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**: Разработка структур и нормализация БД

Выполнил студент гр. 43501/1 Чан Ван Кует

(подпись)

Руководитель А.В. Мяснов

(подпись)

“ ” 2015 г.

Санкт-Петербург

2015

1. **Цель работы**

Познакомить студентов 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)

3. **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. **Ход работы**

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

|  |
| --- |
| CONNECT 'E:/FILMS.FDB' user 'SYSDBA' password 'masterkey';  drop database;  CREATE DATABASE 'E:/FILMS.FDB' user 'SYSDBA' password 'masterkey';  CONNECT 'E:/FILMS.FDB' user 'SYSDBA' password 'masterkey';  commit;  CREATE TABLE director  ( director\_id int not null primary key,  name varchar(32) not null  );  CREATE TABLE studio  ( studio\_id int not null primary key,  title varchar(32) not null,  location varchar(32) not null  );  CREATE TABLE films  ( films\_id int not null primary key,  title varchar(32) not null,  director\_id int not null references director(director\_id),  studio\_id int references studio(studio\_id),  rating int  );  CREATE TABLE actors  ( actors\_id int not null primary key,  names varchar(32)  );  CREATE TABLE award\_type  (  award\_type\_id int not null primary key,  title varchar(32) not null  );  CREATE TABLE award  ( award\_id int not null primary key,  award\_name varchar(32) not null,  label int not null,  award\_type\_id int references award\_type(award\_type\_id)  );  CREATE TABLE award\_actors  ( award\_actors\_id int not null primary key,  actors\_id int references actors(actors\_id),  award\_id int references award(award\_id)  );  CREATE TABLE award\_films  ( award\_films\_id int not null primary key,  films\_id int references films(films\_id),  award\_id int references award(award\_id)  );  CREATE TABLE films\_actors  ( films\_id int not null references films(films\_id),  actors\_id int not null references actors(actors\_id)  ); |

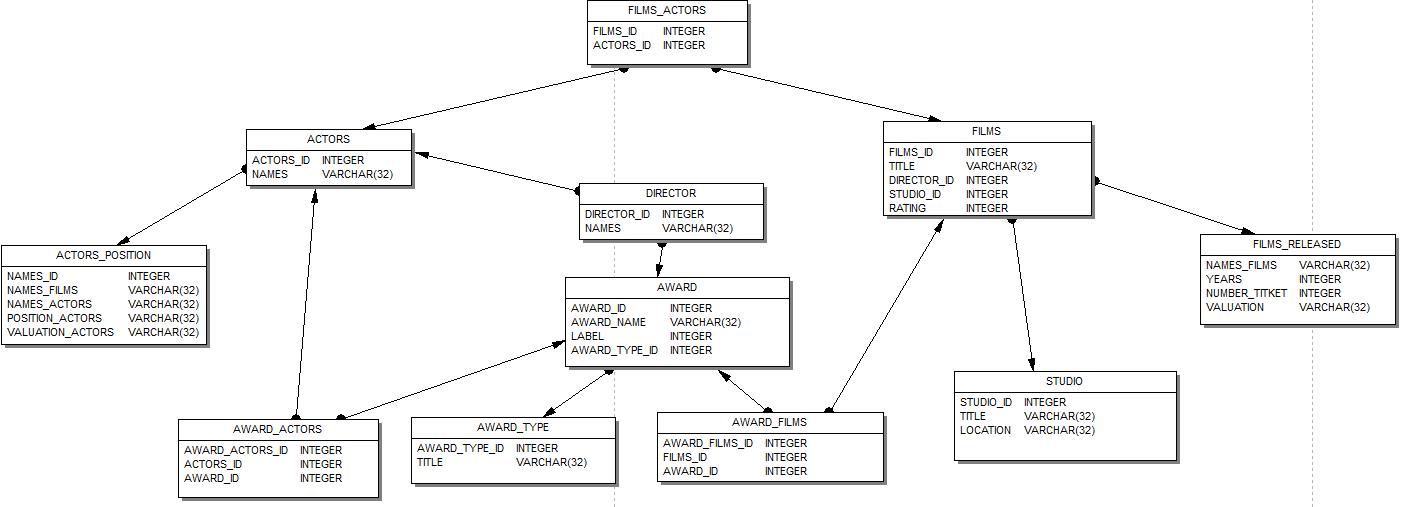
Затем был создан скрипт, модифицирующий базу данных согласно заданию:

Модифицировать схему БД для удовлетворения следующим требованиям:

1. Единого учета людей в фильмах с возможностью указания должности человека в конкретном фильме.
2. Ввести учет проката фильмов в кинотеатрах.

