Санкт-Петербургский политехнический университет Петра Великого

Институт информационных технологий и управления

Кафедра компьютерных систем и программных технологий

**Отчёт лабораторной работе №2**

**Дисциплина**: Базы данных

**Тема**: SQL-DDL

Выполнил студент гр. 43501/1 С.И.Бочкова

(подпись)

Руководитель А.В. Мяснов

(подпись)

“ ” 2015 г.

Санкт - Петербург

2015

**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 работы:**

Ниже приведен скрипт создания базы данных:

|  |
| --- |
| create database 'localhost:C:/Users/Sony/Documents/Bd/Films.fdb'  user 'SYSDBA' password 'masterkey'  default character set win1251;  commit;  CREATE TABLE country  (  c\_id INT PRIMARY KEY,  name VARCHAR(80) NOT NULL UNIQUE  );  CREATE TABLE prokat  (  pro\_id INT PRIMARY KEY,  name VARCHAR(255),  money INT NOT NULL  );  CREATE TABLE man  (  man\_id INT PRIMARY KEY,  name VARCHAR(80) NOT NULL,  surname VARCHAR(80) NOT NULL,  second\_name VARCHAR(80),  birthday DATE NOT NULL,  sex CHAR CHECK (sex IN ('M','F')),  city VARCHAR(80) NOT NULL,  country INT NOT NULL REFERENCES country(c\_id)  );  CREATE TABLE career(  id INT PRIMARY KEY,  name VARCHAR(50) NOT NULL UNIQUE  );  CREATE TABLE genre(  id INT PRIMARY KEY,  name VARCHAR(50) NOT NULL UNIQUE  );    CREATE TABLE film  (  f\_id INT PRIMARY KEY,  name VARCHAR(100) NOT NULL,  year\_of\_creation INT NOT NULL CHECK (year\_of\_creation >= 0),  duration INT NOT NULL CHECK (duration >= 0),  budget INT NOT NULL CHECK (budget >= 0),  prok INT REFERENCES prokat(pro\_id)  );  CREATE TABLE soundtrack  (  sound\_id INT PRIMARY KEY,  name VARCHAR(255),  author INT NOT NULL REFERENCES man(man\_id),  film INT NOT NULL REFERENCES film(f\_id),  duration TIME  );  CREATE TABLE company  (  com\_id INT PRIMARY KEY,  name VARCHAR(255) NOT NULL,  year\_of\_foundation INT NOT NULL CHECK (year\_of\_foundation >= 0)  );  CREATE TABLE part  (  film\_id INT NOT NULL REFERENCES film(f\_id),  man\_id INT NOT NULL REFERENCES man(man\_id),  role\_id INT NOT NULL REFERENCES career(id),  CONSTRAINT role PRIMARY KEY (film\_id,man\_id,role\_id)  );  CREATE TABLE premium  (  film\_id INT NOT NULL REFERENCES film(f\_id),  man\_id INT REFERENCES man(man\_id),  CONSTRAINT film\_award PRIMARY KEY (film\_id,man\_id)  );  CREATE TABLE made\_in  (  film\_id INT NOT NULL REFERENCES film(f\_id),  country\_id INT NOT NULL REFERENCES country(c\_id),  CONSTRAINT film\_country PRIMARY KEY (film\_id,country\_id)  );  CREATE TABLE owner  (  man\_id INT NOT NULL REFERENCES man(man\_id),  com\_id INT NOT NULL REFERENCES company(com\_id),  CONSTRAINT owner\_company PRIMARY KEY (man\_id,com\_id)  );  CREATE TABLE production  (  film\_id INT NOT NULL REFERENCES film(f\_id),  com\_id INT NOT NULL REFERENCES company(com\_id),  CONSTRAINT film\_company PRIMARY KEY (film\_id,com\_id)  );  CREATE TABLE style  (  film\_id INT NOT NULL REFERENCES film(f\_id),  genre\_id INT NOT NULL REFERENCES genre(id),  CONSTRAINT film\_genre PRIMARY KEY (film\_id,genre\_id)  );  commit; |

Ниже приведен скрипт заполнения базы данных:

|  |
| --- |
| INSERT INTO country VALUES (1,'Russia');  INSERT INTO country VALUES (2,'USA');  INSERT INTO country VALUES (3,'Germany');  INSERT INTO country VALUES (4,'France');  INSERT INTO country VALUES (5,'United Kingdom');  INSERT INTO country VALUES (6,'Canada');  INSERT INTO country VALUES (7,'New Zealand');  INSERT INTO country VALUES (8,'Austria');  INSERT INTO prokat VALUES(1,'mirash',20000000);  INSERT INTO prokat VALUES(2,'Oazis',30000000);  INSERT INTO prokat VALUES(3,'PIC',40000000);  INSERT INTO prokat VALUES(4,'NewFilms',50000000);  INSERT INTO prokat VALUES(5,'LikeCinima',60000000);  commit;  INSERT INTO man (man\_id,name,surname,birthday,sex,city,country)  VALUES(1,'Steven','Spielberg','1946-12-18','M','Cincinnati',2);  INSERT INTO man (man\_id,name,surname,second\_name,birthday,sex,city,country)  VALUES(2,'James','Cameron','Francis','1954-08-16','M','Kapuskasing',6);  INSERT INTO man (man\_id,name,surname,birthday,sex,city,country)  VALUES(3,'Woody','Allen','1935-12-01','M','New York',2);  INSERT INTO man (man\_id,name,surname,birthday,sex,city,country)  VALUES(4,'Peter','Jackson','1961-10-31','M','Pukerua Bay',7);  INSERT INTO man (man\_id,name,surname,birthday,sex,city,country)  VALUES(5,'Lyv','Tyler','1977-07-01','F','New York',2);  INSERT INTO man (man\_id,name,surname,birthday,sex,city,country)  VALUES(6,'Kate','Winslet','1975-10-05','F','Reading',5);  INSERT INTO man (man\_id,name,surname,birthday,sex,city,country)  VALUES(7,'Leonardo','DiCaprio','1974-11-11','M','Hollywood',2);  INSERT INTO man (man\_id,name,surname,second\_name,birthday,sex,city,country)  VALUES(8,'Arnold','Schwarzenegger','Alois','1947-07-30','M','Thal',8);  INSERT INTO man (man\_id,name,surname,birthday,sex,city,country)  VALUES(9,'James','Horner','1953-08-14','M','Los Angeles',2);  commit;  INSERT INTO career VALUES (1,'director');  INSERT INTO career VALUES (2,'writer');  INSERT INTO career VALUES (3,'actor');  INSERT INTO career VALUES (4,'producer');  INSERT INTO career VALUES (5,'composer');  INSERT INTO career VALUES (6,'operator');  INSERT INTO career VALUES (7,'designer');  INSERT INTO genre VALUES (1,'action');  INSERT INTO genre VALUES (2,'biography');  INSERT INTO genre VALUES (3,'western');  INSERT INTO genre VALUES (4,'crime');  INSERT INTO genre VALUES (5,'documentary');  INSERT INTO genre VALUES (6,'drama');  INSERT INTO genre VALUES (7,'comedy');  INSERT INTO genre VALUES (8,'melodrama');  INSERT INTO genre VALUES (9,'animation');  INSERT INTO genre VALUES (10,'musical');  INSERT INTO genre VALUES (11,'thriller');  INSERT INTO genre VALUES (12,'horror');  INSERT INTO genre VALUES (13,'fantasy');  INSERT INTO genre VALUES (14,'adventure');  INSERT INTO genre VALUES (15,'mystery');  INSERT INTO genre VALUES (16,'romance');  commit;  INSERT INTO film (f\_id,name,year\_of\_creation,duration,budget,prok)  VALUES (1, 'The Lord of the Rings: The Fellowship of the Ring',2001,178,93000000,3);  INSERT INTO film (f\_id,name,year\_of\_creation,duration,budget,prok)  VALUES (2, 'The Lord of the Rings: The Two Towers',2002,179,94000000,2);  INSERT INTO film(f\_id,name,year\_of\_creation,duration,budget,prok)  VALUES (3, 'Titanic',1997,194,200000000,4);  INSERT INTO film(f\_id,name,year\_of\_creation,duration,budget,prok)  VALUES (4, 'Avatar',2009,162,237000000,5);  INSERT INTO film (f\_id,name,year\_of\_creation,duration,budget,prok)  VALUES (5,'The Terminator 2: Judgment Day',1991,137,102000000,1);  INSERT INTO film (f\_id,name,year\_of\_creation,duration,budget,prok)  VALUES (6, 'Catch Me If You Can',2002,141,52000000,2);  INSERT INTO film (f\_id,name,year\_of\_creation,duration,budget,prok)  VALUES (7, 'Manhattan',1979,96,39000000,4);  commit;  INSERT INTO soundtrack (sound\_id,name,author,film,duration)  VALUES (1,'My Heart Will Go ON',9,3,'05:08');  INSERT INTO soundtrack (sound\_id,name,author,film)  VALUES (2,'You Dont Dream In Cryo',9,4);  INSERT INTO soundtrack (sound\_id,name,author,film,duration)  VALUES (3,'Rose',9,3,'03:03');  INSERT INTO company(com\_id,name,year\_of\_foundation)  VALUES(1,'Paramount Pictures',1912);  INSERT INTO company(com\_id,name,year\_of\_foundation)  VALUES(2,'Warner Bros.Pictures',1918);  INSERT INTO company(com\_id,name,year\_of\_foundation)  VALUES(3,'20th Century Fox',1935);  INSERT INTO company(com\_id,name,year\_of\_foundation)  VALUES(4,'Universal Studios',1912);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(1,4,1);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(2,4,1);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(3,2,1);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(4,2,1);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(5,8,3);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(6,7,3);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(6,1,1);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(3,6,3);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(3,7,3);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(3,9,5);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(1,5,3);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(2,5,3);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(7,3,1);  INSERT INTO part(film\_id,man\_id,role\_id)  VALUES(7,3,2);  commit;  INSERT INTO premium(film\_id,man\_id)  VALUES (3,2);  INSERT INTO premium(film\_id,man\_id)  VALUES (3,9);  INSERT INTO premium(film\_id,man\_id)  VALUES (3,7);  INSERT INTO premium(film\_id,man\_id)  VALUES (5,2);  INSERT INTO premium(film\_id,man\_id)  VALUES (7,3);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (3,2);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (1,2);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (1,7);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (2,2);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (2,7);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (4,2);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (4,5);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (5,2);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (6,2);  INSERT INTO made\_in(film\_id,country\_id)  VALUES (7,2);  INSERT INTO production(film\_id,com\_id)  VALUES (3,3);  INSERT INTO production(film\_id,com\_id)  VALUES (3,1);  INSERT INTO production(film\_id,com\_id)  VALUES (4,3);  INSERT INTO style(film\_id,genre\_id)  VALUES(1,13);  INSERT INTO style(film\_id,genre\_id)  VALUES(1,14);  INSERT INTO style(film\_id,genre\_id)  VALUES(2,13);  INSERT INTO style(film\_id,genre\_id)  VALUES(2,1);  INSERT INTO style(film\_id,genre\_id)  VALUES(3,8);  INSERT INTO style(film\_id,genre\_id)  VALUES(3,6);  INSERT INTO style(film\_id,genre\_id)  VALUES(4,13);  INSERT INTO style(film\_id,genre\_id)  VALUES(5,1);  INSERT INTO style(film\_id,genre\_id)  VALUES(5,13);  INSERT INTO style(film\_id,genre\_id)  VALUES(6,6);  INSERT INTO style(film\_id,genre\_id)  VALUES(6,2);  INSERT INTO style(film\_id,genre\_id)  VALUES(6,4);  commit; |

