Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра Компьютерных систем и программных технологий

Отчет по лабораторной работе 3

**«Язык SQL - DDL»**

по дисциплине «Базы данных»

Работу выполнил:

студент гр. 43501/3

Родина В.В.

Руководитель:

Мяснов А.В.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2016 г

Санкт-Петербург

2016

1. **Цель работы**

Познакомиться 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)
4. Создание скрипта, заполняющего все таблицы БД данными
5. Выполнение SQL-запросов, изменяющих схему созданной БД по заданию преподавателя
6. Изучение основных возможностей IBExpert. Получение ER-диаграммы созданной БД с помощью Database Designer.
7. Автоматическая генерация данных при помощи IBExpert (для трех или большего числа таблиц, не менее 100000 записей в каждой из выбранных таблиц)
8. **Ход работы**

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.

Разработан скрипт, создающий БД, согласованную со схемой:

connect 'C:\BD\FITNESSCLUB.FDB' user 'SYSDBA' password 'masterkey';

drop database;

create database 'C:\BD\fitnessclub' user 'SYSDBA' password 'masterkey';

create table Clients(

ID\_CLIENT int primary key,

clientName varchar(255),

visitNumbers int,

ID\_DISCOUNT int not null

);

create table Discounts(

ID\_DISCOUNT int primary key,

nameOfDiscount varchar(255),

valueOfDiscount int

);

create table Trainers(

ID\_TRAINER int primary key,

nameOfTrainer varchar(255),

callNumberOfTrainer int,

workingTimeOfTrainer int

);

create table TypeOfClasses(

ID\_TYPEOFCLASS int primary key,

typeClass int

);

create table Place(

ID\_PLACE int primary key,

nameOfPlace varchar(255)

);

create table Classes(

ID\_CLASS int primary key,

ID\_TYPEOFCLASS int not null,

durationOfClass int,

ID\_TRAINER int not null,

ID\_PLACE int not null

);

create table TypeOfTickets(

ID\_TYPEOFTICKET int primary key,

nameOfTicket varchar(255),

priceOfTicket int

);

create table Tickets(

ID\_TICKET int primary key,

ID\_TYPEOFTICKET int not null,

validity int,

ID\_CLIENT int not null

);

create table ClientToClass(

ID\_CLASS int not null,

ID\_CLIENT int not null

);

create table ClassToTicket(

ID\_TYPEOFTICKET int not null,

ID\_TYPEOFCLASS int not null

);

alter table Clients add constraint clients\_to\_discounts

foreign key (ID\_DISCOUNT) references Discounts (ID\_DISCOUNT);

alter table Classes add constraint classes\_to\_place

foreign key (ID\_PLACE) references Place (ID\_PLACE);

alter table Classes add constraint classes\_to\_typeofclasses

foreign key (ID\_TYPEOFCLASS) references TypeOfClasses (ID\_TYPEOFCLASS);

alter table Tickets add constraint tickets\_to\_typeoftickets

foreign key (ID\_TYPEOFTICKET) references TypeOfTickets (ID\_TYPEOFTICKET);

alter table Tickets add constraint tickets\_to\_clients

foreign key (ID\_CLIENT) references Clients (ID\_CLIENT);

alter table Classes add constraint classes\_to\_trainers

foreign key (ID\_TRAINER) references Trainers (ID\_TRAINER);

alter table ClientToClass add constraint clienttoticket\_to\_classes

foreign key (ID\_CLASS) references Classes (ID\_CLASS);

alter table ClientToClass add constraint clienttoticket\_to\_clients

foreign key (ID\_CLIENT) references Clients (ID\_CLIENT);

alter table ClassToTicket add constraint classtoticket\_to\_typeoftickets

foreign key (ID\_TYPEOFTICKET) references TypeOfTickets (ID\_TYPEOFTICKET);

alter table ClassToTicket add constraint classtoticket\_to\_typeofclasses

foreign key (ID\_TYPEOFCLASS) references TypeOfClasses (ID\_TYPEOFCLASS);

commit;

Далее написан скрипт, содержащий SQL – запросы, которые изменяют схему созданной БД по заданию преподавателя:

1. Ввести учет фактических посещений занятий клиентами.
2. Ввести учет дополнительных услуг к абонементам с возможностью их приобретений клиентами.
3. Для абонементов добавить стоимость.

connect 'C:\BD\FITNESSCLUB.FDB' user 'SYSDBA' password 'masterkey';

create table Visits(

ID\_CLIENT int not null,

dateOfVisit date

);

alter table Visits add constraint visits\_to\_clients

foreign key (ID\_CLIENT) references Clients (ID\_CLIENT);

alter table TypeOfTickets drop priceOfTicket;

alter table Tickets add ticketPrice int not null;

create table AddServices(

ID\_SERVICE int primary key,

nameService varchar(255),

priceService int

);

create table ServicesToTypes(

ID\_SERVICE int,

ID\_TYPEOFTICKET int

);

alter table ServicesToTypes add constraint servicetttypes\_to\_serv

foreign key (ID\_SERVICE) references AddServices (ID\_SERVICE);

alter table ServicesToTypes add constraint servicetttypes\_to\_types

foreign key (ID\_TYPEOFTICKET) references TypeOfTickets (ID\_TYPEOFTICKET);

create table AccountServices(

ID\_CLIENT int not null,

ID\_SERVICE int not null,

buyServiceDate date

);

alter table AccountServices add constraint accountservices\_to\_clients foreign key (ID\_CLIENT)

references Clients(ID\_CLIENT);

alter table AccountServices add constraint accountservices\_to\_services foreign key (ID\_SERVICE)

references AddServices(ID\_SERVICE);

commit;

Создадим скрипт, заполняющий все таблицы БД данными:

connect 'C:\BD\FITNESSCLUB.FDB' user 'SYSDBA' password 'masterkey';

insert into discounts (ID\_DISCOUNT, nameOfDiscount, valueOfDiscount)

values (1, 'one friend', 10);

insert into discounts (ID\_DISCOUNT, nameOfDiscount, valueOfDiscount)

values (2, 'VIP', 30);

insert into discounts (ID\_DISCOUNT, nameOfDiscount, valueOfDiscount)

values (3, 'your birthday', 20);

insert into discounts (ID\_DISCOUNT, nameOfDiscount, valueOfDiscount)

values (4, 'no any discounts', 0);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (1, 'lev', 3, 2);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (2, 'maks', 2, 4);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (3, 'lera', 3, 1);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (4, 'vladmir', 10, 3);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (5, 'jay', 20, 4);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (6, 'katya', 75, 2);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (7, 'john', 29, 4);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (8, 'marina', 33, 1);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (9, 'sasha', 0, 3);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (10, 'peter', 50, 4);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (11, 'nastya', 35, 2);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (12, 'matvey', 20, 1);

insert into Clients (ID\_CLIENT, clientName, visitNumbers, ID\_DISCOUNT) values (13, 'galina', 100, 4);

insert into Trainers (ID\_TRAINER, nameOfTrainer, callNumberOfTrainer, workingTimeOfTrainer)

values (1, 'Aleks', 892189, 1);

insert into Trainers (ID\_TRAINER, nameOfTrainer, callNumberOfTrainer, workingTimeOfTrainer)

values (2, 'Lena', 891187, 1);

insert into Trainers (ID\_TRAINER, nameOfTrainer, callNumberOfTrainer, workingTimeOfTrainer)

values (3, 'Rustam', 4558910, 5);

insert into Trainers (ID\_TRAINER, nameOfTrainer, callNumberOfTrainer, workingTimeOfTrainer)

values (4, 'Mariya', 890456, 10);

insert into Trainers (ID\_TRAINER, nameOfTrainer, callNumberOfTrainer, workingTimeOfTrainer)

values (5, 'Mike', 8922156, 3);

insert into Place (ID\_PLACE, nameofPlace) values (1, 'gym');

insert into Place (ID\_PLACE, nameofPlace) values (2, 'small hall');

insert into Place (ID\_PLACE, nameofPlace) values (3, 'big hall');

insert into typeofclasses (ID\_TYPEOFCLASS, typeclass) values (1, 'hard');

insert into typeofclasses (ID\_TYPEOFCLASS, typeclass) values (2, 'cardio');

