Федеральное агентство по образованию РФ

Государственное образовательное учреждение

Высшего профессионального образования

Владимирский Государственный Университет

**Кафедра информационных систем и информационного менеджмента**

**Лабораторная работа №1**

По дисциплине «Управление данными»

**Выполнил:**

студент группы ИСГ-108

Кутырев Д.

**Приняла:**

Проскурина Г.В.

Владимир 2010

**ЗНАКОМСТВО С СУБД INTERBASE. ВЫПОЛНЕНИЕ ЭЛЕМЕНТАРНЫХ ОПЕРАЦИЙ ПРИ РАБОТЕ С СУБД**

1. **Цель работы.**

Изучить технические возможности системы управления базами данных (СУБД) InterBase при работе с простейшими базами данных (БД) и средства формирования SQL запросов к СУБД.

1. **Вариант индивидуального задания к работе.**

Вариант 14. ИС «Персональные ЭВМ». Минимальный список характеристик:

фирма-изготовитель, тип процессора, тактовая частота, объем оперативной памяти, объем жесткого диска, дата выпуска; сведения о фирмах, реализующих ЭВМ: наименование, адрес, телефон.

1. **Анализ индивидуального задания с описанием таблиц.**

Create database PEVM;

Create table EVM (

Id integer not null primary key, firma char(20), tip\_processora char(30), tactovaya\_castota char(10), OP char(10), zhestkii\_disk char(10), date\_vypuska integer);

Create table Firma(

Id integer not null primary key, naimenovanie char(20), adres char(30), telefon integer, id firms integer);

Create table Magazin(

Id not null primary key, id firma integer, stoimost integer);

**Заполнение таблиц:**

Insert into EVM values(1,’HP’,’intel’,’2\_GHz’,’3\_gb’,’250\_gb’,2008);

Insert into EVM values(2,’Acer’,’intel’,’2\_GHz’,’2\_gb’,’500\_gb’,2009);

Insert into EVM values(3,’Asus’,’intel’,’3\_GHz’,’3\_gb’,’160\_gb’,2010);

Insert into EVM values(4,’Apple’,’intel’,’1\_GHz’,’4\_gb’,’250\_gb’,2009);

Insert into EVM values(5,’Toshiba’,’intel’,’1\_GHz’,’2\_gb’,’360\_gb’,2010);

Insert into EVM values(6,’Samsung’,’intel’,’2\_GHz’,’1\_gb’,’250\_gb’,2008);

Insert into Firma values(1,’HP’,’London’,223322,1);

Insert into Firma values(2,’Acer’,’Cardiff’,532670,2);

Insert into Firma values(3,’Asus’,’Moskva’,123456,3);

Insert into Firma values(4,’Apple’,’Vashington’,234567,4);

Insert into Firma values(5,’Toshiba’,’Tokio’,345678,5);

Insert into Firma values(6,’Samsung’,’Paris’,456789,6);

Insert into Magazin values(1,1, 30000);

Insert into Magazin values(2,2, 25000);

Insert into Magazin values(3,3, 21500);

Insert into Magazin values(4,4, 23000);

Insert into Magazin values(5,5, 32000);

Insert into Magazin values(6,6, 27500);

**Таблицы с исходными данными:**

**Вывод:**

Были изучены технические возможности СУБД MySQL, при работе с БД и средства формирования SQL запросов к СУБД. Была создана БД средствами DDL, состоящая из трех таблиц.

**Ответы на контрольные вопросы:**

1. Реляционная алгебра — формальная система манипулирования отношениями в реляционной модели данных. Существует в двух несколько различающихся вариантах:

алгебра Кодда (Э. Кодд, 1970)

алгебра A (К. Дейт, Х. Дарвен)

Наряду с реляционным исчислением является способом получения результирующего отношения в реляционной модели данных.

1. **Теоретико-множественные операторы**

**Объединение**

Отношение с тем же заголовком, что и у совместимых по типу отношений A и B, и телом, состоящим из кортежей, принадлежащих или A, или B, или обоим отношениям.

Синтаксис:

A UNION B

**Пересечение**

Отношение с тем же заголовком, что и у отношений A и B, и телом, состоящим из кортежей, принадлежащих одновременно обоим отношениям A и B.

Синтаксис:

A INTERSECT B

**Вычитание**

Отношение с тем же заголовком, что и у совместимых по типу отношений A и B, и телом, состоящим из кортежей, принадлежащих отношению A и не принадлежащих отношению B.