По заданию преподавателя:

1. Добавить учет гонораров для артистов, снявшихся в фильмах.

|  |
| --- |
| CREATE TABLE money(  money\_id INT PRIMARY KEY,  mon\_man INT NOT NULL  );  CREATE TABLE ganorar  (  film\_id INT NOT NULL REFERENCES film(f\_id),  man\_id INT NOT NULL REFERENCES man(man\_id),  money\_id INT NOT NULL REFERENCES money(money\_id),  CONSTRAINT monn PRIMARY KEY (film\_id,man\_id,money\_id)  );  INSERT INTO money VALUES(1,20000000);  INSERT INTO money VALUES(2,30000000);  INSERT INTO money VALUES(3,40000000);  INSERT INTO money VALUES(4,50000000);  INSERT INTO money VALUES(5,60000000);  INSERT INTO ganorar(film\_id,man\_id,money\_id)  VALUES(1,4,1);  INSERT INTO ganorar(film\_id,man\_id,money\_id)  VALUES(2,3,2);  INSERT INTO ganorar(film\_id,man\_id,money\_id)  VALUES(4,2,5);  INSERT INTO ganorar(film\_id,man\_id,money\_id)  VALUES(5,1,3);  INSERT INTO ganorar(film\_id,man\_id,money\_id)  VALUES(3,5,4); |

2.Добавить учет прокатных компаний, с возможностью указания выручки за прокат фильма.

|  |
| --- |
| CREATE TABLE prokat\_company(  film\_id INT NOT NULL REFERENCES film(f\_id),  pro\_id INT NOT NULL REFERENCES prokat(pro\_id),  CONSTRAINT prokats PRIMARY KEY (film\_id,pro\_id)  );  INSERT INTO prokat\_company(film\_id,pro\_id)  VALUES(1,1);  INSERT INTO prokat\_company(film\_id,pro\_id)  VALUES(2,3);  INSERT INTO prokat\_company(film\_id,pro\_id)  VALUES(4,5);  INSERT INTO prokat\_company(film\_id,pro\_id)  VALUES(1,2);  INSERT INTO prokat\_company(film\_id,pro\_id)  VALUES(4,3);  INSERT INTO prokat\_company(film\_id,pro\_id)  VALUES(6,4); |

**Вывод:**

В процессе проведения работы были изучены команды по созданию, редактированию структуры БД, установки ограничений на диапазоны значений.