insert into typeofclasses (ID\_TYPEOFCLASS, typeclass) values (3, 'stretch');

insert into classes (ID\_CLASS, ID\_TYPEOFCLASS, durationOfClass, ID\_TRAINER, ID\_PLACE, nameofclass)

values (1, 1, 60, 3, 1, 'crossfit');

insert into classes (ID\_CLASS, ID\_TYPEOFCLASS, durationOfClass, ID\_TRAINER, ID\_PLACE, nameofclass)

values (2, 3, 90, 1, 3, 'yoga');

insert into classes (ID\_CLASS, ID\_TYPEOFCLASS, durationOfClass, ID\_TRAINER, ID\_PLACE, nameofclass)

values (3, 2, 75, 2, 3, 'aerobica');

insert into classes (ID\_CLASS, ID\_TYPEOFCLASS, durationOfClass, ID\_TRAINER, ID\_PLACE, nameofclass)

values (4, 2, 60, 3, 2, 'dancemix');

insert into classes (ID\_CLASS, ID\_TYPEOFCLASS, durationOfClass, ID\_TRAINER, ID\_PLACE, nameofclass)

values (5, 3, 90, 2, 2, 'pilates');

insert into classes (ID\_CLASS, ID\_TYPEOFCLASS, durationOfClass, ID\_TRAINER, ID\_PLACE, nameofclass)

values (6, 1, 60, 4, 1, 'tabata');

insert into classes (ID\_CLASS, ID\_TYPEOFCLASS, durationOfClass, ID\_TRAINER, ID\_PLACE, nameofclass)

values (7, 2, 45, 5, 3, 'step');

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (1, 'unlimited 1 year', 15000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (2, 'unlimited 1 year only gym', 12000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (3, 'unlimited half of year', 9000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (4, 'yoga', 6000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (5, 'aerobica', 5000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (6, 'dancemix', 4000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (7, 'pilates', 6000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (8, 'tabata', 7000 );

insert into typeoftickets (ID\_TYPEOFTICKET, nameOfTicket, ticketprice)

values (9, 'step', 8000 );

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (1, 1, '2015-09-21', 1);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (2, 3, '2016-01-10', 2);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (3, 2, '2015-12-14', 3);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (4, 6, '2015-06-05', 4);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (5, 9, '2015-11-27', 5);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (6, 4, '2015-09-21', 6);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (7, 8, '2015-10-17', 7);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (8, 5, '2065-01-12', 8);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (9, 7, '2015-06-26', 9);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (10, 6, '2015-11-01', 10);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (11, 9, '2016-01-04', 11);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (12, 1, '2015-07-15', 12);

insert into tickets (ID\_TICKET, ID\_TYPEOFTICKET, validity, ID\_CLIENT)

values (13, 4, '2015-12-07', 13);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (1, 1);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (1, 2);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (1, 3);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (1, 4);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (1, 5);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (1, 6);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (1, 7);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (2, 1);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (3, 1);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (4, 2);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (5, 3);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (6, 4);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (7, 5);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (8, 6);

insert into classtoticket (ID\_TYPEOFTICKET, ID\_CLASS) values (9, 7);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (1, 2);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (1, 3);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (1, 5);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (1, 10);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (1, 6);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (1, 12);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 1);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 2);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 4);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 8);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 13);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 10);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 5);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (2, 7);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (3, 3);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (3, 13);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 1);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 3);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 4);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 5);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 10);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 9);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 7);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 12);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 8);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (4, 11);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (5, 1);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (5, 2);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (5, 6);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (6, 2);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (6, 4);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (6, 10);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (6, 8);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (6, 13);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 1);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 2);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 3);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 4);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 5);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 6);

