Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

Отчет по лабораторной работе №3

По дисциплине «Базы данных»

«Язык SQL-DDL»

Работу выполнил студент группы №43501/1 Кричевский В.А. \_\_\_\_\_\_\_\_

Работу принял преподаватель Мяснов А.В.\_\_\_\_\_\_\_\_\_\_\_\_\_

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. Ход работы
9. Изучены основы языка SQL-DDL.
10. База данных создана при помощи следующего скрипта:

|  |
| --- |
| create database 'C:\Users\Vladik\Documents\Database\MyHotel1.fdb' user 'SYSDBA' password 'masterkey';  connect 'C:\Users\Vladik\Documents\Database\MyHotel1.fdb' user 'SYSDBA' password 'masterkey';  create table Zakaz(  id int primary key,  Tour\_id int not null,  Client\_id int not null,  Cost int not null  );  create table Client(  id int primary key,  Name varchar(40) not null,  Phone\_number int not null,  Email varchar(30),  Country\_id int not null  );  create table Country(  id int primary key,  Name varchar(30) not null,  Allow char(1) not null  );  create table Tour(  id int primary key,  Tourfirm\_id int not null,  Persons int not null,  Arrival\_date date not null,  Out\_date date not null,  TypeTour\_id int not null    );  create table Tourfirm(  id int primary key,  Comission int not null,  Name varchar(30) not null  );  create table Hotel(  id int primary key,  Zakaz\_id int not null,  Room\_id int not null  );  create table Room(  id int primary key,  Room\_number int not null,  Beds int not null,  Busy\_status char(1) not null,  Repair\_status char(1) not null  );  alter table Zakaz add constraint Zakaz\_to\_tour foreign key (Tour\_id) references Tour(id);  alter table Zakaz add constraint Zakaz\_to\_client foreign key (Client\_id) references Client(id);  alter table Client add constraint Client\_to\_country foreign key (Country\_id) references Country(id);  alter table Tour add constraint Tour\_to\_firm foreign key (Tourfirm\_id) references Tourfirm(id);  alter table Hotel add constraint Hotel\_to\_zakaz foreign key (Zakaz\_id) references Zakaz(id);  alter table Hotel add constraint Hotel\_to\_room foreign key (Room\_id) references Room(id); |

1. Заполнение БД происходит следующим скриптом:

|  |
| --- |
| connect 'C:\Users\Vladik\Documents\Database\MyHotel1.fdb' user 'SYSDBA' password 'masterkey';  insert into AdditionalServ (id, Name) values (1, 'Ice cream');  insert into AdditionalServ (id, Name) values (2, 'Digit TV in Room');  insert into AdditionalServ (id, Name) values (3, 'Umbrella in Room');  insert into AdditionalServ (id, Name) values (4, 'Standart');  insert into Country (id, Name,Allow) values (1, 'Russia','T');  insert into Country (id, Name,Allow) values (2, 'Ukraine','T');  insert into Country (id, Name,Allow) values (3, 'Belorussia','T');  insert into Tourfirm (id, Comission,Name) values (1, 700, 'Neva');  insert into Tourfirm (id, Comission,Name) values (2, 1000, 'Marlis');  insert into Tourfirm (id, Comission,Name) values (3, 800, 'Booking.com');  insert into TypeTour (id, Name) values (1, 'Sport');  insert into TypeTour (id, Name) values (2, 'Culture');  insert into TypeTour (id, Name) values (3, 'Business');  insert into Room (id,Room\_number,Beds,Busy\_status,Repair\_status)  values (1, 1, 2, 'F','F');  insert into Room (id,Room\_number,Beds,Busy\_status,Repair\_status)  values (2, 2, 3, 'F','F');  insert into Room (id,Room\_number,Beds,Busy\_status,Repair\_status)  values (3, 3, 2, 'F','F');  insert into Client (id,Name,Phone\_number,Email,Country\_ID )  values (1, 'Ericson A.S.',55675777,'eric@mail.ru',2);  insert into Client (id,Name,Phone\_number,Email,Country\_ID )  values (2, 'Molotov A.A.',44545345,'qdw@mail.ru',2);  insert into Client (id,Name,Phone\_number,Email,Country\_ID )  values (3, 'Krusov A.T.',44543355,'qrer@mail.ru',1);  insert into Tour (id, Tourfirm\_id, Persons, Arrival\_date, Out\_date, Trip\_id)  values (1, 1, 1, '21.11.16','25.11.16',2);  insert into Tour (id, Tourfirm\_id, Persons, Arrival\_date, Out\_date, Trip\_id)  values (2, 1, 2, '18.11.16','26.11.16',1);  insert into Tour (id, Tourfirm\_id, Persons, Arrival\_date, Out\_date, Trip\_id)  values (3, 3, 1, '10.11.16','25.11.16',2);  insert into Zakaz (id,Tour\_id,Client\_id,Cost)  values (1, 1, 1, 3000);  insert into Zakaz (id,Tour\_id,Client\_id,Cost)  values (2, 2, 2, 4500);  insert into Zakaz (id,Tour\_id,Client\_id,Cost)  values (3, 3, 3, 6000);  insert into Hotel (id, Zakaz\_id, Room\_id) values (1, 1, 1);  insert into Hotel (id, Zakaz\_id, Room\_id) values (2, 2, 2);  insert into Hotel (id, Zakaz\_id, Room\_id) values (3, 3, 3);  insert into Dostp (id, Name) values (1, 'Museum');  insert into Dostp (id, Name) values (2, 'Bar');  insert into Dostp (id, Name) values (3, 'Stadium');  insert into Trip (id, Dostp\_id, TypeTour\_id) values (1,3,1);  insert into Trip (id, Dostp\_id, TypeTour\_id) values (2,2,3);  insert into Trip (id, Dostp\_id, TypeTour\_id) values (3,1,2);  insert into Addsv (id, Zakaz\_id, AdditionalServ\_id) values (1,1,2);  insert into Addsv (id, Zakaz\_id, AdditionalServ\_id) values (2,1,1);  insert into Addsv (id, Zakaz\_id, AdditionalServ\_id) values (3,2,2);  commit; |

1. Реализовать учет типов туров.

Реализовать учет достопримечательностей со связью с типами туров.  
Реализовать учет заказа дополнительных услуг в гостиницах.  
Скрипт модификации базы данных.

|  |
| --- |
| create table additionalServ(  id int primary key,  Name varchar(30) not null    );  create table TypeTour(  id int primary key,  Name varchar(30)    );  create table Addsv(  id int primary key,  Zakaz\_id int not null,  AdditionalServ\_id int not null  );  CREATE TABLE Dostp(  ID INT PRIMARY KEY ,  Name varchar(30) NOT NULL  );  CREATE TABLE Trip(  ID INT PRIMARY KEY,  Dostp\_id INT NOT NULL,  TypeTour\_id INT NOT NULL  );  ALTER TABLE Tour ADD CONSTRAINT Tour\_to\_trip FOREIGN KEY (Trip\_id) REFERENCES Trip(ID);  ALTER TABLE Trip ADD CONSTRAINT Trip\_to\_dostp FOREIGN KEY (Dostp\_id) REFERENCES Dostp(ID);  ALTER TABLE Trip ADD CONSTRAINT Trip\_to\_typetour FOREIGN KEY (TypeTour\_id) REFERENCES TypeTour(ID);  ALTER TABLE Addsv ADD CONSTRAINT Addsv\_to\_Zakaz FOREIGN KEY (Zakaz\_id) REFERENCES Zakaz(ID);  ALTER TABLE Addsv ADD CONSTRAINT Addsv\_to\_serv FOREIGN KEY (AdditionalServ\_id) REFERENCES AdditionalServ(ID); |

Для учета типа тура и заказа дополнительных услуг были созданы отдельные таблицы.

Добавлено поле в таблицу с типами туров

1. С помощью IBExpert построена схема БД рис.1.

