Санкт-Петербургский Политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

**Отчёт о лабораторной работе №3**

**Дисциплина**: Базы данных

**Тема**: Язык SQL-DDL

Выполнил студент гр. 43501/1 С.В. Смирнов

(подпись)

Руководитель А.В. Мяснов

(подпись)

“ ” 2016 г.

Санкт-Петербург

2016

1. **Цель работы**

Познакомить студентов с основами проектирования схемы БД, языком описания сущностей и ограничений БД SQL-DDL.

1. **Программа работы**
2. Самостоятельное изучение SQL-DDL
3. Создание скрипта БД в соответствии с согласованной схемой (должны присутствовать первичные и внешние ключи, ограничения на диапазоны значений). Продемонстрировать скрипт преподавателю.
4. Создайте скрипт, заполняющий все таблицы БД данными
5. Выполнение SQL-запросов, изменяющих схему созданной БД по заданию преподавателя. Продемонстрировать их работу преподавателю.
6. Изучите основные возможности IBExpert. Получите ER-диаграмму созданной БД с помощью Database Designer.
7. Автоматически сгенерируйте данные при помощи IBExpert (для трех или большего числа таблиц, не менее 100000 записей в каждой из выбранных таблиц).
8. **Язык SQL**

Язык SQL (Structured Query Language) - язык структурированных запросов. Он позволяет формировать весьма сложные запросы к базам данных. В SQL определены два подмножества языка:

SQL-DDL (Data Definition Language) - язык определения структур и ограничений целостности баз данных. Сюда относятся команды создания и удаления баз данных; создания, изменения и удаления таблиц; управления пользователями и т.д.

SQL-DML (Data Manipulation Language) - язык манипулирования данными: добавление, изменение, удаление и извлечение данных, управления транзакциями.

1. **Выполнение работы.**

Был создан скрипт, создающий базу данных в соответствии со схемой:

create database 'D:\7\_sem\carservice.fdb' user 'SYSDBA' password 'masterkey';

commit;

connect 'D:\7\_sem\carservice.fdb' user 'SYSDBA' password 'masterkey';

commit;

CREATE TABLE Client (

ClientID int PRIMARY KEY,

BID int,

Surname VARCHAR(20),

Name VARCHAR(20),

Phone DECIMAL(11));

CREATE TABLE Bonus (

BID int PRIMARY KEY,

SUMM int,

discount int);

CREATE TABLE Ser\_info (

SID int PRIMARY KEY,

Price int,

Date\_in DATE,

Date\_craft DATE,

CarID int);

CREATE TABLE Car (

CarID int PRIMARY KEY,

VIN DECIMAL(15),

ODO DECIMAL(6),

ClientID int);

CREATE TABLE Work\_Parts (

WorkID int,

SID int,

WPID int PRIMARY KEY);

CREATE TABLE Parts (

PartID int PRIMARY KEY,

Price DECIMAL,

Descr VARCHAR(10));

CREATE TABLE Works (

WorkID int PRIMARY KEY,

Price DECIMAL,

Descr VARCHAR(10));

CREATE TABLE Id\_Part (

WPID int,

PartID int,

id int PRIMARY KEY);

ALTER TABLE Client ADD CONSTRAINT Client\_fk0 FOREIGN KEY (BID) REFERENCES Bonus(BID);

ALTER TABLE Ser\_info ADD CONSTRAINT Ser\_info\_fk0 FOREIGN KEY (CarID) REFERENCES Car(CarID);

ALTER TABLE Car ADD CONSTRAINT Car\_fk0 FOREIGN KEY (ClientID) REFERENCES Client(ClientID);

ALTER TABLE Work\_Parts ADD CONSTRAINT Work\_Parts\_fk0 FOREIGN KEY (WorkID) REFERENCES Works(WorkID);

ALTER TABLE Work\_Parts ADD CONSTRAINT Work\_Parts\_fk1 FOREIGN KEY (SID) REFERENCES Ser\_info(SID);

ALTER TABLE Id\_Part ADD CONSTRAINT Id\_Part\_fk0 FOREIGN KEY (WPID) REFERENCES Work\_Parts(WPID);

