: создайте запрос с параметром, отображающий список сотрудников того или иного отдела (как предыдущий), но в нем должно указываться поле ФИО и возраст (рис.11). Подумайте, какие таблицы или запросы нужно добавить в этом случае в верхнюю часть запроса. Имя запроса – «Параметрический ФИО возраст».

Контрольные вопросы:

1. Какие манипуляции возможны с данными в БД?
2. Какие виды запросов вы знаете?
3. Что такое вычисляемое поле?

Практическая работа №5 Сортировка, поиск и фильтрация данных. Построение запросов к СУБД (различного уровня сложности)

Цель занятия: познакомиться с инструментами сортировки и фильтрации данных в БД.

Краткие теоретические сведения

Сортировка − упорядочение данных по какому-либо признаку. Для сортировки и поиска (фильтрации) информации в Access 2007 предусмотрен целый блок команд *Сортировка и фильтр* на карточке *Главная* (рис. 1).

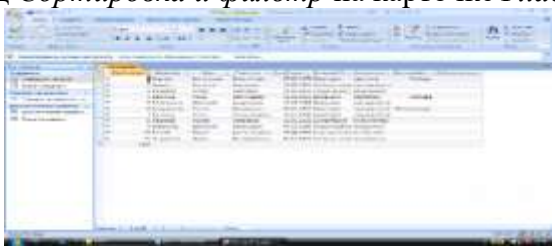


Рис. 1

При сортировке все строки таблицы перестраиваются в указанном порядке. Сортировка позволяет упорядочить данные любого типа: числа (в порядке возрастания), текст (по алфавиту), даты (в порядке возрастания года в дате, при одинаковых годах в порядке возрастания месяца).

Практические задания

Задание 1. Выполните следующие виды сортировок:

- Сортировка списка сотрудников по фамилиям в алфавитном порядке.
- Сортировка списка сотрудников по датам рождения в порядке убывания возраста.
- Сортировка списка сотрудников по ключевому полю в порядке возрастания.

Последовательность работы:

1. Откройте таблицу Сотрудники.
2. Выделите поле сортировки (Фамилия) щелчком на названии поля: при этом выделяется весь столбец с заголовком.
3. Щелкните на кнопке *Сортировка* по возрастанию. Просмотрите результаты сортировки: все фамилии расположены в алфавитном порядке.
4. Проведите другие виды сортировки, указанные в задании.

Задание 2. Поиск с использованием фильтра «*Выделение*».

Поиск (фильтрация) − выбор данных, удовлетворяющих некоторому условию. Выбор из базы данных тех записей, которые удовлетворяют требованиям пользователя, осуществляется с помощью фильтров - условий, по которым производится поиск и отбор записей.

Одним из самых простых способов отбора записей является использование фильтра «*Выделение*».

Порядок работы:

1. Откройте таблицу с данными о сотрудниках.
2. В какой-нибудь записи выделите значение одного из полей или его часть, например, первую букву фамилии Белкина.
3. Нажмите кнопку Выделение. Вам будут предложены варианты: .
4. После применения фильтра в таблице останутся только записи, удовлетворяющие выбранному условию. К уже отобранным записям можно вновь применить другой фильтр. Тогда останутся только записи, удовлетворяющие двум последовательно примененным критериям отбора.
5. Чтобы просмотреть все записи, надо нажать на кнопку *Применить фильтр*, которая включает и отключает фильтрацию. Среда баз данных помнит последний установленный фильтр.

Фильтр можно задать также в форме или запросе. Технология работы аналогична приведенной выше. Проведите в таблице Сотрудники отбор записей, удовлетворяющих следующим условиям:

- Фамилия сотрудника начинается на букву «Б».
- День рождения сотрудника в декабре.
- Сотрудники, работающие в подразделении Дирекция.
- Сотрудники, имеющие должность «менеджер».
- Менеджеры, работающие в отделе снабжения.

Задание 3. Простой фильтр.

Использование простого фильтра − другая возможность отбора данных. Простой фильтр позволяет задать сразу несколько критериев отбора по разным полям.

Последовательность работы:

1. Откройте таблицу с данными о сотрудниках**.**
2. Выберите команду *Фильтр*. В зависимости от положения курсора (типа поля, текстовое или числовое) появятся различные варианты, рис.2.

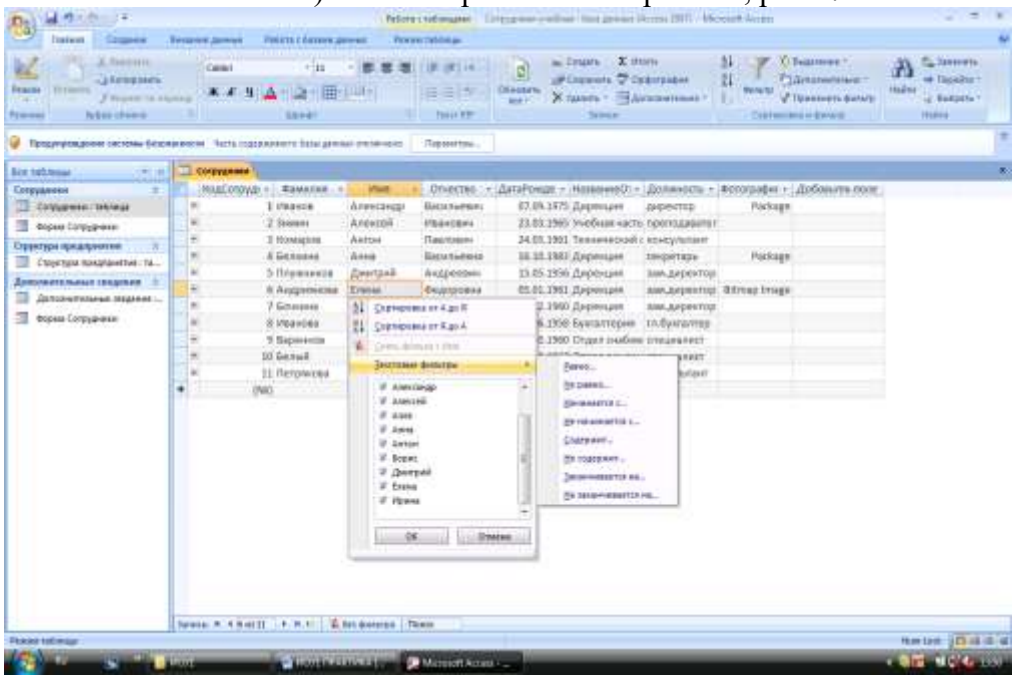


Рис.1

3. Выберите критерии отбора.
4. Дальнейшие действия аналогичны применению фильтра «*По выделенному*»**.**
5. Выполните фильтрацию, используя простой фильтр, по критериям задания 2.

Контрольные вопросы:

1. В чем суть сортировки данных
2. В чем суть фильтрации данных
3. Чем отличается сортиров5ка от фильтрации данных

Практическая работа №6 Построение концептуальной модели базы данных.

Цель занятия: Научиться создавать концептуальную схему базы данных для решения Конкретной задачи в соответствии с индивидуальным вариантом.

Краткие теоретические сведения

Проектирование структуры базы данных начнем с построения концептуальной модели. Концептуальная модель представляет собой высокоуровневый взгляд на предметную область. На данном этапе не учитывается модель данных и физические аспекты представления и хранения данных, проектирование одинаково для любой базы данных.

Одним из самых распространённых способов проектирования базы данных является построение модели «сущность-связь», также известных как ER-модели (англ. entity-relationship model). Модель, построенная таким образом, называется ER-диаграммой. В данной курсовой работе для этой цели используется нотация «Crow's Foot».

Моделирование с использованием модели «сущность-связь» предполагает:

- выделение в предметной области важных сущностей;
- описание их атрибутов и взаимосвязей.

Связи характеризуют в том числе мощность отношений между объектами сущностей. Наиболее важными типами таких отношений являются функциональные бинарные отношения:

- «один-к-одному»,
- «один-ко-многим»,
- и «многие-ко-многим».

При рассмотрении предметной обрасти деятельности туристических агентств можно выделить семь информационных сущностей:

1. Страны;
2. Города;
3. Виды транспорта;
4. Туристы;
5. Отели;
6. Туры;
7. Путевки.

Опишем детально предназначение каждой сущности и ее атрибутов.

Сущность «Страны».

Отвечает за хранение перечня стран мира, в которые совершаются туристические туры. Важным атрибутом этой сущности является «Название страны».

Сущность «Города».

Отвечает за хранение перечня городов, в которые совершаются туристические туры. Важными атрибутами этой сущности являются:

- Название города;
- Название страны, которой принадлежит город.

Атрибут сущности «Название страны» имеет связь «один-ко-многим» с сущностью «Страны».

Сущность «Виды транспорта».

Отвечает за хранение перечня видов транспорта, которым туристы доставляются от транспортных развязок к отелям. Важным атрибутом этой сущности является «Название транспорта».

Сущность «Туристы».

Отвечает за хранение перечня туристов, которые совершили туристические туры. Важными атрибутами этой сущности являются:

- ФИО туриста;
- Возраст.

Сущность «Отели».

Отвечает за хранение перечня отелей, которые принимают туристов на отдых. Важными атрибутами этой сущности являются:

- Название отеля;
- Класс обслуживания;
- Суточная плата за проживание в отеле;
- Название города, где размещен отель.

Атрибут сущности «Название города» имеет связь «один-ко-многим» с сущностью «Города».

Сущность «Туры».

Отвечает за хранение перечня туров в отелях, с указанием продолжительности заезда. Важными атрибутами этой сущности являются:

- Название тура;
- Продолжительность;
- Описание;
- Вид транспорта для доставки туристов в отель.

Атрибут сущности «Вид транспорта» имеет связь «один-ко-многим» с сущностью «Виды транспорта».

Сущность «Путевки».

Основная сущность информационной системы, хранящая информацию о распределении туристов по отелям и заездам. Важными атрибутами этой сущности являются:

- Дата вылета на отдых;
- Тур;
- Отель;
- Турист.

Атрибут сущности «Тур» имеет связь «один-ко-многим» с сущностью «Туры».

Атрибут сущности «Отель» имеет связь «один-ко-многим» с сущностью «Отели».

Атрибут сущности «Турист» имеет связь «один-ко-многим» с сущностью «Туристы».

Построенная ER-модель в графической нотации «Crow's Foot» представлена на рис. 3.

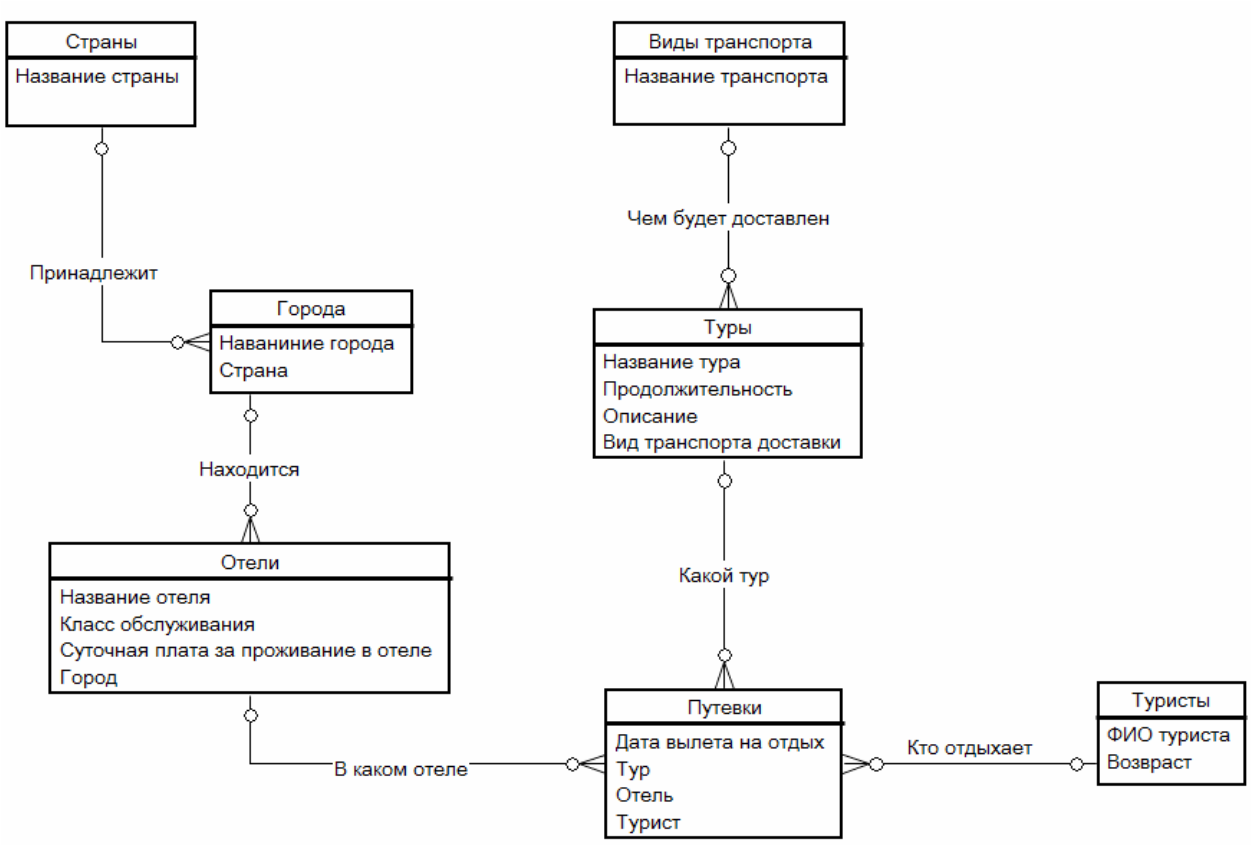


Рис. 3. ER-модель информационной системы

Таким образом, при помощи модели «сущность-связь» на высоком уровне проанализирована предметная область, выявлены её важнейшие сущности, а также их атрибуты и характер взаимосвязей. Результат представлен в соответствующей графической нотации.

Практические задания

Задание:

В соответствии с индивидуальным вариантом создайте концептуальную схему базы данных.

Варианты индивидуальных заданий:

1. Отделение коммерческого банка
2. Поликлиника
3. Колледж
4. Отделение полиции
5. Дизайнерская фирма
6. Офис интернет-провайдера
7. Агентство недвижимости
8. Туристическое агентство
9. Офис благотворительного фонда
10. Издательство
11. Рекламное агентство
12. Отделение налоговой службы
13. Редакция газеты
14. Гостиница
15. Праздничное агентство
16. Городской архив
17. Диспетчерская служба такси

Порядок отчета практической работы

При отчете практической работы необходимо:

1)Продемонстрировать выполненные задания по индивидуальному варианту, прокомментировать порядок его выполнения и объяснить полученные результаты

2)Ответить на контрольные вопросы.

Контрольные вопросы:

1) Перечислите основные этапы проектирования БД?

2) Определите соотношение понятия “сущность”, “связь”?

3) В чем заключается концептуальное проектирование для конкретной предметной области?

Практическая работа №7 Создание логической модели данных с помощью утилиты автоматизированного проектирования базы данных.

Цель занятия: научиться использовать операторы языка SQL для работы с данными БД.

Краткие теоретические сведения

Логическая модель данных или логическая схема-это модель данных конкретной предметной области, выраженная независимо от конкретного продукта управления базами данных или технологии хранения (физической модели данных), но в терминах структур данных, таких как реляционные таблицы и столбцы, объектно-ориентированные классы или теги XML.

Логическими уровень – это абстрактный взгляд на данные, на нем данные представляются так, как выглядят в реальном мире, и могут называться так, как они называются в реальном мире.

Объекты модели, представляемые на логическом уровне, называются *сущностями и атрибутами*. Логическая модель данных может быть построена на основе другой логической модели, например на основе модели процессов. Логическая модель данных является универсальной и никак не связана с конкретной реализацией СУБД.

Различают три уровня логической модели, отличающихся по глубине представления информации о данных:

- диаграмм сущность-связь (Entity Relationship Diagram, ERD);
- модель данных, основанная на ключах (Key Based model, KB);
- полная атрибутная модель (Fully Attributed model, FA).

Диаграмма сущность-связьпредставляет собой модель данных верхнего уровня. Она включает сущности и взаимосвязи, отражающие основные бизнес-правила предметной области. Такая диаграмма не слишком детализирована, в нее включаются основные сущности и связи между ними, которые удовлетворяют основным требованиям, предъявляемым к ИС.

Диаграмма сущность-связь может включать связи многие-ко-многим и не включать описание ключей. Как правило, ERD используется для презентаций и обсуждения структуры данных с экспертами предметной области.

Модель данных, основанная на ключах, - более подробное представление данных. Она включает описание всех сущностей и первичных ключей и предназначена для представления структуры данных и ключей, которые соответствуют предметной области.

Полная атрибутивная модель – наиболее детальное представление структуры данных: представляет данные в третьей нормальной форме ивключает все сущности, атрибуты и связи.

Практические задания

На основании лекционного материала подготовить схемы баз данных разных логических моделей – реляционной, иерархической, сетевой.

Темы баз данных:

корабли второй мировой

автосервис

компьютерная фирма

аэропорт

регистратура поликлиники

журнал колледжа

склад готовой продукции фирмы вторсырья

анкеты студентов

магазин автозапчастей

2. В реляционной модели данных должны быть связи всех типов – один к одному, один ко многим и многие ко многим.

Контрольные вопросы:

1. Что такое логическая модель баз данных?
2. Какие уровни логической модели базы данных вы знаете?
3. Что включает в себя диаграмма сущность-связь?
4. В чем особенность полной атрибутивной модели?

Практическая работа №8 Создание физической модели данных с помощью утилиты автоматизированного проектирования базы данных

Цель занятия:

Краткие теоретические сведения

Различают два уровня физической модели:

- трансформационная модель (Transformation Model);
- модель СУБД (DBMS Model).

Физическая модель содержит всю информацию, необходимую для реализации конкретной БД. Трансформационная модель содержит информацию для реализации отдельного проекта, который может быть частью общей ИС и описывать подмножество предметной области. ERwin поддерживает ведение отдельных проектов, позволяя проектировщику выделять подмножество модели в виде предметных областей (Subject Area). Трансформационная модель позволяет проектировщикам и администраторам БД лучше представлять, какие объекты БД хранятся в словаре данных, и проверить, насколько физическая модель данных удовлетворяет требованиям к ИС.

Модель СУБД автоматически генерируется из трансформационной модели и является точным отображением системного каталога СУБД. ERwin непосредственно поддерживает эту модель путем генерации системного каталога.

Физический уровень представления модели зависит от выбранного сервера. Для смены базы данных, в которой будет реализована физическая модель, следует в меню **Database** выбрать режим **Choose Database**. В открывшемся окне ***Computer Associates ERwin – Target Server*** (см. рис. 1) в блоке **Target SQL DBMS** следует установить переключатель на имени требуемой БД. В нижней части данного окна сразу будет отражен стандарт данной БД по длине и формату полей.

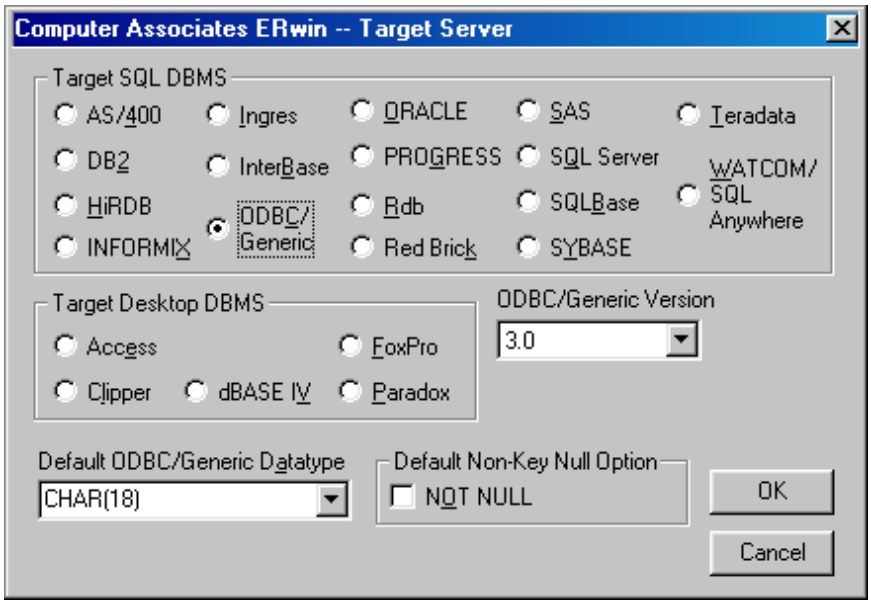


Рис. 1. Окно для выбора базы данных.

Пример физической модели данных представлен на рис. 2.

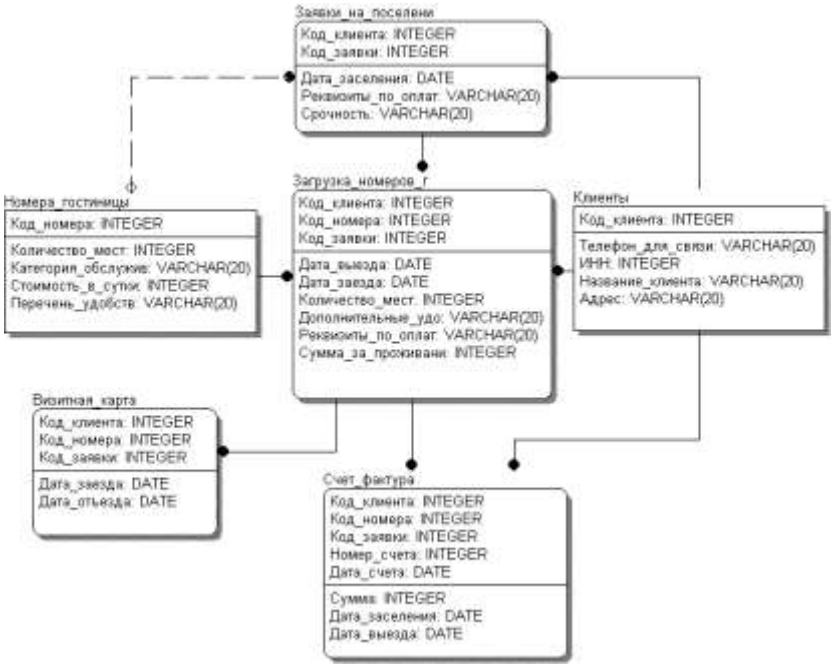


Рис. 2. Пример физической модели данных

По завершении работы над информационной моделью, как правило, распечатываются логический и физический уровни диаграммы, а также отчет по соответствиям сущность-таблица, атрибут-имя колонки, сущность-атрибуты. Для этого в меню **Tools** следует выбрать пункт **Report Builder**, в котором – подпункт **Report Builder**. Откроется диалоговое окно ***Report Templates*** (см. рис. 3), в котором следует выбрать тип отчета (например, *Standard.erp*) и тип представления выходных данных (HTML, RTF, TEXT) и нажать кнопку Run. Откроется диалоговое окно ***Import From ERP*** (рис. 3), в котором следует нажать кнопку Select All, а затем – кнопку OK.

Диаграмма физической модели является необходимым, почти достаточным и очень удобным материалом для разработчиков программ.

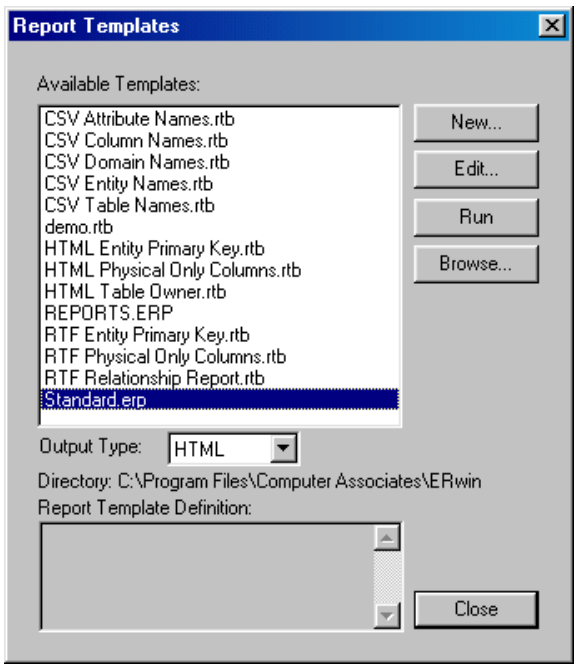


Рис. 3. Окно для выбора типа отчета

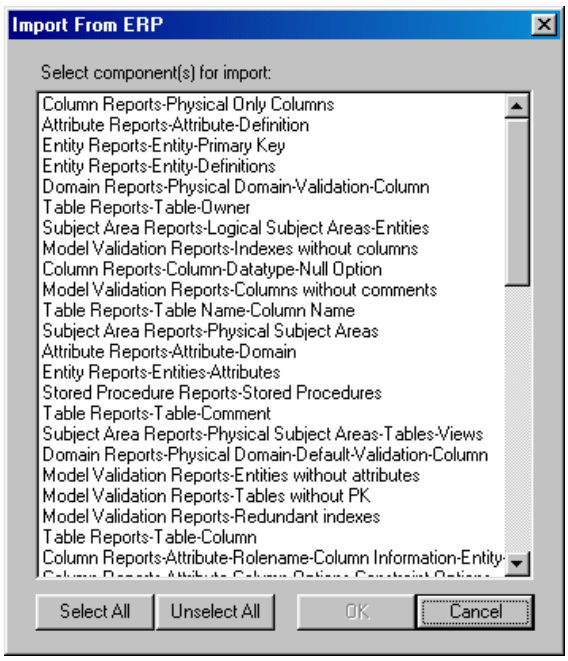


Рис. 4. Окно для выбора компонентов отчета

Сгенерированный отчет может быть сохранен на диск (колонки разделяются запятыми, выравниваются или разделяются табуляцией) или передан в текстовый процессор (или электронную таблицу) через интерфейс DDE.

Практические задания

1. Написать запрос для выбора автомобилей определенного цвета. Цвет задается в виде параметра в условии WHERE (например, ‘белый’).
2. Определить количество автомобилей, у которых номер фондового извещения начинается на "10" и не заканчивается на "39"
3. По каждой штатной группе а/м определить, сколько а/м каждой марки было выпущено в заданном году. Вывести названия групп и названия марок на экран.
4. Определить, какие а/м данного класса переданы в подразделения после указанной даты. Указать также номер автомобиля и дату документа передачи каждого а/м.
5. Произвести выборку автомобилей из двух полей «номер авто», «класс авто» (подставлять название из отношения MENU). Если поле «класс» в таблице MENU не существует, то выводить строку «Класс средства неизвестен» с помощью функции iif.
6. Определить, сколько а/м каждой марки имеют год выпуска меньший, чем округленный до целого средний год выпуска а/м заданной пользователем марки.
7. Определить какое количество а/м каждой марки в каком году было произведено (перекрестный запрос: марки а/м на год производства)

Контрольные вопросы:

1. Что такое физическая модель данных?
2. Виды физических моделей данных?
3. Какую информацию содержит в себе трансформационная модель?

Практическая работа №9 Разработка серверной части базы данных в инструментальной оболочке. Разработка клиентской части базы данных в инструментальной оболочке.

Цель занятия: научиться формировать на языке SQL простейшие запросы к базе данных, использовать в запросах выражения, включающие в себя арифметические операции, функции для работы со строками и датами, агрегатные функции.

Краткие теоретические сведения

Любое клиент-серверное приложение состоит из клиентского и серверного приложений. Соответственно этому имеются инструментальные среды разработки клиентской части и серверной. В качестве первых обычно выступают интегрированные среды разработки, ИСР (Integrated Development Environment, IDE). В качестве вторых - системы управления базами данных, СУБД.

Инструментальные средства для разработки клиентской части

Клиентской называется часть приложения, с которой напрямую взаимодействует конечный пользователь. Это может быть либо приобретенное компанией серийное коммерческое программное обеспечение, либо прикладная программа, разработанная внутри компании с помощью инструментальных средств третьих фирм. В следующих абзацах мы кратко рассмотрим такое программное обеспечение.

Для того, чтобы воспользоваться инструментальными средствами, предназначенными для создания клиентской части приложений, которые доступны сегодня на рынке программного обеспечения, разработчики должны уметь программировать на таких языках, как C++ и HTML, или на одном из множества других процедурных языков программирования, предназначенных для разработки Web-приложений. Раньше для разработки пользовательских корпоративных программ, работающих в основном в символьном режиме, использовались такие языки программирования, как ANSI C, COBOL, FORTRAN и Pascal. Сегодня большинство вновь разрабатываемых клиентских прикладных программ является GUI-приложениями - они содержат графический интерфейс пользователя. Большинство из доступных сегодня инструментальных средств являются дружественными по отношению к пользователю и объектно- ориентированными. В них широко используются пиктограммы, различного рода мастера, а также технология drag-and-drop. Наиболее популярными средствами для создания Web-приложений являются C++-Builder и IntraBuilder фирмы Borland, а также Visual J++ и Visual C++ компании Microsoft. Другие популярные средства разработки корпоративных приложений для локальных вычислительных сетей – PowerBuilder компании Powersoft, Developer/2000 корпорации Oracle, Visual Basic компании Microsoft и Delphi фирмы Borland.

Инструментальные средства для разработки серверной части

Ядром любой прикладной программы является ее серверная часть. Именно здесь незаметно для конечного пользователя базы данных происходит вся основная работа. Серверная часть приложения включает сам сервер БД, источники данных, а также связующее программное обеспечение, с помощью которого приложение подключается к Web-серверу или удаленной базе данных в локальной сети. (важнейшими серверами баз данных являются Oracle, Informix, Sybase, Microsoft SQL Server и Borland InterBase.)

Обычно это является первым шагом при подключении любого приложения к корпоративной среде предприятия или окружению Intemet/intranet. Сервер баз данных устанавливается в этом случае местным администратором БД, хорошо представляющим себе как нужды компании, так и требования, предъявляемые прикладной программой. Подключением приложения называется процесс его реализации в доступном пользователям окружении. Связующее программное обеспечение для подключаемого приложения включает Web-сервер и какое-нибудь инструментальное средство,

предназначенное для соединения Web-сервера с сервером баз данных. Главным требованием в этом случае является наличие на Web-сервере прикладной программы, способной общаться с корпоративной базой данных.

Практические задания

Запрос на выборку всей таблицы. Можно упростить вид запроса, если вместо запроса

SELECT Заказы.* FROM Заказы;

написать запрос

SELECT * FROM Заказы

Язык SQL позволяет опускать имя таблицы перед именем поля в тех случаях, когда в запросе используется одна таблица, или имя поля не повторяется в нескольких таблицах в многотабличном запросе.

Создание запросов на SQL в Access начинается вызовом конструктора запросов. Для этого в окне базы данных нужно выбрать объект «Запросы», пункт меню «Создать» и в окне «Новый запрос» пункт «Конструктор». Далее выберите таблицу «Заказы» и перейдите в режим SQL. Переход в режим SQL: меню Access Вид Режим SQL.

Зокончите формирование запроса и выполните его.

Вывод избранных полей, замена имён полей псевдонимами, сортировка записей. Поля таблицы выводятся на экран дисплея в том порядке, в котором они перечислены в запросе. Имена полей при выводе результатов запроса часто неудобны для чтения. Их можно заменить в запросе псевдонимами, как показано в примере:

```
SELECT КодЗаказа AS Заказ, НазваниеПолучателя AS Получатель,
АдресПолучателя AS Адрес, ДатаИсполнения AS Дата
FROM Заказы
ORDER BY НазваниеПолучателя ASC;
```

В примере *КодЗаказа, НазваниеПолучателя, АдресПолучателя* и *ДатаИсполнения* – имена полей в таблице «Заказы». При выводе результатов запроса на экран дисплея имена полей будут заменены соответствующими псевдонимами, указанными после слова *AS*.

Предложение

ORDER BY НазваниеПолучателя ASC

служит для сортировки отобранных записей по возрастанию (т.е. в алфавитном порядке) значения поля *НазваниеПолучателя.* Если нужно сортировать по убыванию, то вместо ASC нужно использовать DESC (сокращение от descending).

Сформируйте и выполните этот запрос.

Вывод записей без дублирования. Сформируйте и выполните следующий запрос

```
SELECT НазваниеПолучателя AS Получатель
FROM Заказы
ORDER BY НазваниеПолучателя DESC.
```

Названия получателей многократно повторяются, так как выбраны все записи таблицы. Чтобы не было дублирования записей, добавьте в запрос после слова SELECT слово DISTINCT. Иногда в СУБД режим DISTINCT установлен по умолчанию. Для вывода *всех* записей в этом случае после слова SELECT вставляется слово ALL.

Использование в запросе выражений. В списке вывода можно указывать не только имена полей и их псевдонимы, но и выражения, включающие в себя арифметические действия и функции.

Умножение. Сформируйте запрос на вывод из таблицы «Заказано» кода товара, цены, количества и общей стоимости заказанного товара. Запрос выглядит так:

```
SELECT КодТовара,Цена,Количество,Цена*Количество AS Стоимость
FROM Заказано;
```

Самостоятельно дополните запрос стоимостью со скидкой.

Использование функций. Функция **STR()** предназначена для преобразования в текстовый тип. Для вывода на экран дисплея стоимости товара в тысячах рублей с указанием единицы измерения служит следующий запрос:

SELECT КодТовара,*str*(Цена*Количество/1000)+' тыс.
руб' AS Стоимость FROM Заказано;

Для того чтобы в колонке «Стоимость» печатались число и текст, нужно преобразовать число в текстовый тип и объединить с текстом 'тыс. руб.'. Для преобразования служат функция *str(<выражение числового типа>)* и операция слияния «+» (конкатенация).

Сформируйте запрос, в котором из таблицы «Заказы» выбираются 5 полей и результат выводится в две колонки. В первую колонку выводится поле «КодЗаказа», а в колонке с псевдонимом «Адрес клиента» объединены следующие поля: ИндексПолучателя, СтранаПолучателя, ГородПолучателя, НазваниеПолучателя.

Не забудьте поставить между объединяемыми полями адреса запятую с пробелом. Результат запроса (показаны две первые строки) должен иметь вид:

Код заказа	Адрес клиента
10248	90110, Финляндия, Оулу, Wartian Herkku
10249	44087, Германия, Мюнстер, Toms Spezialitaten

Функция выделения части даты DATEPART(). Познакомьтесь с описанием этой функции в справке Access (Содержание, раздел «Справочник по языку Visual Basic», пункт «Functions», буква D).

Определите с помощью запроса к таблице «Заказы», за какие годы были поставки товаров.

Агрегатные функции. (В Access они называются статистическими). Подсчитаем общее количество записей в таблице «Заказы» и количество записей содержащих данные в поле «ОбластьПолучателя», то есть, количество записей с непустым полем «ОбластьПолучателя». Для этого выполним следующий запрос:

SELECT count(*),count(ОбластьПолучателя)
FROM Заказы;

В запросе используется агрегатная функция *COUNT()*. Используя агрегатные функции MAX(), MIN() и AVG(), составьте запрос для подсчёта максимальной минимальной и средней цены товара в таблице «Товары».

Используя агрегатную функцию SUM(), составьте запрос для подсчёта общей стоимости доставки всех заказанных товаров в таблице «Заказы».

Сохраните все созданные Вами запросы и покажите их преподавателю.

Контрольные вопросы:

1. Инструментальные средства для разработки клиентской части
2. Инструментальные средства для разработки серверной части

Практическая работа №10 Разработка технических требований к серверу баз данных

Цель занятия: освоить технологии оценки требований к серверу баз данных.

Краткие теоретические сведения

При проектировании системы автоматизации принимаются во внимание следующие требования:

– система должна нормально функционировать на стандартных персональных компьютерах клона IBM с процессором Pentium IV (минимальные требования), подсоединенных к локальной офисной вычислительной сети в режиме невыделенных серверов;

– система не должна иметь привязки к аппаратной части для возможности переноса ее на новую платформу из-за неизбежного морального старения компьютерной техники;

– архитектура системы должна быть выбрана таким образом, чтобы минимизировать вероятность нарушения штатного режима работы системы (выход системы из строя, разрушение информационной базы данных, потеря или искажение информации) при случайных или сознательных некорректных действиях пользователей;

– система должна обеспечивать защиту информационной базы данных от несанкционированного доступа;

– основная программная оболочка системы должна устанавливаться на рабочие места с любого компьютера, подсоединенного к локальной офисной вычислительной сети;

– основная программная оболочка должна иметь интуитивно ясный дружественный интерфейс и не должна требовать от пользователей специальной подготовки, не связанной с их профессиональными обязанностями;

– система должна иметь возможность наращивания программ.

Практические задания

1. Повторить теоретический материал.
2. Создать документ в формате Ms Excel, сохранить документ в родной папке ТЗ_Сервер_№варианта.xlsx.

	A	B	C
1	Исходная нагрузка(количество пользователей)		150
2	От 100 пользователей рекомендуется разделять физически сервер базы данных и сервер приложений.		
3	Аппаратное обеспечение		
4	Требования к серверу баз данных		
5	Минимальные требования		Рекомендуемые требования
6	Процессор (количество ядер)	2	Процессор: 2х ядерный процессор;
7	Процессор (тактовая частота в	3,2 и выше	Тактовая частота: 3,2 ГГц или выше;
8	Оперативная память (Гб)	3	Оперативная память: 5 ГБ или выше;
9	Дисковое пространство RAID10 (Тб)	0,32	Дисковое пространство: рабочие - RAID10 не менее 4 дисков (SCSI или SAS) суммарным объемом не менее 1 Тб
10	Дисковое пространство для хранения архивных копий (Тб)	0,64	Для хранения архивных копий - RAID1 2 диска (SCSI или SAS) суммарным объемом не менее 2 Тб
11			Диск для горячей замены (Hot Spare)
12	**Требования к серверу приложений (web-сервер) аналогичные**		
13	**Каналы связи:**	В случае развертывания системы в пределах локальной сети необходимо обеспечить пропускную способность сети в 100 Мбит/сек. Пропускная способность внешнего канала связи не менее 20 Мбит/сек с возможностью расширения.	

3. Расчёт минимальных требований к серверу базы данных осуществить по формулам в ячейках B6, B8, B9, B10 исходя из нагрузки-ячейка C1
4. Выбрать рекомендуемые требования, исходя из нагрузки. Кратки теоретические сведения

Для поддержания бесперебойной работы крупных проектов используют производительные сервера или целые кластеры серверных машин, где стоит, как правило, СУБД — комплекс программ для создания и манипулирования данными. Главное назначение выделенного сервера БД состоит в размещении, обработке и хранении информации силами достаточно производительной конфигурации, при этом все это происходит посредством одной из предустановленных СУБД. Непосредственно сама система управления базами предоставляет доступ к ним клиентам и приложениям и обеспечивает оперативную обработку запросов. Описанный формат взаимодействия также называют архитектурой типа «клиент- сервер».

Любое обращение к реляционной БД происходит в большинстве случаев на самом распространенном языке запросов SQL. В свою очередь платформа, на которой запущена СУБД, «понимающая» этот язык, и называется SQL-сервером.

При небольших нагрузках допустимо (а иногда и оправданно) разместить базу данных на основной вычислительной машине. Более крупные проекты, где число ежедневных запросов к базе превышает 500, разумнее реализовывать уже на отдельном SQL-сервере. Это позволяет оборудованию не распыляться на сторонние задачи, а сосредоточиться на выполнении типовых процессов, под которые заранее рассчитаны ресурсы и мощность оборудования.

Требования к обеспечению сервера баз данных

Требования для сервера БД и WEB-сервера идентичны. Рекомендованы физически разные машины.

Формулы расчета требований, исходя из количества зарегистрированных пользователей при условии, что одновременно работать будут максимум 50% пользователей, следующие:

- **Процессор**: количество одновременно работающих пользователей /100 ядер (3,2 ГГц и выше)
- **Оперативная память**: количество одновременно работающих пользователей /50
- **Дисковое пространство RAID10: не менее 4 дисков (SCSI или SAS) суммарным объемом**: 9мб * 100 создаваемых записей в день одним пользователем * количество зарегистрированных пользователей * 2,5
- **Для хранения архивных копий**: 9мб * 100 создаваемых записей в день

одним пользователем * количество зарегистрированных пользователей * 5

Аппаратное обеспечение	
Рекомендуемые требования к аппаратному обеспечению	
До 100 зарегистрированных пользователей	
Требования к серверу базы данных:	
• Процессор: 1 ядерный процессор; • Тактовая частота: 3,2 ГГц или выше; • Оперативная память: 1 ГБ или выше; • Дисковое пространство: рабочие - RAID10 не менее 4 дисков (SCSI или SAS) суммарным объемом не менее 200 Гб • Для хранения архивных копий - RAID1 2 диска (SCSI или SAS)	В количестве1 шт.

суммарным объемом не менее 500 Гб в расчете на 1 год работы в системе • Диск для горячей замены (Hot Spare)	
Требования к серверу приложений (web-сервер) аналогичные	
От 100 пользователей рекомендуется разделять	
До 500 зарегистрированных пользователей:	
Требования к серверу базы данных:	
• Оперативная память: 5 ГБ или выше; • Дисковое пространство: рабочие - RAID10 не менее 4 дисков (SCSIили SAS) суммарным объемом не менее 1 Тб • Для хранения архивных копий - RAID1 2 диска (SCSI или SAS) суммарным объемом не менее 2 Тб • Диск для горячей замены (Hot Spare)	1 шт.
Требования к серверу приложений (web-сервер) аналогичные	
До 1000 зарегистрированных пользователей:	
Требования к серверу базы данных:	
• Процессор: 5ти ядерный процессор; • Тактовая частота: 3,2 ГГц или выше; • Оперативная память: 12 ГБ или выше; • Дисковое пространство: рабочие - RAID10 не менее 4 дисков (SCSIили SAS) суммарным объемом не менее 2 Тб. • Для хранения архивных копий - RAID1 2 диска (SCSI или SAS) суммарным объемом не менее 4,5 Тб • Диск для горячей замены (Hot Spare)	В количестве 1 шт.
Требования к серверу приложений (web-сервер) аналогичные	
До 5000 зарегистрированных пользователей:	
Требования к серверу базы данных:	

• Процессор: 25ти ядерный процессор; • Тактовая частота: 3,2 ГГц или выше; • Оперативная память: 50 ГБ или выше; • Дисковое пространство: рабочие - RAID10 не менее 4 дисков (SCSIили SAS) суммарным объемом не менее 11 Тб • Для хранения архивных копий - RAID1 2 диска (SCSI или SAS) суммарным объемом не менее 21 Тб • Диск для горячей замены (Hot Spare)	В количестве 1шт.
Требования к серверу приложений (web-сервер) аналогичные	
До 10000 зарегистрированных пользователей:	
Требования к серверу базы данных:	
• Процессор: 50ти ядерный процессор; • Тактовая частота: 3,2 ГГц или выше; • Оперативная память: 100 ГБ или выше; • Дисковое пространство: рабочие - RAID10 не менее 4 дисков (SCSIили SAS) суммарным объемом не менее 22 Тб • Для хранения архивных копий - RAID1 2 диска (SCSI или SAS) суммарным объемом не менее 44 Тб • Диск для горячей замены (Hot Spare)	В количестве 1 шт.
Требования к серверу приложений (web-сервер) аналогичные	

ВАРИАНТ1

150 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

ВАРИАНТ2

2700 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

ВАРИАНТ3

430 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

ВАРИАНТ4

8000 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

ВАРИАНТ5

1500 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

ВАРИАНТ6

270 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

ВАРИАНТ7

4300 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

ВАРИАНТ8

800 зарегистрированных пользователей. Разработать технические требования к серверам базы данных: минимальные и рекомендуемые.