|  |
| --- |
| CONNECT 'E:/FILMS.FDB' user 'SYSDBA' password 'masterkey';  create table actors\_position (  number\_id int not null ,  names\_films varchar(32) not null,  names\_actors varchar(32) not null,  position\_actors varchar(32) not null,  valuation\_actors );  create table films\_released (  names\_films varchar(32) not null,  years int not null ,  number\_titket int not null ,  valuation varchar(32) not null);  insert into director (director\_id, name) values (0, 'Ludovski');  insert into director (director\_id, name) values (1, 'Gorenov');  insert into director (director\_id, name) values (2, 'Lomakov');  insert into studio (studio\_id, title, location) values (0, 'WB', 'USA');  insert into studio (studio\_id, title, location) values (1, 'TF', 'USA');  insert into studio (studio\_id, title, location) values (2, 'MF', 'RF');  insert into films (films\_id, title, director\_id, studio\_id, rating) values (0, 'Svecha', 0, 0, 10);  insert into films (films\_id, title, director\_id, studio\_id, rating) values (1, 'Istoria', 1, 1, 8);  insert into films (films\_id, title, director\_id, studio\_id, rating) values (2, 'Magic', 2, 2, 7);  insert into actors (actors\_id, names) values (0, 'Fedor');  insert into actors (actors\_id, names) values (1, 'Loveski');  insert into actors (actors\_id, names) values (2, 'Maria');  insert into actors (actors\_id, names) values (3, 'Kochan');  insert into actors (actors\_id, names) values (4, 'Pavel');  insert into actors (actors\_id, names) values (5, 'Koren');  insert into actors (actors\_id, names) values (6, 'Polina');  insert into award\_type (award\_type\_id, title) values (0,'Oscar');  insert into award\_type (award\_type\_id, title) values (1,'Palm');  insert into award\_type (award\_type\_id, title) values (2,'MTV');  insert into award\_type (award\_type\_id, title) values (3,'Kanskiy');  insert into award (award\_id, award\_name, label, award\_type\_id) values (0, 'Luchshaya Jenskaya Rol', 0, 0);  insert into award (award\_id, award\_name, label, award\_type\_id) values (1, 'Luchshaya Reshiserskaya Rabota', 1, 1);  insert into award (award\_id, award\_name, label, award\_type\_id) values (2, 'Luchshaya Mushskaia Rol', 1, 2);  insert into award (award\_id, award\_name, label, award\_type\_id) values (3, 'Luchshaya Musika', 1, 3);  insert into award\_actors (award\_actors\_id, actors\_id, award\_id) values (0, 2, 0);  insert into award\_actors (award\_actors\_id, actors\_id, award\_id) values (1, 3, 2);  insert into award\_films (award\_films\_id, films\_id, award\_id) values (0, 0, 1);  insert into award\_films (award\_films\_id, films\_id, award\_id) values (1, 2, 0);  insert into films\_actors (films\_id, actors\_id) values (2, 1);  insert into films\_actors (films\_id, actors\_id) values (2, 0);  insert into films\_actors (films\_id, actors\_id) values (2, 2);  insert into films\_actors (films\_id, actors\_id) values (2, 4);  insert into films\_actors (films\_id, actors\_id) values (1, 6);  insert into films\_actors (films\_id, actors\_id) values (1, 5);  insert into films\_actors (films\_id, actors\_id) values (0, 3);  insert into films\_released (names\_films , years) values ('ionki' , 2010);  insert into films\_released (names\_films , years) values ('octrop',2006);  insert into films\_released (names\_films , years) values ('prirrac',2011);  insert into films\_released (names\_films , years) values ('octrop' ,2006);  insert into films\_released (names\_films , years) values ('vovraxenie' ,2003);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors) values (1,’ionki’, 'timur bekmvmbetov ', 'derected',’otlitrno’);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors) values (2,’ionki’, 'alekcandr voitinxki ', 'derected',’otlitrno’);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors) values (3,’ionki’, ' ivan ergant ', ' actors ',’otlitrno’);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors)) values (4,’ionki’, ' cergay xvenlfrov ', ' actors ',’otlitrno’);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors) values (5,’ionki’, ' cergay poxodaev ', ' actors ',’xoroso’);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors) values (6,’ionki’, ' alina belunko ', ' actors ',’xoroso’);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors) values (7’ionki’, ' maria porosina ', ' actors ',’xoroso’);  insert into actors\_position (names\_id, name\_films , names\_actors,position\_actors,valuation\_actors) values (8,’ionki’, ' Ekaterina vikova ', ' actors ',’otlitrno’);  commit; |

С помощью Database Designer в IBExpert была сгенерирована ER-диаграмма нашей БД:



С помощью Test data generator сгенерировали данные (10000 записей) для 2-х таблиц (actors\_id,names ).

1. **Вывод**

В результате работы было проведено знакомство и изучение языка SQL-DDL. Также были написаны скрипты для создания таблиц в БД, для заполнения их данными и для модификации БД по заданию преподавателя.

С помощью IBExpert была сгенерирована ER-диаграмма нашей БД, было произведено заполнение таблиц случайными данными.

При выполнении работы проблем не возникло.