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 Чинь Тхань Нам

(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.

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. **Ход работы**

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

SET NAMES CYRL;

connect 'd:/library\_films.fdb'

user 'SYSDBA' password 'masterkey';

drop database;

commit;

create database 'd:/library\_films.fdb' user 'SYSDBA' password 'masterkey';

connect 'd:/library\_films.fdb'

user 'SYSDBA' password 'masterkey';

create table film (

film\_id int primary key,

title varchar(255) not null,

year0 int not null,

country varchar(16) not null,

film\_company\_id int not null,

genre\_id int not null,

duration int not null,

rating int not null,

price float not null);

CREATE TABLE Viewer (

Viewer\_id INT NOT NULL PRIMARY KEY,

Name varchar(16),

Birthday date,

Adress VARCHAR(255),

Number\_telephone VARCHAR(16)

);

CREATE TABLE Genre (

Genre\_id INT NOT NULL PRIMARY KEY,

Title varchar(32) NOT NULL);

CREATE TABLE Request (

Request\_id INT NOT NULL PRIMARY KEY,

Viewer\_id INT NOT NULL,

Film\_id INT NOT NULL,

Price float NOT NULL,

Date\_supply\_request date

);

CREATE TABLE Film\_company (

Film\_company\_id INT NOT NULL PRIMARY KEY,

Title varchar(32) NOT NULL,

Year\_establish INT NOT NULL,

Positioning VARCHAR(255) NOT NULL

);

CREATE TABLE Film\_person (

Id INT NOT NULL PRIMARY KEY,

Film\_id INT NOT NULL,

Person\_id INT NOT NULL

);

CREATE TABLE Person (

Person\_id INT NOT NULL PRIMARY KEY,

Name varchar(32) NOT NULL,

birthday date,

Place\_birth varchar(16),

Hight INT NOT NULL,

Character\_id int NOT NULL

);

CREATE TABLE Charactor (

Character\_id int NOT NULL PRIMARY KEY,

Actor blob not null,

Director blob not null,

soundman blob not null

);

ALTER TABLE Film ADD CONSTRAINT Films\_fk0 FOREIGN KEY (Film\_company\_id) REFERENCES Film\_company (Film\_company\_id);

ALTER TABLE Film ADD CONSTRAINT Films\_fk1 FOREIGN KEY (Genre\_id) REFERENCES Genre(Genre\_id);

ALTER TABLE Request ADD CONSTRAINT Request\_fk0 FOREIGN KEY (Viewer\_id) REFERENCES Viewer(Viewer\_id);

ALTER TABLE Request ADD CONSTRAINT Request\_fk1 FOREIGN KEY (Film\_id) REFERENCES Film(Film\_id);

ALTER TABLE Film\_person ADD CONSTRAINT Film\_person\_fk0 FOREIGN KEY (Film\_id) REFERENCES Film(Film\_id);

ALTER TABLE Film\_person ADD CONSTRAINT Film\_person\_fk1 FOREIGN KEY (Person\_id) REFERENCES Person(Person\_id);

ALTER TABLE Person ADD CONSTRAINT Person\_fk0 FOREIGN KEY (Character\_id) REFERENCES Charactor(Character\_id);

Был создан скрипт, заполняющий БД данными:

connect 'd:/library\_films.fdb' user 'SYSDBA' password 'masterkey';

DELETE FROM FILM\_PERSON;

DELETE FROM REQUEST;

DELETE FROM PERSON;

delete FROM CHARACTOR;

delete FROM VIEWER;

DELETE FROM FILM;

delete from FILM\_COMPANY;

delete from GENRE;

insert into GENRE (GENRE\_ID,TITLE) values (1, 'ACTION');

insert into GENRE (GENRE\_ID,TITLE) values (2, 'HORROR');

insert into GENRE (GENRE\_ID,TITLE) values (3, 'FICTION');

insert into GENRE (GENRE\_ID,TITLE) values (4, 'COMEDY');

insert into GENRE (GENRE\_ID,TITLE) values (5, 'ROMANCE');

insert into GENRE (GENRE\_ID,TITLE) values (6, 'CRIME');

insert into FILM\_COMPANY (FILM\_COMPANY\_ID,TITLE,YEAR\_ESTABLISH,POSITIONING) values (1,'EUROPACORP',1999,'PARIS');