ALTER TABLE Id\_Part ADD CONSTRAINT Id\_part\_fk1 FOREIGN KEY (PartID) REFERENCES Parts(PartID);

commit;

show tables;

В результате исполнения создаётся БД с соответствующими таблицами:

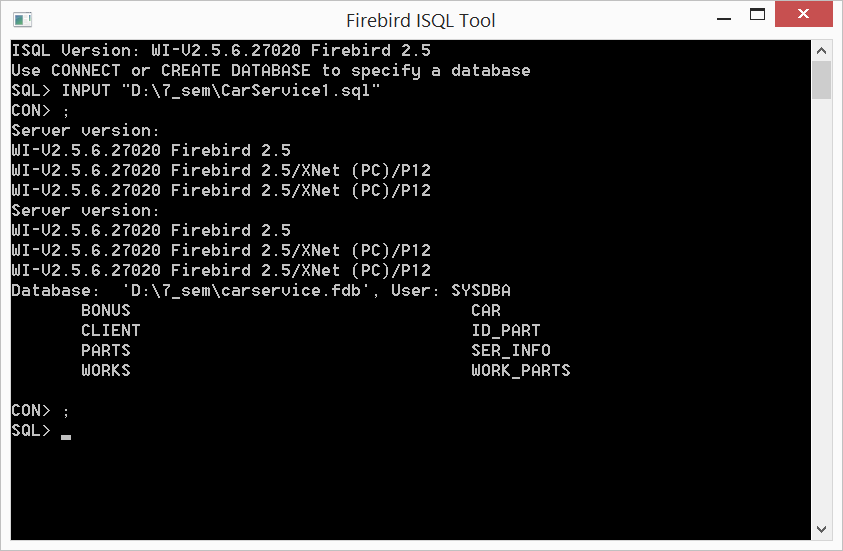


Рис.1.Результат исполнения скрипта

Так же был создан скрипт заполняющий базу данными:

INSERT INTO Bonus VALUES(1,1000,1);

INSERT INTO Bonus VALUES(2,2000,2);

INSERT INTO Bonus VALUES(3,3000,3);

INSERT INTO Bonus VALUES(4,4000,4);

INSERT INTO Bonus VALUES(5,5000,5);

INSERT INTO Bonus VALUES(6,10000,7);

commit;

INSERT INTO Client VALUES(1,1,'Smirnov','Sergei',89210987330);

INSERT INTO Client VALUES(2,3,'Goremykina','Olga',89210987330);

INSERT INTO Client VALUES(3,6,'Drozdovski','Alex',89210987330);

INSERT INTO Client VALUES(4,2,'Evseev','Evgeni',89210987330);

INSERT INTO Client VALUES(5,4,'Kolpinski','Sasha',89210987330);

INSERT INTO Client VALUES(6,5,'Kalugina','Maria',89210987330);

commit;

INSERT INTO Car VALUES(1,999999999999999,999999,1);

INSERT INTO Car VALUES(2,000000000000000,000000,2);

INSERT INTO Car VALUES(3,111111111111111,999900,3);

INSERT INTO Car VALUES(4,222222222222222,400000,5);

INSERT INTO Car VALUES(5,333333333333333,500000,4);

INSERT INTO Car VALUES(6,444444444444444,200000,6);

commit;

INSERT INTO Ser\_info VALUES(1,1,'2013-09-01','2013-09-02',1);

INSERT INTO Ser\_info VALUES(2,10,'2016-09-01','2016-09-02',1);

INSERT INTO Ser\_info VALUES(3,100,'2013-09-01','2016-09-02',4);

INSERT INTO Ser\_info VALUES(4,1000,'2014-09-01','2014-09-02',3);

INSERT INTO Ser\_info VALUES(5,10000,'2015-09-01','2015-09-02',5);

INSERT INTO Ser\_info VALUES(6,20000,'2014-09-03','2014-09-07',6);

commit;

INSERT INTO Works VALUES(1,500,'Diagnostika');

INSERT INTO Works VALUES(2,5000,'Pokraska elementa');

INSERT INTO Works VALUES(3,700,'Zamena masla');

INSERT INTO Works VALUES(4,1200,'Zamena stoiki stabilizatora');

INSERT INTO Works VALUES(5,2300,'Zamena podshipnikov generatora');