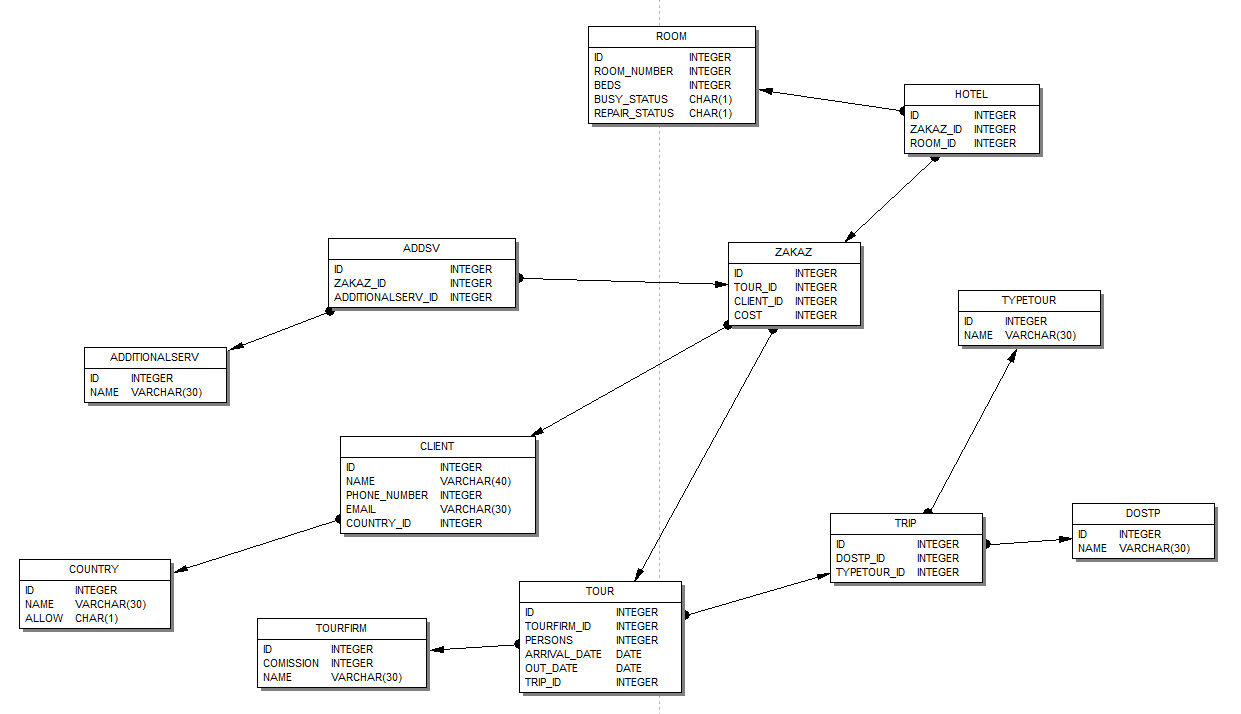


Рис.1. Схема БД.

1. С помощью IBExpert автоматически сгенерированы данные для трех таблиц:

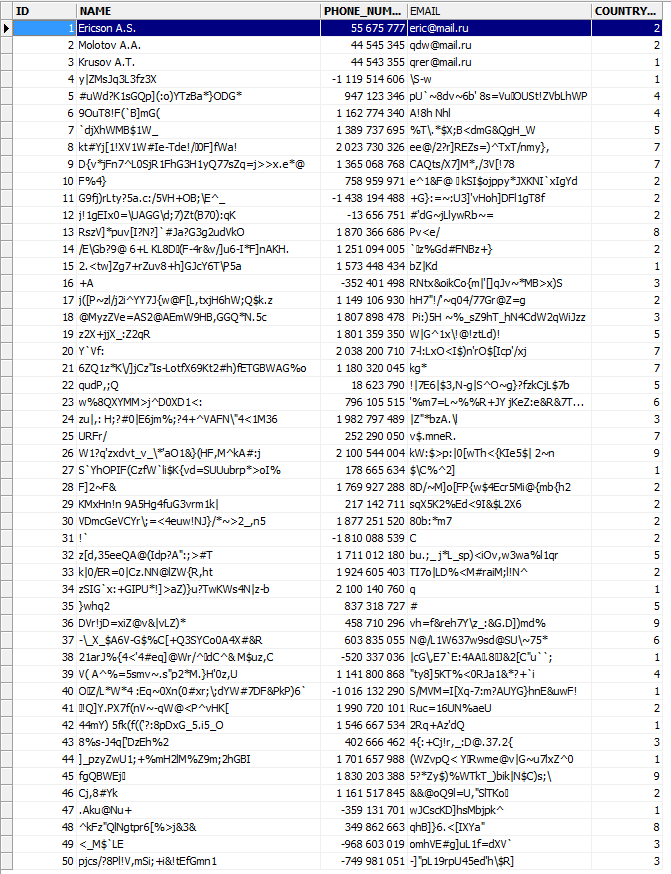


Рис.2. Данные в таблице “Client”

