

Федеральное агентство по образованию

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Институт информационных технологий и управления

Кафедра компьютерных систем и программных технологий

ОТЧЕТ

По лабораторной работе №2

Язык SQL-DDL

Студентка гр.43501/4 Перетяцько Е.В.

Преподаватель Мяснов А. В.

Санкт-Петербург

2015

Цели работы

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

Программа работы

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

Выполнение программы работы

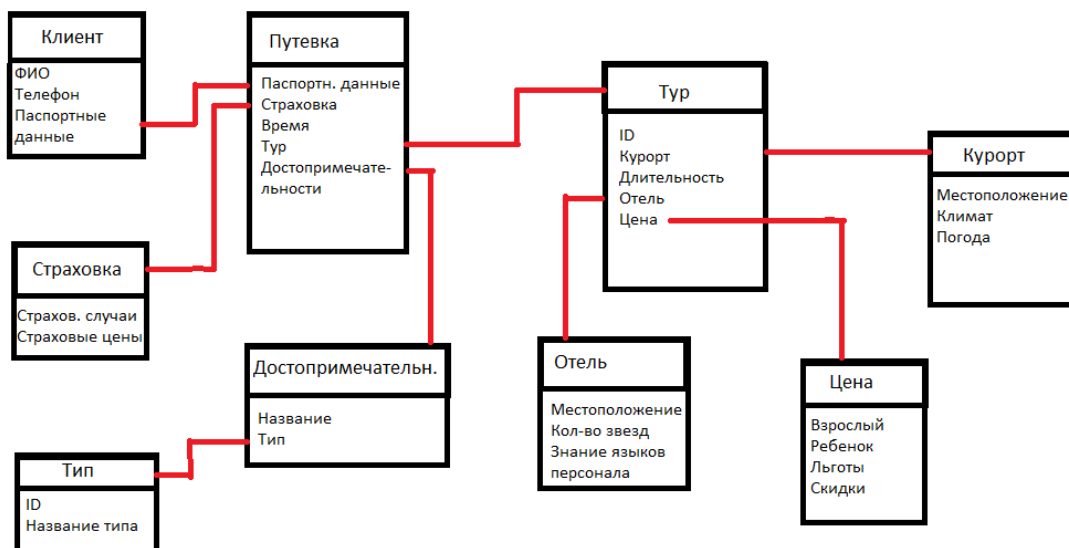


Рис. 1 Туристическое агентство

Ниже представлен скрипт БД в соответствии с заданной схемой.

```
create database 'd:/Turism.fdb' user 'SYSDBA' password 'masterkey';

commit;

CONNECT 'd:/Turism.fdb' user 'SYSDBA' password 'masterkey';


create table type (
    id int primary key,
    typeName varchar(30) not null
);


create table place_of_interest (
    id int primary key,
    type_id int not null,
    FOREIGN KEY (type_id) REFERENCES type(id),
    namePlace varchar(30) not null
);


create table hotel (
    id int primary key,
    location_hotel varchar(30) not null,
    stars int not null,
    language varchar(30) not null
);


create table price (
    id int primary key,
    adult int not null,
```

child int not null

);

create table kurort (

id int primary key,

location_kurort varchar(30) not null,

climate varchar(30) not null,

weather varchar(30) not null

);

create table tour (

id int primary key,

time_tour varchar(30) not null,

id_kurort int not null,

FOREIGN KEY (id_kurort) REFERENCES kurort(id),

id_hotel int not null,

FOREIGN KEY (id_hotel) REFERENCES hotel(id),

id_price int not null,

FOREIGN KEY (id_price) REFERENCES price(id)

);