INSERT INTO Works VALUES(6,5000,'Promyvka forsunok');

commit;

INSERT INTO Work\_Parts VALUES(1,1,1);

INSERT INTO Work\_Parts VALUES(2,1,2);

INSERT INTO Work\_Parts VALUES(3,6,3);

INSERT INTO Work\_Parts VALUES(3,3,4);

INSERT INTO Work\_Parts VALUES(5,4,5);

INSERT INTO Work\_Parts VALUES(4,5,6);

commit;

INSERT INTO Parts VALUES(1,2500,'Maslo Castrol 5w30');

INSERT INTO Parts VALUES(2,500,'Filter maslyanii 914/2 MAN');

INSERT INTO Parts VALUES(3,1500,'Stoika stab volvo xc70/ffocus2');

INSERT INTO Parts VALUES(4,200,'Podshipnik 203');

INSERT INTO Parts VALUES(5,800,'Kolodki Largus 16v');

commit;

INSERT INTO Id\_Part VALUES(1,1,1);

INSERT INTO Id\_Part VALUES(2,5,2);

INSERT INTO Id\_Part VALUES(2,2,3);

INSERT INTO Id\_Part VALUES(3,1,4);

INSERT INTO Id\_Part VALUES(4,3,5);

INSERT INTO Id\_Part VALUES(5,4,6);

commit;

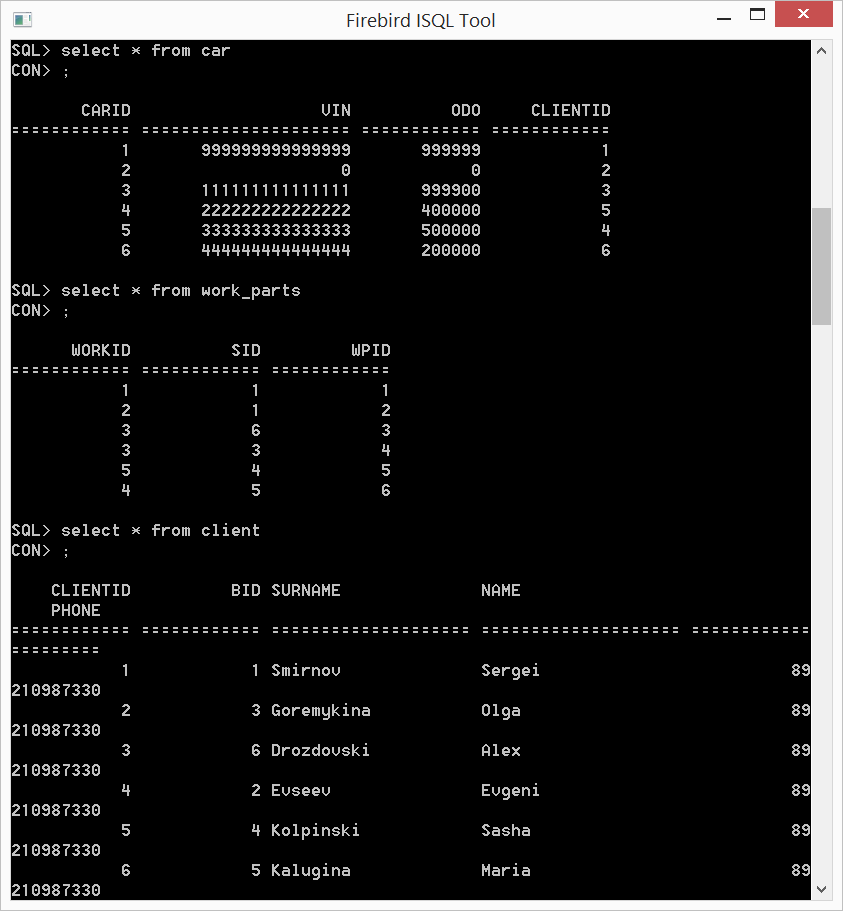


Рис.2. Пример заполненных таблиц

Затем был создан скрипт, модифицирующий базу данных согласно заданию:

1. Для каждой работы и запчасти должна фиксироваться стоимость и количество в рамках конкретного заказа.
2. Реализовать учет "акций" - наборов работ (и запчастей) по фиксированной цене.