insert into FILM\_COMPANY (FILM\_COMPANY\_ID,TITLE,YEAR\_ESTABLISH,POSITIONING) values (2,'FUJISHOJI',1993,'TOKYO');

insert into FILM\_COMPANY (FILM\_COMPANY\_ID,TITLE,YEAR\_ESTABLISH,POSITIONING) values (3,'GOTHAM GROUP',1993,'CALIFORNIA');

insert into FILM\_COMPANY (FILM\_COMPANY\_ID,TITLE,YEAR\_ESTABLISH,POSITIONING) values (4,'BILL AND BEN PRODUCTIONS',2000,'NEWYORK');

insert into FILM\_COMPANY (FILM\_COMPANY\_ID,TITLE,YEAR\_ESTABLISH,POSITIONING) values (5,'COLUMBIA PICTURES',1919,'CALIFORNIA');

insert into FILM\_COMPANY (FILM\_COMPANY\_ID,TITLE,YEAR\_ESTABLISH,POSITIONING) values (6,'MARVEL STUDIOS',1993,'CALIFORNIA');

insert into FILM\_COMPANY (FILM\_COMPANY\_ID,TITLE,YEAR\_ESTABLISH,POSITIONING) values (7,'DNA FILMS',1983,'UNITED KINGDOM');

insert into VIEWER (VIEWER\_ID, NAME, BIRTHDAY, ADRESS, NUMBER\_TELEPHONE) values (1,'MAXIM','1995-01-07','LESNAYA','89118567291');

insert into VIEWER (VIEWER\_ID, NAME, BIRTHDAY, ADRESS, NUMBER\_TELEPHONE) values (2,'TRINHNAM','1992-04-30','AKADEMYTRESKAYA','89118164968');

insert into VIEWER (VIEWER\_ID, NAME, BIRTHDAY, ADRESS, NUMBER\_TELEPHONE) values (3,'LERA','1995-02-22','LESNAYA','89117771345');

insert into VIEWER (VIEWER\_ID, NAME, BIRTHDAY, ADRESS, NUMBER\_TELEPHONE) values (4,'KOLIA','1994-09-27','GRAZDANSKY PROSPEKT','89116879485');

insert into VIEWER (VIEWER\_ID, NAME, BIRTHDAY, ADRESS, NUMBER\_TELEPHONE) values (5,'IANA','1994-06-13','SAINT PETEBURG','89118564567');

insert into FILM(FILM\_ID,TITLE,YEAR0,COUNTRY,FILM\_COMPANY\_ID,GENRE\_ID,DURATION,RATING,PRICE) values (1,'THE TRANSPORTER REFUELED',2015,'AMERICA',1,1,96,8,15.0);

insert into FILM(FILM\_ID,TITLE,YEAR0,COUNTRY,FILM\_COMPANY\_ID,GENRE\_ID,DURATION,RATING,PRICE) values (2,'THE FINAL CURSE',2015,'JAPAN',2,2,90,9,10);

insert into FILM(FILM\_ID,TITLE,YEAR0,COUNTRY,FILM\_COMPANY\_ID,GENRE\_ID,DURATION,RATING,PRICE) values (3,'MAZE RUNNER: THE SCORCH TRIALS',2015,'EURO',3,3,132,9,10);

insert into FILM(FILM\_ID,TITLE,YEAR0,COUNTRY,FILM\_COMPANY\_ID,GENRE\_ID,DURATION,RATING,PRICE) values (4,'ABSOLUTELY ANYTHING',2015,'AMERICA',4,4,85,9,0);

insert into FILM(FILM\_ID,TITLE,YEAR0,COUNTRY,FILM\_COMPANY\_ID,GENRE\_ID,DURATION,RATING,PRICE) values (5,'007: SPECTRE',2015,'AMERICA',5,1,148,10,0);

insert into FILM(FILM\_ID,TITLE,YEAR0,COUNTRY,FILM\_COMPANY\_ID,GENRE\_ID,DURATION,RATING,PRICE) values (6,'ANT-MANT',2015,'AMERICA',6,3,117,10,0);

insert into FILM(FILM\_ID,TITLE,YEAR0,COUNTRY,FILM\_COMPANY\_ID,GENRE\_ID,DURATION,RATING,PRICE) values (7,'NEVER LET ME GO',2010,'EURO',7,5,103,6,0);