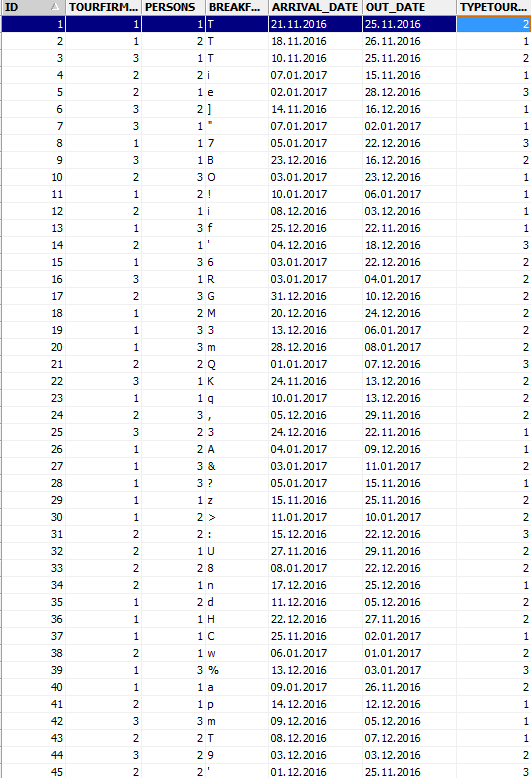


Рис.3. Данные в таблице "Tour"

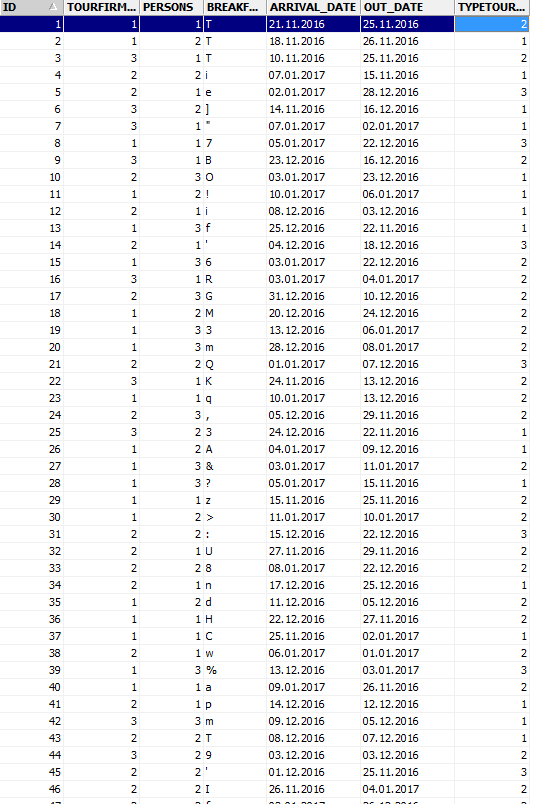


Рис. 4 Данные в таблице "Zakaz"

4. Выводы

С помощью языка проектирования баз данных SQL-DDL была создана таблица, отвечающая заданию. Основной используемой инструкцией была инструкция «create table”. Далее, с помощью инструкции “insert” таблицы заполнены некоторыми данными.

Для модификации уже имеющейся таблицы используется команда “alter table” позволяющая добавлять и удалять столбцы из таблиц.

С помощью среды IBExpert построена диаграмма получено базы данных, а также сгенерировано большое количество случайных данных для нескольких таблиц.

Язык программирования SQL позволяет достаточно просто создавать необходимые таблицы и связи между ними. Язык SQL позволяет описать нужную структуру более простым языком, а также с помощью дополнительных средств, таких как IBExpert графически отобразить. Большинство языков программирования уже имеют в себе средства для работы с базами данных, так что взаимодействие вашего приложения с созданной извне базой ничем не ограничено.