Ниже представлена доработанная схема БД:

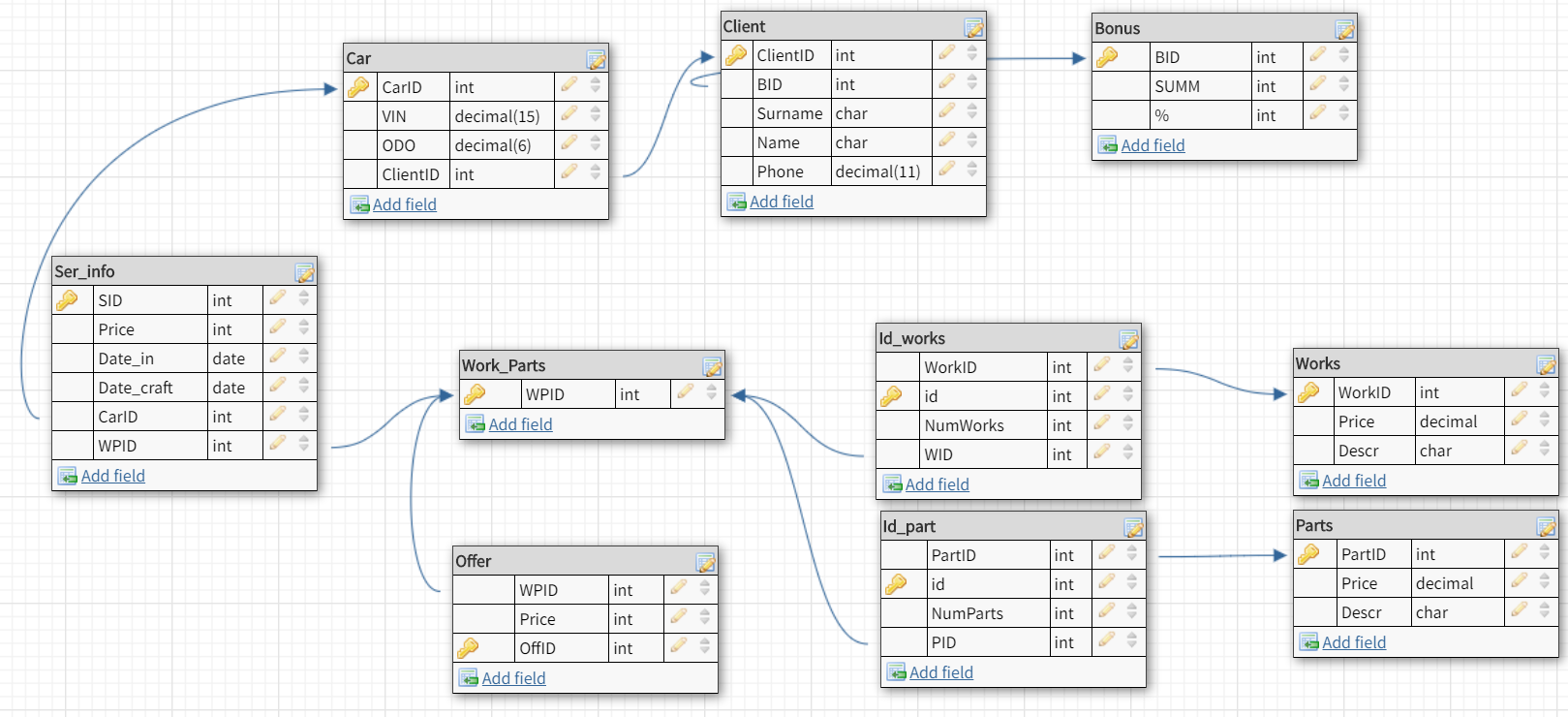


Рис.3.Доработанная схема

Скрипт для корректировки БД:

connect 'D:\7\_sem\carservice.fdb' user 'SYSDBA' password 'masterkey';

commit;

CREATE TABLE Offer (

OffID int PRIMARY KEY,

Price int,

WPID int

);

CREATE TABLE Id\_Works (

WID int,

WorkID int,

NumWorks int,

id int PRIMARY KEY

);