create table insurance (

id int primary key,

insurance_cases varchar(30) not null,

insurance_price int not null

);

```

create table client (

Name varchar(30) not null,

Surname varchar(30) not null,

number_phone int not null,

passport_id int primary key,

insurance_price int not null

);


create table trip (

id_client int not null,

FOREIGN KEY (id_client) REFERENCES client(passport_id),

id_insurance int not null,

FOREIGN KEY (id_insurance) REFERENCES insurance(id),

id_tour int not null,

FOREIGN KEY (id_tour) REFERENCES tour(id),

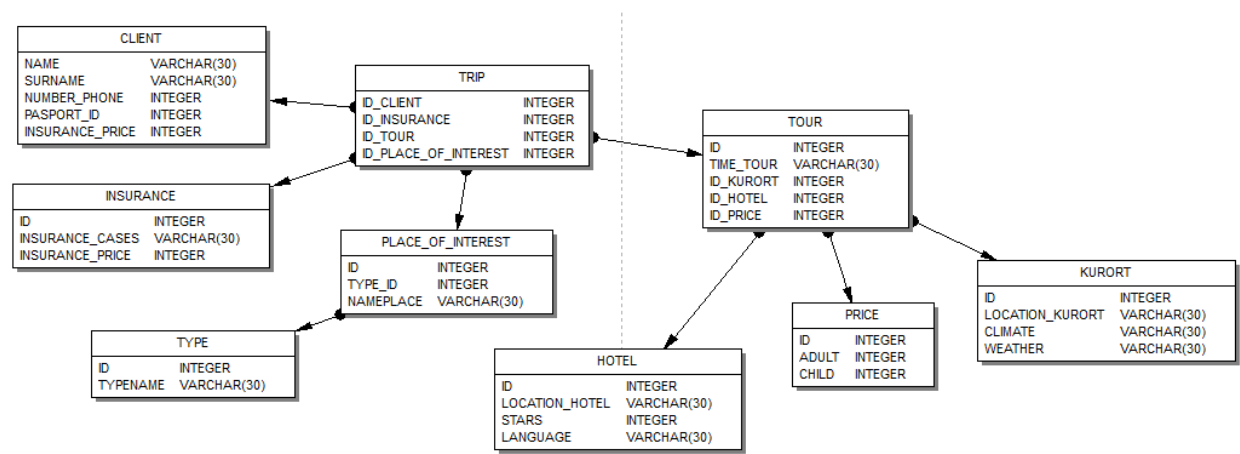
id_place_of_interest int not null,

FOREIGN KEY (id_place_of_interest) REFERENCES place_of_interest(id)

);

```

Получение ER диаграммы с помощью дизайнера БД.



Модифицировать схему БД для удовлетворения следующим требованиям:

1. Привязать достопримечательности к курортам.
2. Добавить учет экскурсий входящих в состав туров и покупаемых индивидуально в путевке.

1) ALTER TABLE kurort

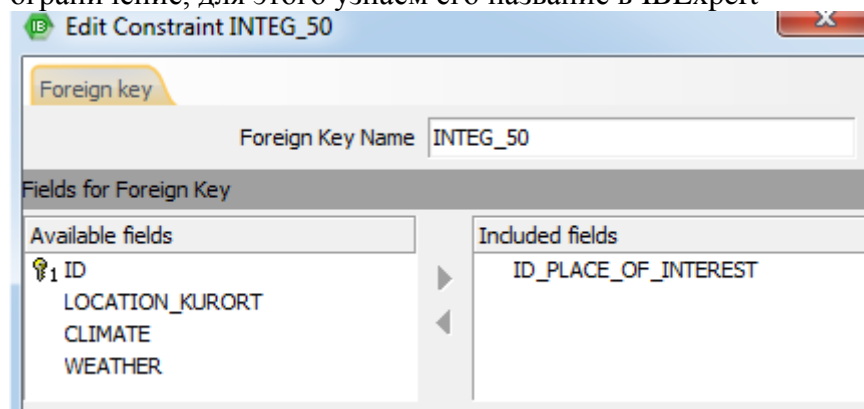
```
ADD id_place_of_interest int;
```

```
ADD FOREIGN KEY (id_place_of_interest) REFERENCES place_of_interest(id);
```

ALTER TABLE trip

```
DROP COLUMN id_place_of_interest;
```

Т.к. на столбец наложены ограничения, удалять такой столбец нельзя. Сначала удалим ограничение, для этого узнаем его название в IBExpert