insert into REQUEST (REQUEST\_ID, VIEWER\_ID,FILM\_ID,PRICE,DATE\_SUPPLY\_REQUEST) values (1,1,1,15.0,'2015-12-12');

insert into REQUEST (REQUEST\_ID, VIEWER\_ID,FILM\_ID,PRICE,DATE\_SUPPLY\_REQUEST) values (2,2,2,10.0,'2016-01-09');

insert into CHARACTOR(CHARACTER\_ID,ACTOR,DIRECTOR,SOUNDMAN) values (1,1,0,0);

insert into CHARACTOR(CHARACTER\_ID,ACTOR,DIRECTOR,SOUNDMAN) values (2,0,1,0);

insert into CHARACTOR(CHARACTER\_ID,ACTOR,DIRECTOR,SOUNDMAN) values (3,0,0,1);

insert into CHARACTOR(CHARACTER\_ID,ACTOR,DIRECTOR,SOUNDMAN) values (4,1,1,0);

insert into CHARACTOR(CHARACTER\_ID,ACTOR,DIRECTOR,SOUNDMAN) values (5,1,0,1);

insert into CHARACTOR(CHARACTER\_ID,ACTOR,DIRECTOR,SOUNDMAN) values (6,1,1,1);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (1,'CAMILLE DELAMARRE','1979-08-03','France',185,2);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (2,'ED SKREIN','1983-03-29','ENGLAND',190,1);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (3,'MASAYUKI OCHIAI','1958-07-01', 'JAPAN',181,2);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (4,'AIRI TAIRA','1984-12-12','JAPAN',175,1);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (5,'WES BALL','1980-11-28','AMERICA',180,2);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (6,'DYLAN OBRIEN','1991-08-16','AMERICA',185,1);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (7,'TERRY JONES','1942-02-01','ENGLAND',180,2);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (8,'KATE BECKINSALE','1973-07-26','ENGLAND',180,1);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (9,'SAM MENDES','1965-08-01','ENGLAND',183,2 );

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (10,'DANIEL CRAIG','1968-03-02','ENGLAND',187,1);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (11,'PEYTON REED','1964-07-03','AMERICA',190,2);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (12,'PAUL RUDD','1969-04-06','AMERICA',179,1);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (13,'MARK ROMANEK','1989-09-18','AMERICA',180,2);

insert into PERSON(PERSON\_ID,NAME,BIRTHDAY,PLACE\_BIRTH,HIGHT,CHARACTER\_ID) values (14,'KEIRA KNIGHTLEY','1985-03-26','ENGLAND',179,1);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (1,1,1);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (2,1,2);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (3,2,3);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (4,2,4);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (5,3,5);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (6,3,6);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (7,3,7);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (8,4,8);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (9,4,9);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (10,5,10);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (11,5,11);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (12,6,12);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (13,7,13);

insert into FILM\_PERSON (ID,FILM\_ID,PERSON\_ID) values (14,7,14);

Затем был создан скрипт, модифицирующий базу данных согласно заданию:

Выполнить работу согласно [плану](http://tiger.ftk.spbstu.ru/trac/edu-db-2015/wiki/LabDdlSql).

Модифицировать схему БД для удовлетворения следующим требованиям:

1. Ввести учет кинотеатров, продажи фильмов в кинотеатры, количества просмотров и цен билетов.

connect 'd:/library\_films.fdb' user 'SYSDBA' password 'masterkey';

drop table bilet;

drop table CINEMA1;

create table cinema1(

cinema\_id int not null primary key,

title varchar(16),

address varchar(255)

);