Контрольные вопросы:

1. Какие требования предъявляются к разработке технических требований к серверу баз данных?

2. Укажите формулы расчета требований, исходя из количества зарегистрированных пользователей?

Практическая работа №11 Модель сервера баз данных

Цель занятия: познакомиться с интерфейсом взаимодействия с PostgreSQL, а также научиться применять некоторые нетривиальные возможности СУБД в качестве сервера БД

Краткие теоретические сведения

Данную модель поддерживают большинство современных СУБД: Informix, Ingres, Sybase, Oracle, MS SQL Server. Основу данной модели составляет механизм хранимых процедур как средство программирования SQL-сервера, механизм триггеров как механизм отслеживания текущего состояния информационного хранилища и механизм ограничений на пользовательские типы данных, который иногда называется механизмом поддержки доменной структуры. Модель сервера баз данных:

В этой модели бизнес-логика разделена между клиентом и сервером. На сервере бизнес-логика реализована в виде хранимых процедур — специальных программных модулей, которые хранятся в БД и управляются непосредственно СУБД. Клиентское приложение обращается к серверу с командой запуска хранимой процедуры, а сервер выполняет эту процедуру и регистрирует все изменения в БД, которые в ней предусмотрены. Сервер возвращает клиенту данные, релевантные его запросу, которые требуются клиенту либо для вывода на экран, либо для выполнения части бизнес-логики, которая расположена на клиенте. Трафик обмена информацией между клиентом и сервером резко уменьшается. Централизованный контроль в модели сервера баз данных выполняется с использованием механизма триггеров. Триггеры также являются частью БД.

Механизм использования триггеров предполагает, что при срабатывании одного триггера могут возникнуть события, которые вызовут срабатывание других триггеров. Этот мощный инструмент требует тонкого и согласованного применения, чтобы не получился бесконечный цикл срабатывания триггеров.

В данной модели сервер является активным, потому что не только клиент, но и сам сервер, используя механизм триггеров, может быть инициатором обработки данных в БД. И хранимые процедуры, и триггеры хранятся в словаре БД, они могут быть использованы несколькими клиентами, что существенно уменьшает дублирование алгоритмов обработки данных в разных клиентских приложениях.

Недостатком данной модели является очень большая загрузка сервера. Действительно, сервер обслуживает множество клиентов и выполняет следующие функции:

- осуществляет мониторинг событий, связанных с описанными триггерами;
- обеспечивает автоматическое срабатывание триггеров при возникновении связанных с ними событий;
- обеспечивает исполнение внутренней программы каждого триггера;
- запускает хранимые процедуры по запросам пользователей;
- запускает хранимые процедуры из триггеров;
- возвращает требуемые данные клиенту;

- обеспечивает все функции СУБД: доступ к данным, контроль и поддержку целостности данных в БД, контроль доступа, обеспечение корректной параллельной работы всех пользователей с единой БД.

Если мы переложили на сервер большую часть бизнес-логики приложений, то требования к клиентам в этой модели резко уменьшаются. Иногда такую модель называют моделью с "тонким клиентом", в отличие от предыдущих моделей, где на клиента возлагались гораздо более серьезные задачи. Эти модели называются моделями с "толстым клиентом".

Практические задания

Выберите предметную область (можно из л/р №1) и опишите структуру БД, используя SQL-запросы. К обязательным требованиям относится использование:

1. объектно-реляционных связей;
2. ограничений в таблицах;
3. массивов;
4. последовательностей;
5. а также backup и restore БД для переноса с домашнего ПК.

Прием работы производится только, если она удовлетворяет всем требованиям.

Прием происходит при наличии оформленного отчета и работающей БД.

Контрольные вопросы:

1. Что такое PostgreSQL, какой язык использует в качестве языка БД, к какому классу ПО (открытое или закрытое) относится? Какая архитектура? Какие клиентские приложения входят в пакет?
2. Как организуется объектно-реляционные связи в СУБД postgreSQL и какие особенности организации могут приводить к визуальному нарушению ограничений установленных в таблицах?
3. Что такое ограничения полей, ограничения таблиц? Как они используются и для чего?
4. Как использовать поля-массивы: как обращаться к элементам массивов, как создавать массивы?
5. Автоматизация стандартных процедур.
6. Что такое последовательности? Как могут быть использованы?
7. Что такое триггеры? На каких языках могут быть реализованы?

Практическая работа №12 Компоненты SQL server

Цель занятия: познакомиться с возможностями языка SQL erver

Краткие теоретические сведения

При работе с информационными системами ранее установленными системными администраторами, достаточно часто можно столкнуться с ситуацией, что на сервере, где требуется исключительно управление базами данных - установлены все компоненты, которые поставляются в дистрибутиве SQL сервера. На резонный вопросе: "Зачем установлены все компоненты?", можно получить ответ "Я всегда ставлю все компоненты" или "Я не знаю зачем нужен каждый из компонентов, поэтому на всякий случай установил все".

Понятно, что такой подход в корне неверный, так как на сервере работают службы, которые никем и никогда не используются, и эти службы, в свою очередь впустую используют вычислительные ресурсы сервера, что может негативно отражаться на производительности самого сервера, так и хоста виртуализации (в случае если SQL сервер виртуализован).

Сисок основных компонентов поставляемых в дистрибутиве SQL сервер, их краткое описание и назначение:

Database Engine Services (Службы компонента Database Engine или Службы ядра СУБД) - это основная служба для хранения, обработки и защиты данных, репликации, полнотекстового поиска, средств управления реляционными и XML-данными, а также Data Quality Services SQL сервера (DQS). К службам Database Engine можно доустановить необязательные компоненты, если этого требует функционал SQL сервера:

- **Replication** (Репликация): Этот компонент представляет собой набор технологий копирования и распространения данных и объектов баз данных между базами данных, а также синхронизации баз данных для поддержания согласованности.
- **Full-Text Search** (Полнотекстовый и семантический поиск): Этот компонент позволяет выполнять полнотекстовые запросы по таблицах SQL сервера для произвольных символьных данных.
- **Data Quality Services** (Служба качества данных): этот компонент, который дает возможность обнаруживать несогласованные и неверные данные в источнике данных и предоставляет компьютеризированные и интерактивные методы очистки данных.

Практические задания

Данная лабораторная работа не является обязательной для выполнения, однако необходима для получения оценки выше «4» на экзамене.

1. Создайте отношение A. Для этого отношения определите два поля:

поле – номер (тип число) и поле – строка_значений (тип одномерная матрица). Таблица будет иметь следующую структур:

номер_матрицы	строка_значений
1	'{2,4}'
1	'{-3,1}'
2	'{0}'
2	'{5}'
3	'{-2,-4}'

Что соответствует матрицам $A1 = \begin{pmatrix} 2 & 4 \\ -3 & 1 \end{pmatrix}$, $A2 = \begin{pmatrix} 0 \\ 5 \end{pmatrix}$ и $A3 = \begin{pmatrix} -2 & -4 \end{pmatrix}$.

2. Создайте отношение B, содержащее четыре поля: номер_операции (целое), номер_первой матрицы (целое), номер_второй матрицы (целое), название_функции (текст).

3. Создайте третье отношение – C, которое имеет поля: номер_операции (целое) и поле результат (numeric).

4. Создайте функции сложения, вычитания, транспонирования, умножения матриц, которые работают с матрицами, определенными в отношении A, т.е. на вход получают номер матрицы (или матриц – для бинарных операций).

5. Создайте функцию вычисления определителя матрицы по ее номеру.

6. Создайте триггер, который:

- при добавлении новой строки в отношение B (с новым номером_операции) производит расчет для этой операции, вызвав соответствующую функцию для переданных матриц, и записывает полученный результат в отношение C в виде нового кортежа;
- при обновлении строки в отношении B производит пересчет, согласно новых переданных параметров, если пересчет произведен без ошибок, то обновляет соответствующий кортеж в отношении C;
- при выборе строки из отношения B производит пересчет, согласно существующих параметров.

Прием работы

Прием происходит при наличии оформленного отчета и работающей БД.

Контрольные вопросы:

1. Чем отличается использование атрибута %ROWTYPE от типа RECORD?
2. Что такое PL/pgSQL и из каких блоков состоит процедура на этом языке?
3. Что такое триггеры и триггерные функции?
4. Как можно вставить данные в переменную типа RECORD?
5. Какие циклы существуют в языке PL/pgSQL.

Практическая работа №13 Модели клиент сервер

Цель занятия: моделирование объектного подхода «Клиент-сервер» на реляционной БД.

Краткие теоретические сведения

Архитектура «Клиент-Сервер» (также используются термины «сеть Клиент-Сервер» или «модель Клиент-Сервер») предусматривает разделение процессов предоставление услуг и отправки запросов на них на разных компьютерах в сети, каждый из которых выполняют свои задачи независимо от других.

В архитектуре «Клиент-Сервер» несколько компьютеров-клиентов (удалённые системы) посылают запросы и получают услуги от централизованной служебной машины – сервера (server – англ. «официант, обслуга»), которая также может называться хост-системой (host system, от host – англ. «хозяин», обычно гостиницы).

Клиентская машина предоставляет пользователю т.н. «дружественный интерфейс» (user-friendly interface), чтобы облегчить его взаимодействие с сервером.

Типы клиент-серверной архитектуры

Архитектуру «клиент-сервер» принято разделять на три класса: одно-, двух- и трёхуровневую. Однако, нельзя сказать, что в вопросе о таком разделении в сообществе ИТ-специалистов существует полный консенсус. Многие называют одноуровневую архитектуру двухуровневой и наоборот, то же можно сказать о соотношении двух- и трёхуровневой архитектур. Постараемся внести ясность в этот вопрос.

Одноуровневая архитектура (1-Tier)

Одноуровневая архитектура «клиент-сервер» (1-Tier) – такая, где все прикладные программы рассредоточены по рабочим станциям, которые обращаются к общему серверу баз данных или к общему файловому серверу. Никаких прикладных программ сервер при этом не исполняет, только предоставляет данные.

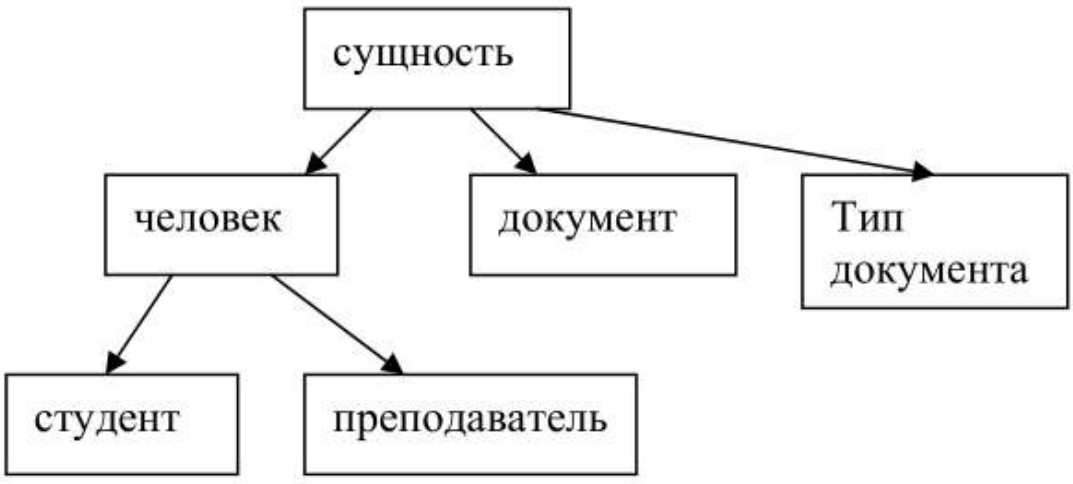
В целом, такая архитектура очень надёжна, однако, ей сложно управлять, поскольку в каждой рабочей станции данные будут присутствовать в разных вариантах. Поэтому возникает проблема их синхронизации на отдельных машинах. В общем, как можно видеть из рисунка, в этой архитектуре просматривается ещё один уровень – базы данных, что даёт повод во многих случаях называть её двухуровневой.

Двухуровневая архитектура (2-Tier)

К двухуровневой архитектуре «клиент-сервер» следует относить такую, в которой прикладные программы сосредоточены на сервере приложений (Application Server), например, сервере 1С или сервере CRM, а в рабочих станциях находятся программы-клиенты, которые предоставляют для пользователей интерфейс для работы с приложениями на общем сервере.

Практические задания

1. Создать несколько отношений, связанных в виде иерархии, как это показано на рисунке:



2. Самостоятельно определить атрибуты этих отношений.

3. Иерархию реализовывать с использованием наследования.

4. Создать представление, которое выбирает все атрибуты объекта и его наследников в один кортеж. В случаи, если для какого-то из атрибутов имеется несколько значений необходимо формировать поле в следующем виде:

{‘<1ое значение атрибута>’, ‘<2ое значение атрибута>’, …}, где <i-ое значение атрибута> - значение атрибута в i-ом кортеже для объекта. Для этого написать собственную агрегатную функцию(ии), работающую с типами integer, text, timestamp.

5. Определить универсальные функции для удаления, добавления, обновления любого объекта, которым передается имя отношения, фильтр (если нужно), массив имен полей (если нужно), массив новых значений полей (если нужно).

6. На основании функций из п.5 определить для каждого объекта БД (кроме «сущность») функции: добавить, изменить, удалить. В функциях должен быть реализован контроль за уникальностью объекта.

7. Запретить добавление данных в отношения с использованием SQL запросов (т.е. не через интерфейсные функции из пункта 5).Для этого определить необходимые триггеры.

8. Написать функции (PL/PGSQL) вывода существующих документов человека (например, паспорт, з/к – если студент, № пропуска – если преподаватель). Функция использует представление созданное в пункте 4. Не использовать внешние ключи (реляционные связи) для связывания отношений находящихся в одной ветке иерархии, но использовать их (если необходимо) для связи объектов на одном уровне иерархии. Значение потенциального ключа в базовых таблицах не должно повторяться даже при выполнении запроса без параметра ONLY.

9. Структура БД, ограничения, правила наследования, процедуры, представления, а также данные должны быть представлены в виде SQL-скрипта.

Практическая работа №14 Системные базы данных

Цель занятия: создать ситемную базаму данных, сформировать запросы к базе данных.

Краткие теоретические сведения

База данных в SQL Server представляет собой логический объект, в котором размещаются таблицы и индексы. Физически база данных содержится в одном или нескольких файлах операционной системы.

Различают таблицы двух типов: постоянные и временные. Постоянные таблицы существуют до тех пор, пока их не удалят. Временные таблицы подразделяют на локальные и глобальные. Первые (локальные временные таблицы) существуют в текущем сеансе и затем уничтожаются. Вторые (глобальные временные таблицы) существуют до завершения всех использующих их сеансов.

Для хранения баз данных используются три типа файлов: основной, вспомогательные и журналов транзакций (рис. 6.4).

Основной (Primary) файл создается один и содержит информацию, требуемую для инициализации;

вспомогательные (Secondary) файлы содержат данные, не умещающиеся в основном файле; использование их не обязательно, но позволяет разместить БД на нескольких носителях;

файлы журналов транзакций хранят информацию для восстановления БД. Кроме того, могут создаваться дополнительные группы файлов для размещения пользовательских данных.

Журнал транзакций представляет собой рабочую область, в которую SQL Server записывает информацию до и после выполнения транзакций. Эта информация может использоваться для отмены выполненной транзакции или для восстановления БД. Журнал транзакций размещается в отдельном файле, создаваемом автоматически при создании базы данных.

При добавлении данных файлы базы данных и журнала транзакций расширяются автоматически.

Для хранения данных используются таблицы, размещаемые в базах данных. В Microsoft SQL Server базы данных делят на два типа — системные и пользовательские. В системных базах данных размещаются метаданные, используемые для управления системой. При инсталляции Microsoft SQL Server создаются следующие системные базы данных: ***master, model, tempdb и msdb***.

Системная база данных ***master*** обеспечивает управление пользовательскими базами данных и работу MS SQL Server. Она содержит системный каталог, или словарь данных, насчитывающий 13 системных таблиц. Названия таблиц и характеристика содержащихся в них данных приведены на рис. 6.5. Ввиду важности этой базы данных рекомендуется иметь ее архив, отражающий самое последнее состояние.

В состав системного каталога входят следующие системные таблицы: *syscharsets, sysconfigures, syscurconfig, sysdatabases, sysdevices, syslanguages, syslocks, sysl ogin, sysmessages, sysprocesses, sysremotelogins, sysservers, sysusages*.

Хранящиеся в таблицах данные:

учетные записи пользователей;

сведения о текущих процессах;

сообщения о системных ошибках;

сведения о базах данных на сервере;

выделенные размеры баз данных;

сведения об активных блокировках;

сведения о доступных устройствах баз данных и резервных;

процедуры системного администрирования.

Практические задания

Предикат IS NULL. Для выяснения смысла значения NULL рассмотрим пример. Пусть в городе N ведётся база данных, в которой хранятся данные обо всех жителях, включая детей. Очевидно, что в графу «профессия» записи о ребёнке поместить нечего, так как у ребёнка ещё нет профессии. Графа профессия может оказаться пустой и в том случае, когда в момент занесения данных профессия жителя не была известна. Предполагается, что графа будет заполнена позже. Для неизвестного значения в SQL применяется специальное обозначение NULL. Значение NULL имеют по умолчанию все поля, в которые ничего не заносилось.

NULL применяется в полях всех типов и само не имеет типа. Значение NULL можно использовать только в специальном предикате IS NULL, имеющем следующий синтаксис:

<выражение> IS [NOT]NULL

Предикат IS NULL принимает значение «истина» только, если выражение равно NULL.

Для работы с NULL-значениями полей создайте в базе данных таблицу NullPusto, состоящую из двух текстовых полей длиной по 30 символов. Назовите поля «ФИО» и «адр». Введите в таблицу данные из табл. 3.

Создайте и выполните следующие запросы к таблице NullPusto:

a) выбрать все записи с NULL;
b) выбрать все записи с “”;
c) выбрать все записи, в которых есть адреса;
d) выбрать все записи, в которых нет адресов;
e) подсчитать количество записей, содержащих NULL;
f) подсчитать количество записей, содержащих NULL и “”.

Сохраните запросы и покажите их преподавателю.

Таблица 3

Поле		Значение в поле «адр»
ФИО	адр	
А	К	“К”
Б		“” (две двойные кавычки)
В		NULL
Г	М	“М”
Д		NULL
Е		“” (две двойные кавычки)
Ж		NULL

Подзапросы. С помощью SQL можно вкладывать один запросы внутрь другого. Внутренний запрос называют подзапросом. Обычно, внутренний запрос генерирует значение, которое проверяется в предикате внешнего запроса, определяющего верно оно или нет. Например, в следующем запросе выбираются из таблицы «Товары» те товары, цена которых меньше средней цены всех товаров таблицы:

```
SELECT  *
FROM Товары
WHERE  Цена<(SELECT  AVG(Цена) FROM Товары);
```

Самостоятельно с помощью подзапроса выберите из таблицы «Заказано» заказы на товары с маркой «Pavlova». Марки товаров хранятся в таблице «Товары».

Предикат EXISTS имеет синтаксис

EXISTS подзапрос

и принимает значение ИСТИНА (TRUE), если подзапрос содержит хотя бы одну строку.

В следующем запросе выбираются фамилии всех сотрудников, оформлявших заказы для клиента ANTON, при условии, что хотя бы один заказ для клиента ANTON был размещён в мае любого года.

```
SELECT DISTINCT  b.Фамилия
FROM Заказы a, Сотрудники b
WHERE EXISTS (SELECT * FROM Заказы WHERE КодКлиента='ANTON' AND DatePart('
m',ДатаРазмещения)=5)
  AND a.КодСотрудника = b.КодСотрудника AND a.КодКлиента='ANTON';
```

Самостоятельно, используя таблицы «Сотрудники», «Клиенты» и «Заказы», создайте и выполните запрос на выборку всех клиентов из Рио-Де-Жанейро, если был сделан хотя бы один заказ из Рио-Де-Жанейро, оформленый сотрудником Кротовым.

Предикаты количественного сравнения ANY, SOME и ALL имеют синтаксис

оператор сравнения {ANY | SOME | ALL} подзапрос.

ANY и SOME – синонимы.

Пример использования предиката ANY:

```
SELECT КодЗаказа
FROM Заказы
WHERE СтоимостьДоставки
< ANY(SELECT СтоимостьДоставки FROM Заказы WHERE ГородПолучателя   ='Ванкув
ер');
```

Для исследования особенностей предиката ANY проделайте следующее упражнение:

a) выберите из таблицы «Товары» цены товаров от поставщика с кодом 2; запишите эти цены;

b) используя ANY, выберите все товары, цены которых больше цен поставщика 2; сравните выбранные цены с записанными;

c) повторите предыдущий пункт, используя вместо ANY предикат ALL; сравните результаты.

Сохраните все выполненные запросы и покажите их преподавателю

Практическая работа №15 Оптимизация запросов, управляемых правилами

Цель занятия: научиться объединять в одной выводимой таблице строки, полученные разными запросами и создавать новую таблицу базы данных из существующих таблиц.

Краткие теоретические сведения

В оптимизацию реляционных запросов входят два различных аспекта. Во-первых, это внутренняя задача СУБД, которая заключается в определении наиболее оптимального (эффективного) способа выполнения реляционных запросов. Во-вторых, это задача программиста (или квалифицированного пользователя): она заключается в написании таких реляционных запросов, для которых СУБД могла бы использовать более эффективные способы нахождения данных. Сначала рассмотрим первый аспект.

Каждая команда языка манипулирования данными может быть выполнена разными способами. Определение наиболее оптимального плана выполнения запроса называется **оптимизацией**. Выбором этого плана занимается оптимизатор – специальная компонента СУБД.

Выполнение запроса состоит из последовательности шагов, каждый из которых либо физически извлекает данные из памяти, либо делает подготовительную работу. Последовательность шагов, которую строит оптимизатор, называется **планом выполнения**.

Обработка запроса, поступившего в систему и представленного на некотором языке запросов, состоит из этапов или фаз, представленных на рис.1.

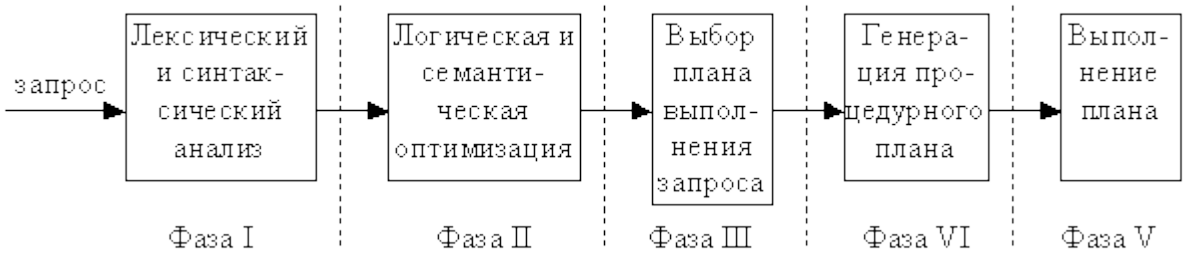


Рис.1. Последовательность выполнения запросов в реляционных СУБД

На первой фазе запрос, представленный на языке запросов, подвергается лексическому и синтаксическому анализу. При этом вырабатывается его внутреннее представление, отражающее структуру запроса и содержащее информацию, которая характеризует объекты базы данных, упомянутые в запросе (отношения, поля и константы). Информация о хранимых в базе данных объектах выбирается из каталогов базы данных (словаря-справочника данных). Внутреннее представление запроса используется и преобразуется на следующих стадиях обработки запроса.

На второй фазе запрос в своем внутреннем представлении подвергается логической оптимизации. При этом могут применяться различные преобразования, "улучшающие" начальное представление запроса. Среди этих преобразований могут быть эквивалентные преобразования, после проведения которых получается внутреннее представление, семантически эквивалентное начальному (например, приведение запроса к некоторой канонической форме). Преобразования могут быть и семантическими, когда получаемое представление не является семантически эквивалентным начальному, но гарантируется, что результат выполнения преобразованного запроса совпадает с результатом запроса в начальной форме при соблюдении ограничений целостности, существующих в базе данных. В любом случае после выполнения второй фазы обработки запроса его

внутреннее представление остается непроцедурным, хотя и является в некотором смысле более эффективным, чем начальное.

Третий этап обработки запроса состоит в выборе на основе информации, которой располагает оптимизатор, набора альтернативных процедурных планов выполнения данного запроса в соответствии с его внутреннем представлением, полученным на второй фазе. Основой является информация о существующих путях доступа к данным. Единственный путь доступа, который возможен в любом случае, – это последовательное чтение (FULL). Возможность использования других путей доступа зависит от способов размещения данных в памяти (например, кластеризация данных), наличия индексов и формулировки самого запроса.

На этом же этапе для каждого плана оценивается предполагаемая стоимость выполнения запроса по этому плану. При оценках используется либо доступная оптимизатору статистическая информация о состоянии базы данных, либо информация о механизмах реализации различных путей доступа. Из полученных альтернативных планов выбирается наиболее оптимальный с точки зрения некоторого (заранее выбранного или заданного) критерия. Внутреннее представление этого плана теперь соответствует обрабатываемому запросу.

На четвертом этапе по внутреннему представлению наиболее оптимального плана выполнения запроса формируется процедурное представление плана. Выполняемое представление плана может быть программой в машинных кодах, если, как в случае System R, система ориентирована на компиляцию запросов в машинные коды, или быть машинно-независимым, но более удобным для интерпретации, если, как в случае INGRES, система ориентирована на интерпретацию запросов. В нашем случае это непринципиально, поскольку четвертая фаза обработки запроса уже не связана с оптимизацией.

Наконец, на последнем, пятом этапе обработки запроса происходит его реальное выполнение в соответствии с выполняемым планом запроса. Это либо выполнение соответствующей подпрограммы, либо вызов интерпретатора с передачей ему для интерпретации выполняемого плана.

Существуют два принципиально разных подхода к оптимизации запросов. Если оптимизатор использует информацию о механизмах реализации путей доступа, то метод оптимизации основан на синтаксисе (на правилах). Если же основой является статистическая информация о распределении данных, то это метод оптимизации, основанный на стоимости (на издержках). Рассмотрим их подробнее.

Практические задания

Предложение UNION применяется для объединения результатов нескольких запросов в одной выводимой таблице. Количество столбцов во всех запросах должно быть одинаковым и типы соответствующих столбцов должны быть сравнимыми. В следующем примере выводятся адреса и города клиентов и заказов. Параметр ALL разрешает выводить дубликаты строк.

```
SELECT ALL Адрес,Город,'Заказы ' AS Источник
FROM Клиенты
UNION
SELECT ALL АдресПолучателя AS Адрес,ГородПолучателя AS Город,'Клиенты
' AS Источник
FROM Заказы;
```

Выполните этот запрос.

Самостоятельно выберите из таблиц «Клиенты» и «Сотрудники» следующие данные:

- фамилию и имя;
- должность;
- город.

В дополнительном столбце укажите , из какой таблицы выбрана запись.

Создание таблицы из существующих таблиц с помощью SELECT … INTO. Во многих СУБД конструкция SELECT … INTO <имя таблицы> используется для создания новой таблицы и вывода в неё результатов запроса. Например, таблица «Страны» с названиями всех стран, в которые направляются заказы, создаётся в результате выполнения следующего запроса:

```
SELECT DISTINCT СтранаПолучателя
INTO Страны
FROM Заказы;
```

Самостоятельно с помощью SELECT … INTO создайте таблицу «Клиенты2», содержащую данные из таблицы «Клиенты» обо всех клиентах, живущих в Лондоне.

Сохраните выполненные запросы и покажите их преподавателю

Практическая работа №16 Объектноориентированные модели данных

Цель занятия: освоить способы редактирования, вставки и удаления записей в рамках объектно-ориентированной модели.

Краткие теоретические сведения

Стройность и мощность реляционных моделей сделали их доминирующими в среде баз данных. Но постоянное усложнение данных позволило выявить ряд неудобств, возникающих при работе с реляционными базами:

Реляционные системы ограничены в структурах представления данных, так как все данные хранятся в них в виде отношений, состоящих из простых атрибутов. Классическая реляционная модель предполагает неделимость данных, хранящихся в полях таблицы, то есть информация в таблице должна быть представлена в первой нормальной форме. Однако на практике иногда возникают ситуации, когда такое ограничение снижает эффективность работы с базой.

Данные в реляционной системе пассивны, и для описания их поведения требуется создавать прикладные программы.

Возможности реляционных баз данных недостаточны в тех случаях, когда объекты данных сложны, например: географические информационные системы, мультимедийные базы, базы с проектной документацией и др.

Все эти требования можно реализовать с помощью реляционных методов, но в результате получается не очень естественное представление требований пользователя.

Постреляционная модель является расширением реляционной модели. Она снимает ограничение неделимости данных, допуская многозначные поля, значения которых состоят из подзначений, и набор значений воспринимается как самостоятельная таблица, встроенная в главную таблицу.

Практические задания

Вставка в таблицу одной или нескольких строк с помощью оператора INSERT. Синтаксис оператора INSERT:

INSERT INTO <имя таблицы>
[(<имя столбца>)]
{VALUES (<значение> .,..)}
|<выражение запроса>
|{DEFAULT VALUES};

Пример. Добавим в созданную в лаб. работе №15 таблицу «книга» книгу М. Горького «Детство». Так как в таблице «писатель» Горькому не присвоен код, то в добавляемой строке будут заполняться только столбцы «КодКн» и «Наим». Описанная строка добавляется с помощью оператора

INSERT INTO книга
(КодКн,Наим)
VALUES (10,'Детство');

Столбец «КодКн» не является счётчиком, поэтому он указан в списках столбцов и добавляемых значений. Счётчик в операторе INSERT указывать не надо.

Самостоятельно добавьте в таблицу «писатель» Толстого А.Н. и в таблицу «книга» - роман «Сёстры».

Изменение (редактирование) данных в таблице с помощью оператора UPDATE. Синтаксис оператора UPDATE:

UPDATE <имя таблицы>
SET {<имя столбца>={<выражение для вычисления значения>
|NULL |DEFAULT}}

[WHERE <предикат>]
Пример. Укажем в таблице «писатель» код Горького:

UPDATE писатель
SET КодП=10
WHERE ФИО='Горький';

Самостоятельно с помощью оператора UPDATE занесите в таблицу «книга» все недостающие значения полей.

Удаление строк таблицы с помощью оператора DELETE. Синтаксис оператора DELETE:

DELETE FROM <имя таблицы>
[WHERE <предикат>]

Пример. Удалим из таблицы «книга» книгу «На дне»

DELETE FROM **книга** WHERE **КодКн=7**

Самостоятельно с помощью оператора DELETE удалите из таблицы «писатель» Тургенева.

Сохраните выполненные запросы и покажите их преподавателю

Практическая работа №17 Разработка требований к корпоративной сети

Цель занятия: Изучение принципывотразработка требований к корпоративной сети

Краткие теоретические сведения

Рост объема передаваемых по сети данных связан с появлением приложений, работающих с данными мультимедиа, а также с развитием технологий обработки и представления данных. В качестве примеров таких технологий можно привести приложения клиент/сервер, позволяющие производить обработку данных большим числом конечных пользователей. Поэтому сеть должна обеспечивать адекватную пропускную способность на уровне доступа (не менее 100 Мбит/с), а также удовлетворять повышенным требованиям надежности и высокой интенсивности трафика на уровне магистралей (не менее 1 Гбит/с).

Практические задания

Задание.

Характеристики разрабатываемой сети следующие:

- количество рабочих предприятия - 500, из них 220 имеют рабочие места, оборудованные компьютерами, сетевые соединения работают по технологии Ethernet;
- в сети имеется несколько больших корпоративных серверов (сервер БД, почтовый, proxy, обработка сообщений, WWW), а также высокопроизводительные сервера рабочих групп, интенсивно использующих полосу пропускания, и мультимедиа-серверы для осуществления мультикастинговой трансляции по протоколу IP;
- в сети может наблюдаться высокая загруженность активного сетевого оборудования, сетевых соединений, начиная от серверных сегментов и заканчивая пользовательскими подсетями;
- предприятие имеет один центральный и 2 региональных офиса, связь между которыми должна осуществляться по защищенному каналу и обеспечивать возможность передачи большого количества данных;
- центральный офис имеет 8 основных отделов: отдел администрации (10 рабочих мест), отдел бухгалтерии (20 рабочих мест), ИТ-отдел (10 рабочих мест), отдел сбыта (20 рабочих мест), отдел снабжения (20 рабочих мест), отдел маркетинга (30 рабочих мест), склады (10 рабочих мест), производственно-технический отдел (100 рабочих мест);
- в сети должна присутствовать возможность подключения мобильных пользователей посредством беспроводного доступа (например, кладовщиков с КПК-модулями либо клиентов имеющих ноутбуки).

Кроме того, для удовлетворения возрастающих требований приложений к пропускной способности сетевых систем необходим некоторый запас быстродействия, который обеспечил бы нормальное функционирование системы в течение нескольких лет. Также необходимо обеспечить возможность эффективного роста сетевой системы при минимальном вложении средств в оборудование и каналы связи.

Практическая работа №18 Cache и WWWтехнологии

Цель занятия: Овладение приемами работы с базой данных MySQL и отображение

данных таблиц на веб-странице.

Краткие теоретические сведения

MySQL – это одна из самых популярных и самых распространенных СУБД (система управления базами данных) в интернете. Она не предназначена для работы с большими объемами информации, но ее применение идеально для интернет сайтов, как небольших, так и достаточно крупных.

MySQL отличатся хорошей скоростью работы, надежностью, гибкостью. Работа с ней, как правило, не вызывает больших трудностей. Поддержка сервера MySQL автоматически включается в поставку PHP. Приложение на PHP, использующее для хранения информации базу данных (в частности MySql) всегда работает быстрее приложения, построенного на файлах.

```
<?php for ($x=0; $x<10; $x++) echo $x; ?>
foreach (массив as $ключ=>$значение) команды;
<?php
$names["Иванов"] = "Андрей"; $names["Петров"] =
"Борис"; $names["Волков"] = "Сергей"; $names["Макаров"]
= "Федор"; foreach ($names as $key => $value) { echo
"<b>$value $key</b><br>";
}
?>
for (инициализирующие_команды; условие_цикла;
команды_после_итерации) { тело_цикла; }
```

Дело в том, что базы данных написаны на языке C++, и написать на PHP программу, которая работала бы с жёстким диском эффективнее базы данных - задача неразрешимая по определению, поскольку программы на PHP в принципе работают медленнее, чем программы на C++, так как PHP - интерпретатор, а C++ - компилятор. Таким образом, основное достоинство базы данных заключается в том, что она берёт на себя всю работу с жёстким диском и делает это очень эффективно

Практические задания

1. С помощью phpMyAdmin создать новую базу данных и таблицу.
2. Занести несколько записей в таблицу
3. С помощью PHP отобразить все записи таблицы.
4. Осуществить выбрку данных по какому-либо критерию(фильтру)
5. Реализовать параметрический запрос(значение параметра
определяется выпадающим списком <select>)
Все ответы от MySQL отображать на странице в виде таблиц с
заголовками отобранных полей(использовать тэг <table>)
6. Оформить и сдать отчёт.

Лабораторная работа№1 Конфигурирование сети

Цель занятия: Научиться моделировать локальные сети, используя программу для моделирования сетей

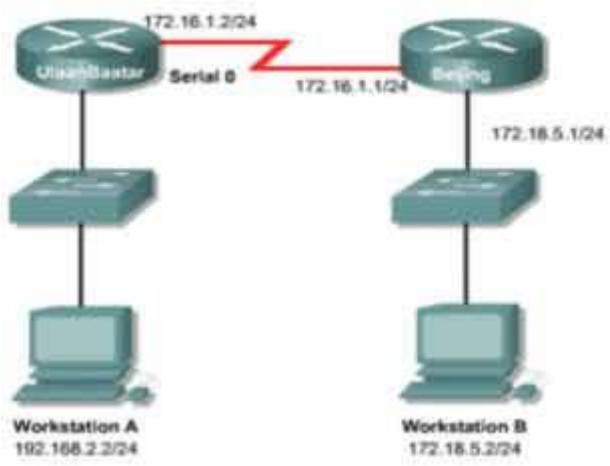
Краткие теоретические сведения

Рост объема передаваемых по сети данных связан с появлением приложений, работающих с данными мультимедиа, а также с развитием технологий обработки и представления данных. В качестве примеров таких технологий можно привести приложения клиент/сервер, позволяющие производить обработку данных большим числом конечных пользователей. Поэтому сеть должна обеспечивать адекватную пропускную способность на уровне доступа (не менее 100 Мбит/с), а также удовлетворять повышенным требованиям надежности и высокой интенсивности трафика на уровне магистралей (не менее 1 Гбит/с).

Практические задания

Шаг 1

a) Создать сеть, показанную на рисунке



b) Убедитесь, что не находитесь в простом режиме (simplemode) – пункт меню Options

Шаг 2

a) Щелкните на BegingRouter
b) Установите привилегированный EXEC режим.
c) Войдитев global configuration режим.
d) Войдите в режим конфигурации маршрутизатора для настройки RIP протокола.
e) Добавьте адреса 172.16.0.0 и 172.18.0.0.
f) Вернитесь в привилегированный EXEC режим (команда exit).
g) Нажмите showrun для просмотра текущих настроек. При правильной настройке должны быть видны сети 172.16.0.0 и 172.18.0.0.

Практическая работа №19 Формирование аппаратных требований и схемы банка данных

Цель занятия: изучить основные принципы формирования аппаратных требований и схемы банка данных

Краткие теоретические сведения

Составной частью АСВУР является хранилище информации: банки данных и знаний.

Банк данных (БД) – автоматизированная информационная система централизованного хранения данных и коллективного пользования ими. В состав БД входят: одна или несколько баз данных, справочник баз данных, система управления базами данных и библиотека запросов и прикладных программ.

База данных (БзД) представляет совокупность массивов информации, которые организованы по определенным правилам, предусматривающим общие принципы описания, хранения и использования. По назначению выделяют централизованную, распределительную и персональную базы данных. Содержимое централизованной БзД размещено в виде единого информационного массива на одном или нескольких носителях в одной ЭВМ. Распределительная (децентрализованная) БзД – совокупность баз данных, распределенных по взаимно-связанным узлам вычислительной системы и доступных для совместного использования в различных приложениях. С персональной (личной) БзД взаимодействует один пользователь.

Банк знаний –автоматизированная система искусственного интеллекта, ориентированная на решение сложных задач, трудно поддающихся однозначному и формализованному описанию и обычно решаемых на основе опыта и неформальной логики (эвристических методов), как правило, с привлечением экспертов. Банк знаний включает базу знаний с набором правил и механизмов вывода, позволяющих на основе правил и представляемых исследователем фактов распознать ситуацию, поставить диагноз, сформулировать решение или дать рекомендацию для выбора действия, Обычно банки знаний и данных представляют собой одну систему – банк данных и знаний (БДиЗ).

База данных и знаний включает различные массивы информации, которые исходя из стадии обработки информации, делятся на входные, промежуточные и выходные.

Во входные массивы переносится информация из первичных документов, содержащих сведения о состоянии управляемого объекта и потребностях в ресурсах. Эти массивы направляются на обработку или хранение. Промежуточные массивы информации размещаются на носителях прямого доступа (магнитных лентах, барабанах, дисках). Они формируются по соответствующим алгоритмам на базе данных входных массивов. К промежуточным массивам относятся: массивы изменений, рабочие массивы, нормативно- сметные, плановые, справочные, отчетно-архивные, массивы знаний и др. Выходные массивы формируются в процессе решения задач. Они содержат информацию, необходимую для выдачи табуляграмм. Массивы изменений и рабочие массивы являются переменными, а остальные – постоянными, содержащими условно-постоянную информацию. В массивах знаний размещаются данные научного и экспертного анализа.

Обслуживание банка данныхзаключается в накоплении, обновлении и корректировке хранимой информации, а также в выдаче ответов при решении задач как на регламентированные, так и на произвольные запросы.

БДиЗ создается на основе соблюдения ряда принципов.

1. Способность его системы к развитию, что достигается абстрагированием от

пользователей и программ решения функциональных задач. Исходя из этого принципа, БДиЗ должен обеспечивать постановку и реализацию новых задач. В этом случае может измениться (увеличится) только база данных, но ни в коем случае не должны меняться программы уже решаемых задач. Из общего принципа следуют частные: допустимость взаимодействия с различными пользователями и максимально возможная интеграция данных. Этот частный принцип особенно актуален, когда обмен информацией на уровне БДиЗ должен вестись автоматизировано между различными подсистемами управления.

2. Уменьшение избыточности (дублирования) хранимых данных.

3. Разнообразие хранимых данных, их структур и взаимосвязей.

4. Достоверность данных и запрет некомпетентного доступа к ним.

Работа БДиЗ обеспечивается единой организацией хранения данных и специальной системой управления. Эта система состоит из двух частей: языковых средств и пакета прикладных программ, который реализует формирование, а также ведение базы данных.

При увязке компьютерных пунктов в единую интегрированную информационно-вычислительную систему управления проектами (целевыми программами) создаются интегрированные распределенные банки данных и зданий. В основу построения этих банков положены три следующих принципа.

1. Модульно-блочный принцип, предусматривающий накопление всей информации в основных базах с распределением ее по рабочим базам не только для каждой подсистемы управления, но и для каждой задачи, проектной процедуры и функции управления, реализуемых математическими и программными модулями. Реализация этого принципа предусматривает комплексную интеграцию систем научных исследований, проектирования и реализации проектов. При этом интеграция проводится на основе создания единой системы сбора, поиска и передачи информации в пределах программы (проекта). Интеграция информации осуществляется на основе единой ее классификации и кодирования, единой системы документации проектирования объектов и управления их строительством.

2. Принцип универсальности, позволяющий на практике обеспечить необходимыми данными решение проектных и управленческих задач различного класса.

3. Принцип совместимости, предусматривающий совмещение информации не только внутри данной системы, но и при ее взаимодействии с другими системами через центральный вычислительный центр.

Распределенный интегрированный банк данных и знаний представляет собой систему иерархически организованных локальных БДиЗ, систем и подсистем, входящих в организационную структуру интегрированной системы. Каждый локальный БДиЗ – это совокупность взаимосвязанных массивов информации, предназначенных для решения отдельных проектных и управленческих задач и их комплексов, а также языковых и программных средств, методов доступа и управления массивами, технических средств, реализующих функции хранения, обновления, поиска и выдачи информации пользователям.

Использование БДиЗ характерно только для мощных ЭВМ, так как их разработка связана с большими затратами.