ALTER TABLE Work\_Parts DROP SID;

ALTER TABLE Work\_Parts DROP WorkID;

ALTER TABLE Offer ADD CONSTRAINT Offer\_fk0 FOREIGN KEY (WPID) REFERENCES Work\_Parts(WPID);

ALTER TABLE Ser\_info ADD WPID INT;

ALTER TABLE Ser\_info ADD CONSTRAINT Ser\_info\_fk1 FOREIGN KEY (WPID) REFERENCES Work\_Parts(WPID);

ALTER TABLE Id\_Works ADD CONSTRAINT Id\_Works\_fk0 FOREIGN KEY (WID) REFERENCES Work\_Parts(WPID);

ALTER TABLE Id\_Works ADD CONSTRAINT Id\_Works\_fk1 FOREIGN KEY (WorkID) REFERENCES Works(WorkID);

commit;

ALTER TABLE Id\_Part ADD NumParts INT;

ALTER TABLE Id\_Part DROP WPID;

ALTER TABLE Id\_Part ADD PID INT ;

ALTER TABLE Id\_Part ADD CONSTRAINT Id\_Part\_fk0 FOREIGN KEY (PID) REFERENCES Work\_Parts(WPID);

commit;

show tables;

Результат:

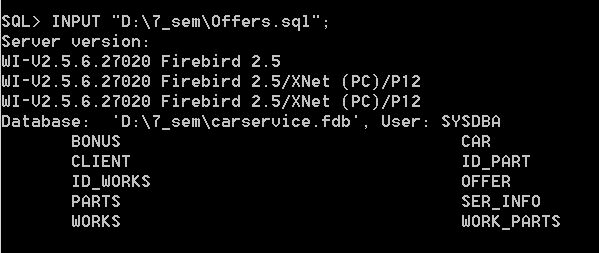


Рис.4. Изменённая база

С помощью Database Designer в IBExpert сгенерируем ER-диаграмму нашей БД:

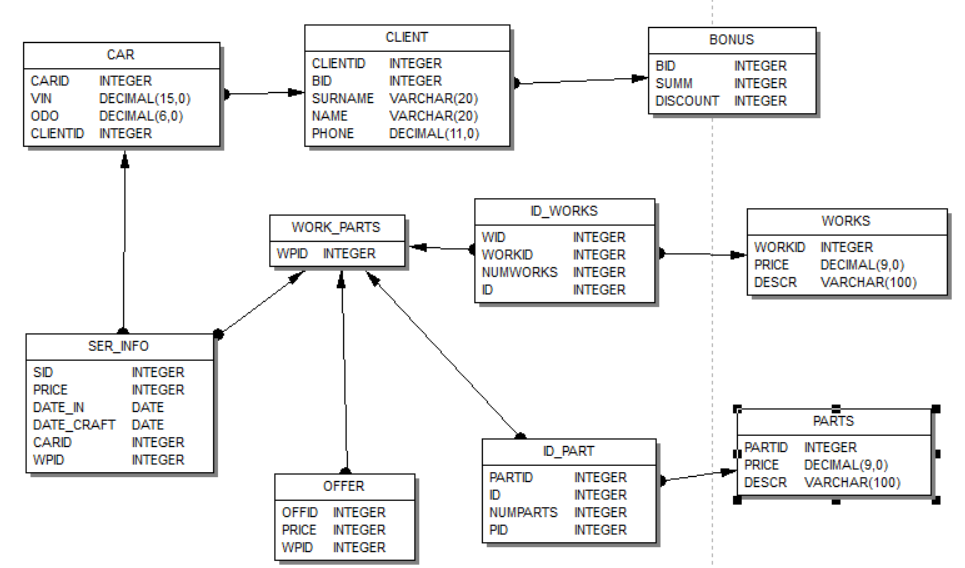
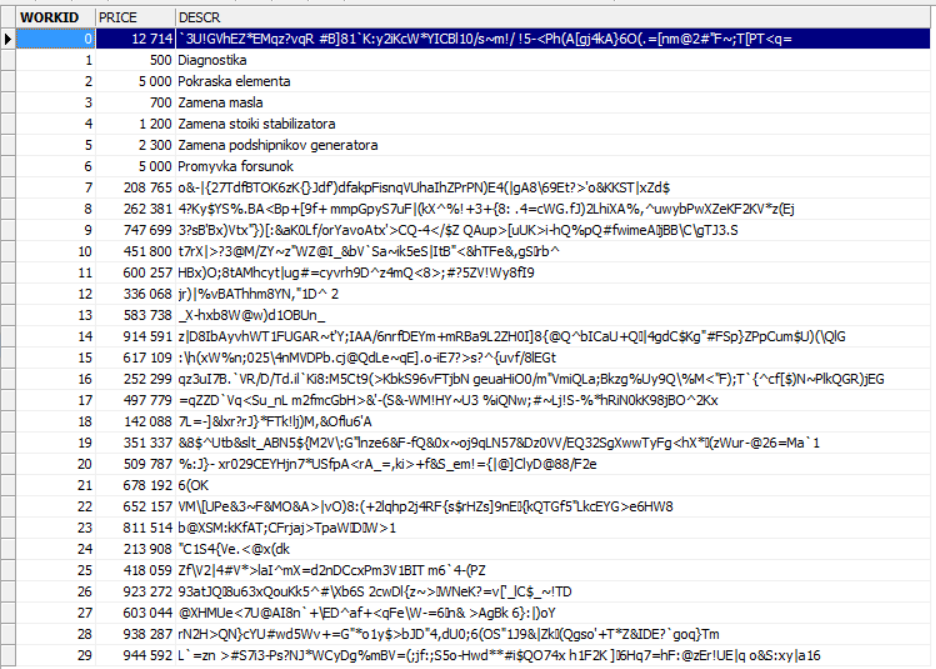