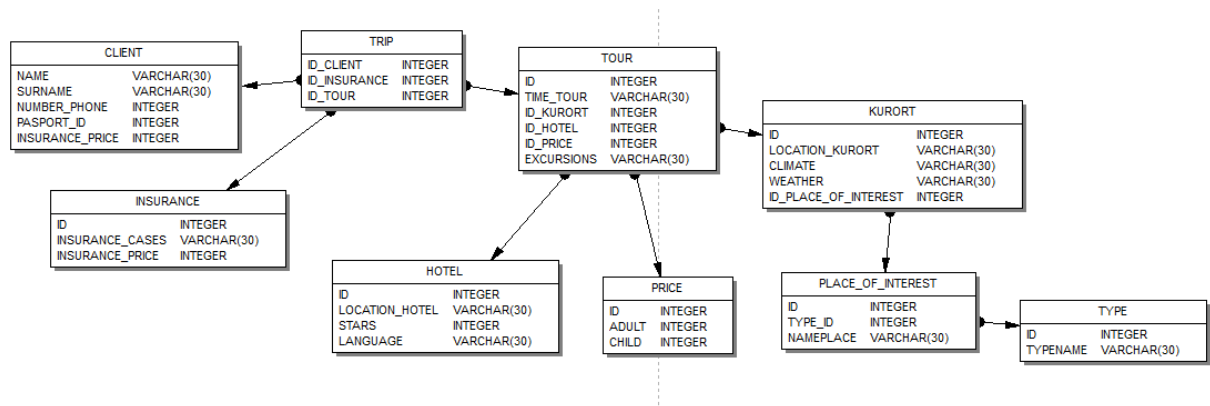


```
ALTER TABLE TRIP  
DROP CONSTRAINT INTEG_50;  
ALTER TABLE TRIP  
DROP ID_PLACE_OF_INTEREST;
```

2) ALTER TABLE tour

```
ADD excursions varchar(30);
```

Модифицированная ER диаграмма, полученная с помощью дизайнера БД.



Ниже представлен скрипт, заполняющий все таблицы БД

```

INSERT INTO TYPE (id, typeName) VALUES (1, 'Tower');

INSERT INTO TYPE (id, typeName) VALUES (2, 'Castle');

INSERT INTO TYPE (id, typeName) VALUES (3, 'Museum');

INSERT INTO TYPE (id, typeName) VALUES (4, 'Palace');

INSERT INTO TYPE (id, typeName) VALUES (5, 'Park');

commit;

INSERT INTO HOTEL (id, location_hotel, stars, language) VALUES (1, 'center', 5, 'English');

INSERT INTO HOTEL (id, location_hotel, stars, language) VALUES (2, 'center', 3, 'English');

INSERT INTO HOTEL (id, location_hotel, stars, language) VALUES (3, 'near the sea', 5, 'English');

INSERT INTO HOTEL (id, location_hotel, stars, language) VALUES (4, 'suburb', 4, 'English');

commit;

INSERT INTO PRICE (id, adult, child) VALUES (1, 25000, 12500);

INSERT INTO PRICE (id, adult, child) VALUES (2, 50000, 25000);

INSERT INTO PRICE (id, adult, child) VALUES (3, 75000, 35000);

INSERT INTO PRICE (id, adult, child) VALUES (4, 25000, 7000);
    
```

commit;

INSERT INTO INSURANCE (id, insurance_cases, insurance_price) VALUES (1, 'accident', 500000);

INSERT INTO INSURANCE (id, insurance_cases, insurance_price) VALUES (2, 'infectious disease', 100000);

commit;

INSERT INTO CLIENT (Name, Surname, number_phone, passport_id, insurance_price) VALUES ('Masha', 'Petrova', 4608345, 789345, 1);

INSERT INTO CLIENT (Name, Surname, number_phone, passport_id, insurance_price) VALUES ('Dasha', 'Vetrova', 4567345, 784567, 2);

INSERT INTO CLIENT (Name, Surname, number_phone, passport_id, insurance_price) VALUES ('Ivan', 'Ivanov', 5672341, 744384, 3);

INSERT INTO CLIENT (Name, Surname, number_phone, passport_id, insurance_price) VALUES ('Pavel', 'Sidorov', 3848432, 734829, 1);

INSERT INTO CLIENT (Name, Surname, number_phone, passport_id, insurance_price) VALUES ('Igor', 'Kuznezov', 7423847, 785499, 2);

INSERT INTO CLIENT (Name, Surname, number_phone, passport_id, insurance_price) VALUES ('Anna', 'Alexeeva', 7348957, 783243, 3);

commit;

INSERT INTO PLACE_OF_INTEREST (id, type_id, namePlace) VALUES (1, 1, 'Tower of London');

INSERT INTO PLACE_OF_INTEREST (id, type_id, namePlace) VALUES (2, 2, 'Hearst Castle');