В интегрированных системах управления, организованных на базе персональных компьютеров, широко используются децентрализованные базы нормативов и смет объектов строительства. Это позволяет организовать хранение локальных баз непосредственно в местах переработки информации. Базы данных и зданий для компьютерных пунктов, оснащенных ПК, создаются открытыми, состоящими из ряда файлов внешней памяти. Эти файлы содержат данные для решения задач в определенных подсистемах управления с использованием прикладных программ.

Практические задания

Задание.Сформируйте аппаратных требованя и постройте схему банка данных к БД разработанноу в ПЗ №1

Лабораторная работа№2 Установка и настройка сервера MySQL

Цель занятия: ознакомление с MySQLServer

Краткие теоретические сведения

База данных сайта MySQL – это система, предназначенная для хранения и обработки информации. Комплекс таблиц, взаимосвязанных между собой, для доступа к которым применяется система управления базами данных (СУБД) MySQL. По сути, MySQL – это специальная программа с открытым кодом, которая используется на сервере SQL. Данная программа не способна обрабатывать большое количество информации, однако она идеальна для небольших и крупных веб-ресурсов.

Практические задания

Провести установку MySQLServer.

Лабораторная работа№3 Конфигурирование SQL Server Agent и SQL Server Enterprise Manager

Цель занятия:

Краткие теоретические сведения

Практические задания

Запустите программу Enterprise Manager (Пуск/Программы/Microsoft SQL Server/ Enterprise Manager).

В дереве консоли раскройте ветви Microsoft SQL Server, а затем SQL Server Group. Обратите внимание, что заданный по умолчанию экземпляр SQL Server регистрируется автоматически с именем вашего компьютера.

Как узнать, запущен ли SQL Server?

Если сервер не зарегистрирован, выполните его регистрацию. Для этого

щелкните правой кнопкой по SQL Server Group, а затем выполните командуNewSQLServerRegistration. Появляется мастер регистрации. Зарегистрируйте ваш сервер заново, используя следующую информацию:

Доступные серверы Ваш сервер

Режим аутентификации WindowsAuthentication

Правой кнопкой мыши щелкните по имени вашего сервера и затем выполните команду **Edit SQL Server Registration Properties**(редактирование свойств регистрации SQL Server). Какие свойства заданы по умолчанию? Какой тип аутентификации используется при соединении с SQL Server? Скопируйте данное окно в ваш отчет.

Щелкните по кнопке ОК для закрытия окна свойств.

Конфигурирование службы SQL Server Agent для автоматического запуска

На панели задач Windows щелкните дважды по значку SQL Server ServiceManager.

В списке Services выберите SQL Server Agent.

Установите флажок AutostartservicewhenOSstarts(Автоматический запуск при запуске операционной системы).

Щелкните по кнопке Start/Continue. Через несколько секунд отметьте появление зеленой стрелки, которая показывает, что службаSQLServerAgentзапущена. Закройте окноSQLServerServiceManager.

Ошибки инсталляции можно выявить с помощью программы EventViewer(журнал приложений Windows) и с помощью просмотра журнала SQL Server.

Просмотр журнала SQL Server

В программе Enterprise Manager раскройте ваш сервер, после чего разверните папку Managent/

Щелкните по рубрике SQL Server Logs.

Откройте двойным щелчком текущий (Current) журнал. Просмотрите его содержимое. Что вызвало появление записей в этом журнале?

Лабораторная работа№4 Управление файлами базы данных

Цель занятия: изучить систему основных компонентов Microsoft SQL Server, понять процесс создания файла данных, освоить управление базами данных при помощи команд языка T-SQL.

Краткие теоретические сведения

Основные компоненты Microsoft SQL Server 2008

Все компоненты Microsoft SQL Server 2008 запускаются из меню «Пуск \ Программы \ Microsoft SQL Server 2008. В Microsoft SQL Server 2008 входят следующие компоненты:

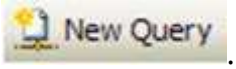
1) Deployment Wizard – мастер по выводу информации хранимой на сервере;
2) SQL Server Installation Center – центр установки SQL Server 2008;
3) Reporting Services Configuration Manager – менеджер службы настройки отчётов;
4) SQL Server Configuration Manager – менеджер настройки сервера;
5) SQL Server Error and Usage Reporting – служба протоколирования работы сервера и служба отчётов об ошибках;
6) Microsoft Samples Overview – ссылка на сайт корпорации Microsoft, где можно просмотреть примеры работы с сервером;
7) SQL Server Books Online – полная справочная система по Microsoft SQL Server 2008. Она содержит справки, как по программированию, так и по администрированию сервера;
8) SQL Server Tutorials – учебники по работе с сервером;
9) Data Profile Viewer – просмотр профилей по работе с данными;
10) Execute Package Utility – инструменты по сжатию данных;
11) Database Engine Tuning Advisor – мастер настройки ядра базы данных;
12) SQL Server Profiler – настройка профилей по работе с данными;
13) Import and Export Data – импорт и экспорт данных;
14) SQL Server Business Intelligence Development Studio – интегрированная среда разработки Business Intelligence Development Studio;
15) SQL Server Management Studio – графическая оболочка для управления сервером и разработки баз данных.

Практические задания

Создание файла данных

Новую БД можно создать, используя стандартные команды языка T-SQL. Для создания новой БД необходимо сделать активную БД «Master». Это можно сделать либо выбором ее из выпадающего списка БД на панели инструментов, либо набором команды USE Master на вкладке нового запроса.

Замечание: все команды языка T-SQL набираются на вкладке нового запроса (SQLQuery). Для того чтобы создать новый запрос на панели инструментов необходимо нажать кнопку



.

Для выполнения команд языка T-SQL на панели инструментов необходимо нажать кнопку

или на вкладке нового запроса набрать команду GO.

Замечание: В Microsoft SQL Server БД состоит из двух частей:

- файл данных – файл, имеющий расширение mdf и где находятся все таблицы и запросы;

- файл журнала транзакций – файл, имеющий расширение ldf, содержит журнал, где фиксируются все действия с БД. Данный файл предназначен для восстановления БД в случае её выхода из строя.

Для создания нового файла данных используется команда CREATE DATABASE, которая имеет следующий синтаксис:

CREATE DATABASE <Имя БД>
(Name=<Логическое имя>, FileName=<Имя файла>
[Size=<Нач.размер>,][Maxsize=<Макс.размер>,][FileGrowth=<Шаг>])
[LOG ON
(Name=<Логическое имя>, FileName=<Имя файла>
[Size=<Нач.размер>,][Maxsize=<Макс.размер >,][FileGrowth=<Шаг>])

Здесь:

- Имя БД – имя создаваемой БД,

- Логическое имя – определяет логическое имя файла данных БД, по которому происходит обращение к файлу данных,

- Имя файла – определяет полный путь к файлу данных,

- Нач.размер – начальный размер файла данных в Мб,

- Макс.размер – максимальный размер файла данных в Мб,

- Шаг – шаг увеличения файла данных, либо в Мб либо в %.

Параметры в разделе LOG ON аналогичны параметрам в разделе CREATE DATABASE. Однако они определяют параметры журнала транзакций.

Управление базами данных при помощи команд языка T-SQL

В языке запросов T-SQL с БД возможны следующие действия:

1) отображение сведений о БД: EXEC sp_helpdb <Имя БД>;
2) изменение параметров БД: EXEC sp_dboption <Имя БД>, <Параметр>, <Значение>;
3) добавления новых файлов, удаление файлов и переименования файлов, входящих в БД:

ALTER DATABASE <Имя БД>
ADD FILE (<Параметры>)|
REMOVE FILE <Логическое имя файла>|
MODIFY FILE (<Параметры>)

где раздел ADD FILE добавляет файл, REMOVE FILE удаляет, а раздел MODIFY FILE изменяет параметры файла;

4) сжатие всей БД: DBCC SHRINKDATABASE <Имя БД>;
5) сжатие конкретного файла БД: DBCC SHRINKFILE <Логическое имя файла>;
6) переименование БД: EXEC SP_RENAMEDB <Имя БД>,<Новое имя БД>;
7) удаление БД: DROP DATABASE <Имя БД>.

Замечание: вышеперечисленные команды используют следующие параметры:

- <Имя БД> - имя БД с которой производится действие;
- <Параметр> - изменяемый параметр;
- <Значение> - новое значение изменяемого параметра;
- <Параметры> - параметры файла БД, аналогичные параметрам, используемым в команде CREATE DATABASE;

- <Логическое имя файла> - логическое имя файла, входящего в БД;
- <Новое имя БД> - новое имя БД.

Для запуска среды разработки «SQL Server Management Studio» в меню «Пуск» выбираем пункт «Программы\Microsoft SQL Server 2008\SQL Server Management Studio».

После запуска среды разработки появится окно подключения к серверу **«Connect to Server»** (см. рисунок 2.1).

Рисунок 2.1 - Окно подключения к серверу

В этом окне необходимо нажать кнопку **«Connect»**

Замечание: если при установке «Microsoft SQL Server 2008» был задан логин и пароль подключения к серверу, то перед нажатием кнопки **«Connect»**, в выпадающем списке **«Authentication»** нужно выбрать **«SQL Server Authentication»**, а затем необходимо ввести заданные при установке логин и пароль.

После нажатия кнопки **«Connect»** появится окно среды разработки **«SQL Server Management Studio»** (см. рисунок 2.2).

Данное окно имеет следующую структуру:

1) **оконное меню** содержит полный набор команд для управления сервером и выполнения различных операций;

2) **панель инструментов** содержит кнопки для выполнения наиболее часто производимых операций. Внешний вид данной панели зависит от выполняемой операции;

3) **панель «Object Explorer»** – обозреватель объектов. Обозреватель объектов – это панель с древовидной структурой, отображающая все объекты сервера, а также позволяющая производить различные операции как с самим сервером, так и с БД. Обозреватель объектов является основным инструментом для разработки БД;

4) **рабочая область**. В рабочей области производятся все действия с БД, а также отображается ее содержимое.

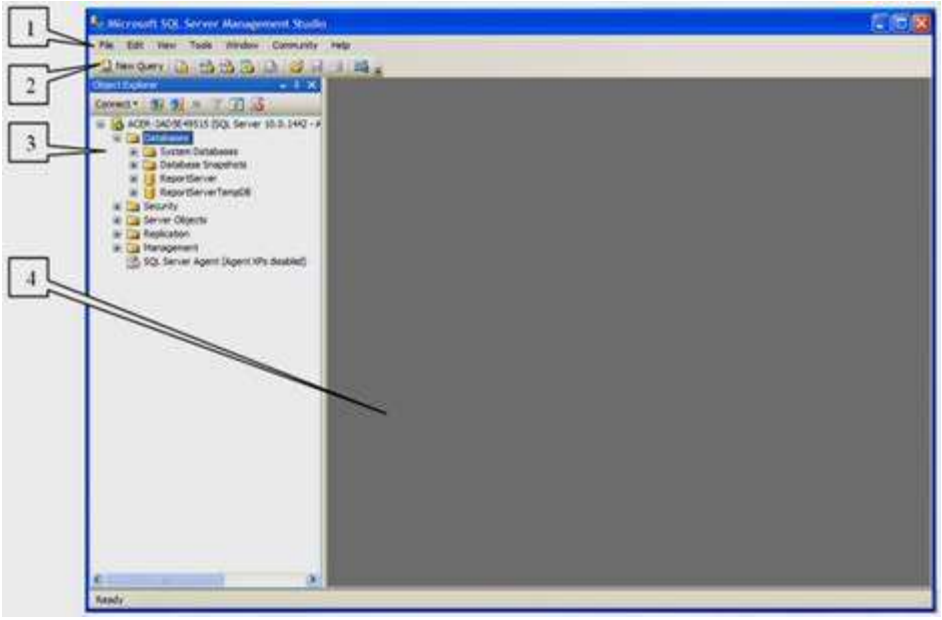


Рисунок 2.2

Замечание: в обозревателе объектов сами объекты находятся в папках. Чтобы открыть папку, необходимо щелкнуть по знаку «+» слева от изображения папки.

Перейдем непосредственно к созданию файла данных. Для этого в обозревателе объектов щелкните **ПКМ** на папке **«Databases»** (Базы данных) (см. рисунок. 2.2) и в появившемся меню выберите пункт **«New Database»** (Новая БД). Появится окно настроек параметров файла данных новой БД **«New Database»** (см. рисунок 2.3). В левой части окна настроек имеется список **«Select a page»**. Этот список позволяет переключаться между группами настроек.

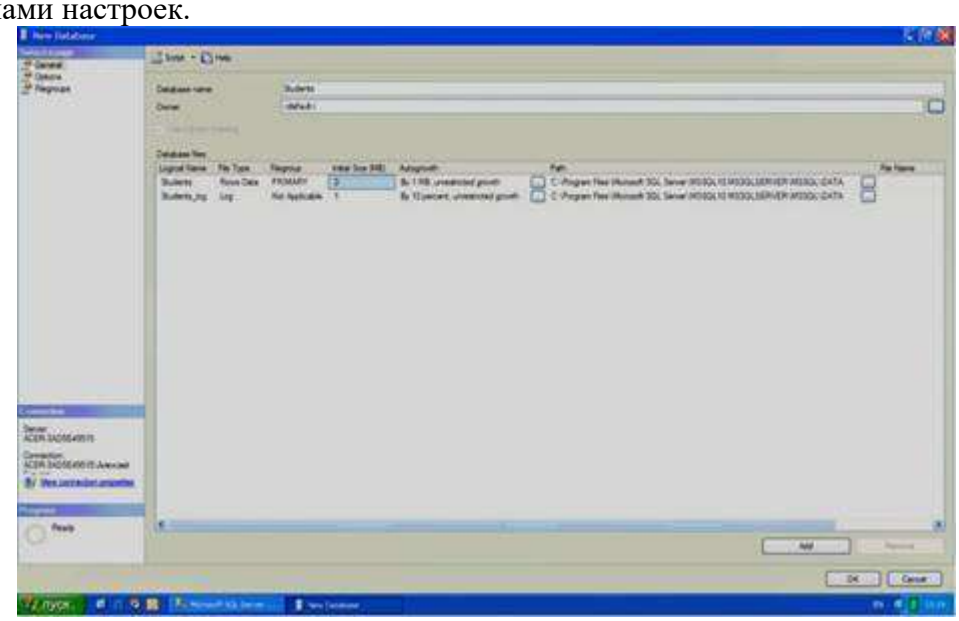


Рисунок 2.3

Настроим основные настройки **«General»**. Для выбора основных настроек нужно просто щелкнуть мышью по пункту **«General»** в списке **«Select a page»**. В правой части окна **«New Database»** появятся основные настройки.

Рассмотрим их более подробно. Верхней части окна расположено два параметра: **«Database name»** (Имя БД) и **«Owner»** (Владелец). Задайте параметр **«Database name»** равным **«Students»**. Параметр **«Owner»** оставьте без изменений.

Под вышеприведенными параметрами в виде таблицы располагаются настройки файла данных и журнала транзакций. Таблица имеет следующие столбцы:

1) **Logical Name** – логическое имя файла данных и журнала транзакций. По этим именам будет происходить обращение к вышеприведенным файлам в БД. Можно заметить,

что файл данных имеет то же имя что и БД, а имя файла журнала транзакций составлено из имени БД и суффикса «_log»;

2) **File Type** – тип файла. Этот параметр показывает, является ли файл файлом данных или журналом транзакций;

3) **Filegroup** – группа файлов, показывает к какой группе файлов относится файл. Группы файлов настраиваются в группе настроек **«Filegroups»**;

4) **Initial Size (MB)** – начальный размер файла данных и журнала транзакций в мегабайтах;

5) **Autogrowth** – автоувеличение размера файла. Как только файл заполняется информацией его размер автоматически увеличивается на величину, указанную в параметре **«Autogrowth»**. Увеличение можно задавать как в мегабайтах так и в процентах. Здесь же можно задать максимальный размер файлов. Для изменения этого параметра надо нажать кнопку **«…»**;

6) **Path** – путь к папке, где хранятся файлы. Для изменения этого параметра также надо нажать кнопку **«…»**;

7) **File Name** – имена файлов. По умолчанию имена файлов аналогичны логическим именам. Однако файл данных имеет расширение «mdf», а файл журнала транзакций – расширение «ldf».

Замечание: для добавления новых файлов данных или журналов транзакций используется кнопка **«Add»**, а для удаления - кнопка **«Remove»**.

Рассмотрим второстепенные настройки файла данных. Для доступа к этим настройкам необходимо щелкнуть мышью по пункту **«Options»** в списке **«Select a page»**. Появится следующее окно (см. рисунок. 2.4).

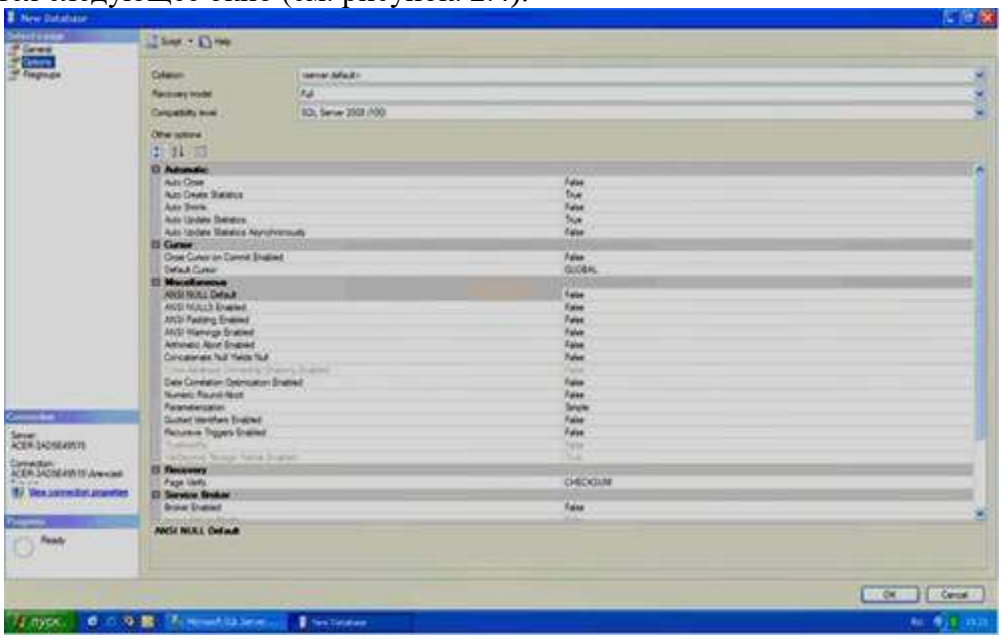


Рисунок 2.4

В правой части окна мы видим следующие настройки:

1) **Collation** – этот параметр отвечает за обработку текстовых строк, их сравнение, текстовый поиск и т.д. Рекомендуется оставить его как **«<server default>»**. При этом данный параметр будет равен значению, заданному на вкладке **«Collation»**, при установке сервера;

2) **Recovery Model** – модель восстановления. Данный параметр отвечает за информацию, предназначенную для восстановления БД, хранящуюся в файле транзакций. При наличии места на диске рекомендуется оставить этот параметр в значении **«Full»**;

3) **Compatibility level** – уровень совместимости, определяет совместимость файла данных с более ранними версиями сервера. Если планируется перенос данных на другую, более раннюю версию сервера, то ее необходимо указать в этом параметре;

4) **Other options** – второстепенные параметры. Данные параметры являются необязательными для изменения.

Рассмотрим последнюю группу настроек **«Filegroups»**. Данная группа настроек отвечает за группы файлов. Для ее отображения в списке **«Select a page»** необходимо щелкнуть мышью по пункту **«Options»**. Отобразятся настройки групп файлов (см. рисунок 2.5).

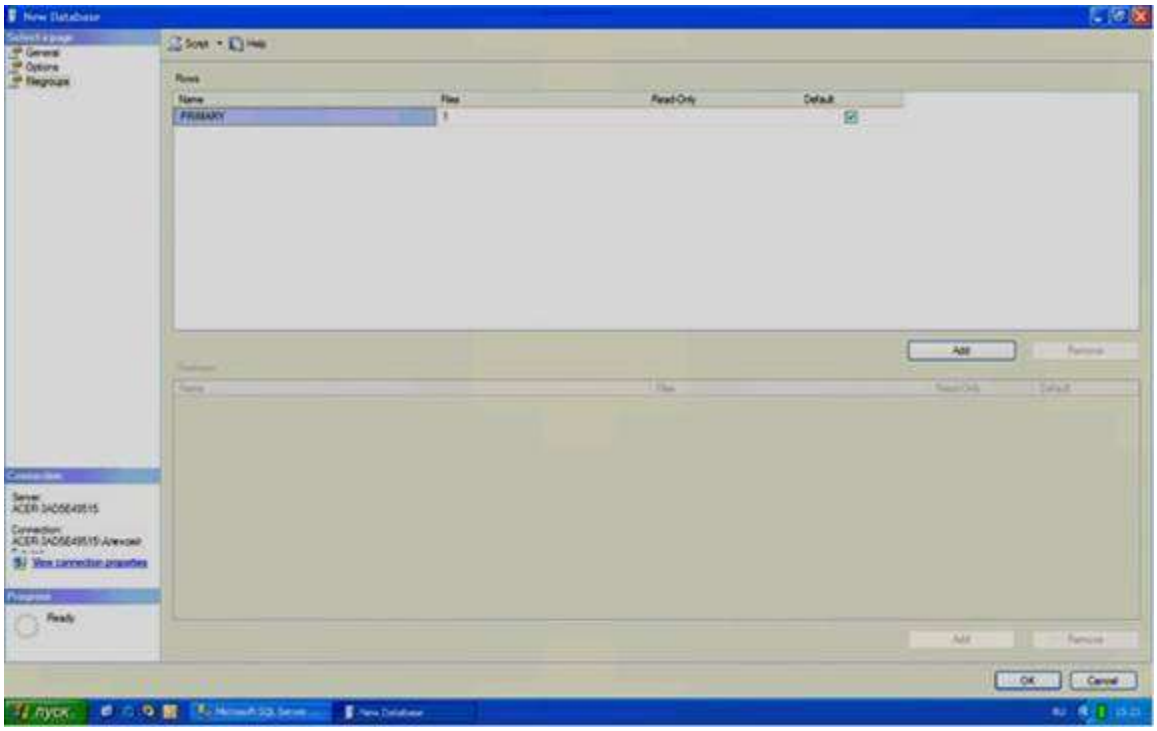


Рисунок 2.5

Группы файлов представлены в таблице **«Rows»** в правой части окна. Данная таблица имеет следующие столбцы:

1) **Name** – имя группы файлов.
2) **Files** – количество файлов входящих в группу.
3) **Read only** – файлы в группе будут только для чтения, их можно только просматривать, но нельзя изменять.
4) **Default** – группа по умолчанию. Все новые файлы данных будут входить в эту группу.

Замечание: как и в случае с файлами данных, для добавления новых групп используется кнопка **«Add»**, а для - удаления кнопка **«Remove»**.

Для переименования БД необходимо в обозревателе объектов щелкнуть по ней **ПКМ** и в появившемся меню выбрать пункт **«Rename»**. Для удаления в это же меню выбираем пункт **«Delete»**, для обновления – пункт **«Refresh»**, а для изменения свойств описанных выше – пункт **«Properties»**.

Лабораторная работа№5 Команды Transact_sqL

Цель занятия: *познакомиться* с возможностями Transact – SQL по созданию схем, логинов, пользователей и определения прав пользователей. Научиться организовывать со стороны клиентского приложения удаленное управление правами доступа к данным БД

Краткие теоретические сведения

Transact-SQL (T-SQL) — процедурное расширение языка SQL, созданное компанией Microsoft (для Microsoft SQL Server) и Sybase (для Sybase ASE).

SQL был расширен такими дополнительными возможностями как:

– управляющие операторы,

– локальные и глобальные переменные,

– различные дополнительные функции для обработки строк, дат, математики и т. п.,

– поддержка аутентификации Microsoft Windows.

Язык Transact-SQL является ключом к использованию MS SQL Server. Все приложения, взаимодействующие с экземпляром MS SQL Server, независимо от их реализации и пользовательского интерфейса, отправляют серверу инструкции Transact-SQL.

Практические задания

Задание №1: Создание логинов, пользователей и предоставление прав пользователям средствами transact-sql.

Указание: Перед выполнением работы ознакомьтесь с теоретическим материалом в презентации к лекции Введение в SQL Server, тема Безопасность.

Ход работы:

1. Запустите MS SQL Server Management Studio, подключитесь к серверу, используя технологию 1
2. Выберите контекстом свою базу данных свою БД, используя технологию 6
3. Найдите на панели инструментов среды кнопку «Создать запрос» и нажмите ее.
4. С помощью команд Transact – SQL создадим новый логин для вашей базы данных с именем «qwerty» и паролем «123456». Для создания логинов используется запрос CREATE LOGIN. Ознакомьтесь с синтаксисом запроса, представленным на рисунке 20.

Синтаксис:

```
CREATE LOGIN login { WITH <option_list1> | FROM <sources> }
<sources> ::=
  WINDOWS [ WITH <windows_options> [ ,...n ] ]
  | CERTIFICATE certificate_name
  | ASYMMETRIC KEY asym_key_name
<option_list1> ::=
  PASSWORD = 'password' [ HASHED ] [ MUST_CHANGE ]
  [ , <option_list2> [ ,...n ] ]
<option_list2> ::=
  SID = sid
  | DEFAULT_DATABASE = database_name
```

```
  | DEFAULT_LANGUAGE = language_id
  | CHECK_EXPIRATION = { ON | OFF }
  | CHECK_POLICY = { ON | OFF }
  [ CREDENTIAL = credential_name ]
<windows_options> ::=
  DEFAULT_DATABASE = database_name
  | DEFAULT_LANGUAGE = language_id
```

Рисунок 20 – Синтаксис запроса Create Login

5. Для создания нашего логина необходимо набрать и выполнить следующий запрос:

CREATE LOGIN qwerty WITH PASSWORD='123456'

6. После создания логина, можно приступить к созданию пользователя для этого логина. Создадим одноименного пользователя «qwerty». Для создания пользователей используется запрос CREATE USER. Ознакомьтесь с синтаксисом запроса, представленным на рисунке 21.

```
CREATE USER user_name
  [ { FOR | FROM }
    { LOGIN login
      | CERTIFICATE certificate_name
      | ASYMMETRIC KEY asym_key_name
    }
  ]
  [ WITH DEFAULT_SCHEMA = schema_name ]
```

Рисунок 21 – Синтаксис запроса Create User

7. Для создания нашего пользователя необходимо набрать и выполнить следующий запрос:

CREATE USER qwerty FOR LOGIN qwerty

8. После добавления логина и пользователя отключитесь от сервера, нажав в обозревателе объектов кнопку «Отключить», изображение которой представлено на рисунке 22.

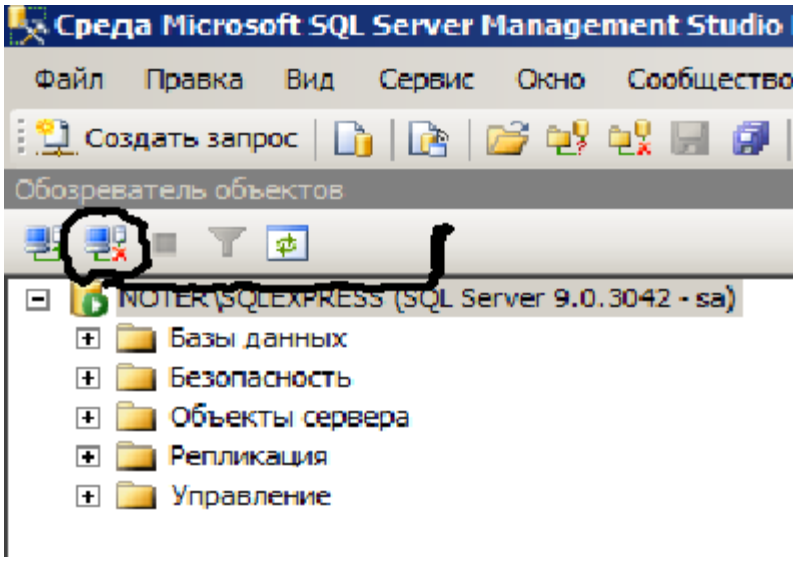


Рисунок 22 – Отключение от сервера

9. Подключитесь к серверу с помощью нажатия на кнопку «Подключение» под именем qwerty с паролем 123456

10. Разверните в обозревателе объектов свою БД, разверните узел «Таблицы» и убедитесь, что данный пользователь не имеет доступа ни к каким таблицам БД.

11. Отключитесь и подключитесь заново к серверу с правами администратора (то есть пользователь sa).

12. Добавим для пользователя qwerty возможность просмотра таблицы Модель и добавления записей в таблицу Модель. Для передачи прав пользователю используется SQL запрос GRANT. Ознакомьтесь с синтаксисом запроса, представленным на рисунке 23.

```
GRANT
  { ALL [ PRIVILEGES ] }
  | permission_name [ ( column_name [ ,...n ] ) ] [ ,...n ]
    [ ON [ class:: ] securable ]
  TO principal [ ,...n ] [ WITH GRANT OPTION ]
  [ AS principal ]
```

Рисунок 23 – Синтаксис запроса GRANT

13. Для передачи прав пользователю qwerty на просмотр и добавления записей в таблице Модель необходимо набрать и выполнить следующие два запроса:

Grant Select On модель to qwerty;

Grant Insert On модель to qwerty;

14. После успешного выполнения предыдущих запросов, отключитесь и подключитесь заново к серверу как пользователь qwerty.

15. Разверните в обозревателе объектов свою БД, разверните узел «Таблицы»◊ таблица Модель, просмотрите записи, добавьте новую и убедитесь, что данный пользователь имеет права просмотра и добавления для таблицы.

Лабораторная работа№6 Обеспечение безопасности в SQL SERVER

Цель занятия: *п*ознакомиться с политикой безопасности MS SQL Server,.

Краткие теоретические сведения

Базовые концепции безопасности SQL Server. MSSQL управляет доступом к объектам через аутентификацию и авторизацию.

Аутентификация — это процесс входа в SQL Server, когда пользователь отправляет свои данные на сервер. Аутентификация устанавливает личность пользователя, который проходит аутентификацию;

Авторизация — это процесс определения того, к каким защищаемым объектам может обращаться пользователь, и какие операции разрешены для этих ресурсов.

Многие объекты SQL Server имеют свои разрешения, которые могут наследоваться от вышестоящего объекта. Разрешения могут быть предоставлены отдельному пользователю, группе или роли.

Практические задания

Задание №1: В среде MS Visual Studio необходимо создать Windows-приложение, которое позволит добавлять новых пользователей для вашей БД вида, представленного на рисунке 24.

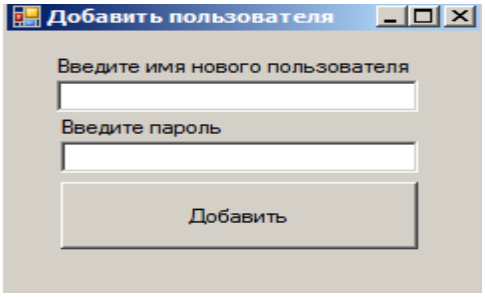


Рисунок 24 – Форма для добавления пользователя

Ход работы:

1. Добавьте в проект новую форму. Добавьте на форму следующие компоненты: 2 Label, 2 textBox, 1 Button и измените их свойства в соответствии с рисунком 24.
2. Добавьте обработчик для открытия только что добавленной формы при выборе в главном меню проекта пункта Администрирование◊Добавить пользователя.
3. Добавьте в код формы ссылку на пространства имен для работы с объектами ADO: Imports System.Data

   Imports System.Data.SqlClient
4. В коде формы «Добавить пользователя», в процедуре-обработчике кнопки «Добавить» опишите переменную и создайте экземпляр объекта Connection и задайте ей параметры подключения, используя технологии 9,10.
5. Опишите переменную и создайте экземпляр объекта Command, используя технологию 11. Создаваемый вами объект Command должен быть подключен к ранее (в четвертом пункте) созданному объекту Connection.
6. Задайте для только что созданного объекта Command текст запроса на добавление логина (CREATE LOGIN) с использованием информации из текстовых полей на форме, используя технологию 12 ИЛИ технологию 13.
7. Откройте подключение к БД, используя технологию 15.
8. Выполните одним из методов объект Command, используя ОДНУ из следующих технологий 16,17,18.
9. Закройте подключение к БД, используя технологию 15.
10. Запустите проект, добавьте нового пользователя к вашей БД.
11. Подключитесь к базе данных МММ в среде MS SSMS, используя технологию 1 под именем только что добавленного пользователя. Убедитесь что это возможно.

Задание №3. В среде MS Visual Studio необходимо создать Windows-приложение, которое позволит добавлять новых пользователей для вашей БД вида, изображенного на рисунке 25.

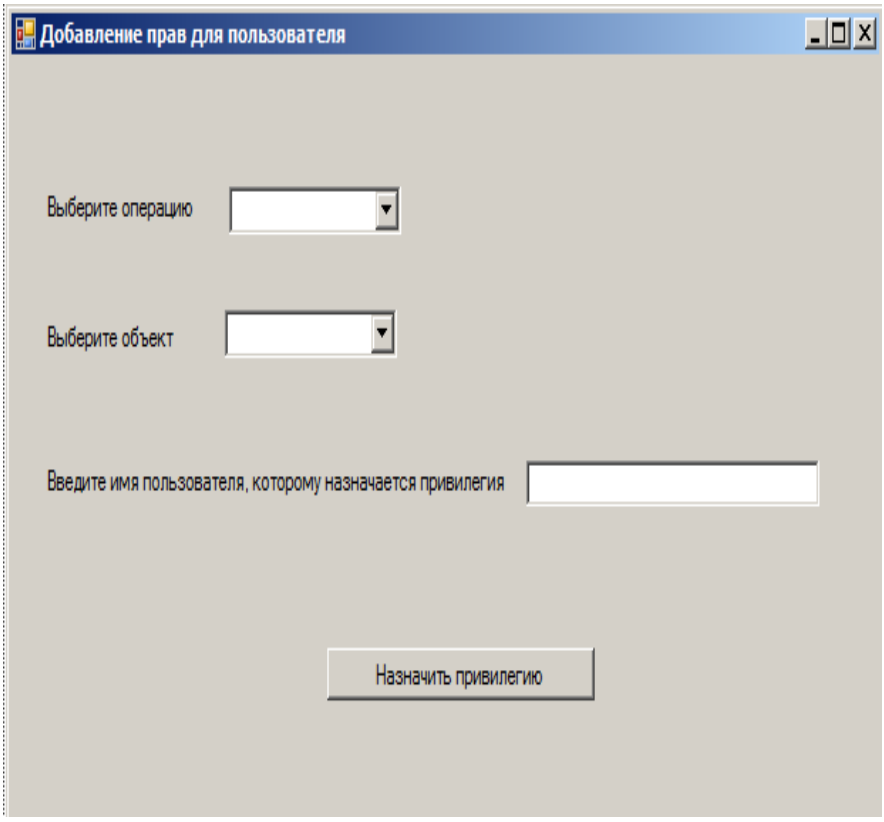


Рисунок 25 – Форма для добавления привилегий пользователя

Ход работы:

1. Добавьте в проект новую форму. Добавьте на форму следующие компоненты: 3 Label, 2 ComboBox, 1 textBox, 1 Button и измените их свойства в соответствии с рисунком 25. В список ComboBox1 занести перечень значений:
   - INSERT;
   - UPDATE;
   - DELETE.

   В список ComboBox2 занести перечень значений = название таблиц Вашей БД, например:
   - Модель;
   - Готовый_продукт;
   - Заказ;
   - Состав_заказа;
   - Магазин.
2. Добавьте обработчик для открытия только что добавленной формы при выборе в главном меню проекта пункта Администрирование◊Добавить права для пользователя
3. Добавьте в код формы ссылку на пространства имен для работы с объектами ADO: Imports System.Data

   Imports System.Data.SqlClient
4. В коде формы «Добавить права для пользователя», в процедуре-обработчике кнопки «Назначить привилегию» опишите переменную и создайте экземпляр объекта Connection и задайте ей параметры подключения, используя технологии 9,10.
5. Опишите переменную и создайте экземпляр объекта Command, используя технологию 11. Создаваемый вами объект Command должен быть подключен к ранее созданному объекту Connection.

6. Задайте для только что созданного объекта Command текст запроса на добавление Прав пользователю (GRANT) с использованием информации из полей со списком и текстовых полей на форме, используя технологию 12 ИЛИ технологию 13.

7. Откройте подключение к БД, используя технологию 15.

8. Выполните одним из методов объект Command, используя ОДНУ из следующих технологий 16,17,18.

9. Закройте подключение к БД, используя технологию 15.

10. Запустите проект, добавьте для какого-либо пользователя права на просмотр таблицы Модели вашей БД.

11. Подключитесь к базе данных MMM в среде MS SSMS под именем только что добавленного пользователя. Убедитесь что возможно просматривать записи из таблицы Модели.

Лабораторная работа№7 Установка и настройка сервера под UNIX

Цель занятия: ознакомление с регламентом установки и настройки сервера под UNIX.

Краткие теоретические сведения

Unix («UNIX» является зарегистрированной торговой маркой организации TheOpenGroup[1]) — семейство переносимых, многозадачных и многопользовательских операционных систем, которые основаны на идеях оригинального проекта AT&T Unix, разработанного в 1970-х годах в исследовательском центре BellLabs Кеном Томпсоном, ДеннисомРитчи и другими.

Операционные системы семейства Unix характеризуются модульным дизайном, в котором каждая задача выполняется отдельной утилитой, взаимодействие осуществляется через единую файловую систему, а для работы с утилитами используется командная оболочка.

Практические задания

Задание.

Провести установку и настройку сервера под UNIX

Лабораторная работа№8 Выполнение запросов к базе данных

Цель занятия: получить навыки формирования SQL запросов на добавление, изменение, извлечение и удаление данных на примере созданной согласно варианту базы данных.

Краткие теоретические сведения

Запрос SQL — это запрос, создаваемый при помощи инструкций SQL. Язык SQL (StructuredQueryLanguage) используется при создании запросов, а также для обновления и управления реляционными БД.

Практические задания

1) Заполнить БД, созданную в ПЗ №1 используя запросы
2) Создать запросы на извлечение данных (Требования: запросы должны отражать потребности реальных пользователей, например, найти самую дорогуюкнигу, самую покупаемую вещь, определить наиболее частых клиентов и т.д.)
3) Создать подзапросы и вложенные запросы (такие же требования как и в П2)

Лабораторная работа№9 Выполнение изменений в базе данных, создание триггеров

Цель занятия: Изучить основы создания простейших триггеров.

Краткие теоретические сведения

Триггер – это сочетание хранимой в базе данных процедуры и события, которое заставляет ее выполняться. Такими событиями могут быть: ввод новой строки таблицы, изменение значений одного или нескольких ее столбцов и (или) удаление строки таблицы. При любом из этих событий автоматически запускаются один или несколько заранее созданных триггеров, которые производят проверку запрограммированных в них условий, и если они не выполняются, отменяют ввод, изменение или удаление, посылая об этом заранее подготовленное сообщение пользователю.

Триггеры похожи на процедуры и функции тем, что также являются именованными блоками и имеют раздел объявлений, выполняемый раздел и раздел обработки исключительных ситуаций. Подобно процедурам и функциям, триггеры хранятся как автономные объекты в базе данных.

Триггеры позволяют:

– реализовывать сложные ограничения целостности данных, которые невозможно реализовать через ограничения, устанавливаемые при создании таблицы;

– контролировать информацию, хранимую в таблице, посредством регистрации вносимых изменений и пользователей, производящих эти изменения;

– автоматически оповещать другие программы о том, что необходимо делать в случае изменения информации, содержащейся в таблице;

– публиковать информацию о различных событиях. Триггеры также делятся на три основных типа.

– Триггеры DML активизируются предложениями ввода, обновления и удаления информации (INSERT, UPDATE, DELETE) до или после выполнения предложения, на уровне строки или таблицы.

– Триггеры замещения (insteadof) можно создавать только для представлений (либо объектных, либо реляционных). В отличие от триггеров DML, которые выполняются в дополнение к предложениям DML, триггеры замещения выполняются вместо предложений DML, вызывающих их срабатывание. Триггеры замещения должны быть строковыми триггерами.

– Системные триггеры активизируется не на предложение DML, выполняемое над таблицей, а на системное событие, например, на запуск или останов базы данных. Системные триггеры срабатывают и на предложения DDL, такие как создание таблицы.

Практические задания

1) Открыть БД, созданную в ПЗ №12

4) Создать триггеры с помощью запросов

5) Отразить созданные запросы в отчете с комментариями

Лабораторная работа№10 Создание запросов и процедур на изменение структуры базы данных

Цель занятия: получить навыки формирования SQL запросов на добавление, изменение, извлечение и удаление данных на примере созданной согласно варианту базы данных. Изучить основы создания простейших триггеров.

Краткие теоретические сведения

Процесс создания любого запроса на изменение начинается с создания запроса на выборку, который после добавления в него необходимых таблиц преобразуется в нужный запрос на изменение.

Практические задания

1) Открыть БД, созданную в ПЗ №12
2) Создать запросы на изменение структуры базы данных
3) Отразить созданные запросы в отчете с комментариями

Лабораторная работа№11 Работа с журналом аудита базы данных

Цель занятия: Изучить основные принципы аудита безопасности, управление политикой и правилами аудита, журналами безопасности системы

Краткие теоретические сведения

Аудит — это одно из основных средств защиты ОС. Аудит позволяет отслеживать и журналировать события, связанные с безопасностью. Примерами событий для аудита можно назвать доступ к файлу, вход в систему или изменение системной конфигурации.

Журнал безопасности представляет собой базу данных или файл, в котором регистрируются события, связанные с безопасностью системы.

Благодаря системе аудита, администратор может узнать, кто, каким образом и когда воспользовался (или пытался воспользоваться, но получил отказ в доступе) интересующими его ресурсами.

Настройка средств аудита позволяет выбрать типы событий, подлежащих регистрации, и определить, какие именно параметры будут регистрироваться.

Практические задания

1) Изучить управление политиками аудита безопасности
2) Изучить управление журналами событий и безопасности

Лабораторная работа№12 Резервное копирование баз данных

Цель занятия: ознакомиться с основными конструкциями SQL, технологиями среды MS SQL Server Management для резервного копирования и восстановления БД.

Краткие теоретические сведения

Компонент резервного копирования и восстановления SQL Server обеспечивает необходимую защиту важных данных, хранящихся в базах данных SQL Server. Чтобы свести к минимуму риск необратимой потери данных, необходимо создавать резервные копии баз данных для сохранения вносимых изменений на регулярной основе. Хорошо спланированная стратегия резервного копирования и восстановления защищает базы от потери данных, вызванной разными сбоями. Протестируйте стратегию, выполнив восстановление набора резервных копий и вернув в исходное состояние базу данных. Так вы будете готовы эффективно реагировать на проблемы.

Практические задания

Задание №1. необходимо создать резервные копии базы данных «MMM» с использованием полного резервного копирования, разностного резервного копирования и резервного копирования журнала транзакций.