create table bilet(

bilet\_id int not null primary key,

film\_id int not null,

cinema\_id int not null,

price float not null,

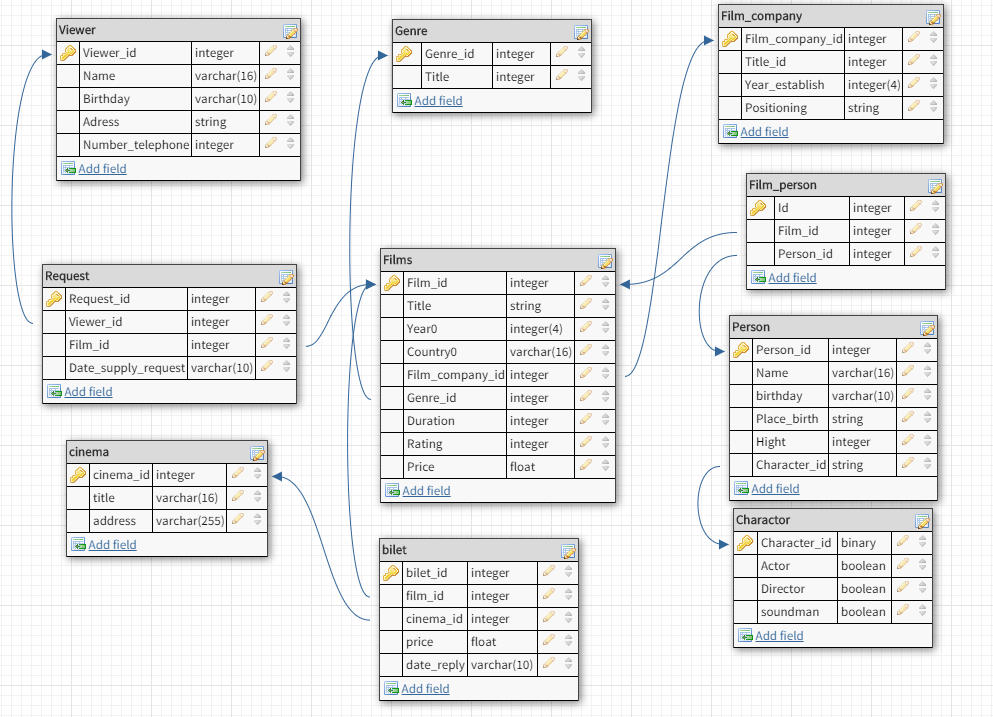
date\_reply date

);

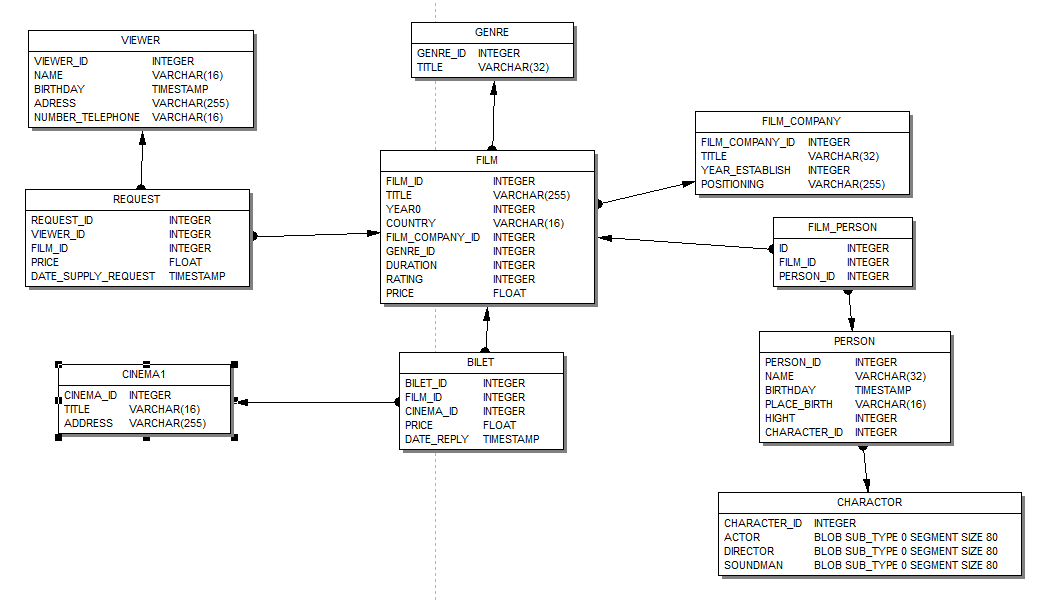
alter table bilet add constraint bilet\_fk0 foreign key (film\_id) references film(film\_id);

alter table bilet add constraint bilet\_fk1 foreign key (cinema\_id) references cinema1(cinema\_id);

Структура БД после модификации:



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



С помощью Test data generator сгенерировали данные (10000 записей) для 3-х таблиц .

1. **Вывод**

В результате работы получается создание базыданых на SQL и изучение языка SQL-DDL. Написал скрипты для созданий таблиц в БД, заполнения их данными и для модификации БД по заданию преподавателя.

С помощью IBexpert, Test data generator сгенерировали данные, получил диаграмму нашей БД