INSERT INTO PLACE_OF_INTEREST (id, type_id, namePlace) VALUES (3, 3, 'Museum of fine arts boston');

INSERT INTO PLACE_OF_INTEREST (id, type_id, namePlace) VALUES (4, 4, 'Crystal Palace');

INSERT INTO PLACE_OF_INTEREST (id, type_id, namePlace) VALUES (5, 5, 'Park in Paris');

commit;

INSERT INTO KURORT (id, location_kurort, climate, weather, id_place_of_interest) VALUES (1, 'London', 'moderate', 'sun', 1);


```
INSERT INTO KURORT (id, location_kurort, climate, weather, id_place_of_interest)
VALUES (2, 'Paris', 'moderate', 'rain', 5);
```

```
INSERT INTO KURORT (id, location_kurort, climate, weather, id_place_of_interest)
VALUES (3, 'Boston', 'moderate', 'snow', 3);
```

```
commit;
```

```
INSERT INTO TOUR (id, time_tour, id_kurort, id_hotel, id_price) VALUES (1,'2 week',
1,2,1);
```

```
INSERT INTO TOUR (id, time_tour, id_kurort, id_hotel, id_price) VALUES (2,'1 week',
2,3,3);
```

```
INSERT INTO TOUR (id, time_tour, id_kurort, id_hotel, id_price) VALUES (3,'2 week',
3,4,3);
```

```
commit;
```

```
INSERT INTO TRIP (id_client, id_insurance, id_tour) VALUES (789345,2,1);
```

```
INSERT INTO TRIP (id_client, id_insurance, id_tour) VALUES (784567,2,2);
```

```
INSERT INTO TRIP (id_client, id_insurance, id_tour) VALUES (744384,2,3);
```

```
INSERT INTO TRIP (id_client, id_insurance, id_tour) VALUES (734829,2,1);
```

```
INSERT INTO TRIP (id_client, id_insurance, id_tour) VALUES (785499,2,2);
```

```
INSERT INTO TRIP (id_client, id_insurance, id_tour) VALUES (783243,2,3);
```

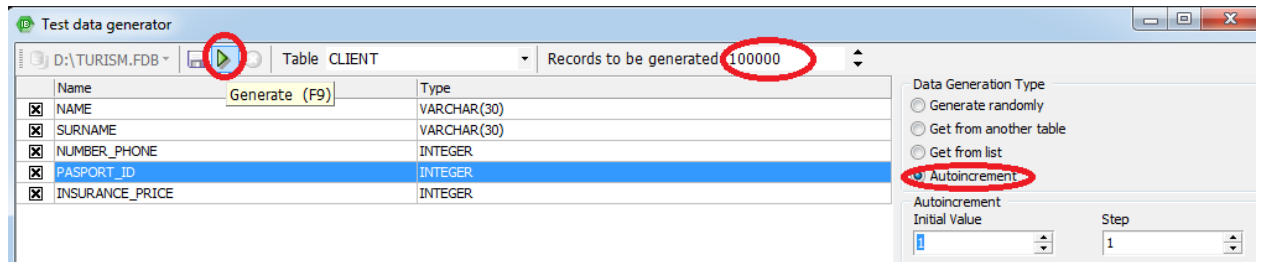
```
commit;
```

ID	TYPENAME
1	Tower
2	'astle
3	Museum
4	Palace
5	Park

ID	LOCATION_HOTEL	STARS	LANGUAGE
1	center	5	English
2	center	3	English
3	near the sea	5	English
4	suburb	4	English

Автоматически сгенерируйте данные при помощи IVExpert (для трех или большего числа таблиц, не менее 100000 записей в каждой из выбранных таблиц)

С помощью IVExpert были сгенерированы данные в Test data generator



Таким образом были сгенерированы данные:

NUMBER_PH...	PASPORT...
91 805	2 137 303 53
37 741	2 137 309 99
44 266	2 137 329 87
12 987	2 137 331 26
42 520	2 137 354 68
21 036	2 137 446 38
71 940	2 137 607 50
32 269	2 137 617 61
58 334	2 137 627 94
25 555	2 137 648 20
95 656	2 137 658 50
29 022	2 137 737 35
36 072	2 137 801 13
41 908	2 137 893 12
72 733	2 137 941 00
91 846	2 137 962 04
90 945	2 137 979 15
74 837	2 138 014 15
33 029	2 138 024 45
78 432	2 138 201 35
7 175	2 138 224 49
59 157	2 138 252 63
2 293	2 138 358 04
47 684	2 138 453 71
59 180	2 138 492 88