Ход работы:

1. Запустите SQL Server Management Studio (SSMS), подключитесь к своему экземпляру SQL Server, используя технологию 1.
2. Создайте папку с именем c:\Student\ВашаПапка\test.
3. Откройте окно нового запроса. Измените контекст на базу данных master, используя технологию 6. Наберите и исполните следующую команду, чтобы создать полную резервную копию базы данных:

BACKUP DATABASE MMM TO DISK = 'C:\.....TEST\AW.BAK’

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

4. Внесите изменение в таблицу «Модель» базы данных MMM. Добавьте одну запись (придумайте сами)/
5. Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать резервную копию журнала транзакций и сохранить только что внесенное изменение:

BACKUP LOG MMM TO DISK = ‘C:\.....TEST\AW1.TRN'

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

6. Внесите еще одно изменение в таблицу«Модель».
7. Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать разностную резервную копию базы данных:

BACKUP DATABASE MMM TO DISK = ‘C:\.....\TEST\AWDIFF1.BAK' WITH DIFFERENTIAL

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

8. Внесите еще одно изменение в таблицу«Модель».
9. Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать полную резервную копию базы данных в указанном месте на диске:

BACKUP LOG MMM TO DISK = ‘C:\....TEST\AW2.TRN'

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

Задание №2. необходимо провести восстановление базы данных «MMM» из сделанных в задании №1 резервных копий.

Ход работы:

1. Если необходимо, запустите SSMS, подключитесь к своему экземпляру SQL Server, используя технологию 1.

2. Выполните восстановление БД из первой полной резервной копии (C:\...TEST\AW.BAK) средствами оболочки SSMS. Для этого выполните:

- В обозревателе объектов вызовите контекстное меню на вашей БД и выберите задачу восстановления базы данных (см. рисунок 6).

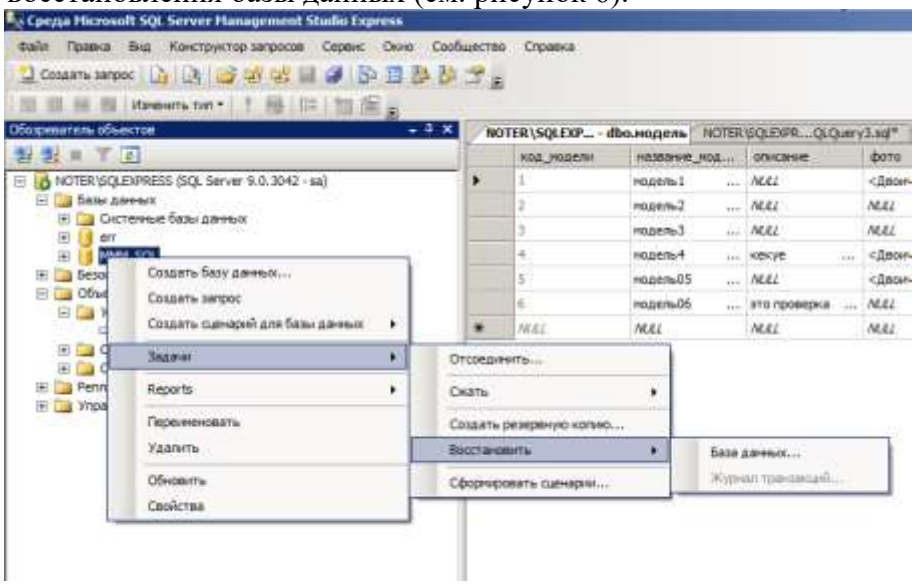


Рисунок 6 – Восстановление БД

- В открывшемся окне необходимо задать следующие параметры восстановления

На закладке «Общие» необходимо выбрать:

1. Базу данных для восстановления (вашу МММ)
2. Выбрать источник набора данных для восстановления с устройства ◊ файл C:\...TEST\AW.BAK
3. После определения файла-источника данных необходимо флажком выбрать базу данных для восстановления (рисунок 7).

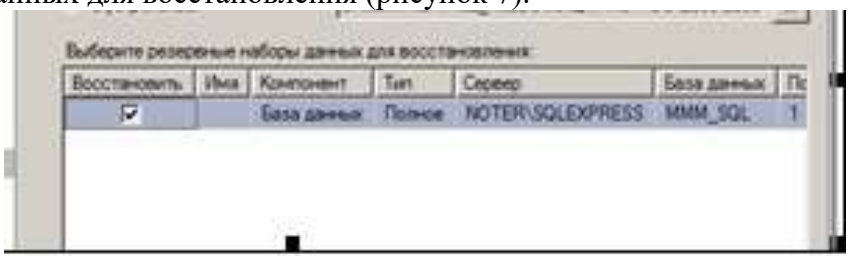


Рисунок 7- Выбор БД для восстановления

На закладке «Параметры»

1. необходимо включить опцию «Перезаписать БД» и «оставить БД готовой к использованию», (рисунок 8).

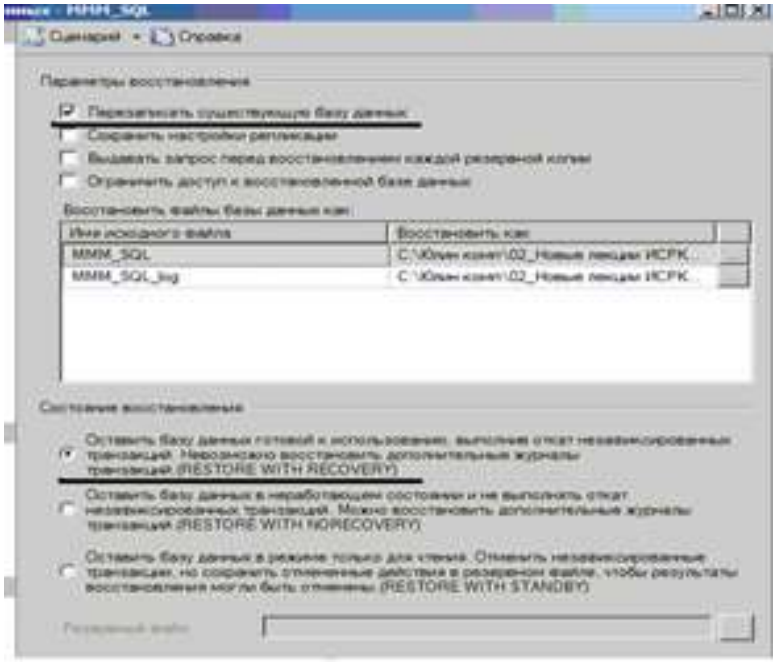


Рисунок 8 – Задание параметров восстановления

3. Нажмите ОК

4. После восстановления БД, откройте таблицу «Модель» и убедитесь, что она не содержит всех добавлений, вносимых вами в процессе выполнения упражнения, так как восстановление происходило из первой резервной копии (без изменений).

Контрольные вопросы:

Лабораторная работа№13 Мониторинг нагрузки сервера

Цель занятия: Изучить принципы работы простейших средств мониторинга сети.

Краткие теоретические сведения

1. Протокол ICMP

Протокол ICMP (Интернет-протокол контрольных сообщений) стека протоколов TCP/IP предназначен для передачи между сетевыми устройствами сообщений об ошибках и контрольных сообщений при помощи IP-пакетов.

В протоколе ICMP определены несколько типов сообщений, в том числе:

DestinationUnreachable	TimetoLiveExceeded	ParameterProblem
SourceQuench	Redirect	Echo
EchoReply	Timestamp	TimestampReply
InformationRequest	InformationReply	AddressRequest
AddressReply		

Например, если маршрутизатор получает пакет, который он не может доставить по указанному в нем адресу, отправителю передается ICMP-сообщение о недостижимости адреса (DestinationUnreachable).

2. PING: Проверка соединения с определенным интерфейсом. Программа ping использует протокол ICMP.

Эта команда посылает пакет эхо-запроса на другой IP-адрес и ожидает ответа. Она чаще всего используется для того, чтобы посмотреть, «жив ли» другой компьютер. Ответ на запрос содержит также данные о том, как долго пакет путешествовал до адресата. Можно использовать команду ping с различными опциями: число посланных пакетов (от 1 до 10), время жизни пакета (timetolive –TTL, от 1 до 255ms), размер пакета (от 16 до 8192 байт), время ожидания (timeout, до 9999 ms) и разрешать или нет фрагментацию каждого пакета.

Практические задания

Изучите принципы работы простейших средств мониторинга сети

Лабораторная работа№14 Автоматизация административных задач

Цель занятия: Изучить принципы работы простейших средств администрирования сети.

Краткие теоретические сведения

1. Протокол ICMP

Протокол ICMP (Интернет-протокол контрольных сообщений) стека протоколов TCP/IP предназначен для передачи между сетевыми устройствами сообщений об ошибках и контрольных сообщений при помощи IP-пакетов.

В протоколе ICMP определены несколько типов сообщений, в том числе:

DestinationUnreachable	TimetoLiveExceeded	ParameterProblem
SourceQuench	Redirect	Echo
EchoReply	Timestamp	TimestampReply
InformationRequest	InformationReply	AddressRequest
AddressReply		

Например, если маршрутизатор получает пакет, который он не может доставить по указанному в нем адресу, отправителю передается ICMP-сообщение о недостижимости адреса (DestinationUnreachable).

2. PING: Проверка соединения с определенным интерфейсом. Программа ping использует протокол ICMP.

Эта команда посылает пакет эхо-запроса на другой IP-адрес и ожидает ответа. Она чаще всего используется для того, чтобы посмотреть, «жив ли» другой компьютер. Ответ на запрос содержит также данные о том, как долго пакет путешествовал до адресата. Можно использовать команду ping с различными опциями: число посланных пакетов (от 1 до 10), время жизни пакета (timetolive –TTL, от 1 до 255ms), размер пакета (от 16 до 8192 байт), время ожидания (timeout, до 9999 ms) и разрешать или нет фрагментацию каждого пакета.

Практические задания

Изучите принципы работы простейших средств мониторинга сети

МДК.07.02 СЕРТИФИКАЦИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

Лабораторная работа№15 Настройка политики безопасности

Цель занятия: изучить структуру и возможности групповых и локальных политик безопасности, научиться настраивать политику безопасности.

Краткие теоретические сведения

Служба каталогов Active Directory является средством для именования, хранения и выборки информации в некоторой распределенной среде, доступное для приложений, пользователей и различных клиентов этой среды. Служба сетевых каталогов хранит информацию об общедоступных приложениях, файлах, принтерах и сведения о пользователях.

Служба каталогов Active Directory обеспечивает эффективную работу сложной корпоративной среды, предоставляя следующие возможности.

Пользователи могут регистрироваться в сети с одним именем и паролем и получать при этом доступ ко всем сетевым ресурсам (серверам, принтерам, приложениям, файлам и т. д.) независимо от их расположения в сети.

Средства аутентификации и управления доступом к ресурсам, встроенные в службу Active Directory, обеспечивают централизованную защиту сети. Права доступа можно определять не только для каждого объекта каталога, но и каждого свойства (атрибута) объекта.

Администраторы могут централизованно управлять всеми корпоративными ресурсами.

При загрузке компьютера или регистрации пользователя в системе выполняются требования групповых политик; их настройки хранятся в объектах групповых политик (GPO) и «привязываются» к сайтам, доменам или организационным единицам. Групповые политики определяют, например, права доступа к различным объектам каталога или ресурсам, а также множество других правил работы в системе.

Служба Active Directory тесно связана с DNS. Этим достигается единство в именовании ресурсов локальной сети и сети Интернет, в результате чего упрощается подключение пользовательской сети к Интернету.

Служба Active Directory может охватывать как один домен, так и множество доменов, один контроллер домена или множество контроллеров домена, т. е. она отвечает требованиям сетей любого масштаба. Несколько доменов можно объединить в дерево доменов, а несколько деревьев доменов можно связать в лес.

Для разработчиков приложений служба каталогов предоставляет доступ ко всем возможностям (средствам) каталога и поддерживает принятые стандарты и интерфейсы программирования (API). Служба каталогов тесно связана с операционной системой, что позволяет избежать дублирования в прикладных программах функциональных возможностей системы, например, средств безопасности.

Каталог состоит из элементов (entries), представляющих собой информацию, или атрибуты, связанные с некоторым реальным объектом, например компьютером, человеком или организацией.

Каждый объект принадлежит хотя бы к одному объектному классу, представляющему собой некоторое семейство объектов с определенными общими характеристиками. Класс объектов определяет тип информации, содержащейся в Active Directory для экземпляров (объектов) данного класса. Атрибуты могут быть как обязательными (mandatory) для данного класса (например, имя), так и дополнительными (optional) (пароль).

Контейнер (container) – это специфический объект службы каталогов, который, в отличие от обычных объектов, не имеет какого-либо физического представления, а служит только структурной организации других объектов каталога. Типичным приме ром контейнеров могут служить организационные единицы, или подразделения, используемые

для упрощения администрирования отдельных групп ресурсов или пользователей в домене.

Элементы каталога организованы в виде иерархического дерева, называемого Directory Information Tree (DIT, Информационное дерево каталога или просто Дерево каталога). Элементы, находящиеся ближе к корню дерева, обычно представляют крупные объекты, например, организации или компании; элементы, располагающиеся на ветвях этого дерева (листья), представляют более простые объекты – пользователей, устройства, компьютеры.

Схема каталога (Directory Schema) – это набор правил, описывающих структуру дерева каталога, объявления и синтаксис объектных классов и типы атрибутов, входящих в каталог.

Схема каталога гарантирует, что все добавления или изменения каталога соответствуют данным правилам, и препятствует появлению некорректных элементов, ошибочных типов атрибутов или классов.

В Active Directory схема реализована как набор экземпляров объектных классов, хранящийся в самом каталоге. Этим Active Directory отличается от многих каталогов, в которых схема хранится в текстовом файле, считываемом при запуске каталога. Когда схема хранится в каталоге, пользовательские приложения могут обращаться к ней и узнавать об имеющихся объектах и свойствах. Схему Active Directory можно динамически обновлять: модифицировать и расширять.

Основные компоненты любой службы каталога – база данных, содержащая нужную информацию, и один или несколько протоколов, обеспечивающих доставку данных пользователям.

Active Directory обеспечивает хранение любой общедоступной информации. Как и другие службы каталогов, Active Directory обеспечивает некоторый механизм хранения информации и протоколы для доступа к ней.

Домены – это известное решение для администрирования групп, предоставляющее каждому пользователю учетную запись в конкретном домене. В Windows NT Server 4.0 доменам давались простые строковые имена (имена NetBIOS), в среде Windows Server каждый домен должен иметь имя, отвечающее соглашениям именования доменов Domain Name System (DNS). В каждом домене один или несколько компьютеров должны выполнять функции контроллеров домена.

В среде Windows Server каждый контроллер домена содержит полную копию базы данных Active Directory этого домена. В Active Directory используются так называемое ядро Extended Storage Engine (ESE) и два различных протокола, обеспечивающих связь между клиентами и базой данных. Для поиска контроллера домена клиент обращается к протоколу, описанному в DNS. Для доступа к данным в Active Directory клиент использует протокол Lightweight Directory Access Protocol (LDAP).

В большинстве современных сетей TCP/IP используется служба DNS, главное назначение которой преобразовывать сим вольные имена в IPадреса. Для этого каждый компьютер-сервер DNS имеет набор записей с информацией о ресурсах. Каждая запись имеет некоторый тип, определяющий характер и назначение хранящейся информации. Интеграцию служб Active Directory и DNS можно рассматривать в трех аспектах:

– домены Active Directory и домены DNS имеют одинаковую иерархическую структуру и схожее пространство имен;

– зоны (zone) DNS могут храниться в Active Directory. Если используется сервер DNS, входящий в состав Windows Server, то первичные зоны (primary zone), занесенные в каталог, реплицируются на все контроллеры домена, что обеспечивает лучшую защищенность службы DNS.

Каждый элемент Active Directory и каждый атрибут любого элемента имеют список управления доступом (ACL), который определяет права и возможности пользователей в отношении до ступа к конкретным элементам и атрибутам. Например, список ACL может

позволить одним пользователям читать атрибуты не которого элемента, другим пользователям – читать и изменять некоторые из атрибутов, а остальным – запретить какой-либо до ступ к элементу. Эффективное управление доступом невозможно без достоверной аутентификации клиентов, Active Directory ис пользует для этой цели протокол Kerberos.

Управление подразделениями, компьютерами, группами и учетными записями пользователей.

Для управления учетными записями пользователей и компьютерами следует вначале войти в раздел администрирования (Administrative Tools) и выбрать Active Directory Users and Computers .

Для создания подразделения, или организационной единицы (Organizational Unit, OU) следует:

1. Выделить объект типа «домен» и нажать правую кнопку мыши. В появившемся меню выбрать команду Создать | Подразделение (New | Organizational Unit). Можно воспользоваться па нелью инструментов и кнопкой Создание нового подразделения в текущем контейнере (Create a new organizational unit in a current container) на панели инструментов.
2. В открывшемся окне указать имя создаваемого подразделения и нажмите кнопку ОК.

В результате в выбранном вами домене будет создано под разделение с заданным именем. В дальнейшем внутри него можно создать вложенные подразделения.

В процессе установки домена Windows в нем создается не сколько встроенных групп, обладающих определенным набором прав. Их можно использовать для присвоения администраторам или пользователям определенных ролей или прав доступа в домене.

К встроенным относятся перечисленные ниже группы. Эти группы служат для назначения разрешений доступа пользователям, на которых возложено выполнение в данном домене каких -либо административных функций.

Локальные группы в домене:

– администраторы (Administrators);
– гости (Guests);
– операторы архива (Backup Operators);
– операторы печати (Print Operators);
– операторы сервера (Server Operators);
– операторы учета (Account Operators);
– пользователи (Users);
– репликатор (Replicator);
– совместимый с предWindows доступ(PreWindows CompatibleAccess).

Глобальные группы:

– администраторы домена (Domain Admins);
– владельцы-создатели групповой политики (Group Policy Creator Owners);
– гости домена (Domain Guests);
– издатели сертификатов (Cert Publishers);
– компьютеры домена (Domain Computers);
– контроллеры домена (Domain Controllers);
– пользователи домена (Domain Users);

Универсальные группы:

– администраторы предприятия (Enterprise Admins);
– администраторы схемы (Schema Admins).

Универсальные группы создаются только на контроллерах корневого (первого в лесе) домена. В зависимости от установленных на сервере служб могут быть и

дополнительные встроенные группы, локальные в домене или глобальные. По умолчанию все встроенные локальные группы домена находятся в папке Builtin объекта домена. Все встроенные глобальные группы находятся в папке Users. Встроенные группы можно переносить в другие контейнеры или подразделения в пределах домена.

По умолчанию каждая созданная в домене учетная запись автоматически становится членом группы Пользователи домена. Кроме того, группа Пользователи домена является членом локальной в домене группы Пользователи.

Любой объект типа Компьютер (Computer) при создании по умолчанию автоматически включается в группу Компьютеры домена.

Группа Администраторы домена объединяет всех пользователей, имеющих полный административный доступ в домене. По умолчанию Администраторы домена являются членами локальной в домене группы Администраторы.

Группа Гости домена объединяет все учетные записи, с помощью которых можно зарегистрироваться в домене без пароля и получить минимальные права доступа. По умолчанию Гости домена являются членами локальной в домене группы Гости.

Помимо перечисленных выше встроенных групп администратор может создать любое количество групп пользователей и предоставить им необходимый набор прав и разрешений. Для со здания группы необходимо выполнить следующее:

1. Выбрать подразделение, где следует создать группу, и нажмите правую кнопку мыши. Выбрать в появившемся меню команду Создать | Группа (Group), либо нажать кнопку Создание новой группы в текущем контейнере (Create New Group in a Current Container) на панели инструментов.
2. В открывшемся окне диалога Новый объект – Группа (New Object – Group) в поле Имя группы (Group name) ввести имя создаваемой группы.
3. Установить переключатель Тип группы (Group type) в одно из положений, соответствующее типу создаваемой группы: Группа безопасности (Security) или Группа распространения (Distribution). Первый тип группы служит для предоставления пользователям определенного набора прав доступа к таким ресурсам сети, как файлы и принтеры. Второй тип группы служит только для распространения информации в сети, например, в качестве списков рассылки электронной почты. Следует отметить, что группы безопасности могут использоваться в качестве групп распространения.
4. Установив в одно из положений переключатель Область действия группы (Group scope), выбирать подходящую область действия создаваемой группы. Область действия группы определяет, где может быть видна данная группа (уровень доступности) и какие типы объектов могут быть ее членами, и может быть выбрана как:

– локальная в домене (Domain Local): пользователи, а также глобальные и универсальные группы из всего леса, другие локальные группы из этого же домена;

– глобальная (Global): пользователи, а также глобальные и универсальные группы;

– универсальная (Universal): пользователи и глобальные группы (только в основном режиме домена).

Для создания в домене учетной записи пользователя, пред положим с идентификатором popov_as, необходимо выполнить следующее:

1. Указать подразделение, в котором следует создать учетную запись, и нажмите правую кнопку мыши. В появившемся меню выбрать команду Создать | Пользователь.
2. В окне диалога Новый объект – Пользователь (New Object

– User) в поле Имя входа пользователя (User logon name) ввести уникальный идентификатор, в поле Имя (First name) – имя пользователя, в поле Фамилия (Last name) – фамилию пользователя, в поле Полное имя (Full name) автоматически появятся имя и фамилия пользователя. После ввода всей необходимой информации нажать кнопку Далее (Next).

3. В следующем окне в полях ввода Пароль (Password) и Подтверждение (Confirm password) ввести с клавиатуры пароль учетной записи пользователя.

4. Если необходима принудительная смена пароля при пер вой регистрации в сети, установить флажок потребовать смену пароля при следующем входе в систему (User must change password at next logon). С целью защиты от атак по подбору пароля, следует установить срок действия пароля пользователя, сбросив флажок Срок действия пароля не ограничен (Password never expires).

6. Установленный флажок Запретить смену пароля пользо вателем (User cannot change password) запрещает пользователю самостоятельно изменять свой пароль.

7. Если только что созданная учетная запись по каким-либо причинам должна быть заблокирована, установить флажок Отключить учетную запись (Account disabled).

8. По завершении настройки создаваемой учетной записи нажимать кнопку Далее.

9. В окне диалога, запрашивающего подтверждение правильности выполняемого действия, нажимать кнопку Готово (Finish).

Для ввода дополнительной информации или изменения не которых данных пользователя:

1. Указать учетную запись пользователя, информацию кото рой следует изменить, и нажмите правую кнопку мыши. В по явившемся меню выбрать команду Свойства.

2. Внести необходимые изменения и нажмите кнопку ОК.

Учетную запись пользователя можно перемещать из одного подразделения в другое в пределах одного домена или между доменами. Для соответствующего перемещения учетной записи этого пользователя следует воспользоваться технологией Drag and Drop (перетаскивание), применяемой практически ко всем визуальным объекта операционной системы семейства Windows.

Для добавления пользователя в группу необходимо выполнить следующие действия:

1. Указать группу, в которую необходимо добавить пользователя, и нажмите правую кнопку мыши. В появившемся меню выбрать команду Свойства. Появится окно свойств группы.

2. Перейти на вкладку Члены группы (Members) окна свойств и нажать кнопку Добавить.

3. Появится окно Выбор: Пользователи, Контакты или Компьютеры (Select Users, Contacts, or Computers). Здесь можно за дать область выполнения запроса: весь каталог, определенный домен или определенная часть дерева подразделения внутри домена. Обратите внимание, что каталог может состоять из множества доменов.

4. Выбрать имя добавляемого пользователя и нажать кнопку Добавить. Обратите внимание, что, нажав клавишу и одновременно выполняя щелчки на нужных объектах, в этом диалоговом окне можно одновременно выбрать несколько пользователей или групп.

В результате все выбранные объекты станут членами соответствующей группы.

После создания объекта «компьютер» можно управлять им удаленно, диагностируя службы, работающие на этом компьютере, просматривая события и т. д.

Для того чтобы управлять компьютером удаленно:

1. В окне оснастки Active Directory – Пользователи и компьютеры укажите имя компьютера и нажать правую кнопку мы ши. В появившемся меню выбрать команду Управление (Manage).

2. Для выбранного компьютера будет запущена оснастка Управление компьютером (Computer Management).

Как правило, сети больших предприятий на платформе Windows обладают чрезвычайно разветвленным деревом катало га. Большое количество ветвей, а также наличие достаточно автономных площадок организации, включенных в общее дерево каталога, усложняют управление. Администрирование сети, ка талог которой состоит из десятков тысяч объектов, не может без опасно осуществляться одним или несколькими администратора ми, имеющими права доступа ко всем объектам.

В подобных случаях следует применять делегирование прав администрирования. Это чрезвычайно мощный инструмент, который в больших организациях позволяет более эффективно сконфигурировать систему безопасного администрирования. С его помощью управление отдельными областями сети смогут осуществлять специально назначенные ответственные лица – администраторы. При делегировании прав администрирования очень важно наделять ответственных лиц полномочиями, позволяющими выполнять функции администратора только в пределах их зоны ответственности, они не должны иметь возможность администрировать объекты каталога, находящиеся в других частях сети организации.

Права на создание новых пользователей или групп предоставляются на уровне подразделения или контейнера, в котором будут создаваться учетные записи. Администраторы групп одного подразделения могут не иметь прав на создание и управление учетными записями другого подразделения в том же домене. Однако, если права доступа и настройки политик получены на более высоком уровне дерева каталога, они могут распространяться вниз по дереву благодаря механизму наследования прав доступа.

С помощью инструментов управления Active Directory администратор может делегировать другим пользователям и группам право управления частью каталога. Это в полной мере относится и к объектам групповой политики, в отношении которых могут быть, в частности, делегированы следующие права:

– управление связями GPO с сайтом, доменом или под разделением (организационной единицей). Для этого с помощью инструмента управления Active Directory следует указать объект (сайт, домен или организационную единицу) и щелкнуть правой кнопкой мыши. В появившемся контекстном меню выбрать ко манду Делегирование управления (Delegate Control). Запустится Мастер делегирования управления (Delegation of Control Wizard). С его помощью можно выбрать объект групповой политики, группу или пользователя, которому должны быть делегированы права, а также и само право (в данном случае Управление ссылками групповой политики (Manage Group Policy links);

– создание и удаление всех дочерних объектов групповой политики. По умолчанию правом создания объектов в GPO обладают администраторы домена (Domain Admins) и администраторы предприятия (Enterprise Admins), а также операционная система. Для делегирования пользователю права управления объектами групповой политики домена необходимо включить его в группу «Создателивладельцы групповой политики» (Group Policy Creator Owners);

– редактирование свойств объектов групповой политики. По умолчанию правом редактирования GPO обладают администраторы домена, администраторы предприятия и операционная система. Для делегирования пользователю права редактирования объекта групповой политики необходимо включить его в одну из указанных групп безопасности.

Чтобы позволить группе или пользователю управлять некоторым подразделением (контейнером):

1. Запустите Active Directory – Пользователи и компьютеры.
2. Укажите подразделение, управление которым необходимо передать, и нажмите правую кнопку мыши. В появившемся меню выберите команду Делегировать

управление (Delegate control). Запустится Мастер делегирования управления (Delegation of Control Wizard). Нажмите кнопку Далее.

3. В следующем окне мастера нажмите кнопку Добавить и выберите пользователя или группу, которой вы хотите разрешить управление подразделением, нажмите кнопку ОК и затем кнопку Далее.

4. В открывшемся окне диалога мастера делегирования управления в окне со списком Делегировать следующие обычные задачи (Delegate the following common tasks) выберите одну или несколько операций, право выполнения которых делегируется указанному пользователю или группе. Если нужно делегировать право выполнения более специализированной задачи, установите переключатель Создать особую задачу для делегирования (Create a custom task to delegate). Нажмите кнопку Далее.

5. Если указана особая задача для делегирования в следующем окне, можно выбрать область применения для этой задачи: положение переключателя Этой папкой и существующими в ней объектами, созданием новых объектов в этой папке (This folder, existing objects in this folder, and creation of new objects in this folder), в этом случае вы передадите группе право на админи стрирование всего контейнера, или положение Только следую щими объектами в этой папке (Only the following objects in the folder) и установить флажки возле нужных объектов, в этом случае группа сможет управлять только выбранными объектами. За тем нажмите кнопку Далее.

6. В открывшемся окне определяются делегируемые разрешения. Можно отображать и устанавливать общие разрешения или разрешения для отдельных свойств или дочерних объектов. В пределах контейнера можно делегировать не все, а только некоторые права администрирования: например, можно делегировать только права на модификацию (чтение-запись) выбранного контейнера без дочерних объектов. Задайте нужные разрешения и нажмите кнопку Далее.

7. В следующем окне сводки выводится информация о вы бранных действиях. Можно вернуться назад и скорректировать параметры. Если все правильно, нажмите кнопку Готово.

Эффективное функционирование многопользовательской операционной системы невозможно без четкого разграничения доступа к ресурсам. Одним из средств, позволяющих настраивать параметры безопасной работы пользователей в сети в операционных системах семейства Windows (NT, 2000, XP и выше), являются политики безопасности.

Реализация политик безопасности в Windows предоставляет достаточно широкие возможности, в том числе настройку поли тики безопасности для всего дерева доменов. Установив политику безопасности в одном месте, администраторы могут контролировать безопасность всех рабочих станций домена. Политики безопасности в Windows реализуются с помощью средств групповых политик (group policy).

Групповая политика имеет следующие преимущества:

– основываясь на службе Active Directory системы Windows, позволяет как централизованно, так и децентрализовано управлять параметрами политики;

– обладает гибкостью и масштабируемостью. Может быть применена в широком наборе конфигураций системы, пред назначенных как для малого бизнеса, так и для больших корпораций;

– обладает высокой степенью надежности и безопасности;

– групповые политики расширяют и используют преимущества Active Directory. Их настройки находятся в объектах групповых политик (Group Policy Object, GPO), которые в свою очередь ассоциируются с такими контейнерами Active Directory, как сайты, домены и подразделения (организационные единицы).

Для запуска объекта Групповая политика следует выполнить следующие действия:

1) выбрать объект Active Directory, для которого необходимо установить групповую политику безопасности и правой кноп кой мыши вызвать контекстное меню;

2) выбрать элемент Свойства (Properties);

3) в диалоговом окне свойств выбрать вкладку Групповая политика (Group Policy);

4) для модификации глобальной политики следует выбрать кнопку Edit, если политика еще не была создана – New и ввести имя, либо воспользоваться тем, которое предлагает система.

Создать групповую политику для контейнера Active Directory можно только при наличии определенного набора условий. Необходимо иметь работающий контроллер домена Windows . Пользователь, который создает групповую политику, должен обладать правами на чтение и запись в системный том контроллеров домена (папка Sysvol). Кроме того, он должен иметь право модификации выбранного контейнера Active Directory.

После выбранных действий загружается корневой узел, представляющий собой GPO, присоединенный к определенному контейнеру.

Имя этого GPO и имя контейнера, к которому он присоединен, отображаются в окне структуры в следующем формате: Имя_политики [Имя_домена] Policy.

Затем пространство имен подразделяется на два узла более низкого уровня: «Конфигурация компьютера» (Computer Configuration) и «Конфигурация пользователя» (User Configuration). Используя их, можно создавать и настраивать групповые политики для компьютера и пользователей.

Узел «Конфигурация компьютера» содержит параметры всех политик, определяющих работу компьютера. Они регулируют функционирование операционной системы, вид рабочего сто ла, задают параметры выполняемых приложений, определяют работу средств обеспечения безопасности и т. д. Групповая политика применяется к рабочей станции домена на этапе загрузки системы и в дальнейшем при выполнении циклов обновления.

Узел «Конфигурация пользователя» содержит параметры всех политик, определяющих работу пользователя на компьютере. Они регулируют вид рабочего стола, как и в предыдущем случае, задают параметры выполняющихся приложений, определяют работу средств обеспечения безопасности и пользовательских сценариев входа и выхода. Групповая политика применяется к пользователю при его регистрации и в дальнейшем при выполнении циклов обновления.

Опишем некоторые расширения объекта «Групповая поли тика»:

– административные шаблоны. (Administrative Templates). Здесь находится групповая политика, определяющая параметры реестра, задающие работу и внешний вид рабочего стола, компонент операционной системы и приложений;

– параметры безопасности (Security Settings). Служит для настройки параметров системы безопасности компьютеров, на которые воздействует данный объект групповой политики. С помощью групповых политик можно настроить безопасность индивидуального компьютера, домена и целой сети;

– установка программ (Software Installation). Служит для централизованного управления программным обеспечением ор ганизации. С его помощью можно задавать различные режимы установки новых программ на компьютеры пользователей;

– сценарии (Scripts). Сценарии используются для автоматического выполнения набора команд при загрузке операционной системы и в процессе завершения ее работы, а также при реги страции и отключении пользователя от сети. Для выполнения сценариев, написанных на Microsoft JScript и Microsoft Visual Basic Scripting Edition, можно применять сервер сценариев (Windows Scripting Host);

– перенаправление папок (Folder Redirection). Позволяет перенаправлять обращение к специальным папкам в сеть.

С помощью расширения «Параметры безопасности» в GPO можно определить параметры политики безопасности, определяющие различные аспекты работы системы безопасности Windows. Созданная в объекте групповой политики конфигурация

воздействует на все компьютеры, находящиеся в контейнере, к которому присоединен данный GPO.

Расширение «Параметры безопасности» позволяет настраивать следующие аспекты системы безопасности компьютера:

– –политики учетных записей (Account Policies). Можно настраивать политики безопасности как учетных записей в масштабах домена, так и локальных учетных записей. Здесь определяются политика паролей, политика блокировки паролей и поли тика Kerberos, распространяющаяся на весь домен;

– локальные политики (Local Policies). Можно настраивать политику аудита, назначать права пользователей и различ ные параметры безопасности, доступные для настройки в системе Windows;

– журнал событий (Event Log). Можно настраивать политики безопасности, определяющие работу журналов событий приложений, системы и безопасности;

– группы с ограниченным доступом (Restricted Groups). Можно регулировать членство пользователей в специфических группах. Сюда обычно включают встроенные группы, такие как Администраторы, Операторы архива и другие, имеющие по умолчанию права администратора. В эту категорию могут быть включены и другие группы, безопасность которых требует особо го внимания и членство в которых должно регулироваться на уровне политики;

– системные службы (System Services). Можно настраивать безопасность и параметры загрузки для работающих на ком пьютере служб. В этом разделе могут быть использованы расши рения, с помощью которых можно осуществлять настройку без опасности, специфическую для данной службы. Например, рас ширение File Sharing Service позволяет настраивать политику безопасности для службы создания общего доступа к файлу (ограничение анонимного доступа к общим ресурсам, формирование безопасности различных сетевых общих ресурсов и т. д.);

– реестр (Registry). Можно настраивать безопасность различных разделов реестра;

– файловая система (File System). Можно настраивать безопасность определенных файлов;

– политики открытого ключа (Public Key Policies). Можно настраивать политики безопасности в отношении шифрования информации с помощью EFS, авторизации корневого сертифика та в масштабах домена, авторизации доверенного сертификата и т. д.;

– политики безопасности IP (IPSEC). Позволяет настраивать политику безопасности IP для компьютеров.

Политики безопасности, определяемые расширением «Параметры безопасности», действуют на компьютеры и частично на пользователей. Поскольку политика безопасности Windows значительно отличается от политик предыдущих версий Windows NT, при переходе к Windows низкоуровневые политики безопасности не переносятся. Если при переходе создается новое дерево доменов, одновременно создается и новая политика безопасности, назначаемая по умолчанию. Если при переходе домен присоединяется к уже существующему дереву, политика безопасности берется от родительского домена.

Для модификации настроек безопасности щелкните на папке

«Параметры безопасности», затем щелчками на соответствующих узлах откройте весь путь, ведущий к интересующим настройкам. В правом подокне окна «Групповая политика» двойным щелчком выберите настраиваемую политику и в открывшемся окне настройте ее.

Рассмотрим работу указанных расширений на конкретных примерах.

1. Настройка политики паролей.

Предположим, нам необходимо установить следующие правила политики паролей и блокировки:

– минимальная длина пароля – 8 символов;
– максимальный срок действия пароля – 30 дней;
– блокировать консоль после трех неудачных попыток входа.

Для реализации указанных правил выполним следующие действия:

1) откроем глобальную политику безопасности домена (см. выше) и расширение «Политики безопасности» (Security Settings);

2) выберем пункт «Политика учетных записей» (Account Policies), а затем политику паролей (Password Policy);

3) в правой части окна появится полный список правил, поддерживаемых политикой безопасности Windows;

4) найдем требуемые правила:

а) минимальная длина пароля (Minimum password length);

б) максимальный срок действия пароля (Maximum password age);

установим требуемые значения, вызвав соответствующие диалоговые окна двойным щелчком мыши на названии правила;

5) для установки параметра блокировки перейдем в раздел политики блокировки учетных записей (Account Lockout Policy), выберем необходимое правило в правой части окна – Account lockout threshold – и установим требуемое значение – три.

2. Политика учетных записей.

Предположим, нам необходимо разрешить всем пользователям домена использовать привилегию изменения системного времени. Для этого следует выполнить:

1) откроем глобальную политику безопасности домена (см. выше) и расширение «Политики безопасности» (Security Settings);

2) выберем пункт «Локальные политики» (Local Policies), а затем политику назначения прав пользователей (User Rights Assignment);

3) в правой части окна выберем требуемое правило – Change the system time и добавим пользователя Все (Evryone).

Применение групповых политик происходит в последовательности, соответствующей иерархии GPO: сначала объект групповой политики сайта, затем домена, затем GPO, связанные с подразделениями в соответствии с их вложенностью. Порядок выполнения групповых политик можно изменить с помощью настроек, блокирующих определенные групповые политики или заставляющих их выполняться принудительно. Кроме того, на порядок выполнения групповых политик влияет применение групп безопасности.

По умолчанию настройки групповой политики, применяемые к контейнеру определенного уровня, наследуются всеми контейнерами более низких уровней и находящимися внутри них пользователями и компьютерами. Если с дочерней организационной единицей (контейнером) связан свой GPO, он может устанавливать для нее индивидуальные настройки групповых поли тик, отменяющие применение к ней наследуемых настроек. Если некоторые настройки групповых политик родительского контейнера не заданы (not defined), то они не наследуются и дочерними контейнерами. Если родительский контейнер обладает сконфигурированными настройками групповых политик, которые не заданы в GPO дочернего контейнера, то такие настройки наследуются.

Наследование настроек групповых политик родительского контейнера дочерним контейнером, с которым связан собственный объект групповой политики, может иметь место только в случае совместимости этих групповых политик. Например, если политика родительского контейнера задает определенную конфигурацию рабочего стола компьютера пользователя, а политика дочернего контейнера дополняет ее, пользователь увидит на сво ем рабочем столе все элементы, заданные обеими политиками. Если же

групповая политика родительского контейнера противоречит групповой политике дочернего контейнера, выполняются только настройки GPO, связанного с дочерним контейнером.

Подобное положение вещей может быть изменено. Установка флажка Блокировать наследование политики (Block Policy inheritance), находящегося на вкладке Групповая политика окна свойств некоторого контейнера, запрещает наследование каких либо групповых политик, установленных для родительского контейнера.

Существует средство, позволяющее установить принуди тельное применение групповой политики, настроенной для некоторого контейнера, всеми контейнерами более низкого уровня. Для этого на вкладке Групповая политика окна свойств контейнера следует нажать кнопку Параметры (Options). В появившемся окне диалога Параметры необходимо установить флажок Не пе рекрывать (No override). В этом случае дочерние контейнеры бу дут наследовать (т. е. не смогут переопределить) все настройки родительского контейнера, даже в том случае, если для дочерних контейнеров установлен флажок Блокировать наследование политики.

По умолчанию групповая политика применяется синхронно, т. е. политики компьютера применяются до появления окна «Вход в Windows» (Log on to Windows), а политики пользователя

– до передачи операционной системой управления оболочке, интерактивно взаимодействующей с пользователем. Подобный по рядок можно изменить, однако делать это не рекомендуется, поскольку асинхронное применение групповых политик может привести к непредсказуемым и нежелательным результатам.

Применение групповых политик не ограничивается только, например, моментом загрузки операционной системы компьютера или регистрацией пользователя в системе. При работе компьютера в сети групповые политики могут измениться, поэтому они применяются периодически (по умолчанию – каждые 90 минут). Длительность периода применения политик можно изменять. Если задать его равным нулю, групповые политики применяются через каждые 7 секунд. Следует учитывать, что при уменьшении периода применения групповых политик значительно увеличивается нагрузка на систему. На контроллерах доменов период применения политик равен 5 минутам.

Настройки расширений Установка программ и Переназначение папки применяются только при загрузке операционной системы или регистрации пользователя в системе, поскольку пери одическое применение этих групповых политик может вызвать нежелательные результаты.

Практические задания

Задание

Данное практическое занятие предполагает выполнение следующих этапов:

1. Создать новую организационную единицу (имя выбрать произвольно, например, my_unit).
2. Создать новую группу.
3. Создать в организационной единице трех новых пользователей: для всех потребовать смену пароля при входе и ограничить срок действия пароля. Одного из пользователей включить в новую группу.
4. Делегировать права на созданную организационную единицу пользователю из новой группы.
5. Установить максимальный срок действия пароля – 30 дней.
6. При вводе нового пароля требовать его неповторяемость. Хранить в системе 2 предыдущих пароля.
7. Установить минимальную длину пароля – 10 символов.
8. Установить аудит успеха для событий входа в систему.
9. Назначить возможность выключения системы только для администраторов.

10. Разрешить вход в систему только для членов группы «Администраторы» и определенного пользователя.

11. Разрешить доступ к компьютеру из сети только для определенного пользователя.

12. Блокировать консоль пользователя после ввода двух неверных паролей на 5 минут.

13. Отображать последнее имя пользователя при диалоге входа в систему.

14. Разрешить определенному пользователю изменять политику аудита системы.

Контрольные вопросы:

1. Назначение Active Directory и основные возможности.
2. Какова структура Active Directory?
3. Для чего используются организационные единицы, когда и с какой целью их следует создавать?
4. Какие группы пользователей операционная система создает по умолчанию?
5. Может ли один и тот же пользовать входить в разные группы?
6. В каких случаях следует использовать делегирование прав?
7. Какие права могут быть делегированы?
8. Пользователям, каких групп можно делегировать права?
9. Какие виды политик безопасности поддерживаются, сферы их применения?
10. Какие параметры безопасности можно настроить в глобальной политике безопасности?
11. Как взаимодействуют между собой глобальная и локальная политики безопасности?
12. Какие правила наследования политик безопасности поддерживаются?

Лабораторная работа№16 Создание резервных копий базы данных

Цель занятия: ознакомиться с основными конструкциями SQL, технологиями среды MS SQL Server Management, объектами SMO (среды MS Visual Studio) для резервного копирования и восстановления БД.