Синтаксис:

A MINUS BСпециальные реляционные операторы

1. Выборка (ограничение)

Отношение с тем же заголовком, что и у отношения A, и телом, состоящим из кортежей, значения атрибутов которых при подстановке в условие c дают значение ИСТИНА. c представляет собой логическое выражение, в которое могут входить атрибуты отношения A и/или скалярные выражения.

Синтаксис:

A WHERE c

Проекция

Отношение с заголовком (X, Y, …, Z) и телом, содержащим множество кортежей вида (x, y, …, z), таких, для которых в отношении A найдутся кортежи со значением атрибута X равным x, значением атрибута Y равным y, …, значением атрибута Z равным z. При выполнении проекции выделяется «вертикальная» вырезка отношения-операнда с естественным уничтожением потенциально возникающих кортежей-дубликатов.

Синтаксис:

A[X, Y, …, Z]

или

PROJECT A {x, y, …, z}

Соединение

Операция соединения есть результат последовательного применения операций декартового произведения и выборки. Если в отношениях и имеются атрибуты с одинаковыми наименованиями, то перед выполнением соединения такие атрибуты необходимо переименовать.

Синтаксис:

(A TIMES B) WHERE c

Деление

Отношение с заголовком (X1, X2, …, Xn) и телом, содержащим множество кортежей (x1, x2, …, xn), таких, что для всех кортежей (y1, y2, …, ym) ∈ B в отношении A(X1, X2, …, Xn, Y1, Y2, …, Ym) найдется кортеж (x1, x2, …, xn, y1, y2, …, ym).

Синтаксис:

A DIVIDEBY B

**Data Definition Language (DDL)** – язык определения данных, позволяющий создавать, удалять и изменять объекты в базах данных; 4. **Синтаксис DDL команд**

Оператор создания таблиц БД позволяет создавать новые таблицы и указывать связи между ними. Синтаксис:

CREATE TABLE <Имя таблицы> {<Определение столбца> |

[<Ограничение таблицы>]},

где:

<Определение столбца>::=

<имя столбца>

{<имя домена>|<тип данных>}

[<ограничение столбца>]

<Ограничение столбца>::=

UNIQUE|

PRIMARY KEY|

FOREIGN KEY (<имя столбца>)|

REFERENCES <имя таблицы>[(<имя столбца>…)] NOT NULL.

Имена таблиц и определения столбцов должны быть уникальными.

Назначение каждого атрибута в <Ограничение столбца> приведено далее:

• UNIQUE – определяет уникальное (не повторяющееся значение атрибута);

• PRIMARY KEY – первичный ключ;

• FOREIGN KEY – внешний ключ;

• REFERENCES – определение ссылки на другую таблицу;

• NOT NULL – указывает, что значение атрибута не должно быть пустым;

Оператор изменения структуры таблицы позволяет изменить структуру таблицы, имена атрибутов и их типы. Синтаксис:

ALTER TABLE <имя таблицы>

{ADD <определение столбца>}|

ALTER <имя столбца>|

DROP <имя столбца>|

ADD <ограничение таблицы>.

Следует заметить, что если пытаться изменить имя или тип столбца, на который имеется ссылка из другой таблицы, то будет выдано сообщение об ошибке, и изменения нельзя будет произвести. Для этого сначала необходимо разорвать связь между таблицами.

Оператор создания БД позволяет создавать новые БД в СУБД для

последующего использования при хранении и доступе к данным. Синтаксис:

CREATE DATABASE '<имя файла БД>'

USER '<имя пользователя>'

PASSWORD '<пароль пользователя>',

где:

<имя файла БД> – имя физического файла, в котором будет храниться БД, расширение в имени указывается явно, как правило “gbd”. Создание базы данных происходит в два этапа. Сначала физически создается файл БД оператором CREATE DATABASE, затем выполняется регистрация БД в СУБД.

1. Язык SQL состоит из набора операторов, которые можно разделить на несколько категорий:

• Data Definition Language (DDL) – язык определения данных, позволяющий создавать, удалять и изменять объекты в базах данных;

• Data Manipulation Language (DML) – язык управления данными, позволяющий модифицировать, добавлять и удалять данные в имеющихся объектах базы данных;

• Data Control Language (DCL) – язык, используемый для управления пользовательскими привилегиями;

• Transaction Control Language (TCL) – язык для управления изменениями, сделанными группами операторов;

• Cursor Control Language (CCL) – язык для определения курсоров, подготовки операторов SQL к выполнению и некоторых других операций