Рис.5. ER-диаграмма БД

Автоматически сгенерируем данные при помощи IBExpert для таблиц Works и Parts:



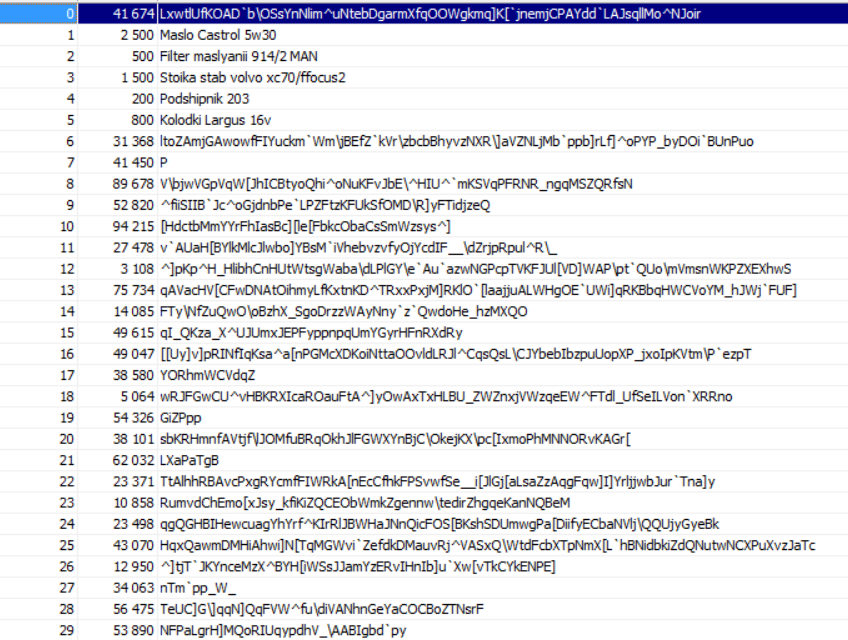


Рис.6-7. Генерация данных

Таким образом было сгенерировано по 10000 записей в двух таблицах.

**Выводы:**

В ходе лабораторной работы были изучены команды языка DDL позволяющие изменять базу в нужной степени (create,drop, alter). В ходе изменений возникли трудности, так как по условию акция может включать в себя несколько работ и несколько деталей, решение состояло в введении дополнительной таблицы и привязки акций к таблице Work\_Parts.

Изучаемая программа IBExpert позволяет загружать созданную ранее БД и с использованием удобного графического интерфейса редактировать её, вводить тестовые данные, создавать er-диаграммы и др.