Краткие теоретические сведения

Предотвращение потерь данных – одна из самых важных проблем, с которой можно столкнуться при управлении системами баз данных. Потери данных могут иметь место в результате множества самых различных проблем:

– неисправности аппаратного обеспечения;
– вирусы;
– некорректное использование инструкций UPDATE и DELETE;
– ошибки программного обеспечения;
– аварийные ситуации, например, пожар или затопление. Чтобы избежать потери данных, можно реализовать для базы данных стратегию восстановления. Стратегию восстановления необходимо спланировать, реализовать и протестировать с учетом возможных неисправностей, с которыми можно встретиться в процессе работы системы, и необходимого уровня защиты данных. В витринах данных, то есть в случаях, когда данные можно восстановить из других систем, вероятно, нет необходимости создавать резервные копии каждой отдельной транзакции. Возможно, будет достаточно выполнять полное резервное копирование данных с регулярными временными интервалами. И, наоборот, для базы данных, в которой хранятся транзакции интернет- магазина, возможно, будет необходимо сохранять резервные копии каждой отдельной транзакции. СУБД SQL Server предоставляет полный комплекс функций для реализации именно того вида резервного копирования, который вам необходим. В данной лекции рассматриваются наиболее широко используемые в Microsoft SQL Server стратегии для защиты данных.

2.2.1 Полное резервное копирование базы данных

Самой распространенной стратегией резервного копирования является резервное копирование всей базы данных через заранее заданные промежутки времени (например, каждую ночь). Благодаря такой стратегии аварийного восстановления можно восстановить базу данных до состояния на момент выполнения последнего резервного копирования. Эта стратегия реализуется посредством выполнения полных резервных копий базы данных, как рассказывается ниже.

Полная резервная копия базы данных содержит все данные и метаинформацию базы данных, которые необходимы для восстановления базы данных полностью, включая полнотекстовые каталоги. При восстановлении базы данных из полной резервной копии восстанавливаются все файлы базы данных, причем данные извлекаются в непротиворечивом состоянии на тот момент времени, в который выполнялось резервное копирование. Пока выполняется резервное копирование, база данных работает в рабочем режиме, и пользователь может выполнять транзакции, изменяя данные обычным путем. Термин "непротиворечивое состояние" означает, что все транзакции, которые были зафиксированы в процессе выполнения резервного копирования базы данных, применяются, а все транзакции, которые не были завершены, подвергаются откату. Для ситуаций, которые могли бы привести к нарушению непротиворечивости данных вследствие выполнения транзакций, изменяющих данные в процессе выполнения резервного копирования, в SQL Server есть особый процесс, который позволяет гарантировать непротиворечивость данных. Этот процесс выполняет запись на устройство резервного копирования как страниц данных, так и журнала транзакций.

Полнотекстовые каталоги были введены в базы данных, чтобы добавить в SQL Server функции полнотекстового индексирования. Полнотекстовое индексирование позволяет быстрее и с большей точностью осуществлять поиск данных в базе данных. Дополнительную информацию о полнотекстовом индексировании см. в Электронной документации по SQL Server в разделе «Полнотекстовые индексы».

Скорость резервного копирования определяется скоростью используемых устройств ввода/вывода (тех устройств ввода вывода, которые используются для сбора и хранения информации). Чтобы добиться наилучшей производительности, SQLServer считывает файлы последовательно. Если ваши устройства ввода/вывода способны одновременно обрабатывать данные ввода/вывода резервного копирования и данные ввода/вывода, поступающие в результате обычного использования системы, то создание резервной копии окажет на производительность системы незначительное воздействие. Тем не менее, лучше выполнять полное резервное копирование базы данных при отсутствии пиковых нагрузок.

Практические задания

Задание.

Данное практическое занятие предполагает выполнение следующих этапов:

1. Необходимо создать резервные копии базы данных «MMM» с использованием полного резервного копирования, разностного резервного копирования и резервного копирования журнала транзакций.

Ход работы:

Запустите SQL Server Management Studio (SSMS), подключитесь к своему экземпляру SQL Server, используя технологию 1.

Создайте папку с именем c:\Student\ВашаПапка\test.

Откройте окно нового запроса. Измените контекст на базу данных master, используя технологию 6. Наберите и исполните следующую команду, чтобы создать полную резервную копию базы данных:

BACKUP DATABASE MMM TO DISK = 'C:\ TEST\AW.BAK'

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

Внесите изменение в таблицу «Модель» базы данных MMM. Добавьте одну запись (придумайте сами).

Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать резервную копию журнала транзакций и сохранить только что внесенное изменение:

BACKUP LOG MMM TO DISK = 'C:\ TEST\AW1.TRN'

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

Внесите еще одно изменение в таблицу «Модель».

Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать разностную резервную копию базы данных:

BACKUP DATABASE MMM TO DISK = 'C:\.....\TEST\AWDIFF1.BAK' WITH DIFFERENTIAL

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

Внесите еще одно изменение в таблицу «Модель».

Откройте окно нового запроса наберите и исполните следующую команду, чтобы создать полную резервную копию базы данных в указанном месте на диске:

BACKUP LOG MMM TO DISK = 'C:\ TEST\AW2.TRN'

Ознакомьтесь с результатами запроса – какая информация обработана, сколько страниц, сколько файлов.

Контрольные вопросы:

1. В каких случаях могут иметь место потери данных?
2. В чем заключается полное резервное копирование базы данных?
3. В чем заключается простая модель восстановления?
4. Виды устройств резервного копирования.

5. Вы выполняете разностное резервное копирование базы данных AdventureWorks каждые четыре часа, начиная с 04:00. полная резервная копия создается в полночь.

Лабораторная работа№17 Восстановление базы данных

Цель занятия: ознакомиться с основными конструкциями SQL, технологиями среды MS SQL Server Management, объектами SMO (среды MS Visual Studio) для резервного копирования и восстановления БД.

Краткие теоретические сведения

2.2.2 Простая модель восстановления

Следует заранее уведомить SQL Server о том, какой тип резервного копирования вы намерены использовать, поэтому надо сконфигурировать базу данных так, чтобы настройки соответствовали выбранному вами типу резервного копирования. Такая настройка выполняется посредством выбора значения параметра «модель восстановления базы данных». Модель восстановления базы данных, которая используется по умолчанию, является производным от модели восстановления модели базы данных, определенной при ее создании. Чтобы реализовать стратегию резервного копирования, которая будет включать только полные резервные копии, следует выбрать простую модель восстановления (SIMPLE).

Выбираем модель восстановления SIMPLE

В меню Start (Пуск) выберите All Programs,. Microsoft SQL Server 2005, SQL Server Management Studio (Все программы, Mi- crosoft SQL Server 2005, Среда SQL Server Management Studio).

В диалоговом окне Connect To Server (Соединение с сервером) нажмите кнопку Connect (Соединить).

В панели инструментов Standard (Стандартная) нажмите кнопку New Query (Новый запрос), чтобы открыть окно New Query (Новый запрос).

Чтобы задать модель восстановления, можно использовать инструкцию ALTER DATABASE. Введите текст следующей инструкции и нажмите кнопку Execute (Выполнить).

```
USE master;
GO
ALTER DATABASE AdventureWorks SET RECOVERY SIMPLE;
GO
```

Проверяем настройки модели аварийного восстановления. Чтобы просмотреть заданную для базы данных модель восстановления, можно использовать функцию DATA BASE PROPERTYEX, которая извлекает параметры текущей базы даты или свойства указанной базы данных. Выполните инструкцию, приведенную ниже, чтобы извлечь информацию о модели восстановления базы данных AdventureWorks.

```
SELECT DATA BASE PROPERTYEX ('AdventureWorks', 'Recovery')
```

Убедитесь, что в результатах запроса указана модель восстановления SIMPLE.

Закройте окно среды SQL Server Management Studio.

2.2.3 Устройства резервного копирования

До начала выполнения операций резервного копирования необходимо определить, где будут храниться резервные копии. Место хранения резервных копий называется устройством резервного копирования. Каждое устройство резервного копирования может хранить несколько резервных копий разных типов. Существует два разных вида устройств резервного копирования:

Ленточные устройства. Могут использоваться для хранения резервных копий на лентах. Ленточные устройства должны быть установлены локально. Резервная копия может занимать несколько лент, а на одной ленте могут находиться одновременно резервные копии SQL Server и Windows.

Дисковые устройства. Файлы на локальном или удаленном диске, или дисковом накопителе. К этим файлам обращаются, указывая путь к файлу, в котором хранится резервная копия. Для обращения к удаленным хранилищам следует использовать путь в формате UNC.

Резервное копирование файлов SQL Server на ленточные устройства в настоящее время используется не очень часто. Если резервные копии SQL Server сохраняются на лентах, то они обычно создаются при помощи программ сторонних разработчиков, которые предлагают дополнительные функции, например, использование удаленного ленточного хранилища. В качестве альтернативы ленточное устройство может использоваться для дополнительного страхования сохранности данных, резервная копия которых уже сохранена на дисковом устройстве.

Устройства резервного копирования идентифицируются по имени устройства. В качестве имени устройства может использоваться имя логического или физического устройства. Имя физического дискового устройства представляет собой путь к файлу резервной копии, например, «\\BACKUPSERVER \Backups\ adv\ AdventureWorks.bak». Этот путь можно включить непосредственно в инструкцию резервного копирования. Имя логического устройства представляет собой имя, указывающее на имя физического устройства резервного копирования и хранящееся в SQL Server. Когда в инструкции резервного копирования используется имя логического устройства, SQL Server осуществляет поиск соответствующего физического устройства в системном каталоге и выполняет резервное копирование, сохраняя резервную копию в указанной папке.

Чтобы добавить в системный каталог логическое устройство, можно использовать хранимую процедуру sp_addumpdevice. В следующем примере определяется логическое устройство с именем Adv_FullDb_Dev.

EXEC sp addumpdevice 'disk', 'AdvFullDbDev', 'T:\BACKUPS\ AdvFullDbDev.bak';

Обязательно измените, путь к файлу, чтобы он соответствовал вашему компьютеру. T:/, то измените эту часть пути к файлу в инструкции так, чтобы он соответствовал букве диска на вашем компьютере. Кроме того, убедитесь в том, что все папки, заданные в этом пути, существуют на вашем компьютере.

Имена логических и физических устройств являются взаимозаменяемыми, для резервного копирования и восстановления базы данных могут использоваться оба имени. Конечно, как правило, лучше все время использовать одно из двух соглашений о назначении имен, чтобы не усложнять код. Следует заранее выбрать соглашение, которое вам больше нравится.

Никогда не следует выполнять резервное копирование на дисковое устройство, которое размещается на том же физическом устройстве хранения, что и сама база данных. Даже если дисковое хранилище отличается устойчивостью против сбоев благодаря наличию RAID, всегда существует возможность возникновения неисправности контроллера и повреждения данных на дисках. Кроме того, следует подумать о сохранении файлов резервной копии устройства резервного копирования на лентах и хранении этих лент в удаленном месте.

Практические задания

Задание

1. Необходимо провести восстановление базы данных «МММ» из сделанных в задании №1 резервных копий.

Ход работы:

Если необходимо, запустите SSMS, подключитесь к своему экземпляру SQL Server, используя технологию 1.

Выполните восстановление БД из первой полной резервной копии (C:\...TEST\AW.BAK) средствами оболочки SSMS. Для этого выполните:

В обозревателе объектов вызовите контекстное меню на вашей БД и выберите задачу восстановления базы данных (рисунок 2.1).

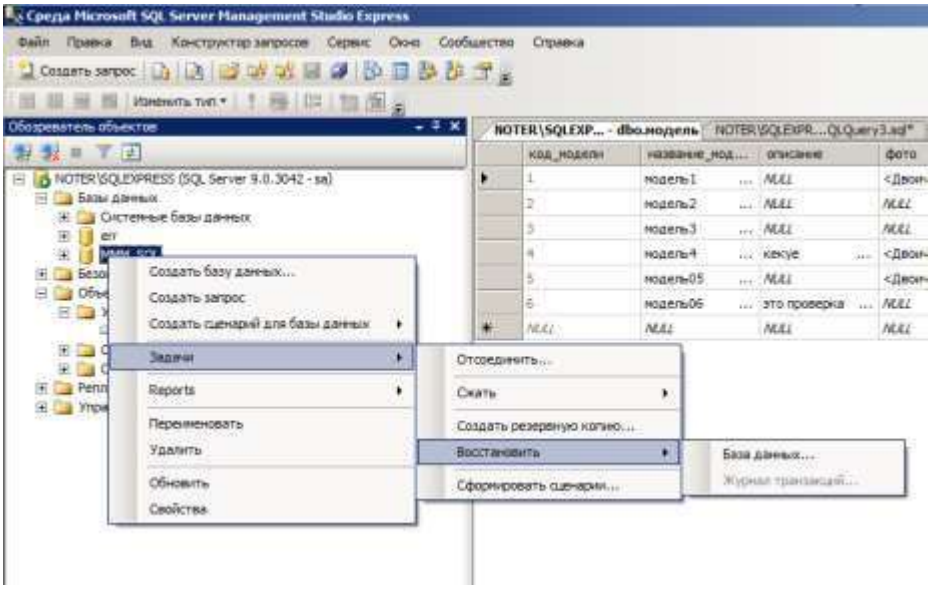


Рисунок 2.1

В открывшемся окне необходимо задать параметры восстановления.

На закладке «Общие» необходимо выбрать:

— базу данных для восстановления (вашу МММ);

— выбрать источник набора данных для восстановления с устройства файл C:\...TEST\AW.BAK.

После определения файла-источника данных необходимо флажком выбрать базу данных для восстановления (рисунок 2.2).

Рисунок 2.2

На закладке «Параметры» необходимо включить опцию «Перезаписать БД» и «оставить БД готовой к использованию»,(рисунок 2.3).

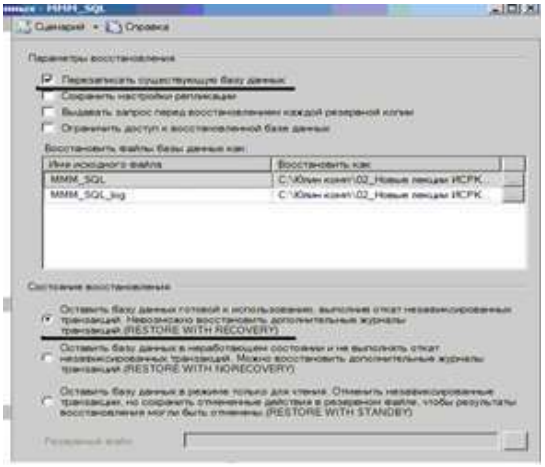


Рисунок 2.3

После восстановления БД, откройте таблицу «Модель» и убедитесь, что она не содержит всех добавлений, вносимых вами в процессе выполнения упражнения, так как восстановление происходило из первой резервной копии (без изменений).

2. Необходимо организовывать со стороны клиентского приложения, созданного

в Visual Studio удаленное администрирование БД (резервное копирование).

Ход работы:

Создайте новый проект Windows Application и сохраните его в своей папке под именем Лабы_МММ_семестр.

В главную форму добавьте меню, изображенное на рисунке 2.4.

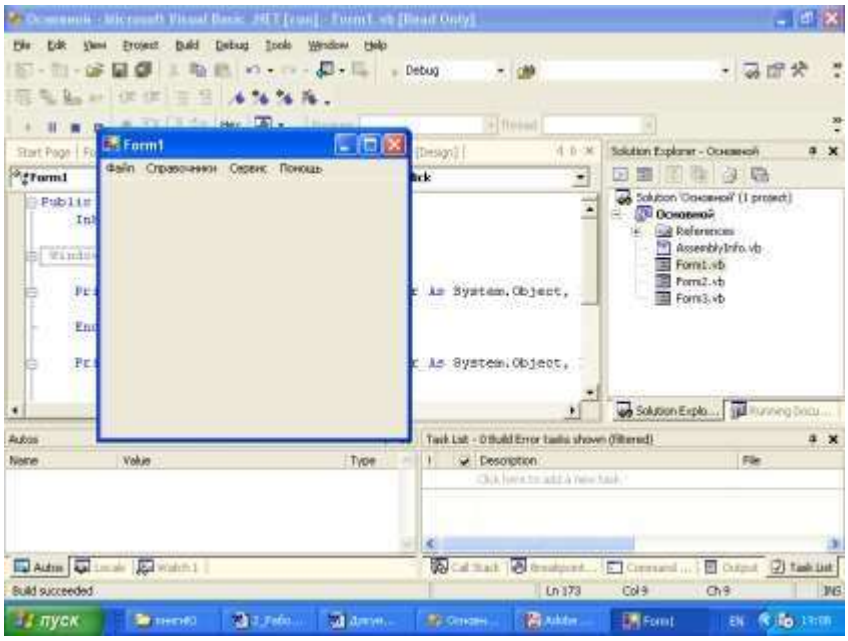


Рисунок 2.4

Добавьте на только что созданную форму компоненты в соответствии с рисунком 2.5.

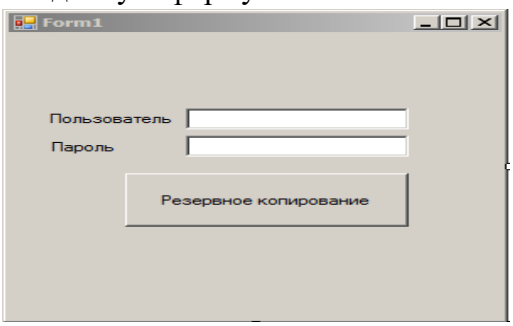


Рисунок 2.5

Обеспечьте функциональную работу формы (напишите обработчик кнопки «Резервное копирование» с использованием объектов SMO.

Добавьте возможность открытия данной формы при выборе в главной форме пункта меню Администрирование БД Резервное копирование.

Запустите проект, проверьте работу формы.Закройте проект

Убедитесь в появлении файла резервной копии на диске (файл, который указан в тексте программы).

Откройте SSMS. Добавьте в таблицу «Модель» новую строку данных.

Средствами оболочки SSMS, выполните восстановление БД из резервной копии, созданной вашей программой

Убедитесь, что после восстановления добавленных строк втаблице «Модель» нет.

Контрольные вопросы:

1.Какие данные будут содержаться в разностной резервной копии, сделанной в полдень?

а) страницы данных, измененные после полуночи;

б) экстенты, измененные после полуночи;

в) страницы данных, измененные после 08:00;

г) экстенты, измененные после 08:00.

3.Вы выполняете полное резервное копирование базы данных AdventureWorks, которое завершается в полночь. Разностное резервное копирование выполняется по расписанию каждые четыре часа, начиная с 04:00. Резервное копирование журнала транзакций происходит по расписанию каждые пять минут.

4.Какую информацию будет содержать резервная копия журнала транзакций, созданная в 09:15?

а) все транзакции, начатые после 09:10;

б) транзакции, завершенные после 09:10;

в) страницы, измененные после 09:10;

г) экстенты, измененные после 09:10.

Лабораторная работа№18 Восстановление носителей информации

Цель занятия: научиться осуществлять восстановление жесткого диска после сбоев.

Краткие теоретические сведения

На сегодняшний день жёсткие диски занимают доминирующее место на рынке накопителей информации. К плюсам жёстких дисков можно отнести низкую стоимость за Гбайт памяти и практичность в использовании. Поэтому возникает необходимость в своевременном обслуживании, тестировании и выявлении критического состояния жесткого диска.

В состав утилит современной операционной системы, входят программы, позволяющие осуществлять дефрагментацию и очистку жесткого диска. Для выполнения программ необходимо выполнить команду Пуск/Стандартные/Служебные и из появившегося списка программ выбрать нужную.

Кроме того, современные накопители имеют систему оперативного наблюдения за своим состоянием – S.M.A.R.T. (Self-Monitoring, Analysis And Reporting Technology) – технология самодиагностики, анализа и отчета. Это набор программ, вшитых в ПЗУ диска.

Данная технология позволяет в любое время оценить такие важные параметры накопителя, как:

– количество отработанных часов, число возникших в процессе чтения/записи ошибок, температуру накопителя

– среднюю производительность, количество циклов запуска/останова шпинделя, время раскрутки шпинделя,

– количество переназначенных секторов, количество ошибок позиционирования головок и т. д.

Технология позволяет предсказать возможный выход из строя накопителя.

Исходя из огромной важности корректной работы жесткого диска, существует большое количество программ, позволяющих восстанавливать удаленные файлы с диска, файловую систему, критически важные структуры жесткого диска, такие как главная загрузочная запись, таблица разделов и т. д.

1. Partition Magic

Power Quest @ Partition Magic – это утилита, которая позволяет быстро и легко создавать, удалять, объединять или преобразовывать файловые системы и разделы на жестком диске, не уничтожая существующие данные. Новый инструмент кластерного анализа исследует FAT дисководы и рекомендует подходящий размер кластера. Кроме того, есть возможность создавать, перемещать и изменять размер разделов типа FAT, FAT 32, файловой системы Windows NT (Windows NT File System, NTFS), HPFS (High-Performance File System – высокопроизводительная файловая система).

Partition Magic помогает надежно устанавливать и использовать несколько операционных систем на одном жестком диске. Partition Magic включает в себя Boot Magic – мощный администратор загрузки, который помогает безопасно устанавливать новые операционные системы и позволяет выбирать через меню систему при загрузке компьютера.

Программа имеет наглядный доброжелательный интерфейс. В версии Partition Magic 8.0 включена новая утилита – Power

Quest Data Keeper. Она поможет защитить ценные данные на диске от системных сбоев, упростить процесс копирования и пересылки в пределах системы, восстановить удаленный файл.

В процессе установки программы можно сделать две загрузочные дискеты – на одной будет DOS от Caldera, а на другой – Partition Magic for DOS. С помощью этих дискет можно подготовить новый диск к работе с нуля, т.к. программа наряду с организацией разделов выполняет и их форматирование, причем эти процедуры выполняются намного быстрее, чем при использовании традиционных программ.

Прежде чем начать работу с программой Partition Magic обязательно нужно выполнить следующие рекомендации:

– Установить самые последние обновления для операционных систем Windows 95/98/Me/NT Workstation/2000/XP Professional. Удостовериться, что самые последние исправления для операционных систем Windows 95/98/Me/NT Workstation/2000/XP Professional установлены и запущены.

– Сделать копию вашего жесткого диска. Данные на диске – самая ценная часть компьютера. Хоть это и маловероятно, чтобы Partition Magic повредил бы данные, но влияние других ошибок типа системных отказов аппаратных средств, программного обеспечения, или питания, могут привести к повреждению данных в момент выполнения программы PartitionMagic. Используя программу Power Quest's Drive Image, можно создать резервную копию раздела, который будет изменяться. Можно также использовать эту программу и для полного восстановления раздела к первоначальному состоянию.

– Создать загрузочный диск Windows. Загрузочный диск позволит загрузить Windows при возникновении проблемы.

– Запустить опцию проверки ошибок на диске. Для раздела, который будет проверяться, нажать Partition – Check for Errors. Небольшие ошибки могут быть исправлены Partition Magic, однако более серьезные ошибки прекратят выполнение программы. Проверить и исправить обычные ошибки на диске. Проверка загрузочного раздела операционной системы Windows невозможно, так как есть всегда открытые файлы. Для этого раздела, можно воспользоваться Partition > MS ScanDisk.

– Закрыть все запущенные приложения. Нельзя запускать Partition Magic вместе с другими приложениями, включая вирусные сканеры. Если осуществляется работа в сети под управлением Windows NT, перед выполнением Partition Magic, необходимо удостовериться что другие пользователи, не подключены к вашему компьютеру.

– Использовать UPS (Источник бесперебойного питания). Partition Magic не способна восстановить данные, если в процессе разделения диска происходит сбой питания. Используя источник бесперебойного питания (UPS) можно избежать проблем, вызванных сбоем питания.

Из-за несовместимости аппаратной и системной конфигурации одного компьютера с другим, не рекомендуется переносить с одного на другой компьютер, жесткий диск, разделенный с помощью программы Partition Magic, во избежание потери данных.

Программа Partition Magic проверяет целостность диска сложной системой анализа и проверки достоверности, которая скрыто начинает свою работу, каждый раз, когда запускается программа или завершается операция. Первоначальная проверка на целостность диска, сообщает о любых проблемах, связанных с разделами, которые могут препятствовать нормальной работе программы Partition Magic. Проверка целостности действует как ранняя система предупреждения, которая сообщит о том, что структура диска полностью проверена и проанализирована еще до изменения.

Если физический диск проходит первоначальную проверку целостности диска, то появляется таблица разделов, и вы можете начинать работу с программой. В случае появления сообщения об ошибке вместо таблицы разделов, указывается проблема с жестким диском, а не с программой Partition Magic (так как никакие изменения с диском еще не проводились).

Необходимо исправить проблему с жестким диском и пере- запустить Partition Magic . Для получения дополнительной информации можно воспользоваться кнопкой помощи на панели инструментов.

В дополнение проверки целостности при запуске программы, Partition Magic выполняет еще две проверки в течение любой операции. До операции разделения диска проверяется файловая система (наподобие CHKDSK или MS ScanDisk), после проверяется целостность данных. Partition Magic анализирует диск и немедленно сообщает о найденных ошибках.

Интерфейс программы Partition Magic состоит из панели действия, строки меню, инструментальной панели, карты жестких дисков, списка разделов, кнопок мастера и строки легенда. Можете показать или скрыть, а также установить размеры для различных частей интерфейса. Выполнить настройку главного окна программы любым удобным способом для различных частей интерфейса. Если выбранный жесткий диск содержит логические разделы, то они показываются внутри расширенного раздела.

Строка меню и Панель инструментов (Menu Bar and Toolbar). В главном окне программы Partition Magic, строка меню и панель инструментов находятся наверху окна. Строка меню дает возможность доступа к любой из настроек Partition Magic, в то время как панель инструментов обеспечивает доступ к обычно используемым вариантам. Можно скрыть панель инструментов, что увеличит видимую область главного окна. Опция «Disks» на строке меню будет видна, только если у установлен второй жесткий диск.

Информация о разделах жесткого диска (Partition Information). Информационная область окна, отображает все данные для выбранного жесткого диска. Информация представлена в виде панели задач, карты диска и списка разделов.

Панель задач (Action Panel). Панель задач позволяет выбрать задачу, а также увидеть текущие незаконченные операции разделения диска.

Карта диска (Disk Map). На карте очень наглядно изображены разделы диска, с возможностью масштабировать. (Для масштабирования нажимать View – Scale Disk Map).

Каждый раздел на карте обозначается цветом (согласно легенде), которая приведена внизу окна. Освобожденное место на карте диска обозначается блоком темно – серого цвета.

Если у вас имеется второй жесткий диск то, возможно вы должны передвинуть карту что бы увидеть всю доступную информацию. Вы можете переместить карту дисков вверх или вниз, для более удобного просмотра.

Список разделов (Partition List) выводит информацию о каждом разделе на вашем жестком диске, конкретно это: имя диска, метки, тип файловой системы, размер в мегабайтах, количество используемого и неиспользуемого пространства в мегабайтах, состоянии, и является раздел первичным или логическим.

Разделы диска обозначаются названием тома, буквой с двоеточием. Звездочка заменяет букву в том случае, если раздел является:

– скрытым разделом;
– расширенным разделом;
– разделом с файловой системой, которая не поддерживается активной операционной системой;
– высвобожденным пространством. Состояние раздела, может быть:
– Активным (Active): Раздел диска, с которого загружается компьютер.
– Скрытым (Hidden): К разделу, который не имеет букву диска, нельзя обратиться из текущей операционной системы. Разделы диска могут быть скрыты операционной системой (возможно, скрыть все первичные разделы кроме активного) или вы можете использовать Partition Magic чтобы самостоятельно скрыть нужный вам раздел. В среде Windows 2000/XP, скрытые разделы могут иметь имя.
– Никакой (None): Разделы, которые ни активны, ни скрыты.

Легенда (Legend) – это цветовые обозначения различных файловых систем, которые должны помочь пользователям понять цвета, которые используются в панели задач, карте диска, списка разделов. Можно скрыть строку легенды, что увеличит видимую область главного окна.

Можно выполнить задачу двумя различными способами. Первый способ – использовать мастер программы Partition Magic из опускающего меню панели задач. Второй способ – это сделать вручную. Чтобы выполнить задачу вручную надо:

1. Выбрать жесткий диск или раздел.
2. Выбрать задачу (operation).

3. Применить выбранные задачи к вашей системе

Выбор жесткого диска и раздела. Можете выделить раздел сразу, не выделяя первый жесткий диск. Для этого необходимо нажать на выбранном разделе на карте диска или выбрать его из списка в главном окне. Есть две задачи, которые всегда могут быть выполнены: удалить все разделы и вывести подробную информацию о жестком диске. Когда выделяется жесткий диск, его разделы отображаются в списке разделов главного окна.

Выбор задачи (Selecting an operation). После того как были выбраны диск и раздел, используя строку меню или панель задач, выбрать операции. Есть несколько вариантов выполнения выбранной операции, для этого надо:

– В строке меню нажать Partition, затем нужную операцию. Справка советует этот метод как предпочтительный.

– На панели инструментов выбрать нужную операцию и нажмите <Enter>.

– На карте диска или в списке, выбрать раздел и щелкнуть на нем правой клавишей, затем выбрать нужную операцию.

Если операция недоступна, значить она не может быть применена к данному разделу.

Partition Magic начинает выполнять немедленно операции по сбору информации, проверки на ошибки, MS ScanDisk. Остальные операции помещаются в очередь в диалоговом окне Текущие действия (Operations Pending) и ожидают нажатия кнопки Применить (Apply).

Можно в любой момент отменить последнюю операцию, которые помещаются в очередь в диалоговом окне Текущие действия (Operations Pending).

Есть несколько вариантов выполнения выбранного действия, для этого надо:

– Нажать General > Undo Last Change.

– На панели задач нажмите кнопку Undo (отмена) которая находиться внизу панели задач.

– Нажмите Click View > Operations Pending – Undo Last.

– На панели инструментов нажмите кнопку Undo (отмена).

– Нажмите клавиши (Ctrl+Z).

Чтобы отменить все операции сразу, которые помещены в очередь в диалоговом окне Текущие действия (Operations Pending) надо:

– Нажать General – Discard All Changes.

– Нажать View – Operations Pending – Discard All.

– Нажать клавиши <Ctrl+D>.

Сделанные изменения отображаются на карте диска, в списке разделов. Однако реальные изменения будут произведены только после нажатия кнопки Apply (применить). Если кнопка на панели задач активна, а текущие операции находятся в ожидании, значит, изменения еще не были произведены.

Для применения выбранных операций надо:

– Нажать General – Apply Changes.

– Нажать кнопку Apply на панели задач главного окна программы PartitionMagic.

– Нажать кнопку Apply на панели инструментов.

– Нажать клавиши <Ctrl+A>.

Индикатор движения процесса может не двигаться в течение нескольких минут.

Изменение настроек (preferences) Partition Magic:

1. Нажать General – Preferences.

2. Поставить галочку напротив надписи «Allow 64K FAT Clusters for Windows NT/2000/XP». Установка этой опции позволит создать файловую систему FAT с размером кластера равного 64К, а также позволит программе Partition Magic создавать FAT разделы размером до 4Гб. Но операционные системы Dos, Windows 3x/95/98/Me не поддерживают

размеры кластеров больше 32К. Поэтому нельзя получить доступ к разделу с размером кластера 64К, используя эти операционные системы.

3. Установка галочки в маленьком окне с надписью «Skip bad sector checks» позволит пропустить проверку жесткого диска на сбойные секторы. Однако если жесткий диск имеет сбойные секторы, то возможна потеря всех данных, поэтому не рекомендуется ее включать (по умолчанию отключена).

4. Установка галочки в маленьком окне с надписью «Set as Read-Only for Partition Magic» не позволит программе произвести какие-либо изменения с жестким диском.

Если установить галочку как в шаге 2, то опция «размер кластера 64К» становится доступной в задачах Изменение/Перемещение раздела (Resize/Move Partition), и в диалоговых окнах Изменения размера кластера (Resize Clusters). Если использовать разные операционные системы, то не рекомендуется использовать кластеры размером 64К. В процессе разделения программа Partition Magic выполняет проверку на сбойные секторы. Дисковые интерфейсы IDE и SCSI устроены так, что часто обрабатывают сбойные секторы внутри, делая излишним дополнительную проверку. Partition Magic позволяет отключать проверку на сбойные секторы. Если проверка отключена, то все операции выполняются гораздо быстрее. Если установлено два диска, то можно отключить изменение одного из них, как в шаге 4. В шаге 4 есть, исключения даже если выбрана эта опция, то некоторые загрузочные файлы Windows NT все равно могут быть изменены.

2. PARAGON PARTITION MANAGER

Функции программы во многом совпадают с возможностями предыдущей программы – любые разделы можно создавать, удалять, форматировать, перемещать, конвертировать между файловыми системами, объединять и изменять их атрибуты, уменьшать или увеличивать размер разделов – и все это без потери данных. Кроме того, программа от отечественных производителей. Программа способна работать практически с любыми накопителями – жесткими дисками (PATA/SATA/SCSI) с неограниченным объемом, внешними жесткими дисками (USB/FireWire), Zip, Jazz и Flash устройствами.

3. ACRONIS DISK DIRECTOR SUITE

Еще одна отечественная разработка. Возможности утилиты по редактированию разделов дублируют функциональность предыдущих. Дополнительно в комплект входит утилита Acronis Disk Editor, благодаря которой можно вручную редактировать огромное количество параметров жесткого диска и содержащихся на нем разделов. В частности, можно править таблицу разделов, загрузочные секторы FAT и NTFS, настройки FAT и даже все данные, хранящиеся на накопителе (в шестнадцатеричном виде).

4. ACRONIS RECOVERY EXPERT

Нередко проблемы потери данных выходят за рамки гибели пары файлов, порой случается и так, что бесследно исчезают и целые разделы. Список причин, в результате которых может случиться подобная неприятность, довольно обширен – простая невнимательность или неосторожность пользователя, сбой в работе жесткого диска, проказы вируса, ошибка в исполняемой программе, скачок напряжения в сети и многое другое. Помочь может эта программа. Сначала она сканирует неразмеченную область диска на предмет нахождения пропавших разделов, затем удостоверяется у пользователя, что конкретно надо восстановить, после чего приступает к окончательной процедуре восстановления. Программа понимает большинство распространенных файловых систем. Утилита распространяется в составе предыдущей программы.

5. PARTITION TABLE DOCTOR

Одна из самых распространенных неприятностей – это частичное повреждение главной загрузочной записи (Master Boot Record), таблицы разделов (Partition Table) или загрузочных секторов (Boot Sectors), в результате чего система может вообще отказаться запускаться. Справиться с этими проблемами, и поможет данная программа. Помимо непосредственного лечения с помощью утилиты можно сделать резервную копию таблицы

разделов и загрузочных секторов. Программа может создать загрузочную дискету или CD со своим полнофункциональным модулем.

6. PARAGON MOUNT EVERYTHING

В последнее время все большую популярность набирают файловые системы NTFS, Ext2, Ext3. Но далеко не у всех установлены ОС, поддерживающие эти системы. Поэтому возникают проблемы совместимости при появлении в системе нового накопителя с другой файловой системой. Данная программа позволяет решить эти проблемы: моментально подключает разделы NTFS, Ext2, Ext3 в любой версии Windows, после чего работа с ними никак не будет отличаться от использования стандартных разделов FAT. Подключенным разделам присваивается буква, на них можно копировать, открывать, редактировать любые файлы и даже запускать приложения. Утилита может управлять разделами

– создавать, удалять и форматировать. Можно создать загрузочную DOS дискету с возможность доступа к NTFS.

Практические задания

Данное практическое занятие предполагает выполнение следующих этапов:

– изучить методические указания;

– ответить на контрольные вопросы.

Контрольные вопросы:

1. В чем назначение программы Partition Magic?
2. Какие действия необходимо выполнить перед началом работы с программой Partition Magic?
3. Как осуществляется проверка целостности жесткого диска с помощью программы Partition Magic?

Лабораторная работа№19 Восстановление удаленных файлов.

Цель занятия: научиться осуществлять восстановление жесткого диска после сбоев.

Краткие теоретические сведения

7. DISK DIRECTORSUITE

Эта программа предназначена для профессиональной работы с жестким диском. Это комплексный программный пакет, который включает в себя менеджер разделов, позволяющий осуществлять копирование, перемещение и изменение любых разделов Windows и Linux без риска потери данных, инструмент для восстановления разделов на жестком диске, а также менеджер загрузки, позволяющий установить несколько ОС на один ПК и управлять их запуском. Уже при загрузке программа производит проверку имеющихся дисков. Есть возможность запустить программу с загрузочного CD или дискеты, что позволяет восстановить разделы даже в ситуациях, когда загрузка компьютера невозможна. Программа оснащена паролем на вход и файлом помощи.

8. EASY RECOVERY PRO

Эта программа предназначена для восстановления утраченных или недоступных (в результате их повреждения) данных. Утилита позволяет без особого труда восстановить данные на жестком диске при утере их вследствие случайного удаления, атаки вирусов, повреждения из-за отключения или резких колебаний напряжения в электросети, ошибок в программе, проблем при создании разделов, неправильного включения ПК, повреждения структуры файловой системы. При помощи команды Drive Test можно проверить диск на наличие физических проблем.

Восстановление удаленных файлов. Общие сведения о программе Easy Recovery Pro. Easy Recovery Pro на сегодняшний день – это одна из лучших программ своего класса. Облегченный вариант – Easy Recovery Lite – входит в состав пакета комплексного обслуживания системы Fix-It Utilities.

Easy Recovery умеет работать почти со всеми более-менее распространенными файловыми системами: FAT12, FAT16, FAT32, NTFS, Novell, стандартами ZIP и JAZприводов, поддерживаются также и SCSI-жесткие диски. Одно из важнейших достоинств программы заключается в том, что у нее не только удобный и понятный Windows-интерфейс, доступный неопытным пользователям, но и есть возможность создать комплект загрузочных дискет с полноценной DOS-версией Easy Recovery. Сделано это для того, чтобы в случае серьезных неполадок, когда нет возможности загрузить Windows (а, соответственно, и "виндовскую" версию Easy Recovery), вас всегда был бы доступ к жесткому диску, и вы могли бы восстанавливать файлы непосредственно из MS-DOS. Такой режим наиболее предпочтителен при крупных сбоях – на сбойный диск ничего не записывается, Easy Recovery работает для него в режиме Read only («Только чтение»), поэтому и файлы на нем будут в большей сохранности.

Первое, что бросается в глаза сразу после запуска программы – очень долгий процесс сканирования диска. Однако это не является недостатком, а совсем наоборот – свидетельствует о ее неслабых возможностях. Дело в том, что быстрые, простые программы получают информацию об удаленных файлах и шансах на их восстановление из структуры директорий таблицы размещения файлов. Времени это, конечно, занимает очень мало, но ведь файл может еще быть на диске даже в том случае, если больше никаких его следов не осталось, да и сама таблица размещения файлов и корневая директория могут быть разрушены. Easy Recovery – она просканирует целиком весь жесткий диск, кластер за кластером, пытаясь собрать все кусочки каждого файла воедино. При этом допускается полная потеря обоих копий таблицы FAT, повреждение Root Folder и загрузочного сектора диска. Разумеется, если что-то из этого все-таки сохранилось, то будет в полной мере использовано. Кстати, если вы регулярно дефрагментируете диск, то шансы на успех еще больше увеличиваются – файл, у которого используемые кластеры идут друг за другом, восстановить проще.

Таким образом, Easy Recovery – это одна из немногих программ, которая справляется не только с восстановлением ошибочно удаленных файлов, но и восстанавливает информацию на диске после повреждения его вирусами, форматирования, пере- разбиения на разделы, порчи при скачках напряжения питания, сбоях аппаратного оборудования или программ.

Целесообразно сделать заранее загрузочные дискеты Easy Recovery – с ними ваши данные будут иметь как бы дополнительный "спасательный круг". Правда, поскольку Easy Recovery с поврежденным диском работает только на чтение, то придется запастись вторым винчестером или другим носителем, прежде чем приступать к восстановлению больших объемов данных. Причем доступ к диску вы, скорее всего, получите, даже если ваша ОС его не обнаруживает.

Конечно, с DOS-вариантом программы работать сложнее, поэтому желательно предварительно изучить инструкцию, чтобы разобраться во всех многочисленных опциях Easy Recovery.

Восстановление файлов с помощью EasyRecovery Запустите EasyRecovery (Пуск – Тема – Осмотр носителя – 4 Восстановление данных – EasyRecovery Professional).

После загрузки программы на экране появляется окно, в левой части которого размещено меню в виде кнопок, обеспечивающих доступ к четырем категориям функций, а также к двум дополнительным сервисам:

– Диагностика диска – утилиты для проверки физических параметров диска и целостности файловой системы;

– Восстановление данных – утилиты для поиска и восстановления удаленных и поврежденных данных;

– Восстановление файлов – специализированные утили- ты для восстановления файлов, созданных приложениями из семейства MS Office (кроме Outlook), а также ZIP архивов;

– Восстановление Email – специализированная утилита для восстановления файлов Outlook;

– Обновление программы – сервисные функции, позволяющие получать информацию и выполнять обновление лицензионной версии Easy Recovery через Интернет;

– Кризисный центр – набор функций, обеспечивающих доступ к сервисным веб-службам компании Ontrack.

В меню выберите Восстановление данных и далее Deleted Recovery. В левой части выберите диск. Если вы удалили один или несколько файлов, быстрое сканирование должно найти эти файлы. Поиск будет производиться только в файловой системе (это должно продолжаться всего несколько секунд). В случае, когда вы удалили целые каталоги, используйте опцию полного поиска. Для этого выберите опцию Complete Scan.

Нажмите кнопку Далее, чтобы начать сканирование диска. Вы увидите окно прогресса сканирования. Processing block показан сканированный блок диска и число всех блоков до момента сканирования, Elapsed time – время, которое прошло от момента начала сканирования, Remaining time – предполагаемое время, которое осталось до окончания операции, Directories found – количество найденных на диске каталогов, Files found – количество найденных файлов, Last file – название последнего найденного файла.

После окончания сканирования вы увидите список найденных файлов. Однако напомнить, что не каждый найденный с помощью Easy Recovery файл возможно восстановить.

Поле Condition в списке файлов показывает в каком состоянии находится найденный файл.

Выберите файлы, которые хотите восстановить и щелкните Далее. Первый символ имени удаленного файла заменен символом подчеркивания. В следующем окне в поле Recovery Statistics находится короткая статистика о восстановленных файлах, включающая количество файлов, которые вы выбрали для восстановления, а также их полный размер.

Выберите директорию, в которую их надо записать (Recover to Local Drive). Вы также можете отправить восстановленные файлы непосредственно на сервер FTP (Recover to an FTP Server). Помните, что Easy Recovery не позволит записать файлы в раздел, с которого происходит восстановление данные. Версия Professional предлагает возможность компрессии восстановленных файлов в архив ZIP (Create ZIP). На ваше усмотрение вы можете установить лимит размера файла ZIP (ZIP File Size Limit), а также создать отчет о восстановлении файлов (Generate Recovery Report). Выберите для восстановления диск C:\, нажмите Далее. В следующем окне нажмите Готово. Easy Recovery может записать установки восстановления, чтобы потом вы смогли продолжить операцию восстановления других файлов. Нажмите кнопку No. Вы восстановили данные. Просмотрите восстановленный файл.

9. FILE RECOVERY

Утилита предназначена для восстановления удаленных или стертых в результате форматирования жесткого диска, данных. Работает с файловыми системами FAT 12/16/32 и NTFS, а также умеет восстанавливать зашифрованные и сжатые файлы. Имеется возможность восстановления информации не только на жестком диске, но и на съемных носителях – дискетах, картах SmartMedia, CompactFlash, Memory Stick и т.д.