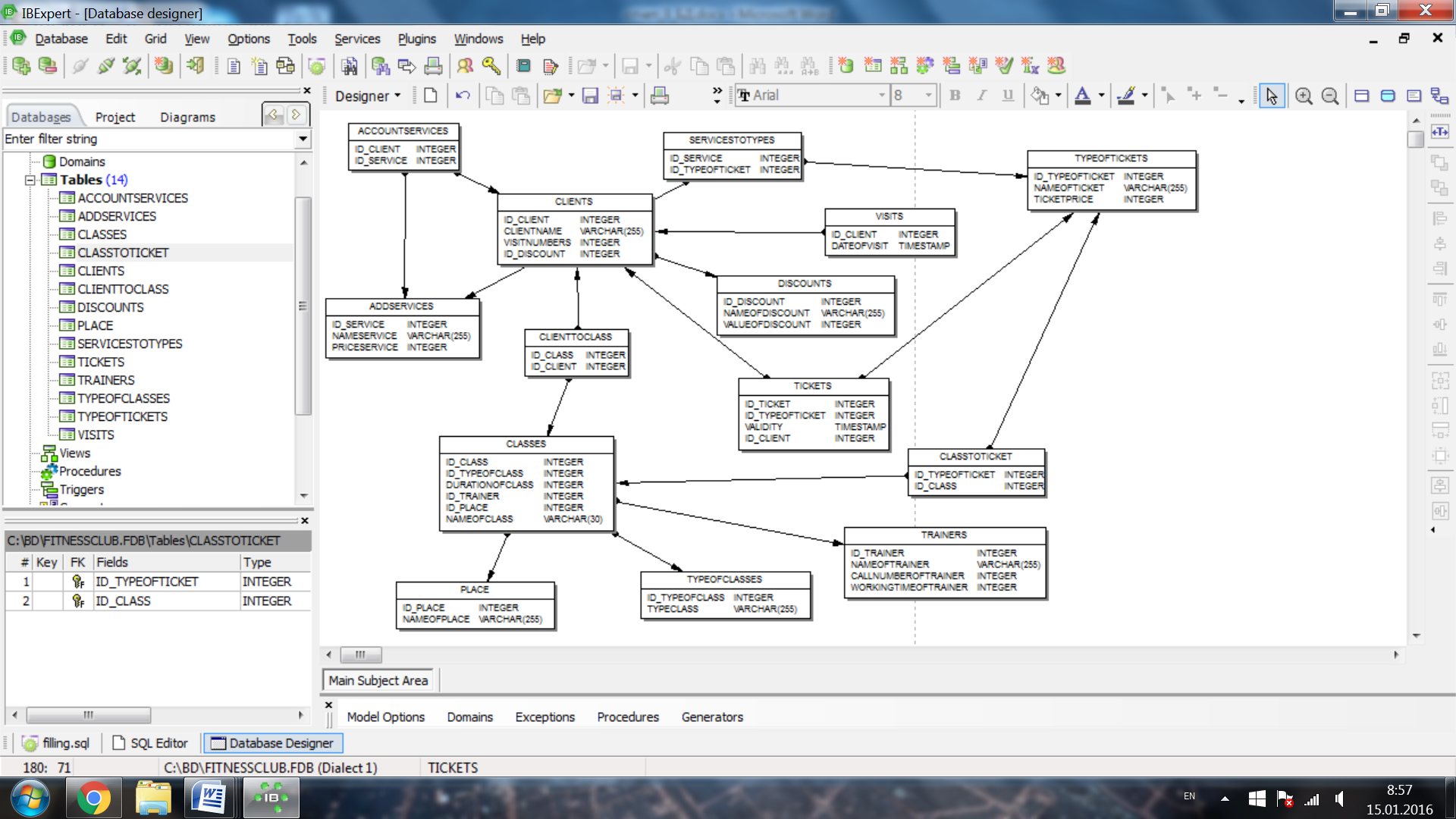
insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 12);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 11);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 9);

insert into clienttoclass (ID\_CLASS, ID\_CLIENT) values (7, 10);

С помощью Database Designer получили ER – диаграмму:



С помощью Test data generator сгенерированы данные - 10000 записей для таблиц: TRAINERS, PLACE, DISCOUNTS.

1. **Выводы**

Таким образом, мы познакомились с языком SQL-DDL. Также написаны скрипты для создания таблиц в БД, для заполнения их данными и для модификации БД по заданию, выданному преподавателем.

С помощью IBExpert сгенерирована ER - диаграмма БД и произведено заполнение таблиц случайными данными.