В схему были введены некоторые исправления

```
connect 'localhost:D:\Turism.fdb' user 'SYSDBA' password 'masterkey';
```

```
alter table trip add ID integer;
```

```
alter table trip add ID_excursions integer;
```

```
alter table tour add ID_excursions integer;
```

```
alter table tour drop purchar;
```

```
alter table tour drop excursions;
```

```
alter table tour add id_excursions integer;
```

```
create table excursions (
```

```

id int primary key,
place varchar(30));

alter table trip
add foreign key (id_excursions) references excursions(id);

alter table tour
add foreign key (id_excursions) references excursions(id);

alter table kurort drop constraint integ_57;
id_kurort int,
id_place_of_interest int);
alter table kurort drop id_place_of_interest;

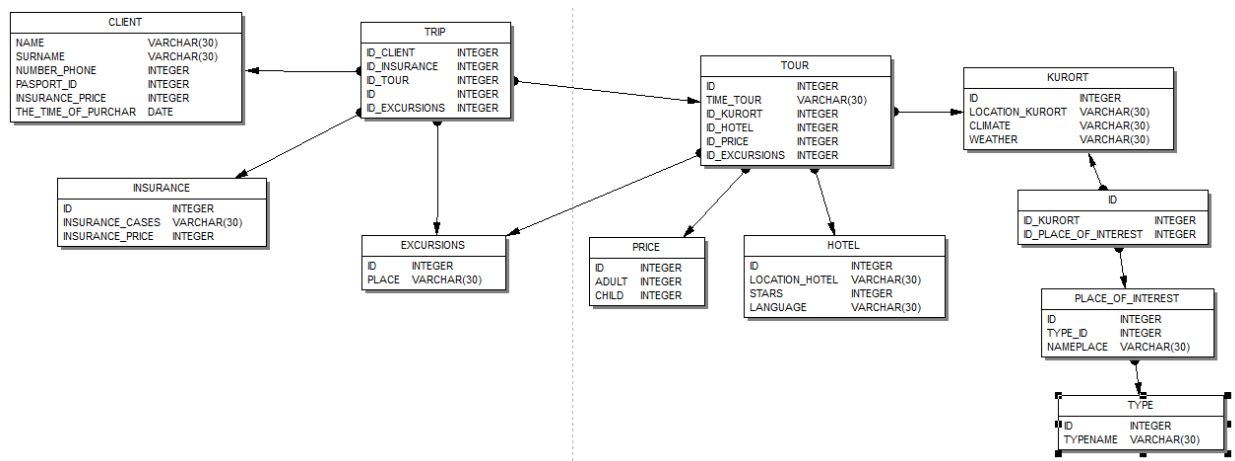
alter table id
dd foreign key (id_place_of_interest) references place_of_interest(id);

alter table id
add foreign key (id_kurort) references kurort(id);

commit;

```

Итоговая ER-диаграмма



Вывод

В результате выполнения лабораторной работы были получены базовые навыки изменения структуры базы данных, а так же навыки создания, изменения таблиц, создания связей между полями таблиц с помощью IVexpert. Также получили навыки заполнения БД тестовыми данными с помощью генератора тестовых данных и научились получать схему отражающую связи таблиц между собой в БД используя дизайнер.

Data Definition Language (DDL) (язык описания данных) — это семейство компьютерных языков, используемых в компьютерных программах для описания структуры баз данных. Запрос «create» используется для создания базы данных, таблицы, индекса, представления или хранимой процедуры. Запрос «alter» используется для изменения существующего объекта базы данных (таблицы, индекса, представления или хранимой процедуры) или самой базы данных. Запрос «drop» используется для удаления существующего объекта базы данных (таблицы, индекса, представления или хранимой процедуры) или самой базы данных. И наконец, в DDL существуют понятия первичного и внешнего ключа, которые осуществляют соблюдение целостности данных. Команды "первичный ключ" primary key, "внешний ключ" foreign key включаются в запросы «create table», «alter table».

Достоинства SQL-DDL: Не большое число команд, простота и лаконичность команд

Недостатки SQL-DDL: В следствие ограниченного числа команд возможны неудобства при реализации специфических структур.