10. RESTORER2000 Data RECOVERY

Это мощная программа, которая поможет быстро и просто восстановить нужные файлы, утерянные в результате случайного удаления, а также восстановить отформатированные или разрушенные диски. Утилита поддерживает возможность создания образа диска, это очень полезно для таких задач, как восстановление жесткого диска с большим количеством неработоспособных секторов. Можно установить размер сканируемой области, в зависимости от этого будет меняться время выполнения, которое программа автоматически подсчитывает.

11. HDD Temperature Pro

Это очень маленькая утилита, предназначена для отслеживания состояния жестких дисков. Используя технологию SMART, встроенную во все современные жесткие диски, она анализирует и показывает текущую температуру диска. Здесь возможна установка максимальной температуры накопителя, при превышении которой программа выдаст сообщение. Можно сделать так, чтобы эта утилита самостоятельно загружалась при входе в ОС, так что она будет незаметна, но в нужный момент предупредит о возможной опасности перегрева диска.

12. TREESIZE

Эта утилита предназначена для мониторинга пространства на жестком диске и его освобождении. Она умет искать старые и неиспользуемые, а также временные файлы и удаляет их. С помощью этой утилиты можно найти папки, которые занимают больше всего места на диске, сравнить их объем в процентном соотношении в виде графика. Примерно такие же возможности имеют программы: FCLEANER, FREESPACE.

Практические задания

Задание

Данное практическое занятие предполагает выполнение следующих этапов:

– изучить методические указания;
– ответить на контрольные вопросы.

Контрольные вопросы:

1.Назначение программы Paragon Partition Manager?

2.Перечислите известные вам программы по обслуживанию жестких дисков в процессе их эксплуатации и определите их назначение.

3.Опишите последовательность восстановления удаленной информации, если: файл удален в корзину; файл удален в корзину и затем корзина была очищена.

Лабораторная работа№20 Мониторинг активности портов

Цель занятия: формирование умений и навыков блокировки и разблокировки портов подключения устройств.

Краткие теоретические сведения

Понятие порта в компьютере многозначно. Самое общее определение: порт – это соединение (физическое или логическое), через которое принимаются и отправляются данные. Об- мен данными между любыми устройствами возможен только при наличии утвержденного стандарта на интерфейс.

В состав аппаратного обеспечения порта входит специализированный разъём, предназначенный для подключения оборудования определённого типа. Часто этот специализированный разъем и называют портом, например, USB-порт, но есть разъемы, которые портами называть не принято, например, RJ11. Как правило, каждый порт имеет обозначение, которое размещается рядом с разъемом.

Основные порты, используемые в компьютерах, ноутбуках:

– USB-порт;
– IEEE 1394 (FireWire);
– порт eSATA и комбинированный порт USB/eSATA;
– сетевой порт Ethernet;
– порт SCSI;
– последовательный порт RS-232;
– порты для подключения внешних мониторов VGA, DVI, S-Video, HDMI, DisplayPort;
– порт для док-станции и порт репликатор;
– порты для модулей расширения PCMCIA, ExpressCard. USB – Universal Serial Bus – универсальная последовательная шина. USB-порты являются своего рода стандартом для подключения внешних устройств, к которому стремятся все производители этих устройств. К портам USB подключаются: мыши, клавиатуры, принтеры, сканеры, модемы, кардридеры, флэш накопители, фотоаппараты, сотовые телефоны, плееры, жёсткие диски, оптические дисководы и др.

IEEE 1394 – высокоскоростной последовательный порт для цифровых видеоустройств. Компания Apple продвигает стандарт IEEE 1394 под маркой FireWire, компания Sony – под маркой i.LINK. IEEE 1394 применяется для подключения видеокамер, цифровых фотоаппаратов и других мультимедийных устройств, а также принтеров, сканеров, внешних жестких дисков.

Основные преимущества по сравнению с USB 2.0 – более высокая скорость передачи, большая стабильность, большая дли- на кабеля до оконечного устройства.

eSATA – External Serial ATA (Advanced Technology Attachment – присоединение по передовой технологии) – последовательный интерфейс для подключения внешних устройств, поддерживающий режим «горячей замены». Стандарт eSATA предусматривает подключение внешних жестких дисков, оптических дисков, RAID-массивов. Скорость передачи данных гораздо выше, чем у USB 2.0 или IEEE 1394.

Недостатки eSATA:

– максимальная длина кабеля не превышает 2 метров;
– жёсткие диски, подключаемые через eSATA, потребуют дополнительного источника питания – это могут быть как разъёмы USB или 1394, так и розетка.

Порт Ethernet предназначен для подключения ноутбука к компьютерной сети с помощью сетевого кабеля через разъем RJ45 (RJ-45). Технология Ethernet описывается стандартами IEEE группы 802.3. Существует несколько стандартов технологии Ethernet.

Стандарты различаются скоростью передачи данных и передающей средой. В ноутбуках обычно устанавливают порт Ethernet 10/100/1000, который поддерживает стандарты 10BASE- T, 100BASE-TX и 1000BASE-T для расстояний до 100 м. Стандарт 10BASE-T позволяет передавать данные со скоростью 10 Мбит/с. Для передачи

используется 4 провода кабеля витой пары категории 3 или категории 5. По стандарту 100BASE-TX скорость передачи данных составляет 100 Мбит/с. Стандарт применяется для построения сетей топологии «звезда». Задействована витая пара категории 5, поддерживается дуплексная передача данных. Стандарт 1000BASE-T – гигабитный (Gigabit, Geth) Ethernet позволяет передавать данные со скоростью до 1 Гбит/с. Стандарт предусматривает использование витой пары категорий 5е.

RS-232 (англ. Recommended Standard) – стандарт последовательной асинхронной передачи двоичных данных между двумя устройствами на расстоянии до 15 метров. Порт RS-232 в последнее время не часто встречается в бизнес-ноутбуках, но может быть полезен в промышленных ноутбуках. Он используется для реализации систем сбора данных в реальном времени, подключения научного ряда контактов. Карты Type III поддерживают 16- или 32-разрядный интерфейс. Они имеют толщину 10,5 мм, что позволяет устанавливать на карту стандартные разъёмы внешних интерфейсов и избавиться, таким образом, от дополнительных кабелей.

Разъем имеет четыре ряда контактов. Разъем PCMCIA представляет собой щель шириной 54 мм, которая закрыта либо откидной шторкой, либо пластиковой заглушкой. Разъем (слот) PCMCIA (вверху) и заглушка, внизу – кардридер.

Большинство ноутбуков оснащается лишь одним разъемом PCMCIA типа II. А современные ноутбуки уже обходятся и вовсе без этих разъемов.

Порт ExpressCard. Стандарт ExpressCard для карт расширения был разработан ассоциацией PCMCIA на смену стандарту PC Card. Новый стандарт был создан на базе новой скоростной последовательной шины PCI Express. Стандарт ExpressCard не только более производительный, чем PC Card, но и более универсальный. Через ExpressCard можно подключаться к шине USB. Карты ExpressCard бывают двух типов, отличающихся по ширине: 34 мм и 54 мм. Соответственно и разъемы бывают двух типов ExpressCard/34 и ExpressCard/54. При этом карты 34 мм можно устанавливать как в разъем ExpressCard/34, так и в разъем ExpressCard/54. Через разъемы ExpressCard подключают ТВ- тюнеры, звуковые карты, карты Wi-Fi, флеш-накопители (они часто подключаются через USB-составляющую интерфейса ExpressCard), модемы для работы в сотовых сетях и др.

Разъем RJ11(RJ-11 Registered jack) – разъем модема ноутбука. Используется для подключения к Интернету через модем по телефонной линии.

Одной из важных особенностей современных корпоративных сетей является их размер, который зачастую исчисляется тысячами, а и иногда и десятками тысяч компьютеров. При этом деятельность пользователей может быть распределена среди различных компьютеров, а одна и та же проблема часто решается группами пользователей.

Важной задачей является контроль работы, как отдельных пользователей, так и групп пользователей.

Основными целями контроля являются: обеспечение информационной безопасности, выявление случаев некорректного, непрофессионального или нецелевого использования ресурсов, оценка характеристик функционирования корпоративной сети и параметров использования ресурсов.

Основной задачей обеспечения информационной безопасности является «раннее обнаружение» внутренних вторжений, т.е. выявление действий пользователей, которые могут предшествовать внутренним вторжениям. Чем крупнее организация, тем актуальней является для нее проблема предотвращения внутренних вторжений, в частности кражи информации, так как именно кража является конечной целью большинства внутренних вторжений. Связано это с тем, что в больших организациях затрудняется контроль над обращением информации и существенно возрастает цена ее утечки. Указанные обстоятельства определяют высокий уровень озабоченности данной проблемой со стороны крупного бизнеса и правительственных организаций. Решение данной проблемы заключается в применении "жесткой" политики информационной безопасности в организации и использовании средств мониторинга действий пользователей.

Spector 360 включает в себя средства для автоматического развертывания и удаленного управления, осуществляет запись разнообразных действий, включая: Email, чаты, мгновенные сообщения, посещаемые веб-сайты, онлайновые поисковые запросы, нажимаемые клавиши и используемые программы. Spector 360 также включает в себя средство для записи образов экрана в режиме видеокамеры.

Все эти инструменты ведут запись одновременно, скрытно, под защитой тройного уровня безопасности. Приложение Recorder хорошо конфигурируется и может быть настроено для записи только интересующих Вас событий.

В дополнение к мониторингу и ведению записи Spector 360 обладает развитой системой определения и обнаружения ключевых слов, которая будет немедленно извещать о каждом случае, когда пользователь контролируемого ПК отклонится от допустимого использования ПК или Интернет.

Регистратор Spector 360 можно перевести в скрытый режим, который обеспечивает невозможность обнаружения программы неуполномоченными пользователями. В скрытом режиме Spector 360 не будет виден пользователю в системном меню задач, диспетчере задач или в меню установки/удаления программ панели управления.

При помощи Spector 360 вы можете сгенерировать высококачественные отчеты для руководства, которые могут регулярно распечатываться или рассылаться по почте.

Spector 360 разработан для коммерческих, образовательных и правительственных организаций, использующих сети на платформе Windows.

Security Curator – это система обеспечения информационной безопасности нового поколения, объединяющая в себе возможность наблюдения за деятельностью сотрудников, контроля их действий и блокировки потенциально опасных путей утечки информации.

Security Curator ведёт мониторинг в реальном времени практически всех действий сотрудников при работе за компьютером. Информация о действиях пользователей обновляется в реальном режиме времени. При этом постоянно производится сохранение снимков экрана при совершении любых действий, также существует возможность наблюдения за рабочим столом пользователя в режиме онлайн. В случае работы пользователем с USB- устройствами производится резервное копирование файлов.

Внедрение Security Curator позволяет ограничить доступ к нежелательным сайтам, программам и приложениям на определенный промежуток времени либо постоянно.

Например, работодатель может разрешить сотрудникам посещать сайты ВКонтакте и Одноклассники только во время обеденного перерыва, а доступ к бухгалтерской программе 1С запретить после окончания рабочего дня и на выходных.

Activity Monitor мощный инструмент, который позволяет отслеживать любые действия в сети и предоставляет вам детальную информацию о том, что, как и когда делали ваши сотрудники. Будь то сеть библиотеки, университета или коммерческой организации, Activity Monitor поможет вам установить эффективный контроль над ней.

Приложение состоит из серверной и клиентской частей. Сервер Activity Monitor может быть установлен на любом компьютере в сети. Модуль-шпион (агент) устанавливается на всех компьютерах, действия на которых вы хотите отслеживать. Он может быть установлен даже удалённо с системы, на которой установлена серверная часть Activity Monitor.

Действия на сетевых компьютерах отслеживаются удалённо. Вы можете настроить программу таким образом, что она будет отслеживать и регистрировать действия на всех компьютерах в сети одновременно. Данные мониторинга могут быть использованы для более глубокого анализа и создания детальных отчётов.

Activity Monitor является эффективным средством повышения общей производительности труда в компаниях, использующих данную программу для мониторинга локальных сетей.

Net Vizor – программа для мониторинга сети. Net Vizor позволяет наблюдать за всей локальной сетью из одного рабочего места. Программа может следить за рабочими станциями и индивидуальными пользователями, которые используют различные компьютеры, находящиеся в сети.

Программа позволяет следить за сетевыми компьютерами, осуществлять фильтрацию контента и управлять сетевыми компьютерами дистанционно.

Существует возможность ведения журналов адресов посещенных сайтов, соединений с интернетом, открываемых файлов, чатов, пересылаемых сообщений электронной почты и так далее. Net Vizor также обезвреживает шпионские программы и помогает следить за безопасностью.

Сравнительный анализ программ приведен в таблицах 4.1–4.5.

Таблица 4.1 – Мониторинг

Наименование	Spector 360	Security Curator	Activity Monitor	Net Visor
Экран	+	+	+	+
Снимки экрана	+	+	+	+
Запущенные процессы	+	+	+	+
Время запуска и выключения программ	+	+	+	+
Бесплатные сервисы электронной почты	+	–	+	+
Нажатие клавиш	+	+	+	+
E-mail	+	+	+	+
Посещенные сайты	+	+	+	+
Переписка в IM агентах	+	+	+	+
Социальные сети	+	+	+	+
Поисковые запросы	+	+	+	+
USB устройства	+	+	+	-
Обнаружение ключевых слов	+	–	-	+
Установка, удаление программ	+	+	+	+
Контроль рабочего времени	+	+	+	+
Загружаемые файлы	+	+	+	+
Доступ к файлам, папкам	+	+	+	+
Активность пользователя	+	+	+	+
FTP	+	+	+	+
Сетевые соединения	+	+	+	+
Выборочный мониторинг	+	+	+	+
Запись по расписанию	+	+	+	+

Таблица 4.2 – Контроль

Наименование	Spector 360	Security Curator	Activity Monitor	Net Visor

Блокировка событий (запуск приложений, сайты, запрет файловых операций)	-	+	+	+
Блокировка запуска любых процессов	–	+	+	+
Блокировка подключения/отключения всех типов USB накопителей и устройств	-	+	–	-
Блокировка сетевых соединений (по порту, ip-адресу)	+	+	+	+
Блокировка сайтов по домену	+	+	+	+
Блокировка чатов и Интернет пейджеров	+	+	+	+
Блокировка доступа в Интернет по протоколу или порту	+	+	+	+
Запрет действий с файлами/папками	–	+	+	+

Таблица 4.3 – Отчетность

Наименование	Spector 360	Security Curator	Activity Monitor	Net Visor
Генерация отчетов с привязкой к отдельному пользователю	+	+	+	+
Поиск по ключевым словам	+	+	+	+
Генерация графических отчетов	+	+	+	+
Конвертация отчетов в PDF	+	+	–	+
Конвертация отчетов в HTML	+	+	+	+
Конвертация отчетов в CSV	+	+	+	+
Конвертация отчетов в Excel	+	–	+	-
Конвертация отчетов в Rich Text	+	–	-	-
Экспорт отчетов	+	–	-	-
Отправка отчетов по электронной почте	+	+	–	+
Отправка отчетов по FTP	+	+	–	-
Печать отчетов	+	+	+	+
Генерация отчетов по расписанию	+	+	–	-

Таблица 4.4 – Управление

Наименование	Spector 360	Security Curator	Activity Monitor	Net Visor
Централизованное управление клиентами	+	+	+	+
Централизованное управление лицензиями	+	+	+	+
Централизованное конфигурирование безопасности	+	+	+	+

Централизованное конфигурирование сети	+	–	-	-
Централизованное конфигурирование WEB фильтра	+	+	+	+
Резервирование и восстановление базы данных	+	–	-	-
Управление резервными копиями	+	–	-	-
Многопользовательский дискреционный контрольдоступа к данным	+	–	-	+

Продолжение таблицы 4.4

Наименование	Spector 360	Security Curator	Activity Monitor	Net Visor
Разделение доступа к функциям администрирования	+	–	-	+
Возможность группировки компьютеров	+	+	+	+
Возможность группировки пользователей	+	–	+	-

Таблица 4.5 – Безопасность

Наименование	Spector 360	Security Curator	Activity Monitor	Net Visor
Контроль компьютеров в сети	+	+	+	+
Удаленная установка	+	+	+	+
Невидимый режим работы	+	+	–	+
Авторизация при запуске административного модуля	+	+	+	+

Spector 360 незаменим в крупных организациях, где реша- ются задачи оперативного мониторинга огромного количества рабочих станции.

Если делать акцент на возможность контроля и блокировки действий пользователей, тут подойдет Security Curator, Net Visor и Activity Monitor.

Рассмотрим два способа улучшения безопасности работы сети.

Шаг 1. Меняем учетную запись администратора (Пользователь Администратор с пустым паролем – это уязвимость) (убираем уязвимость 1).

При установке Windows XP в автоматическом режиме с настройками по умолчанию, мы имеем пользователя Администратор с пустым паролем и любой User может войти в такой ПК с правами администратора. Чтобы решить проблему, выполним команду Мой компьютер – Панель управления- Администрирование – Управление компьютером – Локальные пользователи – Пользователи.

Здесь по щелчку правой кнопкой мыши на Администраторы зададим администратору пароль, например, 12345. Теперь в окне Администрирование зайдем в Локальную политику безопасности. Далее идем по веткам дерева: Локальные политики – Параметры безопасности – Учетные записи – Переименование учетной записи – Администратор.

Пользователя Администратор заменим на Admin.

Перезагружаем ОС. После наших действий получилась учетная запись Admin с паролем 12345 и правами администратора.

Теперь мы имеем пользователя Администратор с паролем, одна из уязвимостей системы устранена.

Операцию по изменению имени пользователя и заданию пароля мы также могли бы выполнить без использования системного реестра, использовав окно Учетные записи пользователей.

Учетная запись Гость позволяет входить в ПК и работать на нем (например, в Интернет) без использования специально созданной учетной записи. Запись Гость не требует ввода пароля и по умолчанию блокирована. Гость не может устанавливать или удалять программы. Эту учетную запись можно отключить, но нельзя удалить.

Шаг 2. Делаем окно приветствия пустым (убираем уязвимость 2).

У нас окно входа в систему содержит подсказку Admin, давайте ее уберем, сделав окно пустым. Для начала в окне Учетные записи пользователей жмем на кнопку Изменение входа пользователей в систему и уберем флажок Использовать страницу приветствия.

Теперь повысим безопасность сети еще на одну условную ступень, сделав оба поля окна приветствия пустыми.

Выполним команду Панель управления – Администрирование – Локальные политики безопасности – Локальные политики – Параметры безопасности – Интерактивный вход: не отображать последнего имени пользователя. Эту запись необходимо включить.

Теперь после завершения сеанса пользователь должен угадать не только пароль, но и имя пользователя.

Злоумышленники используют сканирование портов ПК для того, чтобы воспользоваться ресурсами чужого ПК в Сети. При этом необходимо указать IP адрес ПК и открытый port, к примеру, 195.34.34.30:23. После этого происходит соединение с удаленным ПК с некоторой вероятностью входа в этот ПК.

TCP/IP port – это адрес определенного сервиса (программы), запущенного на данном компьютере в Internet. Каждый открытый порт – потенциальная лазейка для взломщиков сетей и ПК. Например, SMTP (отправка почты) – 25 порт, WWW – 80 порт, FTP – 21 порт.

Хакеры сканируют порты для того, чтобы найти дырку (баг) в операционной системе. Пример ошибки, если администратор или пользователь ПК открыл полный доступ к сетевым ресурсам для всех или оставил пустой пароль на вход к компьютеру.

Одна из функций администратора сети – выявить недостатки в функционировании сети и устранить их. Для этого нужно просканировать сеть и закрыть (блокировать) все необязательные (открытые без необходимости) сетевые порты. Ниже, для примера, представлены службы TCP/IP, которые можно отключить:

– finger – получение информации о пользователях
– talk – возможность обмена данными по сети между пользователями
– bootp – предоставление клиентам информации о сети
– systat – получение информации о системе
– netstat – получение информации о сети, такой как текущие соединения
– rusersd – получение информации о пользователях, зарегистрированных в данный момент.

Просмотр активных подключений утилитой Netstat. Команда netstat обладает набором ключей для отображения портов, находящихся в активном и/или пассивном состоянии. С ее помощью можно получить список серверных приложений, работающих на данном компьютере. Большинство серверов находится в режиме LISTEN– ожидание запроса на соединение. Состояние CLOSE_WAIT означает, что соединение разорвано. TIME_WAIT

– соединение ожидает разрыва. Если соединение находится в состоянии SYN_SENT, то это означает наличие процесса, который пытается, установить соединение с сервером. ESTABLISHED – соединения установлены, т. е. сетевые службы работают (используются).

Итак, команда netstat показывает содержимое различных структур данных, связанных с сетью, в различных форматах в зависимости от указанных опций. Для сокетов (программных интерфейсов) TCP допустимы следующие значения состояния:

CLOSED – закрыт, сокет не используется; LISTEN – ожидает входящих соединений;

SYN_SENT – активно пытается установить соединение; SYN_RECEIVED – идет начальная синхронизация соединения;

ESTABLISHED – соединение установлено;

CLOSE_WAIT – удаленная сторона отключилась; ожидание закрытия сокета;

FIN_WAIT_1 – сокет закрыт; отключение соединения; CLOSING – сокет закрыт, затем удаленная сторона отключилась; ожидание подтверждения;

LAST_ACK – удаленная сторона отключилась, затем сокет закрыт; ожидание подтверждения;

FIN_WAIT_2 – сокет закрыт; ожидание отключения удаленной стороны;

TIME_WAIT – сокет закрыт, но ожидает пакеты, ещё находящиеся в сети для обработки.

Обнаружение открытых на ПК портов утилитой Netstat.

Для выполнения практического задания на компьютере необходимо выполнить команду Пуск – Выполнить. Откроется окно Запуск программы, в нем введите команду cmd.

Чтобы вывести все активные подключения TCP и прослушиваемые компьютером порты TCP/ UDP введите команду netstat. Мы видим Локального адреса (это ваш ПК) прослушиваются 6 портов. Они нужны для поддержки сети. На двух портах мы видим режим ESTABLISHED – соединения установлены, т. е. сетевые службы работают (используются). Четыре порта используются в режиме TIME_WAIT – соединение ожидает разрыва.

Запустите на вашем ПК Интернет и зайдите, например на www.yandex.ru. Снова выполните команду netstat. Как видим, добавилось несколько новых активных портов с их различными состояниями.

Опции команды netstat приведены в таблице 4.6.

Таблица 4.6 – Ключи для команды netstat

Опция (ключ)	Назначение
-a	Показывать состояние всех сокетов; обычно сокеты, используемые серверными процессами, не показываются
-A	Показывать адреса любых управляющих блоковпротокола, связанных с сокетами; используется для отладки
-i	Показывать состояние автоматически сконфигурированных (auto-configured) интерфейсов. Интерфейсы, статически сконфигурированные в системе, но не найденные во время загрузки, не показываются
-n	Показывать сетевые адреса как числа. netstat обычнопоказывает адреса как символы. Эту опцию можно использовать с любым форматом показа
-r	Показать таблицы маршрутизации. При использовании с опцией -s, показывает статистику маршрутизации
-s	Показать статистическую информацию по протоколам. При использовании с опцией -r, показывает статистику маршрутизации

-f	Семейство адресов. Ограничить показ статистики или адресов управляющих блоков только указаннымсемейством адресов, в качестве которого можно указывать: inet – для семейства адресов AF_INET, или unix – для семейства адресов AF_UNIX
-I	Интерфейс. Выделить информацию об указанном интерфейсе в отдельный столбец; по умолчанию (для третьей формы команды) используется интерфейс с наибольшим объёмом переданной информации с момента последней перезагрузки системы
-p	Отобразить идентификатор/название процесса создавшего сокет (-p, -programs display PID/Program name for sockets)

С помощью программы NetStat Agent вы сможете найти причину проблемы и заблокировать ее. Иначе говоря, NetStat Agent – полезный набор инструментов для мониторинга Интер- нет соединений и диагностики сети. Программа позволяет отслеживать TCP и UDP соединения на ПК, закрывать нежелательные соединения, завершать процессы, обновлять и освобождать DHCP настройки адаптера, просматривать сетевую статистику для адаптеров и TCP/IP протоколов, а также строить графики для команд Ping иTraceRoute.

В состав программы NetStat Agent вошли следующие утилиты:

– NetStat – отслеживает TCP и UDP соединения ПК (при этом отображается географическое местоположение удаленного сервера и имя хоста);

– IPConfig – отображает свойства сетевых адаптеров и конфигурацию сети;

– Ping – позволяет проверить доступность хоста в сети;

– TraceRoute – определяет маршрут между вашим компьютером и конечным хостом, сообщая все IP-адреса маршрутизаторов;

– DNS Query – подключается к DNS серверу и находит всю информацию о домене (IP адрес сервера, MX-записи (Mail Exchange) и др.).

– Route – отображает и позволяет изменять IP маршруты на ПК;

– ARP – отслеживает ARP изменения в локальной таблице;

– Whois – позволяет получить всю доступную информацию об IP-адресе или домене;

– HTTP Checker – помогает проверить, доступны ли Ваши веб-сайты;

– Statistics – показывает статистику сетевых интерфейсов и TCP/IP протоколов.

Сканер портов Nmap (Zenmap) – популярный сканер портов, который обследует сеть и проводит аудит защиты. Сканером портов Nmap можно определить открытые порты компьютера, а для безопасности сети пользователям рекомендуется закрыть доступ к этим портам с помощью брандмауэра.

Обычно для того, чтобы просканировать все порты какого- либо компьютера в сети вводится команда nmap –p1-65535 IP- адрес_компьютера или nmap –sV IP-адрес компьютера, а для ска- нирования сайта – команда nmap –sS –sV –O – P0 адрес сайта.

Монитор портов TCPView показывает все процессы, использующие Интернет-соединения.

Запустив TCPView, можно узнать, какой порт открыт и какое приложение его использует, а при необходимости и немедленно разорвать соединение.

Практические задания

Задание

Данное практическое занятие предполагает выполнение следующих этапов:

– изучить методические указания;

– ответить на контрольные вопросы.

Контрольные вопросы:

1. Какие виды мониторинга рабочих операций пользователя существуют?

2. Дайте характеристику современных программных средств мониторинга действий пользователей.
3. Какое программное средство вы порекомендовали? Почему?
4. Какие уязвимости операционной системы Windows были устранены в данной работе и какими путями?
5. Как узнать закрытые порты?
6. Как открыть нужный порт?
7. Для чего используется программа NetStat Agent?
8. Для чего используется программа Nmap?
9. Для чего используется программа TCPView?

Лабораторная работа№21 Блокирование портов

Цель занятия: формирование умений и навыков блокировки и разблокировки портов подключения устройств

Краткие теоретические сведения

Понятие порта в компьютере многозначно. Самое общее определение: порт - это соединение (физическое или логическое), через которое принимаются и отправляются данные. Обмен данными между любыми устройствами возможен только при наличии утвержденного стандарта на интерфейс.

В состав аппаратного обеспечения порта входит специализированный разъём, предназначенный для подключения оборудования определённого типа. Часто этот специализированный разъем и называют портом, например USB-порт, но есть разъемы, которые портами называть не принято, например, RJ11. Как правило, каждый порт имеет обозначение, которое размещается рядом с разъемом.

Основные порты, используемые в компьютерах, ноутбуках:

- USB-порт;
- IEEE 1394 (FireWire) ;
- Порт eSATA и комбинированный порт USB/eSATA;
- Сетевой порт Ethernet;
- Порт SCSI;
- Последовательный порт RS-232;
- Порты для подключения внешних мониторов VGA, DVI, S-Video, HDMI, DisplayPort;
- Порт для док-станции и порт репликатор;
- Порты для модулей расширения PCMCIA, ExpressCard.

USB - Universal Serial Bus - универсальная последовательная шина. USB-порты являются своего рода стандартом для подключения внешних устройств, к которому стремятся все производители этих устройств. К портам USB подключаются: мыши, клавиатуры, принтеры, сканеры, модемы, кардридеры, флэш-накопители, фотоаппараты, сотовые телефоны, плееры, жёсткие диски, оптические дисководы и др.

IEEE 1394 - высокоскоростной последовательный порт для цифровых видеоустройств. Компания Apple продвигает стандарт IEEE 1394 под маркой FireWire, компания Sony – под маркой i.LINK. IEEE 1394 применяется для подключения видеокамер, цифровых фотоаппаратов и других мультимедийных устройств, а также принтеров, сканеров, внешних жестких дисков.

Основные преимущества по сравнению с USB 2.0 - более высокая скорость передачи, большая стабильность, большая длина кабеля до оконечного устройства.

eSATA - External Serial ATA (Advanced Technology Attachment - присоединение по передовой технологии) – последовательный интерфейс для подключения внешних устройств, поддерживающий режим «горячей замены». Стандарт eSATA предусматривает подключение внешних жестких дисков, оптических дисков, RAID-массивов. Скорость передачи данных гораздо выше, чем у USB 2.0 или IEEE 1394.

Недостатки eSATA:

- максимальная длина кабеля не превышает 2 метров;
- жёсткие диски, подключаемые через eSATA, потребуют дополнительного источника питания - это могут быть как разъёмы USB или 1394, так и розетка.

Порт Ethernet предназначен для подключения ноутбука к компьютерной сети с помощью сетевого кабеля через разъем RJ45 (RJ-45). Технология Ethernet описывается стандартами IEEE группы 802.3. Существует несколько стандартов технологии Ethernet. Стандарты различаются скоростью передачи данных и передающей средой. В ноутбуках обычно устанавливают порт Ethernet 10/100/1000, который поддерживает стандарты

10BASE-T, 100BASE-TX и 1000BASE-T для расстояний до 100 м. Стандарт 10BASE-T позволяет передавать данные со скоростью 10 Мбит/с. Для передачи используется 4 провода кабеля витой пары категории 3 или категории 5. По стандарту 100BASE-TX скорость передачи данных составляет 100 Мбит/с. Стандарт применяется для построения сетей топологии «звезда». Задействована витая пара категории 5, поддерживается дуплексная передача данных. Стандарт 1000BASE-T - гигабитный (Gigabit, Geth) Ethernet позволяет передавать данные со скоростью до 1 Гбит/с. Стандарт предусматривает использование витой пары категорий 5e.

RS-232 (англ. Recommended Standard) - стандарт последовательной асинхронной передачи двоичных данных между двумя устройствами на расстоянии до 15 метров. Порт RS-232 в последнее время не часто встречается в бизнес-ноутбуках, но может быть полезен в промышленных ноутбуках.

Он используется для реализации систем сбора данных в реальном времени, подключения научного ряда контактов. Карты Type III поддерживают 16- или 32-разрядный интерфейс. Они имеют толщину 10,5 мм, что позволяет устанавливать на карту стандартные разъёмы внешних интерфейсов и избавиться, таким образом, от дополнительных кабелей. Разъем имеет четыре ряда контактов. Разъем PCMCIA представляет собой щель шириной 54 мм, которая закрыта либо откидной шторкой, либо пластиковой заглушкой.

Разъем (слот) PCMCIA (вверху) и заглушка, внизу – кардридер.

Большинство ноутбуков оснащается лишь одним разъемом PCMCIA типа II. А современные ноутбуки уже обходятся и вовсе без этих разъемов.

Порт ExpressCard. Стандарт ExpressCard для карт расширения был разработан ассоциацией PCMCIA на смену стандарту PC Card. Новый стандарт был создан на базе новой скоростной последовательной шины PCI Express. Стандарт ExpressCard не только более производительный, чем PC Card, но и более универсальный. Через ExpressCard можно подключаться к шине USB. Карты ExpressCard бывают двух типов, отличающихся по ширине: 34 мм и 54 мм. Соответственно и разъемы бывают двух типов ExpressCard/34 и ExpressCard/54. При этом карты 34 мм можно устанавливать как в разъем ExpressCard/34, так и в разъем ExpressCard/54. Через разъемы ExpressCard подключают ТВ-тюнеры, звуковые карты, карты Wi-Fi, флеш-накопители (они часто подключаются через USB-составляющую интерфейса ExpressCard), модемы для работы в сотовых сетях и др.

Разъем RJ11(RJ-11 Registered jack) – разъем модема ноутбука. Используется для подключения к Интернету через модем по телефонной линии.

Сравнение средств мониторинга действий пользователей

Одной из важных особенностей современных корпоративных сетей является их размер, который зачастую исчисляется тысячами, а и иногда и десятками тысяч компьютеров. При этом деятельность пользователей может быть распределена среди различных компьютеров, а одна и та же проблема часто решается группами пользователей. Важной задачей является контроль работы, как отдельных пользователей, так и групп пользователей.

Основными целями контроля являются: обеспечение информационной безопасности, выявление случаев некорректного, непрофессионального или нецелевого использования ресурсов, оценка характеристик функционирования корпоративной сети и параметров использования ресурсов.

Основной задачей обеспечения информационной безопасности является «раннее обнаружение» внутренних вторжений, т.е. выявление действий пользователей, которые могут предшествовать внутренним вторжениям. Чем крупнее организация, тем актуальней является для нее проблема предотвращения внутренних вторжений, в частности кражи информации, так как именно кража является конечной целью большинства внутренних вторжений. Связано это с тем, что в больших организациях затрудняется контроль над

обращением информации и существенно возрастает цена ее утечки. Указанные обстоятельства определяют высокий уровень озабоченности данной проблемой со стороны крупного бизнеса и правительственных организаций. Решение данной проблемы заключается в применении "жесткой" политики информационной безопасности в организации и использовании средств мониторинга действий пользователей.

Spector 360

Spector 360 включает в себя средства для автоматического развертывания и удаленного управления, осуществляет запись разнообразных действий, включая: Email, чаты, мгновенные сообщения, посещаемые веб-сайты, онлайновые поисковые запросы, нажимаемые клавиши и используемые программы. Spector 360 также включает в себя средство для записи образов экрана в режиме видеокамеры.

Все эти инструменты ведут запись одновременно, скрытно, под защитой тройного уровня безопасности. Приложение Recorder хорошо конфигурируется и может быть настроено для записи только интересующих Вас событий.

В дополнение к мониторингу и ведению записи Spector 360 обладает развитой системой определения и обнаружения ключевых слов, которая будет немедленно извещать о каждом случае, когда пользователь контролируемого ПК отклонится от допустимого использования ПК или Интернет.

Регистратор Spector 360 можно перевести в скрытый режим, который обеспечивает невозможность обнаружения программы неуполномоченными пользователями. В скрытом режиме Spector 360 не будет виден пользователю в системном меню задач, диспетчере задач или в меню установки/удаления программ панели управления.

При помощи Spector 360 вы можете сгенерировать высококачественные отчеты для руководства, которые могут регулярно распечатываться или рассылаться по почте.

Spector 360 разработан для коммерческих, образовательных и правительственных организаций, использующих сети на платформе Windows.

Security Curator

Security Curator – это система обеспечения информационной безопасности нового поколения, объединяющая в себе возможность наблюдения за деятельностью сотрудников, контроля их действий и блокировки потенциально опасных путей утечки информации.

Security Curator ведёт мониторинг в реальном времени практически всех действий сотрудников при работе за компьютером. Информация о действиях пользователей обновляется в реальном режиме времени. При этом постоянно производится сохранение снимков экрана при совершении любых действий, также существует возможность наблюдения за рабочим столом пользователя в режиме онлайн. В случае работы пользователем с USB-устройствами производится резервное копирование файлов.

Внедрение Security Curator позволяет ограничить доступ к нежелательным сайтам, программам и приложениям на определенный промежуток времени либо постоянно. Например, работодатель может разрешить сотрудникам посещать сайты ВКонтакте и Одноклассники только во время обеденного перерыва, а доступ к бухгалтерской программе 1С запретить после окончания рабочего дня и на выходных.

Activity Monitor

Этот мощный инструмент позволяет отслеживать любые действия в сети и предоставляет вам детальную информацию о том, что, как и когда делали ваши сотрудники. Будь то сеть библиотеки, университета или коммерческой организации, Activity Monitor поможет вам установить эффективный контроль над ней.

Приложение состоит из серверной и клиентской частей. Сервер Activity Monitor может быть установлен на любом компьютере в сети. Модуль-шпион (агент) устанавливается на всех компьютерах, действия на которых вы хотите отслеживать. Он может быть установлен даже удалённо с системы, на которой установлена серверная часть Activity Monitor.

Действия на сетевых компьютерах отслеживаются удалённо. Вы можете настроить программу таким образом, что она будет отслеживать и регистрировать действия на всех компьютерах в сети одновременно. Данные мониторинга могут быть использованы для более глубокого анализа и создания детальных отчётов.

Activity Monitor является эффективным средством повышения общей производительности труда в компаниях, использующих данную программу для мониторинга локальных сетей. Проще говоря, этот мощный инструмент от Softactivity экономит ваши деньги.

NetVizor

NetVizor — Программа для мониторинга сети. NetVizor позволяет наблюдать за всей локальной сетью из одного рабочего места. Программа может следить за рабочими станциями и индивидуальными пользователями, которые используют различные компьютеры, находящимся в сети.

Программа позволяет следить за сетевыми компьютерами, осуществлять фильтрацию контента и управлять сетевыми компьютерами дистанционно.

Существует возможность ведения журналов адресов посещенных сайтов, соединений с интернетом, открываемых файлов, чатов, пересылаемых сообщений электронной почты и так далее. NetVizor также обезвреживает шпионские программы и помогает следить за безопасностью.

Мониторинг	**Spector 360**	**Security Curator**	**Activity Monitor**	**Net Visor**
Экран	+	+	+	+
Снимки экрана	+	+	+	+
Запущенные процессы	+	+	+	+
Время запуска и выключения программ	+	+	+	+
Бесплатные сервисы электронной почты	+	-	+	+
Нажатие клавиш	+	+	+	+
E-mail	+	+	+	+
Посещенные сайты	+	+	+	+
Переписка в IM агентах	+	+	+	+
Социальные сети	+	+	+	+
Поисковые запросы	+	+	+	+
USB устройства	+	+	+	-
Обнаружение ключевых слов	+	-	-	+
Установка, удаление программ	+	+	+	+
Контроль рабочего времени	+	+	+	+
Загружаемые файлы	+	+	+	+
Доступ к файлам, папкам	+	+	+	+
Активность пользователя	+	+	+	+
FTP	+	+	+	+
Сетевые соединения	+	+	+	+
Выборочный мониторинг	+	+	+	+
Запись по расписанию	+	+	+	+
Блокировка событий (запуск приложений, сайты,запрет файловых	-	+	+	+

операций)				
Блокировка запуска любых процессов	-	+	+	+
Блокировка подключения/отключения всех типовUSB накопителей и устройств	-	+	-	-
Блокировка сетевых соединений (по порту, ipадресу)	+	+	+	+
Блокировка сайтов по домену	+	+	+	+
Блокировка чатов и Интернет пейджеров	+	+	+	+
Блокировка доступа в Интернет по протоколу илипорту	+	+	+	+
Запрет действий с файлами/папками	-	+	+	+
Отчетность	**Spector 360**	**Security Curator**	**Activity Monitor**	**Net Visor**
Генерация отчетов с привязкой к отдельномупользователю	+	+	+	+
Поиск по ключевым словам	+	+	+	+
Генерация графических отчетов	+	+	+	+
Конвертация отчетов в PDF	+	+	-	+
Конвертация отчетов в HTML	+	+	+	+
Конвертация отчетов в CSV	+	+	+	+
Конвертация отчетов в Excel	+	-	+	-
Конвертация отчетов в Rich Text	+	-	-	-
Экспорт отчетов	+	-	-	-
Отправка отчетов по электронной почте	+	+	-	+
Отправка отчетов по FTP	+	+	-	-
Печать отчетов	+	+	+	+
Генерация отчетов по расписанию	+	+	-	-
Управление	**Spector 360**	**Security Curator**	**Activity Monitor**	**Net Visor**
Централизованное управление клиентами	+	+	+	+
Централизованное управление лицензиями	+	+	+	+
Централизованное конфигурирование безопасности	+	+	+	+
Централизованное конфигурирование сети	+	-	-	-
Централизованное конфигурирование WEB-фильтра	+	+	+	+
Резервирование и восстановление базы данных	+	-	-	-
Управление резервными копиями	+	-	-	-
Многопользовательский дискреционный контроль доступа к данным	+	-	-	+
Разделение доступа к функциям администрирования	+	-	-	+

Возможность группировки компьютеров	+	+	+	+
Возможность группировки пользователей	+	-	+	-
Безопасность	**Spector 360**	**Security Curator**	**Activity Monitor**	**Net Visor**
Контроль компьютеров в сети	+	+	+	+
Удаленная установка	+	+	+	+
Невидимый режим работы	+	+	-	+
Авторизация при запуске административного модуля	+	+	+	+
Стоимость	**Spector 360**	**Security Curator**	**Activity Monitor**	**Net Visor**
Цена за 1 лицензию (от 5 до 99 хостов)	~ 5200 руб.	~ 1800 руб.	~ 1400 руб.	~ 1600 руб.
Цена за 1 лицензию (от 100 до 249 хостов)	~ 4000 руб.	~ 1600 руб.	~ 700 руб.	~ 1100 руб.
Цена за 1 лицензию (от 250 до 1000 хостов)	~	~	~	~
3500 руб.		1500 руб.	600 руб.	200 руб.

Определенно, Spector 360 незаменим в крупных организациях, где решаются задачи оперативного мониторинга огромного количества рабочих станции.

Если делать акцент на возможность контроля и блокировки действий пользователей, тут подойдет Security Curator, NetVisor и Activity Monitor.

Рассмотрим два способа улучшения безопасности работы сети.

Шаг 1. Меняем учетную запись администратора (Пользователь Администратор с пустым паролем — это уязвимость) **(убираем уязвимость 1)**

При установке Windows XP в автоматическом режиме с настройками по умолчанию мы имеем пользователя **Администратор** с пустым паролем и любой **User** может войти в такой ПК с правами администратора. Чтобы решить проблему выполним команду **Мой компьютер-Панель управления-Администрирование-Управление компьютером-Локальные пользователи-Пользователи** (рис. 1).

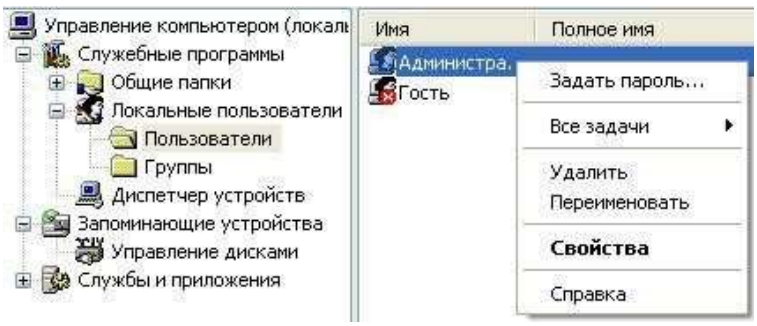


Рис. 1 - Окно Управление компьютером

Здесь по щелчку правой кнопкой мыши на **Администраторы** зададим администратору пароль, например, 12345. Теперь в окне **Администрирование** зайдем в **Локальную политику безопасности**. Далее идем по веткам дерева: **Локальные политики-Параметры безопасности-Учетные записи: Переименование учетной записи Администратор** (рис..2).

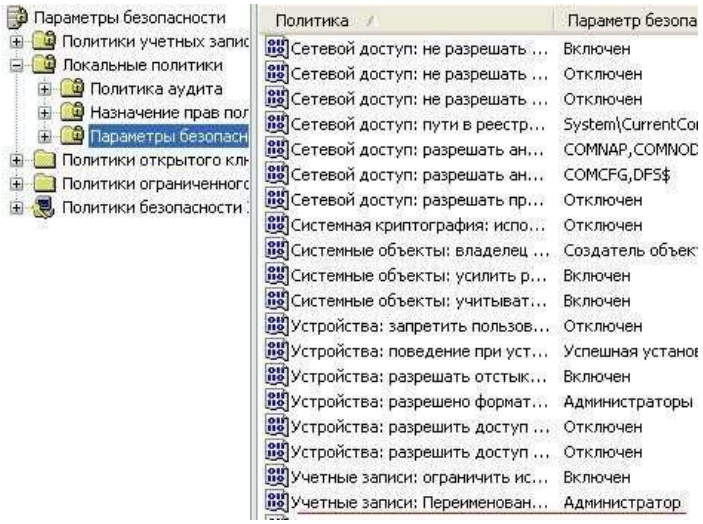


Рис. 2 - Находим в системном реестре запись Переименование учетной записи Администратор

Здесь пользователя **Администратор** заменим на **Admin** (рис. 3).

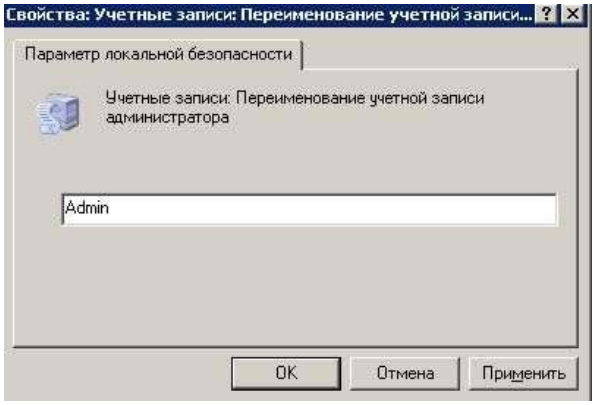


Рис. 3 - Пользователю Администратор присваиваем новое имя

Перезагружаем ОС. После наших действий получилась учетная запись Admin с паролем 12345 и правами администратора (рис. 4).

Рис. 4 - Окно входа в ОС Windows XP

Все, теперь мы имеем пользователя **Администратор** с паролем, одна из уязвимостей системы устранена.

Операцию по изменению имени пользователя и заданию пароля мы также могли бы выполнить без использования системного реестра, использовав окно **Учетные записи пользователей**, что гораздо проще (рис. 5).

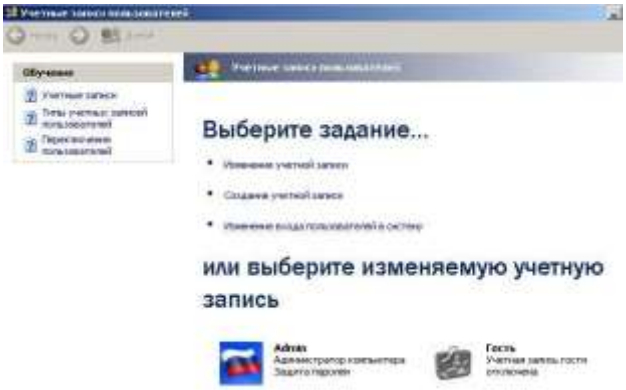


Рис. 5 - Окно Учетные записи пользователей

Примечание

Учетная запись Гость позволяет входить в ПК и работать на нем (например, в Интернет) без использования специально созданной учетной записи. Запись Гость не требуетввода пароля и по умолчанию блокирована. Гость не может устанавливать или удалятьпрограммы. Эту учетную запись можно отключить, но нельзя удалить.

Шаг 2. Делаем окно приветствия пустым (убираем уязвимость 2)

У нас окно входа в систему содержит подсказку Admin, давайте ее уберем, сделав окно пустым. Для начала в окне **Учетные записи пользователей** жмем на кнопку **Изменение входа пользователей в систему** и уберем флажок **Использоватьстраницу приветствия** (рис. 6 и рис. 7).

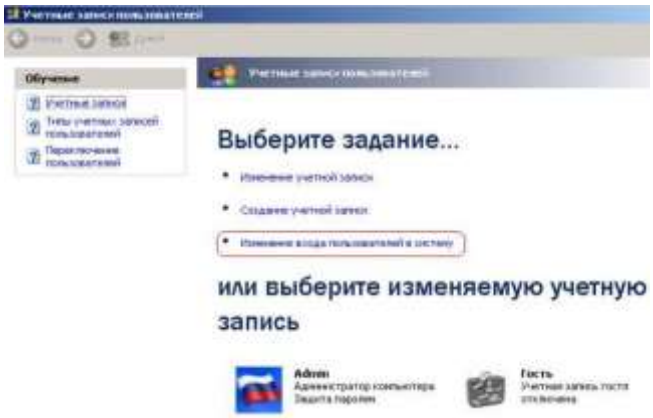


Рис. 6 - Окно Учетные записи пользователей

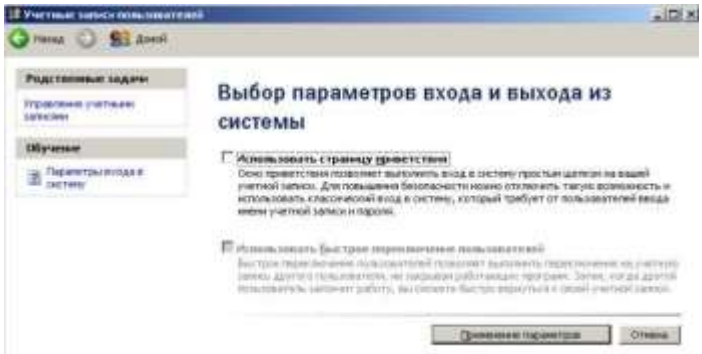


Рис. 7 - Убираем флажок Использовать страницу приветствия

Теперь повысим безопасность сети еще на одну условную ступень, сделав оба поля окна приветствия пустыми (рис. 8).

Рис. 8 - Обе строки данного окна сделаем пустыми

Выполним команду **Панель управления - Администрирование – Локальные политики безопасности - Локальные политики - Параметры безопасности - Интерактивный вход: не отображать последнего имени пользователя**. Эту запись необходимо включить (рис. 9).

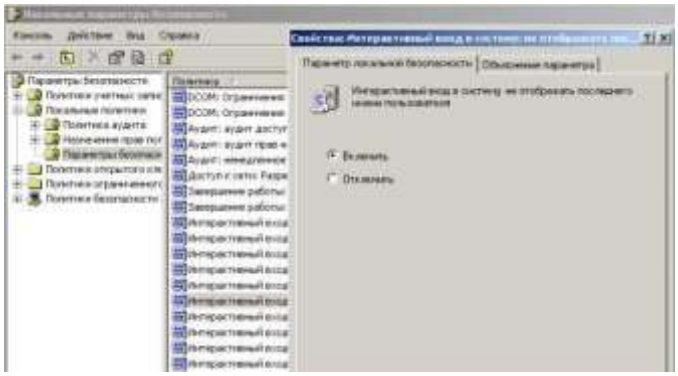


Рис. 9 - Активируем переключатель Включить

Теперь после завершения сеанса пользователь должен угадать не только пароль, но и имя пользователя (рис. 10).

Рис. 10 - Обе строки окна приветствия пусты

Выявление сетевых уязвимостей сканированием портов ПК

Злоумышленники используют сканирование портов ПК для того, чтобы воспользоваться ресурсами чужого ПК в Сети. При этом необходимо указать **IP** адрес ПК и открытый **port,** к примеру, **195.34.34.30:23**. После этого происходит соединение с удаленнымПК с некоторой вероятностью входа в этот ПК.

· TCP/IP port — это адрес определенного сервиса (программы), запущенного на данном компьютере в Internet. Каждый открытый порт — потенциальная лазейка для взломщиков сетей и ПК. Например, SMTP (отправка почты) — 25 порт, WWW — 80 порт, FTP —21 порт.

· Хакеры сканируют порты для того, чтобы найти дырку (баг) в операционной

системе. Пример ошибки, если администратор или пользователь ПК открыл полный доступ ксетевым ресурсам для всех или оставил пустой пароль на вход к компьютер.

Одна из функций администратора сети - выявить недостатки в функционировании сети и устранить их. Для этого нужно просканировать сеть и закрыть (блокировать) все необязательные (открытые без необходимости) сетевые порты. Ниже, для примера, представлены службы TCP/IP, которые можно отключить:

- finger — получение информации о пользователях
- talk— возможность обмена данными по сети между пользователями
- bootp — предоставление клиентам информации о сети
- systat — получение информации о системе
- netstat — получение информации о сети, такой как текущие соединения
- rusersd — получение информации о пользователях, зарегистрированных в данный момент

Просмотр активных подключений утилитой Netstat

Команда **netstat** обладает набором ключей для отображения портов, находящихся в активном и/или пассивном состоянии. С ее помощью можно получить список серверных приложений, работающих на данном компьютере. Большинство серверов находится в режиме **LISTEN**— ожидание запроса на соединение. Состояние **CLOSE_WAIT** означает, что соединение разорвано. **TIME_WAIT** — соединение ожидает разрыва. Если соединение находится в состоянии **SYN_SENT**, то это означает наличие процесса, который пытается, установить соединение с сервером. **ESTABLISHED** — соединения установлены, т. е. сетевые службы работают (используются).

Итак, команда netstat показывает содержимое различных структур данных, связанных с сетью, в различных форматах в зависимости от указанных опций. Для сокетов (программных интерфейсов) TCP допустимы следующие значения состояния

- CLOSED — Закрыт. Сокет не используется.
- LISTEN — Ожидает входящих соединений.
- SYN_SENT — Активно пытается установить соединение.
- SYN_RECEIVED — Идет начальная синхронизация соединения.
- ESTABLISHED — Соединение установлено.
- CLOSE_WAIT — Удаленная сторона отключилась; ожидание закрытия сокета.
- FIN_WAIT_1 — Сокет закрыт; отключение соединения.
- CLOSING — Сокет закрыт, затем удаленная сторона отключилась; ожиданиеподтверждения.
- LAST_ACK — Удаленная сторона отключилась, затем сокет закрыт; ожиданиеподтверждения.
- FIN_WAIT_2 — Сокет закрыт; ожидание отключения удаленной стороны.
- TIME_WAIT — Сокет закрыт, но ожидает пакеты, ещё находящиеся в сети дляобработки

Примечание

Что такое «сокет» поясняет рис. 11. Пример сокета – 194.86.6..54:21

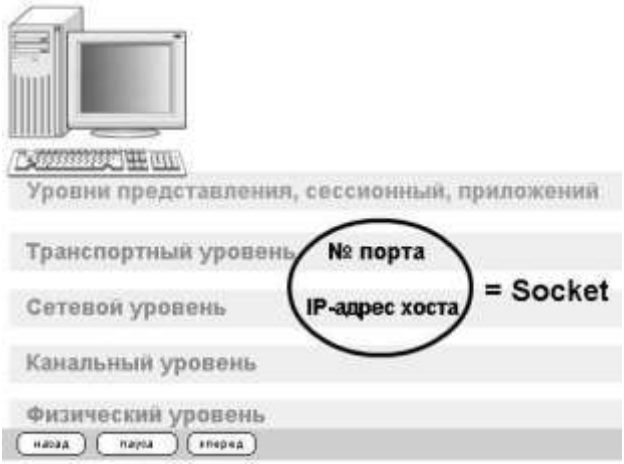


Рис. 11 - Сокет это № порта + IP адрес хоста

Практический пример. Обнаружение открытых на ПК портов утилитой Netstat

Для выполнения практического задания на компьютере необходимо выполнить команду **Пуск-Выполнить**. Откроется окно **Запуск программы**, в нем введите команду **cmd** (рис. 12).

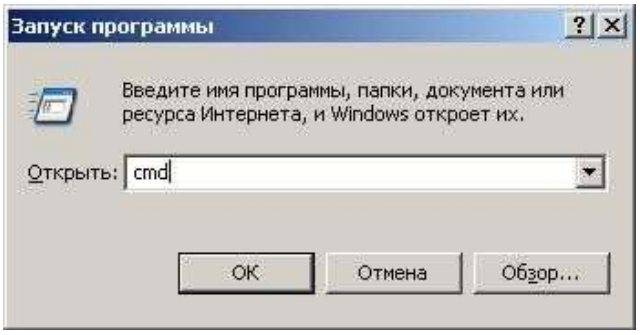


Рис. 12 - Окно Запуск программы

Чтобы вывести все активные подключения TCP и прослушиваемые компьютером порты TCP/ UDP введите команду **netstat** (рис. 13). Мы видим Локального адреса (это ваш ПК) прослушиваются 6 портов. Они нужны для поддержки сети. На двух портах мы видим режим **ESTABLISHED** — соединения установлены, т. е. сетевые службы работают (используются). Четыре порта используются в режиме **TIME_WAIT** — соединение ожидает разрыва.

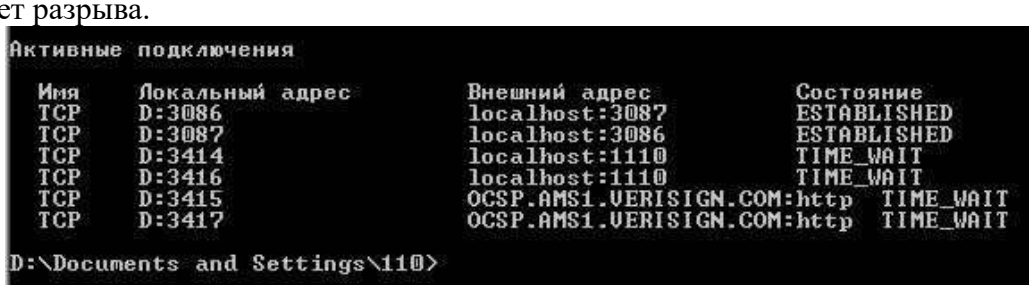


Рис. 13 - Список активных подключений на тестируемом ПК

Запустите на вашем ПК Интернет и зайдите, например на **www. yandex. ru**. Снова выполните команду **netstat** (рис. 14). Как видим, добавилось несколько новых активных портов с их различными состояниями.

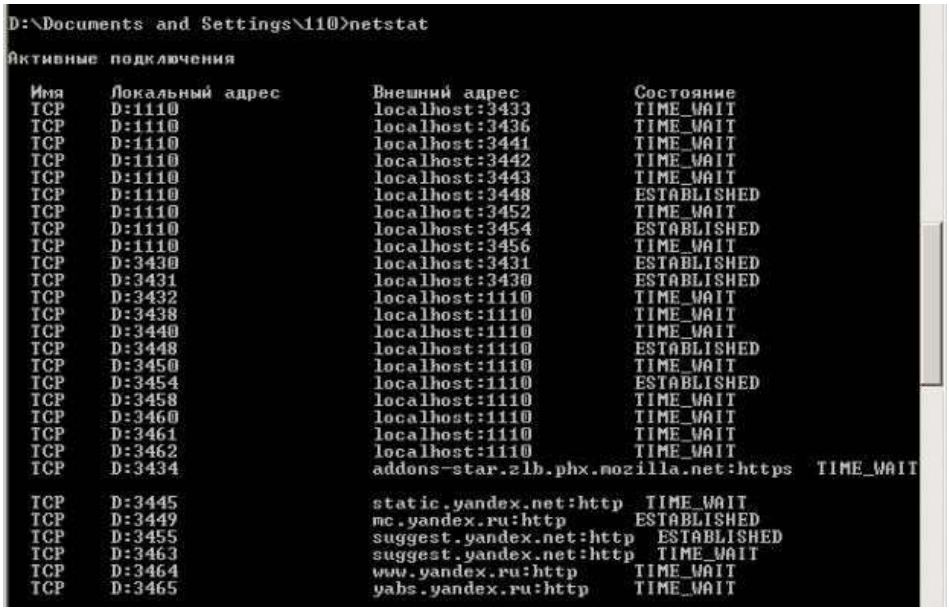


Рис. 14 - Активные подключения при работе ПК в Интернет

Команда **netstat** имеет следующие опции – табл. 1.Таблица 1 - Ключи для команды netstat

Опция (ключ)	**Назначение**
-a	Показывать состояние всех сокетов; обычно сокеты, используемые серверными процессами, не показываются.
-A	Показывать адреса любых управляющих блоков протокола, связанных с сокетами; используется для отладки.
-i	Показывать состояние автоматически сконфигурированных (auto-configured) интерфейсов. Интерфейсы, статически сконфигурированные в системе, но ненайденные во время загрузки, не показываются.
-n	Показывать сетевые адреса как числа. netstat обычно показывает адреса как символы.Эту опцию можно использовать с любым форматом показа.
-r	Показать таблицы маршрутизации. При использовании с опцией - s, показывает статистику маршрутизации.
-s	Показать статистическую информацию по протоколам. При использовании с опцией - r, показывает статистику маршрутизации.
-f семейство адресов	Ограничить показ статистики или адресов управляющих блоков только указаннымсемейством_адресов, в качестве которого можно указывать:**inet** Для семейства адресов **AF_INET**,или **unix** Для семейства адресов**AF_UNIX**.
-I интерфейс	Выделить информацию об указанном интерфейсе в отдельный столбец; по умолчанию(для третьей формы команды) используется интерфейс с наибольшим объёмом переданной информации с момента последней перезагрузки системы. В качестве интерфейса можно указывать любой из интерфейсов, перечисленных в файлеконфигурации системы, например, emd1 или lo0.
-p	Отобразить идентификатор/название процесса создавшего сокет (-p, — programs display PID/Program name for sockets)

Программа NetStat Agent

Представьте ситуацию: ваше Интернет-соединение стало работать медленно, компьютер постоянно что-то качает из Сети. Вам поможет программа NetStat Agent. С ее помощью вы сможете найти причину проблемы и заблокировать ее. Иначе говоря, **NetStat Agent** — полезный набор инструментов для мониторинга Интернет соединений и диагностики сети. Программа позволяет отслеживать TCP и UDP соединения на ПК, закрывать нежелательные соединения, завершать процессы, обновлять и освобождать DHCP настройки адаптера, просматривать сетевую статистику для адаптеров и TCP/IP протоколов,а также строить графики для команд **Ping** и**TraceRoute** (рис. 15).

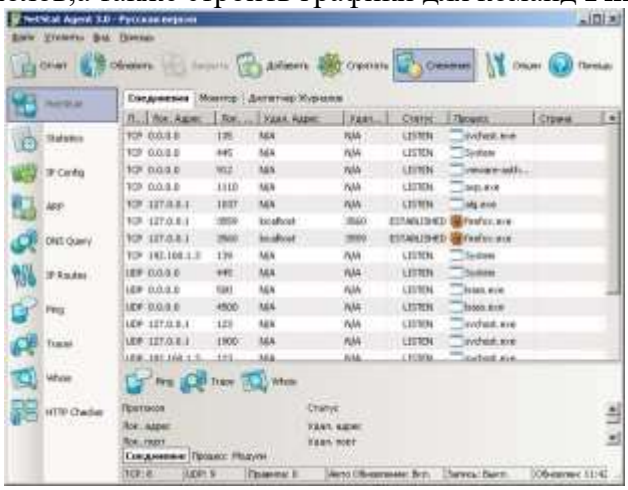


Рис. 15 - Главное окно программы NetStat Agent

В состав программы NetStat Agent вошли следующие утилиты:

· **NetStat** — отслеживает TCP и UDP соединения ПК (при этом отображается географическое местоположение удаленного сервера и имя хоста).

· **IPConfig** — отображает свойства сетевых адаптеров и конфигурацию сети.

· **Ping** — позволяет проверить доступность хоста в сети.

· **TraceRoute** — определяет маршрут между вашим компьютером и конечнымхостом, сообщая все IP-адреса маршрутизаторов.

· **DNS Query** — подключается к DNS серверу и находит всю информацию о домене(IP адрес сервера, MX-записи (Mail Exchange) и др.).

· **Route** — отображает и позволяет изменять IP маршруты на ПК.

· **ARP** — отслеживает ARP изменения в локальной таблице.

· **Whois** — позволяет получить всю доступную информацию об IP-адресе или домене.

· **HTTP Checker** — помогает проверить, доступны ли Ваши веб-сайты.

· **Statistics** — показывает статистику сетевых интерфейсов и TCP/IP протоколов.

Сканер портов Nmap (Zenmap)

Nmap — популярный сканер портов, который обследует сеть и проводит аудит защиты. Использовался в фильме «Матрица: Перезагрузка» при взломе компьютера. Наша задача не взломать, а защитить ПК, поскольку одно и то же оружие можно использовать как для защиты, так и для нападения. Иначе говоря, сканером портов **nmap** можно определить открытые порты компьютера, а для безопасности сети пользователям рекомендуется закрытьдоступ к этим портам с помощью брандмауэра (рис. 16).

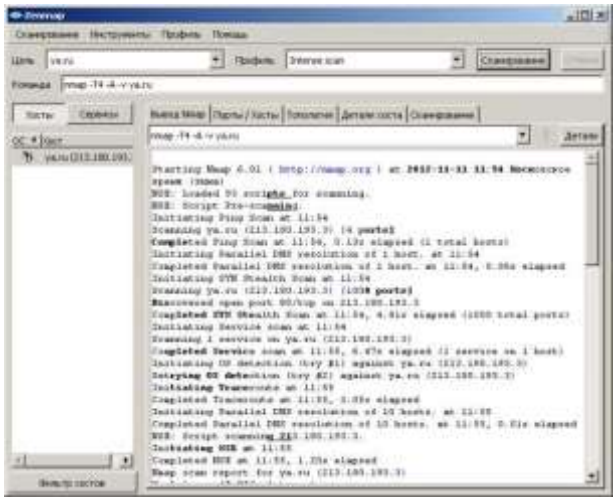


Рис. 16 - Интерфейс программы Nmap

Обычно для того, чтобы просканировать все порты какого-либо компьютера в сети вводится команда **nmap –p1-65535 IP-адрес**_компьютера или **nmap –sV IP-адрес компьютера,** а для сканирования сайта — команда**nmap –sS –sV –O - P0 адрес сайта.**

Монитор портов TCPView

TCPView — показывает все процессы, использующие Интернет-соединения. Запустив **TCPView**, можно узнать, какой порт открыт и какое приложение его использует, а при необходимости и немедленно разорвать соединение – рис. 17.

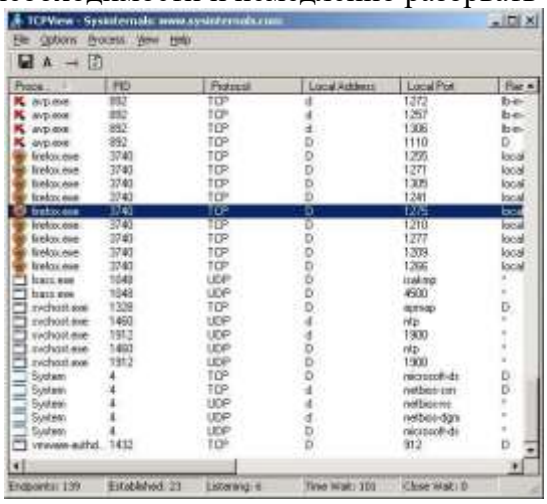


Рис. 17 - Главное окно программы TCPView

Просмотрите активные сетевые подключения локального ПК с помощью монитора портов tpiview. Определите потенциально возможные угрозы (какие порты открыты, и какие приложения их используют). При необходимости можно закрыть установленное приложением TCP-соединение или процесс правой кнопкой мыши.

Практические задания

Задание

Данное практическое занятие предполагает выполнение следующих этапов:

– изучить методические указания;
– ответить на контрольные вопросы.

Контрольные вопросы:

1. Какие виды мониторинга рабочих операций пользователя существуют?

2. Дайте характеристику современным программным средств мониторинга действий пользователей. Какое программное средство вы порекомендовали бы нашей организации? Почему?

3. Какие уязвимости ОС Windows были устранены в данной работе и какими путями?

4. Как узнать закрытые порты? Как открыть нужный порт?

5. Для чего используется программа NetStat Agent? Nmap? TCPView?

Лабораторная работа№22 Проверка наличия и сроков действия сертификатов

Цель занятия: познакомимся с вопросами использования цифровых сертификатов.

Краткие теоретические сведения

Начнем с использования сертификатов протоколами SSL и TSL (это два разных протокола, но т.к. TSL разработан на базе SSL 3.0, принцип использования сертификатов один и тот же).

Эти протоколы широко применяются в сети Интернет для защиты данных передаваемых между web-серверами и браузером клиента. Для аутентификации сервера в нем используется сертификат X.509.

Для примера обратимся на сайт «Нордеа Банка», в раздел «Войти в Интернет-банк», предназначенный для ведения банковских операций через Интернет (рисунок 5.1).

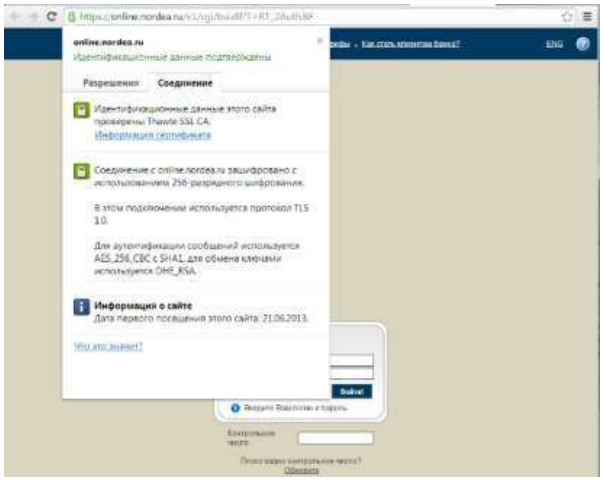


Рисунок 5.1 – Защищенное соединение

Префикс https в строке адреса и изображение закрытого замка справа от строки указывают, что установлено защищенное соединение. Если щелкнуть мышью по изображению замка, то увидим представленное на рисунке 5.1 сообщение о том, что подлинность узла с помощью сертификата подтверждает центр сертификации Thawte. Значит, мы на самом деле обратились на сайт Нордеа Банка (а не подделанный нарушителями сайт) и можем безопасно вводить логин и пароль.

Выбрав «Просмотр сертификата» можно узнать подробности о получателе и издателе, другие параметры сертификата (рисунок 5.2).

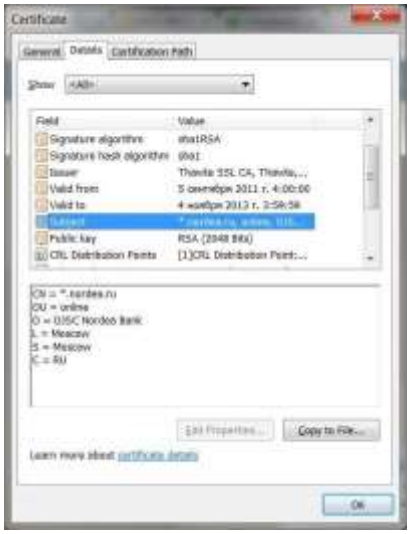


Рисунок 5.2 – Параметры сертификата

Операционная система Windows обеспечивает защищенное хранилище ключей и сертификатов. Работать с хранилищем можно используя настройку консоль управления MMC «Сертификаты».

Из меню Пуск – Выполнить (или «Командная строка») запустите консоль командой mmc.

Таким образом, мы можем просматривать сертификаты текущего пользователя. Если ранее сертификаты не запрашивались,то в разделе «Личное» элементов не будет. В разделе «Доверенные корневые центры сертификации» представлен достаточно обширный список центров, чьи сертификаты поставляются вместе с операционной системой.

Найдите в нем сертификат thawte Timestamping CA. Благо- даря тому, что он уже был установлен, в рассмотренном в начале работы примере с подключением к системам Интернет-банкинга браузер мог подтвердить подлинность узла.

Теперь перейдем к разделу «Сертификаты, к которым нет доверия». Там находятся отозванные сертификаты. Как минимум,там будут находиться два сертификата, которые по ошибке или злому умыслу кто-то получил от имени корпорации Microsoft. Когда это выяснилось, сертификаты отозвали. Сейчас этот списокнамного больше.

Теперь рассмотрим другой вариант – мы подключаемся по SSL к web-серверу, а браузер не может проверить его подлинность. Подобная ситуация произошла при подключении в раздел Интернет-обслуживания Санкт-Петербургского филиала опера- тора мобильной связи Tele2 -https:// www.selfcare.tele2.ru/ work.html (рисунок 5.3).

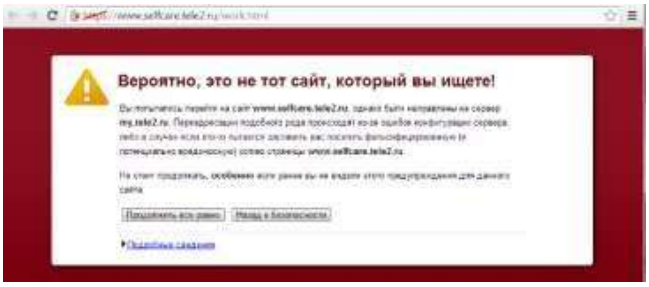


Рисунок 5.3 – Сообщение о проблеме с сертификатом

Если нажать ссылку «Продолжить всё равно» можно будет просмотреть сертификат.

Рассмотрим возможности, которые предоставляет Windows Server по созданию собственно центра сертификации (Certification Authority – CA) на предприятии.

Соответствующие службы присутствовали в серверных операционных системах семейства Windows, начиная с Windows 2000 Server.

В Windows Server 2012 для того, чтобы сервер смог работать как центр сертификации, требуется сначала добавить серверу роль Службы сертификатов Active Directory. Делается это помощью оснастки Диспетчер серверов (рисунок 5.4), которую можно запустить из меню Пуск.

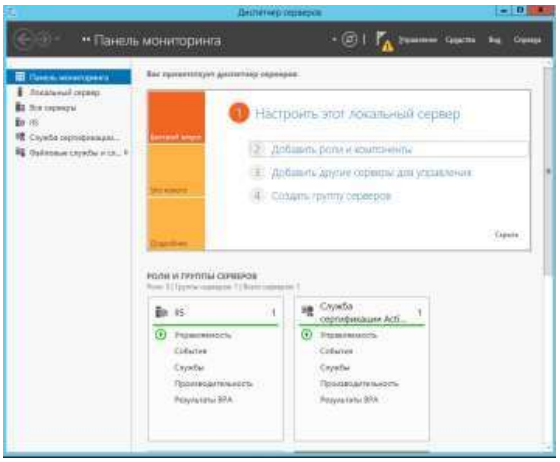


Рисунок 5.4 – Диспетчер серверов

В Server Manager раскроем список ролей и выберем добавление роли (рисунок 5.5) – Служба сертификатов Active Directory.

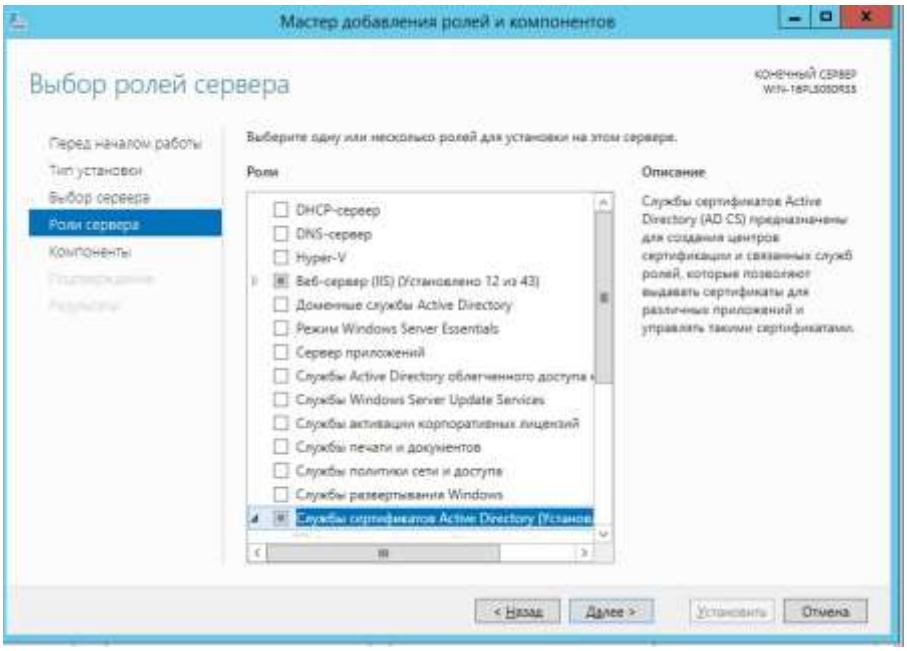


Рисунок 5.5 – Добавление роли

В нашем примере, роль добавляется серверу, который будет также контроллером домена Windows. Так как это первый CA в домене, он в нашей сети будет играть роль корневого (Root).

Контроллер домена в данной работе настраивать не требуется. Рассмотрим по шагам процедуру установки.

В дополнение к обязательному компоненту «Служба сертификатов Active Directory», могут быть установлены дополнительные средства, предоставляющие web-интерфейс для работы пользователей с CA (рисунок 5.6). Это может понадобиться, например, для выдачи сертификатов удаленным или внешним, не зарегистрированным в домене, пользователям. Для выполнения данной лабораторной работы это не понадобится.

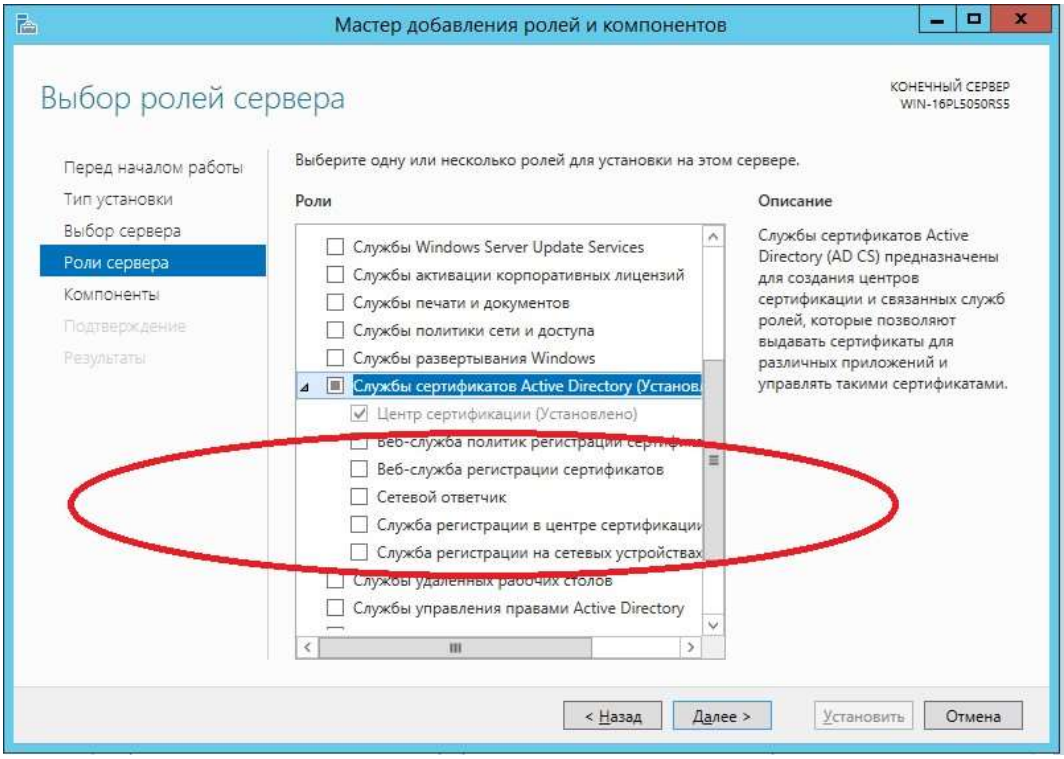


Рисунок. 5.6 – Выбор добавляемой роли

Следующий шаг – определения типа центра сертификации. Он может быть корпоративным (Enterprise) или отдельностоящим (Standalone). Разница заключается в том, что Enterprise CA может быть установлен только на сервер, являющийся членом домена, т. к. для его работы требуется служба каталога Active Directory. Standalone CA может работать вне домена, например, обрабатывая запросы пользователей, полученные через web-интерфейс.

Практические задания

Задание

Данное практическое занятие предполагает выполнение следующих этапов:

– изучить методические указания;
– ответить на контрольные вопросы.

Контрольные вопросы:

1. Какие протоколы применяются для защиты данных?
2. Типы центров сертификации.
3. Алгоритм создания центра сертификации.
4. Как осуществляется работа с хранилищем сертификатов?
5. Как можно узнать подробности о получателе и издателе, другие параметры сертификата?

Лабораторная работа№23 Разработка политики безопасности корпоративной сети

Цель занятия: получить навыки использования системы защиты информации в корпоративной сети.

Краткие теоретические сведения

Информация о пользователе информационной системы должна содержать следующее:

– имя пользователя;
– сведения о парольной защите;
– тип учетной записи.

Кроме того, в этом окне возможна настройка подсказки о пароле и мастера забытых паролей, позволяющих выполнять сброс и изменение забытого пароля, а также пароль доступа к сетевым ресурсам (для администратора).

Чтобы удалить зарегистрированного пользователя, необходимо выполнить следующее:

– выбрать имя нужного пользователя в списке;
– выбрать пункт «Удаление учетной записи».

Для того чтобы зарегистрировать нового пользователя в системе, необходимо произвести следующие действия:

– находясь в меню «Учетные записи пользователей» выбрать пункт «Создание учетной записи»;
– ввести с клавиатуры имя нового пользователя, например, «Начальник»;
– выбрать тип учетной записи;
– если необходимо, установить и подтвердить пароль. Чтобы изменить информацию о пользователе, нужно выполнить следующие действия:
– выбрать имя нужного пользователя в списке;
– изменить нужные параметры.

Управление учетными записями может выполняться в двух режимах: классического запроса пароля и приглашения, когда вход в систему выполняется простым кликом мышью на иконке пользователя. Изменение параметров входа доступно в меню «Учетные записи пользователей».

Настройка элементов политики безопасности. В операционной системе имеется возможность настройки элементов политики безопасности, регулирующей доступ к файлам и папкам.

Единственным условием является наличие версии Professional и файловой системы NTFS. Настройка производится пользовательским интерфейсом, доступ к которому открыт администратору системы.

Для того чтобы использовать его, следует открыть «Пуск/Панель управления/Администрирование/Локальная политика безопасности/Параметры безопасности/Политика учетных записей/Политика паролей/».

Рассмотрим основные параметры, необходимые для выполнения практического занятия.

Максимальный срок действия пароля. Этот параметр безопасности определяет период времени (в днях), в течение которого можно использовать пароль, прежде чем система потребует от пользователя заменить его. Срок действия пароля может быть установлен в пределах от 1 до 999 дней. При установке значения 0 срок действия пароля не ограничен. Если максимальный срок действия пароля составляет 1–999 дней, значение параметра

«Минимальный срок действия пароля» должно быть меньше этого значения. При установке значения 0 для максимального срока действия пароля для минимального срока действия может быть установлено любое значение в пределах от 0 до 998 дней.

Минимальная длина пароля. Этот параметр безопасности определяет наименьшее число символов, которое может содержать пароль учетной записи пользователя. Можно

задать значение в диапазоне от 1 до 14 символов или отменить использование пароля, установив число символов равным 0.

Минимальный срок действия пароля. Этот параметр безопасности определяет период времени (в днях), в течение которого необходимо использовать пароль, прежде чем пользователь сможет заменить его. Можно задать значение в диапазоне от 1 до 998 дней или разрешить немедленное изменение, установив число дней равным 0.

Пароль должен отвечать требованиям сложности. Этот параметр безопасности определяет требования сложности для паролей.

Если эта политика включена, пароли должны удовлетворять следующим минимальным требованиям:

1) пароль не может содержать имя учетной записи пользователя или какую-либо его часть;

2) пароль должен состоять не менее чем из шести символов;

3) в пароле должны присутствовать символы трех категорий из числа следующих четырех:

– прописные буквы английского алфавита от A до Z;

– строчные буквы английского алфавита от a до z;

– десятичные цифры (от 0 до 9);

– неалфавитные символы (например: !, $, #, %).

Проверка соблюдения этих требований выполняется при изменении или создании паролей.

Требование неповторяемости паролей. Этот параметр безопасности определяет число новых паролей, которые должны быть сопоставлены учетной записи пользователя, прежде чем можно будет снова использовать старый пароль. Это значение должно принадлежать диапазону от 0 до 24.

Данная политика позволяет администраторам повышать уровень безопасности, запрещая все время использовать одни и те же старые пароли.

Хранение паролей с использованием обратимого шифрования. Этот параметр безопасности определяет, использует ли операционная система обратимое шифрование для хранения паролей. Такая политика обеспечивает поддержку приложений, использующих протоколы, которым для проверки подлинности нужно знать пароль пользователя. Хранить пароли, зашифрованные обратимыми методами, – это все равно, что хранить их открытым текстом. Поэтому данную политику следует использовать лишь в исключительных случаях, если потребности приложения оказываются важнее, чем защита пароля.

Блокировка учетных записей. В системе существует возможность блокировки учетных записей. Задать условия блокировки можно, открыв интерфейс настройки: «Параметры безопасности/Политика учетных записей/Политика блокировки учетных записей/».

Рассмотрим параметры настройки.

Блокировка учетных записей. Этот параметр безопасности определяет число минут, в течение которых учетная запись остается блокированной, прежде чем будет автоматически разблокирована. Этот параметр может принимать значения от 0 до 99 999 мин. Если установлено значение 0 для длительности блокировки учетной записи, она останется заблокированной до тех пор, пока не будет явно разблокирована администратором.

Если пороговое значение блокировки определено, данный интервал блокировки должен быть больше или равен интервалу сброса.

Пороговое значение блокировки. Этот параметр определяет число неудачных попыток входа в систему, после которых учетная запись пользователя блокируется. Блокированную учетную запись нельзя использовать до тех пор, пока она не будет инициализирована администратором или пока не истечет интервал ее блокировки. Число

неудачных попыток входа в систему можно задать в интервале от 0 до 999. При установке значения 0 учетная запись пользователя никогда не будет блокироваться.

Сброс счетчика блокировки. Параметр определяет, сколько минут должно пройти после неудачной попытки входа в систему, прежде чем счетчик неудачных попыток будет сброшен в 0. Этот параметр может принимать значения от 1 до 99 999 мин.

Аудит (регистрация и учет событий). Важным компонентом системы защиты является система регистрации и учета, реализующая фиксирование событий доступа, в том числе несанкционированного.

Получить доступ к этим настройкам можно следующим образом: «Параметры безопасности/Локальные политики/Политика аудита/».

Рассмотрим их более подробно.

Аудит входа в систему. Позволяет контролировать корректность доступа пользователя в систему, в частности, количество неудачных попыток входа.

Аудит доступа к объектам. Этот параметр безопасности определяет, подлежит ли аудиту событие доступа пользователя к объекту – например, к файлу, папке, разделу реестра, принтеру и т. п., – для которого задана собственная системная таблица управления доступом (SACL).

Аудит доступа к службе каталогов. Определяет, подлежит ли аудиту событие доступа пользователя к объекту каталога Active Directory, для которого задана собственная системная таблица управления доступом (SACL).

Аудит изменения политики. Этот параметр определяет, подлежит ли аудиту каждый факт изменения политик назначения прав пользователей, политик аудита или политик доверительных отношений.

Аудит использования привилегий. Определяет, подлежит ли аудиту каждая попытка пользователя воспользоваться предоставленным ему правом.

Аудит отслеживания процессов. Определяет, подлежат ли аудиту такие события, как активизация программы, завершение процесса, повторение дескрипторов и косвенный доступ к объекту.

Аудит системных событий. Определяет, подлежат ли аудиту события перезагрузки или отключения компьютера, а также события, влияющие на системную безопасность или на журнал безопасности.

Аудит событий входа в систему. Определяет возможность отслеживания доступа пользователей.

Аудит управления учетными записями. Этот параметр безопасности определяет, подлежат ли аудиту все события, связанные с управлением учетными записями на компьютере.

Кроме того, возможна и настройка прав доступа пользователя с помощью отдельного инструмента политики безопасности, но настройка этих элементов в рамках работы не выполняется.

Контроль запуска программ. Операционная система Windows XP SP2 позволяет управлять доступом к файлам, запускающим приложения. Получить доступ к этим настройкам можно следующим путем: «Параметры безопасности/Локальные политики/Политика ограниченного использования программ/». По умолчанию таких политик не существует, поэтому необходимо создать собственную в меню «Действие». Рассмотрим параметры ее настройки. Уровни безопасности. Позволяет контролировать программы исходя из запрета или полного доступа к отдельной области.

Дополнительные правила. Этот параметр безопасности определяет, заданы ли пользователем дополнительные параметры безопасности. Для выполнения практического задания необходимо создать правило для пути в меню «Действие».

Контроль доступа к файлам и папкам. Позволяет контролировать доступ к файлам и папкам каждому пользователю, зарегистрированному в системе. Для доступности такой функции необходима система версии Professional и файловая система NTFS. При

соблюдении этого условия также проконтролируйте снятие флажка «Простой общий доступ к файлам и папкам», доступ к которому можно получить следующим образом: «Панель управления/Свойства папки/Вид».

Для настройки прав доступа пользователя к файлу или папке при выполненных условиях необходимо выполнить следующие действия:

– открыть вкладку «Безопасность» в свойствах файла или папки;

– перейти по кнопке «Дополнительно» в меню настройки;

– добавить или изменить запреты и разрешения с помощью кнопки «Изменить»;

– добавить или изменить правила аудита с помощью вкладки «Аудит».

Если пользователь отсутствует в списке, необходимо добавить его, используя следующую последовательность действий «Безопасность/Добавить/Дополнительно/Поиск» и выбрать из списка внизу нужного пользователя.

Можно заблокировать изменение настроек, включив флажок «Использовать простой общий доступ к файлам и папкам».

Система контроля доступа может применяться для задач блокирования, но она не предполагает установки парольной защиты на файлах и папках. Для этих целей используются дополнительные средства защиты информации.

Практические задания

Задание.

Создайте политику безопасности операционной системы для трех пользователей с заданными правами использования файлов и папок при данных условиях:

Администратор компьютера имеет доступ ко всем папкам пользователей на чтение, но не имеет право удалять или изменять файлы. Папка администратора C:\Admin. Бухгалтер – пользователь, имеет право записи и чтения в своей папке C:\bukh, не имеет доступа к папкам администратора, имеет доступ на запись к папке пользователя Начальник. Начальник – пользователь, имеет право записи и чтения в своей папке C:\chief, не имеет доступа к папкам администратора, имеет доступ на чтение к папке пользователя

Бухгалтер. Все нарушения системы защиты записываются в журнал безопасности системы.

Установите следующие настройки: количество циклов затирания (3), ограничения по паролю пользователей (минимальная длина – 6 символов, требования к сложности, требования к неповторяемости, время действия – 45 дней), максимальное число попыток входа – 5, аудит событий НСД (контролируется в журнале «Безопасность», находящемся «Панель управления /Администрирование/ Просмотр событий»), невозможность запуска программ из папки пользователя.

В каждой папке должны быть два файла, содержащие текстовую информацию, и один файл программы (с расширением .exe).

Контрольные вопросы:

1. В чем заключается сущность принципов функционирования механизма контроля доступа?
2. Как реализовано управление пользователями?
3. Как выполняется блокировка и разблокировка пользователя?
4. В чем заключаются функции персонального идентификатора?
5. В чем заключается сущность принципов функционирования системы аудита безопасности?
6. Какие существуют правила настройки и условия работы системы контроля доступа к файлам и папкам?

Лабораторная работа№24 Получение сертификата

Цель занятия: изучить цели, задачи и методологические проблемы обеспечения качества информационных технологий (ИТ), программных средств (ПС) и баз данных (БД) при сертификационных испытаниях.

Краткие теоретические сведения

7.2.1 Основные понятия и цели сертификации информационных технологий, программных средств и баз данных

Эффективность использования информационных технологий (ИТ) во многом определяется их качеством и доверием к ним пользователей. Возрастание роли важнейших компонент ИТ программных средств (ПС) и баз данных (БД) в народном хозяйстве, широты их применения и ответственности решаемых задач вызвало резкое повышение требований к их качеству.

По мере расширения применения ИТ выделились области, в которых ошибки или недостаточное качество программ может нанести ущерб, значительно превышающий положительный эффект от их использования. В таких критических случаях недопустимы аномалии функционирования программ при любых искажениях исходных данных, сбоях, частичных отказах аппаратуры и других нештатных ситуациях. Для этого испытания ИТ должны специально организовываться и документироваться, что объединяется понятием и процессом сертификации.

Архитектурная, техническая и программно-информационная совместимость современных сложных информационных систем (ИС) может быть обеспечена только путем стандартизации программно-технических средств в соответствии с требованиями международных стандартов. Для этого также необходима сертификация используемых средств, процессов и услуг, а также проведение единой технической политики при создании совместимых аппаратных средств и ПС, организации взаимодействия и комплексирования ИС различных уровней.

Сертификация соответствия ИТ, ПС и БД заключается в их формализованных испытаниях особо выделенным третейским коллективом специалистов, имеющим право на официальный государственный или ведомственный контроль функций и качества ИТ и гарантирующим их соответствие стандартам и другим нормативным документам, а также безопасность применения. Эти специалисты имеют право на расширение условий испытаний и создание различных критических и стрессовых ситуаций в пределах нормативной документации, при которых должны обеспечиваться заданное качество и безопасность результатов решения предписанных задач.

Если все испытания проходят успешно, то на соответствующую версию ИТ, ПС и БД оформляется и выдается специальный документ сертификат соответствия. Этот документ официально подтверждает соответствие стандартам, нормативным и эксплуатационным документам функций и характеристик испытанных средств, а также допустимость их применения в определенной области. Сертификат соответствия документально утверждает право на использование знаков соответствия требованиям сертификации, гарантирует безопасность применения, а также юридически допускает ИТ к эксплуатации и использованию по прямому назначению.

В зависимости от области применения ИТ, назначения и класса ПС и БД их сертификация может быть обязательной или факультативной. Эти виды сертификации близки концептуально и технологически, однако значительно различаются характеристиками объектов, правовым и экономическим взаимодействием между поставщиками, испытателями и пользователями. Обязательная (жесткая) сертификация ИТ необходима для ИС, выполняющих особо ответственные функции, в которых недостаточное качество, ошибки или отказы могут нанести большой ущерб или опасны для жизни и здоровья людей. Этот ущерб может определяться степенью безопасности применения ИТ в авиации, для управления в космосе и атомной энергетике или большими экономическими потерями вследствие недопустимого искажения служебной информации

в системах управления органов власти, банковских системах, системах управления войсками и др. В подобных системах сертификация ИТ способствует значительному снижению риска от их применения и повышению безопасности функционирования до необходимого уровня. В этих случаях разработчики и поставщики ИТ обязаны подвергать свои изделия независимой экспертизе на соответствие стандартам и конкретным требованиям качества для получения разрешения сертификационных центров на их реальную эксплуатацию по прямому назначению.

Факультативная (мягкая) сертификация применяется для удостоверения качества ИТ в целях повышения их конкурентоспособности, расширения сферы использования и получения дополнительных экономических преимуществ на рынке. Таким сертификационным испытаниям подвергаются компоненты операционных систем и пакеты прикладных программ широкого применения, повышение гарантий качества которых выгодно как для поставщиков, так и для пользователей ИТ. Затраты на сертификацию ИТ оправдываются повышением их цены, сокращением претензий пользователей, ростом тиража продаж и др. В этих случаях разработчики и поставщики добровольно предоставляют ИТ для сертификации с учетом экономических оценок выгодности ее проведения для их изделий.

При анализе процессов сертификационных испытаний ИТ, ПС и БД следует выделить ряд базовых компонент методологии сертификации, подлежащих последующему рассмотрению:

– цели сертификации формальные, технологические, правовые, экономические;

– проблемы, которые необходимо решать для обеспечения высокой эффективности и достоверности результатов сертификационных испытаний ИТ, ПС и БД;

– исходные данные и документы, необходимые для проведения сертификации стандарты и нормативные документы, их структура и содержание;

– характеристики и классификация программ и БД как объектов испытаний и сертификации, их показатели качества, позволяющие выделять однородные группы ПС и БД при проведении сертификации;

– ресурсы обеспечения испытаний финансовые, кадры специалистов, аппаратурная оснащенность, нормативно- технические и программно-инструментальные средства.

Проблемы сертификации ИТ, ПС и БД в принципе близки к тем, которые приходится решать для других видов изделий. Однако вследствие их новизны, высокой сложности объектов сертификации и многообразия их показателей качества выявился ряд особенностей этих проблем. При анализе сертификации ИТ, ПС и БД целесообразно выделить следующие проблемы:

– научно-методические, состоящие в создании эффективных по затратам ресурсов методов сертификационных испытаний ИТ, ПС и БД, которые гарантируют достоверное определение заданных показателей их качества и соответствие документации;

– технологические, заключающиеся в обеспечении реализации методов испытаний ИТ средствами автоматизации, тестирования и организации регламентированных проверок качества объектов и документации на разных этапах их создания и при непосредственных сертификационных испытаниях;

– проблемы стандартизации и нормативной документации, которые сводятся к созданию, последующему выбору и адаптации исходных документов, применяемых при сертификационных испытаниях определенных видов ИТ, ПС и БД;

– организационные, состоящие в создании международных, государственных и ведомственных органов, ответственных за сертификацию ИТ и их компонент, определении их прав и обязанностей, оснащении их необходимыми нормативно-методическими и инструментально-технологическими средствами;

– экономические, которые сводятся к выявлению, оценке и применению экономически эффективных методов использования ресурсов испытаний ИТ, ПС и БД, обеспечивающих заданную достоверность определения их качества, разработке экономических механизмов взаимодействия организаций и специалистов по сертификации с разработчиками, заказчиками и пользователями этих средств;

– правовые, сосредоточивающие в себе прежде всего создание юридических механизмов процессов сертификации и использования их результатов, создание нормативов, правил взаимодействия и распределения экономической и юридической ответственности между разработчиками, производителями, сертифицирующими организациями и поставщиками ИТ, ПС и БД за несоответствие реальных показателей качества гарантированным характеристикам сертифицированных изделий.

Ниже рассматриваются проблемы, которые наиболее близки к процессам непосредственных испытаний и определению качества ИТ, ПС и БД. Это методические, технологические, организационные проблемы, а также проблемы стандартизации и нормативной документации. Им сопутствуют задачи распределения экономической и юридической ответственности между испытателями, разработчиками, поставщиками и заказчиками за качество сертифицированной продукции и возможный ущерб при ее несоответствии документированному и объявленному качеству. Экономическими целями сертификации могут быть большие тиражи изделий при производстве, большая длительность жизненного цикла с множеством версий, снижение налогов за высокое качество и высокая прибыль разработчиков и поставщиков ИТ, ПС и БД. Результаты сертификации должны оправдывать затраты на ее проведение вследствие получения пользователями продукции более высокого и гарантированного качества при возможном повышении ее стоимости. Юридические проблемы сертификации и распределения ответственности за соответствие продукции, купленной пользователем, гарантиям, закрепленным сертификатом соответствия, должны решаться правоведами. При их решении необходимо отработать юридические механизмы распределения прибыли и затрат за обеспечение качества ИТ и нарушение их гарантированных значений.

Методически процесс сертификации представляет особую совокупность испытаний ИТ и их компонент, для которых необходимы специальные стандарты, методики, средства автоматизации и подготовленные специалисты. Далее раздельно рассматриваются особенности сертификации основных компонент ИТ ПС и БД. При этом в большинстве случаев, если это не может вызвать сомнений, подразумеваются и ИТ, которые они обеспечивают.

Работы по сертификации ПС объединяются в технологический процесс, на каждом этапе которого регистрируются документы, отражающие состояние и качество результатов контроля программ. В результате процесс сертификации отличается от обычных испытаний ПС более высоким уровнем формализации и документального оформления всех условий и результатов испытаний, проводимых специальным испытательным органом. Необходимость сертификации программ широкого класса привела к появлению ряда следующих научных и методических задач, тесно связанных с процессами разработки ПС высокого качества:

– для каждого вида ПС необходимо определять представительный набор характеристик качества и их значений, его категорию критичности, требуемую достоверность измерения показателей качества и организационный уровень удостоверения сертификата;

– в соответствии с требованиями к достоверности показателей качества должны определяться и минимизироваться содержание и объемы сертификационных испытаний версий ПС;

– для обеспечения качества и ответственности за результаты испытаний должны быть разработаны эффективные методы и методические нормативные документы, регламентирующие процессы сертификации различных видов ПС;

– технологические процессы сертификационных испытаний и измерений качества ПС и их компонент должны быть поддержаны достаточно эффективными средствами автоматизации и определения достоверности измеренных характеристик;

– исследование и обобщение опыта сертификации ПС должны способствовать минимизации затрат на такие испытания, а также улучшению технологических процессов разработки программ, гарантирующих достижение требуемых показателей качества для различных видов программ.

Кроме сертификации объектов разработки в некоторых случаях целесообразна сертификация технологии и средств автоматизации создания комплексов программ. Процесс сертификации технологии разработки ПС в принципе подобен испытаниям программ. Его важная особенность состоит в необходимости регулярного контроля за соблюдением всех характеристик качества технологического процесса всеми его участниками.

Исходные данные для сертификации ПС опираются на совокупность документов, выбираемых и адаптируемых с учетом конкретных объектов сертификации. Наиболее общие исходные данные сосредоточены в стандартах, посвященных непосредственно сертификации, аттестации, тестированию, испытаниям и обеспечению качества различных изделий и, в частности, компонент ИТ. Конкретные нормативные документы должны создаваться в соответствии с базовыми стандартами и содержать методики организации и проведения испытаний, а также контролируемые характеристики сертифицируемых объектов. Эти документы должны отражать все сведения, необходимые для корректного применения ПС по прямому назначению с показателями качества, гарантированными сертификатом соответствия.

Методология принятия решений о допустимости выдачи сертификата на ПС основывается на оценке степени его соответствия действующим и специально разработанным документам:

– международным и национальным стандартам на тестирование, испытания, аттестацию программ, требования которых не ниже требований, регламентируемых отечественными документами;

– международным и государственным стандартам на технологию создания ПС, взаимосвязь открытых систем, языки программирования и др.;

– стандартам на сопровождающую программную документацию с учетом необходимости и достаточности номенклатуры документов, семантической полноты и однозначности понимания содержания документов;

– нормативным документам на испытанное ПС техническим условиям, техническим описаниям, спецификациям требований и другим регламентирующим документам по выбору заказчика, разработчика и испытателя.

В исходных нормативных документах должны быть сосредоточены все функциональные и эксплуатационные характеристики проверяемого ПС, обеспечивающие заказчику и пользователям возможность корректного применения сертифицированного объекта во всем многообразии его функций и показателей качества. Номенклатура характеристик, сертифицируемых ПС строится на применении многоуровневых систем показателей качества, организованных по принципам квалиметрии и таксономических методов анализа. Выбор и ранжирование показателей должны производиться с учетом классов ПС, их функционального назначения, режимов эксплуатации, степени ответственности и жесткости требований к результатам функционирования и проявлениям возможных ошибок в программах. Для сертификации необходимо подготовить следующие исходные данные:

– критерии и четко определенные значения показателей качества, которые должны быть достигнуты для выдачи в последующем сертификата соответствия;

– значения исходных и результирующих данных, в пределах которых должны удовлетворяться заданные показатели качества;

– стандарты, нормативные документы и методики точных и воспроизводимых измерений показателей качества программ, а также состав и значения исходных и результирующих данных, обязательных для использования сертификации.

Имеется необходимость вносить в модифицированные версии отдельные небольшие изменения без полных повторных сертификационных испытаний ПС. При любых изменениях необходимы подтверждение сертификата и проведение некоторого минимума испытаний, удостоверяющих их корректность. Для этого используется система официальных уведомлений о проведенных изменениях и подтверждении сертификата. Для инициализации изменений также необходимы официальные уведомления пользователей о выявленных недостатках ПС или о предложениях по его совершенствованию. Таким образом, обычный процесс сопровождения ПС для сертифицируемых программ дополняется соответствующей системой последовательных официальных уведомлений и контрольных испытаний. При характеристиках и классификации программ как объектов сертификации основная цель классификации состоит в выделении однородных групп программ, имеющих такие показатели или признаки объекта, которые позволяют эффективно применять одинаковые или весьма близкие наборы показателей качества, технологии, методы и средства автоматизации испытаний и сертификации программ. Разнообразие объектов разработки не позволяет обеспечить достаточный уровень качества и технико-экономических показателей при единственной универсальной технологии и комплексе автоматизации испытаний программ. С другой стороны, нерентабельно для каждого нового типа программ создавать собственную технологию и средства автоматизации испытаний. Вследствие этого необходима классификация программ как база для рационального выбора методов и технологий сертификации, обеспечивающих необходимое качество программ и достаточно высокие технико-экономические показатели испытаний программ.

– Классификация программ и соответствующих технологий их испытаний прежде всего базируется на анализе риска от их недостаточного качества и возможного ущерба от проявления не выявленных ошибок при их функционировании у пользователей. С этой позиции по степени ответственности выполняемых функций можно выделить три группы программ:

– критические программы, от которых требуется особенно высокое качество функционирования, так как ошибки могут привести к катастрофическим последствиям порче ценного оборудования или даже угрозе здоровью и жизни людей;

– важные программы, которые должны обладать особенно высоким качеством, так как экономический ущерб от ошибок в них может быть велик, но невозможны особо катастрофические последствия;

– ординарные программы, недостатки которых не угрожают пользователям большим ущербом, являющиеся наиболее массовыми и широко распространенными, их качество и области применения изменяются в широких пределах и к ним принадлежат многие программы, отнесенные ниже к первой и второй категориям.

При оценке целесообразности сертификации необходимо учитывать возможный ущерб не только от кратковременного однократного неудачного применения ПС, но и возможный суммарный потенциальный ущерб от искажений и сбоев при длительной эксплуатации большого тиража версий ПС. Таким образом, в категорию важных ПС, подлежащих сертификации, могут попадать широко тиражируемые, длительно и активно применяемые программы, каковыми являются стандартизированные операционные системы, компиляторы, некоторые компоненты CASE-систем и др. В этих случаях испытания должны проводить также специализированные третейские организации, которые своим авторитетом и; соответствующим документом утверждают высокое качество программ для многочисленных пользователей.

Примером может служить аттестация компиляторов с языка Ада, проводимая специализированной организацией AJPO.

У некоторых специалистов сложилось отрицательное отношение к планированию обеспечения качества и детальным формализованным испытаниям программ. Это характерно для создания программ инженерных и научных расчетов, при некоторых вычислительных экспериментах, разработке программ обучения или бытового применения, которые не подвергают сертификационным испытаниям. В категорию ординарных программ входят программы, разрабатываемые и применяемые в отраслях народного хозяйства, как продукция производственно-технического назначения. Основная особенность этой категории программ состоит в промышленном характере их разработки, испытаний, производства и применении в виде ПС. Программы данной категории в большинстве своем принадлежат к группам важных или критических. Это обусловливает определенные регламентированные организационные формы их жизненного цикла, особенно высокие требования к качеству и документации, необходимость применения типовых проблемно-ориентированных технологий и средств автоматизации при разработке, испытаниях и производстве.

7.2.2 Методы достижения высокого качества ПС

При ограниченных ресурсах на разработку ПС для достижения заданных требований необходимо управление обеспечением качества в течение всего цикла создания программ. Адекватный набор показателей качества программ зависит от функционального назначения и свойств каждого ПС. В соответствии с принципиальными особенностями ПС выбираются номенклатура и значения показателей качества, которые отражаются в техническом задании и спецификации требований на конечный продукт. Каждый критерий может использоваться, если определена его метрика, может быть указан способ ее измерения и сопоставления с требующимся значением. Основным методом измерения качества программ на любых этапах разработки является тестирование. Результаты тестирования и измерения показателей сравниваются с требованиями технического задания или спецификаций для определения степени соответствия предъявлявшимся требованиям, полученным разработчиком от заказчика. Такие достаточно полные эталоны, как совокупность требований технического задания и поэтапная их декомпозиция в спецификациях, являются необходимой базой тестирования при промежуточных и завершающих испытаниях.

За ограниченный, относительно короткий, период сертификационных испытаний трудно провести достаточно обширное тестирование, достоверно демонстрирующее достигнутые показатели качества, и гарантировать выполнение всех технических требований к сложному ПС. Поэтому для обеспечения высокого качества программ целесообразно проводить испытания не только завершенного разработкой ПС, но на ряде промежуточных этапов разработки проверять состояние и характеристики компонент проекта. Критические ПС невозможно создавать без проведения 68 этапов промежуточных испытаний и применения целой системы поэтапного контроля качества. Для этого до начала разработки в процессе формирования технического задания формулируются план и основные положения методики обеспечения качества, поэтапных испытаний компонент и определения характеристик, допустимых для продолжения разработки на следующем этапе. Одновременно происходит поэтапное уточнение технического задания и методик сертификационных испытаний программ. В этом случае испытатели и представители заказчика получают возможность более полно ознакомиться с создаваемым ПС, а также контролировать качество его компонент и достаточно полно их учитывать при заключительных сертификационных испытаниях.

7.2.3 Достоверность результатов испытаний ПС

Сравнение результатов функционирования проверяемого комплекса программ и его компонент на соответствие эталонам предполагает использование критериев оценки величин отклонения от эталонов и принятие решений о степени корректности. Величины

допусков зависят от типа проверяемой программы, метода и этапа проверки ее корректности.

Степень соответствия проверяемых программ эталонам зависит от достоверности функционирования всех компонент, участвующих в установлении корректности. Отклонение проверяемых результатов от эталонов за допустимые пределы может произойти не только вследствие некорректности программ, но и из-за недостаточной точности средств сравнения или эталонов.

Процессы испытаний происходят во времени и их динамические характеристики могут служить частными критериями для оценки достигнутого качества тестирования. Таким критерием может быть интенсивность обнаружения ошибок или количество ошибок, выявляемых в программах в процессе тестирования за единицу времени при постоянных усилиях на его проведение.

7.2.4 Проблемы сертификации БД

БД это набор записей информации, который определен по- средством схемы, не зависящей от программ, которые к ней обращаются. Цель сертификации БД защитить требования потребителей к качеству используемой информационной продукции, содержащейся в БД, по полноте, достоверности, актуальности, защищенности и другим показателям. Обобщенным показателем качества информации в БД является степень ее соответствия существующим стандартам и другим нормативно-техническим документам как в содержательной, так и в форматно-структурной части. Сертификационные испытания БД в некоторых случаях, например, при применении их в критических информационных системах, должны проводиться обязательно. Однако чаще сертификация БД имеет факультативный характер, позволяющий пользователям иметь дополнительную гарантию ее качества.

При испытаниях и сертификации возникает проблема определения состава и использования реально существующих международных и отечественных стандартов и других нормативно- технических документов, которым должны соответствовать сертифицированные БД. Стандарты и документы должны охватывать:

– терминологию в области ИТ и систем;
– порядок организации и создания БД;
– концепции структурного построения, взаимодействия компонент и языки описания БД;
– комплектность документов, сопровождающих БД, и требования к ним; показатели качества БД, ИТ и ПС;
– методы и руководства по испытаниям, аттестации и сертификации компонент и БД в целом.

Непосредственно БД посвящены международные стандарты только на языки БД и на некоторые принципы построения БД. Однако имеются развитые системы стандартов по обработке информации и ИТ, в которых не упоминаются конкретно БД, но их положения по терминам и определениям, кодированию и документированию, жизненному циклу и показателям качества могут быть успешно применены при испытаниях и сертификации БД. Кроме того, при испытаниях й сертификации БД по мере необходимости следует учитывать стандарты в областях защиты информации, текстовых и учрежденческих систем, издательского дела, управления торговлей и транспортом и др. В некоторых стандартах имеются разделы, регламентирующие аттестацию ИТ и БД на соответствие данному стандарту. Эти разделы должны использоваться при подготовке методик сертификации БД на соответствие международным стандартам. Наиболее трудными проблемами при организации и проведении сертификации БД являются:

– классификация БД по характеристикам и сферам их применения;
– определение номенклатуры и требуемых показателей качества БД;

– создание методик тестирования и испытаний БД и их компонент, а также методов и средств достоверного измерения показателей качества БД;
– организация, регламентирование и документирование сертификации БД.

7.2.5 Назначение и особенности современных БД

В ИТ и процессах обработки информации на ЭВМ всегда присутствуют две базовые компоненты: программы, которые реализуют функции обработки, и данные, используемые в процессе обработки. В предшествующих разделах акцент был сосредоточен на анализе и испытаниях ИТ, основные особенности которых заключаются в ПС. При анализе БД на передний план выходит информация, подлежащая накоплению, хранению, обработке и использованию. Соответственно смещается акцент при испытаниях качества БД и при их эксплуатации. Однако при этом сохраняется достаточно важная роль ПС, реализующих все процедуры обработки данных.

Таким образом, при анализе БД как объектов испытаний й сертификации целесообразно рассматривать две компоненты: ПС управления данными и совокупность данных, упорядоченных по некоторым правилам. При этом одна и та же система управления БД может обрабатывать различные по структуре, составу и содержанию данные, а одни и те же данные могут управляться ПС различных СУБД. Хотя эти компоненты тесно взаимодействуют при реализации конкретной прикладной БД, первоначально они создаются независимо и могут рассматриваться как два объекта испытаний. Однако, в конечном счете пользователей интересуют совокупные характеристики качества конкретной используемой БД. Поэтому завершающие испытания и окончательная сертификация БД должны проводиться для проверки функционирования и удостоверения показателей качества во взаимодействии с предполагаемой для использования СУБД, с вполне определенным наполнением БД.

БД графической, речевой, мультимедиа и другой нетрадиционной информации только входят в практику и носят преимущественно экспериментальный характер. Результаты функционирования таких БД отражаются графическими, звуковыми или визуальными образами, качество которых оценивается человеком в значительной степени субъективно. Вследствие этого испытания подобных систем пока слабо формализованы и их сертификация производится редко. Наиболее широко вошли в практику БД для фактографической, документальной, словарной и текстовой информации, для которых накоплен большой опыт использования и испытаний. Поэтому ниже рассматривается сертификация таких БД.

7.2.6 Показатели качества БД

Особенности современных БД и обеспечивающих их СУБД являются следствием возрастающего спектра функций по обработке информации и разнообразия обрабатываемых данных. Отсюда появилось множество наборов показателей качества, определяющих функциональную пригодность каждой БД. Формирование таких наборов представляет собой сложную задачу системного анализа, характеризующуюся оригинальным решением для каждой прикладной проблемно-ориентированной области. Вследствие этого испытания и определения в процессе проектирования достигнутых показателей качества БД отличаются большим разнообразием методов и средств автоматизации.

В рассматривавшихся выше ИТ основное внимание сосредоточено на испытаниях ПС. В системах БД доминирующее значение приобретают сами данные, их хранение и обработка. Поэтому БД при анализе их качества целесообразно разделить на две компоненты:

– ПС СУБД, не зависимые от сферы их применения и смыслового содержания накапливаемых и обрабатываемых данных;

– БД, доступные для обработки и использования в конкретной проблемно-ориентированной сфере применения.

Первой компонентой для испытаний является комплекс программ СУБД. Сертификации должно подвергаться ПС обработки данных на соответствие стандартам и нормативно-техническим документам, адекватным функциям и характеристикам области использования. Методы и технология сертификации, в основном, подобны применяемым при испытаниях других сложных ПС. При этом специфика испытаний так же, как и для других типов ПС, сосредоточивается на выборе адекватных показателей качества из стандартной номенклатуры, особенностях генерации тестов и обработке результатов тестирования. Поставщиками информации для СУБД чаще всего являются специалисты-пользователи и они же должны выступать в роли генераторов тестов. Часть тестов может носить достаточно абстрактный характер и автоматизировано формироваться для заполнения БД и испытания основных операций обработки данных.

Часть функций, связанных с телекоммуникацией, должна испытываться на соответствие стандартам и протоколам телекоммуникации и взаимосвязи открытых систем. Другая значительная часть функций непосредственно обусловлена спецификой применения, распределенной СУБД. Некоторые из этих функций регламентируются специальными стандартами, в которых, в частности, представлены рекомендации по их аттестации. Кроме того, для распределенных СУБД значительно возрастает номенклатура сочетаний типов ЭВМ, выполняющих роль клиентов и серверов. Комбинаторика подобных типов ЭВМ и их операционных систем может быть весьма велика, и при испытаниях СУБД трудно охватить и проверить все особенности взаимодействия в таких распределенных СУБД. Поэтому сертификаты распределенных СУБД должны отражать номенклатуру типов ЭВМ и операционных систем, для которых они предназначены.

Второй компонентой для испытаний БД является собственно накапливаемая и обрабатываемая информация в БД. Показатели качества для БД значительно отличаются от применяемых при испытаниях ПС. Однако может сохраняться общий подход к определению и выделению адекватной номенклатуры показателей качества и их упорядочению. Он состоит в том, что выделяемые показатели качества должны иметь практический интерес для пользователей БД и быть упорядочены в соответствии с приоритетами практического применения. Кроме того, каждый выделяемый для проверки показатель должен быть пригоден для достаточно достоверного измерения и сравнения с требуемым значением при испытаниях и сертификации.

При этом подлежат тестированию, испытаниям и измерению показатели качества информации в БД и определение их соответствия стандартам и технической документации, а также проверяются состав и содержание сопровождающих БД документов. Для сертификации разрабатываются программа и методики испытаний, обеспечивающие достоверную проверку реальных показателей качества БД. Результаты испытаний оформляются протоколами и актом. При положительных результатах заявителю выдается сертификат соответствия.

Так же, как и для ПС, показатели качества БД можно разделить на функциональные и конструктивные. Функциональные показатели качества БД включают:

– полноту накопленных описаний объектов относительное число объектов или документов, имеющихся в БД, к общему числу объектов по данной тематике или по отношению к числу объектов в аналогичных БД по той же тематике;

– достоверность степень соответствия данных об объектах в БД реальным объектам вне ЭВМ в данный момент времени, определяющаяся изменениями самих объектов, некорректностями записей о их состоянии или некорректностями расчетов их характеристик;

– идентичность данных относительное число описаний объектов, не содержащих ошибки, к общему числу документов об объектах в БД;

– актуальность данных относительное число морально устаревших данных об объектах в БД к общему числу накопленных и обрабатываемых данных.

К конструктивным показателям качества информации в БД относятся, в основном, объемно-временные характеристики сохраняемых и обрабатываемых данных:

– объем базы данных число записей описаний объектов или документов, доступных для хранения и обработки в БД;

– оперативность степень соответствия динамики изменения данных в процессе сбора и обработки состояниям реальных объектов или величина запаздывания между появлением реального объекта и его отражением в банке данных;

– периодичность промежуток времени между поставками двух последовательных достаточно различающихся версий БД;

– глубина ретроспективы интервал времени от даты выпуска и/или записи в БД самого раннего документа до настоящего времени;

– динамичность относительное число изменяемых описаний объектов к общему числу записей в БД за некоторый интервал времени, определяемый периодичностью издания версий БД.

Кроме того, к конструктивным относятся все показатели защищенности информации. Защищенность реализуется, в основном, ПС СУБД, однако в сочетании с поддерживающими их средствами организации данных. В распределенных БД показатели защищенности тесно связаны с характеристиками целостности данных. Эти показатели отражают степень тождественности данных в памяти удаленных компонент распределенной БД.

К конструктивным относятся также показатели, определяющие форматную, лингвистическую и физическую совместимость БД. Форматная совместимость характеризуется степенью соответствия данных в БД требованиям стандартов на форматы представления данных для документальных, фактографических и словарных БД. Лингвистическая совместимость определяется степенью использования в БД единых лингвистических средств (классификаторов, рубрикаторов, словарей), формализованных соответствующими стандартами. Физическая совместимость заключается в соответствии БД на машиночитаемых носителях информации. Кроме того, каждая БД должна содержать идентификационные признаки (наименование, тематику, область применения, тип), правовые характеристики авторов разработки СУБД и БД и авторское право пользователей на информацию, содержащуюся в БД, в соответствии с конвенцией и законами об охране авторских прав.

Перечисленные характеристики отражают качество совокупности данных без учета динамики их использования пользователями в процессе эксплуатации. При реальном функционировании БД важную роль играют временные характеристики взаимодействия конечных пользователей и администраторов БД в процессе эксплуатации БД по прямому назначению. Эти характеристики зависят от качества СУБД, а также от структуры и показателей качества используемой информации. Они отражаются критерием эффективности использования ресурсов ЭВМ ПС, в данном случае СУБД. Для БД важнейшим ресурсом является память ЭВМ, занимаемая информацией БД. Эти показатели качества влияют на время реакции БД на разные виды запросов пользователей и пропускную способность БД при эксплуатации. Значения ряда других показателей качества ПС, составляющих СУБД, существенно зависят от характеристик и организации информации в БД. Поэтому при испытаниях и сертификации БД номенклатура показатели качества не может ограничиваться характеристиками информации в БД, а должна включать ряд дополнительных показателей, отражающих комплексную эффективность и функциональную полезность применения СУБД и БД пользователями в реальных условиях.

7.2.7 Ресурсы для сертификации ИТ

В зависимости от характеристик объекта сертификации на ее выполнение выделяются ресурсы различных видов. В результате сложность ИТ и доступные ресурсы становятся косвенными критериями или факторами, влияющими на выбор методов испытаний и достигаемое качество компонент ИТ.

Наиболее общим видом ресурсов, используемых при испытаниях ИТ, являются допустимые финансовые затраты или договорная стоимость сертификации компонент ИТ. При анализе эти показатели могут применяться как вид ресурсных ограничений или как оптимизируемый критерий.

Кадры специалистов можно оценивать численностью, а также тематической и технологической квалификацией. В испытаниях сложных ИТ участвуют системные аналитики и руководители различных рангов, программисты и вспомогательный обслуживающий персонал в некотором рациональном сочетании. Определяющими являются совокупная численность и структура коллектива, а также его подготовленность к коллективной проверке конкретного типа ИТ.

Аппаратурная оснащенность испытателей конкретных ПС или БД определяется прежде всего ресурсами и другими характеристиками реализующей и технологической ЭВМ, доступных для использования коллективу специалистов при сертификации. Тип реализующей ЭВМ, ее ресурсы, архитектура и система команд определяют возможность размещения на ней комплекса автоматизации контроля и регистрации результатов испытаний. Ресурсы технологической ЭВМ важны для сертификации не только по своим абсолютным значениям, но также и относительно численности коллектива специалистов, участвующих в испытаниях. Абсолютные ресурсы технологической ЭВМ определяют принципиальную возможность размещения и функции комплекса автоматизации сертификации, а относительные достигаемую эффективность его использования коллективом специалистов.

Практические задания

Задание

Данное практическое занятие предполагает выполнение следующих этапов:

– изучить методические указания;
– ответить на контрольные вопросы.

Контрольные вопросы:

1. В чем заключается сертификация соответствия программных средств?
2. Что подтверждает сертификат соответствия программного средства?
3. Виды сертификации программных средств.
4. В чем заключается факультативная сертификация?
5. Базовые компоненты методологии сертификации.
6. Проблемы при анализе сертификации.
7. Методы достижения высокого качества программных средств.
8. Достоверность результатов испытаний программных средств.
9. Проблемы сертификации баз данных.
10. Ресурсы для сертификации информационных технологий.

Перечень учебных изданий, Интернет ресурсов, дополнительной литературы

Основные источники

1. Беспалов, Д. А. Администрирование баз данных и компьютерных сетей : учебное пособие / Д. А. Беспалов, А. И. Костюк ; Южный федеральный университет. РостовнаДону ; Таганрог : Южный федеральный университет, 2020. 127 с. : ил., табл. URL: https://biblioclub.ru/index.php?page=book&id=612220 (дата обращения: 29.03.2022). Режим доступа: ЭБС Университетская библиотека онлайн, для зарегистрир. пользователей. Библиогр. в кн. ISBN 9785927535774. – Текст : электронный.
2. Илюшечкин, В. М. Основы использования и проектирования баз данных : учебник для среднего профессионального образования / В. М. Илюшечкин. — испр. и доп. — Москва : Издательство Юрайт, 2022. — 213 с. — (Профессиональное образование). — ISBN 9785534012835. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://ezpro.fa.ru:3217/bcode/491755
3. Казарин, О. В. Программноаппаратные средства защиты информации. Защита программного обеспечения : учебник и практикум для среднего профессионального образования / О. В. Казарин, А. С. Забабурин. Москва : Юрайт, 2022. 312 с. (Профессиональное образование). URL: https://ezpro.fa.ru:3217/bcode/497433 (дата обращения: 29.03.2022) Режим доступа: ЭБС Юрайт, для зарегистрир. пользователей. ISBN 9785534132212. Текст : электронный.
4. Лагоша, О. Н. Сертификация информационных систем : учебное пособие для спо / О. Н. Лагоша. 2е изд., стер. СанктПетербург : Лань, 2021. 112 с. URL: https://e.lanbook.com/book/156616 (дата обращения: 09.03.2022). Режим доступа: ЭБС Лань, для зарегистрир. пользователей. ISBN 9785811472123. Текст : электронный.
5. Тенгайкин, Е. А. Организация сетевого администрирования. Сетевые операционные системы, серверы, службы и протоколы. Практические работы : учебное пособие для спо / Е. А. Тенгайкин. 2е изд., стер. СанктПетербург : Лань, 2022. 100 с. URL: https://e.lanbook.com/book/198497 (дата обращения: 29.03.2022). Режим доступа: ЭБС Лань, для зарегистрир. пользователей. ISBN 9785811497836. Текст : электронный.
6. Черников, Б. В. Управление качеством программного обеспечения : учебник / Б. В. Черников. Москва : ФОРУМ : ИНФРАМ, 2022. 240 с. (Среднее профессиональное образование). URL: https://znanium.com/catalog/product/1850732 (дата обращения: 29.03.2022). Режим доступа: ЭБС Znanium.com, для зарегистрир. пользователей. ISBN 9785819909027. Текст : электронный.

Дополнительные источники

1. Гвоздева, Т. В. Проектирование информационных систем. Стандартизация, техническое документирование информационных систем : учебное пособие для спо / Т. В. Гвоздева, Б. А. Баллод. 2е изд., стер. СанктПетербург : Лань, 2021. 216 с. URL: https://ezpro.fa.ru:3178/book/176672 (дата обращения: 29.03.2022). Режим доступа: ЭБС Лань, для зарегистрир. пользователей. ISBN 9785811484140. Текст : электронный.
2. Губин, А. Н. Проектная оценка надежности информационных систем : учебное пособие / А. Н. Губин. СанктПетербург : СПбГУТ им. М.А. БончБруевича, 2019. 77 с. URL: https://e.lanbook.com/book/180062 (дата обращения: 09.03.2022). Режим доступа: ЭБС Лань, для зарегистрир. пользователей. Текст : электронный.
3. Даева, С. Г. Основы системного администрирования и администрирования СУБД : учебнометодическое пособие / С. Г. Даева. Москва : РТУ МИРЭА, 2021. 75 с. URL: https://e.lanbook.com/book/171547 (дата обращения: 29.03.2022). Режим доступа: ЭБС Лань, для зарегистрир. пользователей. Текст : электронный.
4. Журавлев, А. Е. Корпоративные информационные системы. Администрирование сетевого домена : учебное пособие для спо / А. Е. Журавлев, А. В. Макшанов, Л. Н. Тындыкарь. СанктПетербург : Лань, 2021. 172 с. URL: https://e.lanbook.com/book/176675 (дата обращения: 29.03.2022). Режим доступа: ЭБС Лань, для зарегистрир. пользователей. ISBN 9785811484171. – Текст : электронный.

5. Мартишин, С. А. Базы данных: Работа с распределенными базами данных и файловыми системами на примере MongoDB и HDFS с использованием Node.js, Express.js, Apache Spark и Scala : учебное пособие / С. А. Мартишин, В. Л. Симонов, М. В. Храпченко. Москва : ИНФРАМ, 2021. 235 с. + Доп. материалы (Среднее профессиональное образование). URL: https://znanium.com/catalog/product/1189321 (дата обращения: 29.03.2022). Режим доступа: ЭБС Znanium.com, для зарегистрир. пользователей. ISBN 9785160156439. Текст : электронный.

Электронные издания (электронные ресурсы)

1. http://www.ed.gov.ru – Министерство образования Российской федерации.
2. http://www.edu.ru – Федеральный портал «Российское образование».
3. http://www.rambler.ru – Русская поисковая система.
4. http://www.yandex.ru – Русская поисковая система.
5. http://biblioteka.net.ru – Библиотека компьютерных учебников.
6. http://www.britannica.com – Библиотека Britannica.
7. http://ict.edu.ru/lib/ Библиотека портала «ИКТ в образовании»
8. Единое окно доступа к образовательным ресурсам http://window.edu.ru/
9. Министерство образования и науки РФ ФГАУ «ФИРО» http://www.firo.ru/
10. Портал «Всеобуч» справочноинформационный образовательный сайт, единое окно доступа к образовательным ресурсам –http://www.eduall.ru/
11. Экономико–правовая библиотека [Электронный ресурс]. — Режим доступа: http://www.vuzlib.net.
12. http://www.consultant.ru. Справочноправовая система «Консультант Плюс»
13. http://www.garant.ru Справочноправовая система «Гарант».
14. http://www.nalog.ru. Официальный сайт Федеральной налоговой службы
15. http://znanium.com – Электроннобиблиотечная система znanium.com
16. http://www.urait.ru – электронная библиотека издательства ЮРАЙТ