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

Выполнила студентка гр. 43501/1 В.В. Ракова

(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:

|  |
| --- |
| connect 'D:\Litpub\ONE'  user 'SYSDBA' password 'masterkey';  drop database;  create database 'D:\Litpub\ONE' user 'SYSDBA' password 'masterkey';  connect 'D:\Litpub\ONE' user 'SYSDBA' password 'masterkey';  create table BOOK(  ID\_BOOK int primary key  ,Books\_name varchar(255)  ,Books\_rating float  ,Numder\_of\_pages int  );  create table AUTHOR(  ID\_AUTHOR int primary key  ,Autors\_name varchar(255)  ,date\_of\_born date  );  create table REDACTOR(  ID\_REDACTOR int primary key  ,redactor\_name varchar(255)  ,experience\_red int  );  create table GENRE(  ID\_GENRE int primary key  ,genre\_name varchar(255)  );  create table AUTHOR\_BOOK(  ID\_COMAB int primary key  ,ID\_AUTHOR int not null  ,ID\_BOOK int not null  );  create table RATING\_SALARY(  ID\_GET int primary key  ,ID\_AUTHOR int not null  ,rating\_author float  ,experience\_author int  ,salary\_author int  );  create table EDITION(  ID\_EDITION int primary key  ,number\_of\_copy int  ,ID\_BOOK int not null  ,number\_of\_issue int  );  create table REDACTOR\_BOOK(  ID\_COMRB int primary key  ,ID\_REDACTOR int not null  ,ID\_BOOK int not null  );  create table GENRE\_BOOK(  ID\_COMGB int primary key  ,ID\_GENRE int not null  ,ID\_BOOK int not null  );  alter table AUTHOR\_BOOK add constraint com\_authot\_book1 foreign key (ID\_BOOK)  references BOOK(ID\_BOOK);  alter table AUTHOR\_BOOK add constraint com\_authot1\_book foreign key (ID\_AUTHOR)  references AUTHOR(ID\_AUTHOR);  alter table REDACTOR\_BOOK add constraint com\_red1\_book foreign key (ID\_REDACTOR)  references REDACTOR(ID\_REDACTOR);  alter table REDACTOR\_BOOK add constraint com\_red\_book1 foreign key (ID\_BOOK)  references BOOK(ID\_BOOK);  alter table GENRE\_BOOK add constraint com\_genre\_book1 foreign key (ID\_BOOK)  references BOOK(ID\_BOOK);  alter table GENRE\_BOOK add constraint com\_genre1\_book foreign key (ID\_GENRE)  references GENRE(ID\_GENRE);  alter table EDITION add constraint adition\_book foreign key (ID\_BOOK)  references BOOK(ID\_BOOK);  alter table RATING\_SALARY add constraint rating\_author foreign key (ID\_AUTHOR)  references AUTHOR(ID\_AUTHOR); |

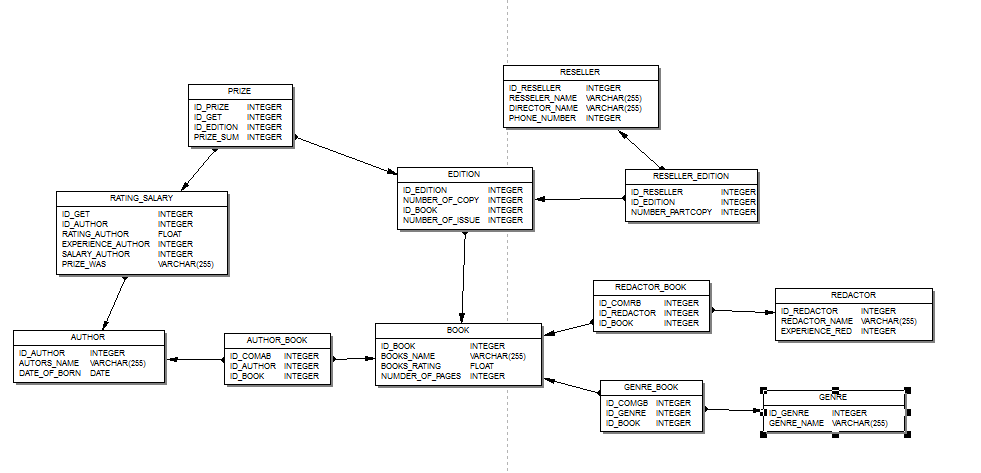
Затем был создан скрипт, модифицирующий базу данных согласно заданию:

Модифицировать схему БД для удовлетворения следующим требованиям:

1. Ввести возможность учета выплат авторам по каждому тиражу.
2. Ввести учет распределения тиражей по реселлерам.

|  |
| --- |
| connect 'D:\Litpub\ONE.fdb'  user 'SYSDBA' password 'masterkey';  alter table RATING\_SALARY add prize\_was varchar(255);  create table PRIZE(  ID\_PRIZE int primary key  ,ID\_GET int not null  ,ID\_EDITION int not null  ,prize\_sum int  );  create table RESELLER(  ID\_RESELLER int primary key  ,resseler\_name varchar(255)  ,director\_name varchar(255)  ,phone\_number int  );  create table RESELLER\_EDITION(  ID\_RESELLER int not null  ,ID\_EDITION int not null  ,number\_partcopy int  );  alter table PRIZE add constraint get\_prize foreign key (ID\_GET)  references RATING\_SALARY(ID\_GET);  alter table PRIZE add constraint edition\_prize foreign key (ID\_EDITION)  references EDITION(ID\_EDITION);  alter table RESELLER\_EDITION add constraint reseller\_edition1 foreign key (ID\_EDITION)  references EDITION(ID\_EDITION);  alter table RESELLER\_EDITION add constraint reseller1\_edition foreign key (ID\_RESELLER)  references RESELLER(ID\_RESELLER); |

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



Был создан скрипт, заполняющий БД данными:

|  |
| --- |
| connect 'D:\Litpub\ONE.fdb'  user 'SYSDBA' password 'masterkey';  insert into BOOK(ID\_BOOK,Books\_name,Books\_rating,Numder\_of\_pages ) values (1,'Flower of god',8.3,453);  insert into BOOK(ID\_BOOK,Books\_name,Books\_rating,Numder\_of\_pages ) values (2,'Lord of the Rings',9.7,1235);  insert into BOOK(ID\_BOOK,Books\_name,Books\_rating,Numder\_of\_pages ) values (3,'Good work',3.5,321);  insert into BOOK(ID\_BOOK,Books\_name,Books\_rating,Numder\_of\_pages ) values (4,'I am Robot',8.9,2679);  insert into BOOK(ID\_BOOK,Books\_name,Books\_rating,Numder\_of\_pages ) values (5,'Cook together ',5.3,128);  insert into AUTHOR(ID\_AUTHOR,Autors\_name,date\_of\_born) values (1,'A.B. Ivanov', '1969-10-12');  insert into AUTHOR(ID\_AUTHOR,Autors\_name,date\_of\_born) values (2,'B.B. Ivanov', '1975-11-14');  insert into AUTHOR(ID\_AUTHOR,Autors\_name,date\_of\_born) values (3,'J.R.R. Tolkien', '1892-01-03');  insert into AUTHOR(ID\_AUTHOR,Autors\_name,date\_of\_born) values (4,'A.K. Litvinov', '1982-9-25');  insert into AUTHOR(ID\_AUTHOR,Autors\_name,date\_of\_born) values (5,'I. Asimov', '1902-01-02');  insert into AUTHOR(ID\_AUTHOR,Autors\_name,date\_of\_born) values (6,'B.B. Kotova', '1956-08-12');  insert into REDACTOR(ID\_REDACTOR,redactor\_name,experience\_red)values(1,'A.A. Trovlev', 5);  insert into REDACTOR(ID\_REDACTOR,redactor\_name,experience\_red)values(2,'A.C. Gotov', 17);  insert into REDACTOR(ID\_REDACTOR,redactor\_name,experience\_red)values(3,'D.E. Kotov', 1);  insert into GENRE(ID\_GENRE,GENRE\_NAME) values(1,'Fiction');  insert into GENRE(ID\_GENRE,GENRE\_NAME) values(2,'Doc');  insert into GENRE(ID\_GENRE,GENRE\_NAME) values(3,'Drama');  insert into EDITION(ID\_EDITION,NUMBER\_OF\_COPY,ID\_BOOK,NUMBER\_OF\_ISSUE) values(1,20000,3,2);  insert into EDITION(ID\_EDITION,NUMBER\_OF\_COPY,ID\_BOOK,NUMBER\_OF\_ISSUE) values(2,50000,1,1);  insert into EDITION(ID\_EDITION,NUMBER\_OF\_COPY,ID\_BOOK,NUMBER\_OF\_ISSUE) values(3,100000,4,5);  insert into EDITION(ID\_EDITION,NUMBER\_OF\_COPY,ID\_BOOK,NUMBER\_OF\_ISSUE) values(4,10000,2,2);  insert into RESELLER(ID\_RESELLER,resseler\_name,director\_name,phone\_number) values(1,'ABGroup','A.B. Igorev', 6543234);  insert into RESELLER(ID\_RESELLER,resseler\_name,director\_name,phone\_number) values(2,'Legend of fiction','C.K. Mohov', 6579230);  insert into RESELLER(ID\_RESELLER,resseler\_name,director\_name,phone\_number) values(3,'PopularBook','A.H. Vasin', 6598981);  insert into RATING\_SALARY(ID\_GET,ID\_AUTHOR,rating\_author,experience\_author,salary\_author,prize\_was) values(1,1,4.2,13,35000, 'Never before');  insert into RATING\_SALARY(ID\_GET,ID\_AUTHOR,rating\_author,experience\_author,salary\_author,prize\_was) values(2,2,4.7,8,47000, 'Last in September');  insert into RATING\_SALARY(ID\_GET,ID\_AUTHOR,rating\_author,experience\_author,salary\_author,prize\_was) values(3,4,5.0,15,89000, 'Last in October');  insert into RATING\_SALARY(ID\_GET,ID\_AUTHOR,rating\_author,experience\_author,salary\_author,prize\_was) values(4,1,3.5,4,21000, 'Never before');  insert into PRIZE(ID\_PRIZE,ID\_GET,ID\_EDITION,prize\_sum) values(1,2,1,5000);  insert into PRIZE(ID\_PRIZE,ID\_GET,ID\_EDITION,prize\_sum) values(2,3,3,13000);  insert into AUTHOR\_BOOK(ID\_COMAB,ID\_AUTHOR,ID\_BOOK) values(1,1,1);  insert into AUTHOR\_BOOK(ID\_COMAB,ID\_AUTHOR,ID\_BOOK) values(2,1,2);  insert into AUTHOR\_BOOK(ID\_COMAB,ID\_AUTHOR,ID\_BOOK) values(3,2,3);  insert into AUTHOR\_BOOK(ID\_COMAB,ID\_AUTHOR,ID\_BOOK) values(4,3,4);  insert into AUTHOR\_BOOK(ID\_COMAB,ID\_AUTHOR,ID\_BOOK) values(5,4,5);  insert into AUTHOR\_BOOK(ID\_COMAB,ID\_AUTHOR,ID\_BOOK) values(6,5,6);  insert into REDACTOR\_BOOK(ID\_COMRB,ID\_REDACTOR,ID\_BOOK) values(1,1,1);  insert into REDACTOR\_BOOK(ID\_COMRB,ID\_REDACTOR,ID\_BOOK) values(2,2,2);  insert into REDACTOR\_BOOK(ID\_COMRB,ID\_REDACTOR,ID\_BOOK) values(3,3,3);  insert into REDACTOR\_BOOK(ID\_COMRB,ID\_REDACTOR,ID\_BOOK) values(4,2,4);  insert into REDACTOR\_BOOK(ID\_COMRB,ID\_REDACTOR,ID\_BOOK) values(5,2,5);  insert into GENRE\_BOOK(ID\_COMGB,ID\_GENRE,ID\_BOOK) values(1,3,1);  insert into GENRE\_BOOK(ID\_COMGB,ID\_GENRE,ID\_BOOK) values(2,1,1);  insert into GENRE\_BOOK(ID\_COMGB,ID\_GENRE,ID\_BOOK) values(3,3,2);  insert into GENRE\_BOOK(ID\_COMGB,ID\_GENRE,ID\_BOOK) values(4,3,3);  insert into GENRE\_BOOK(ID\_COMGB,ID\_GENRE,ID\_BOOK) values(5,1,4);  insert into GENRE\_BOOK(ID\_COMGB,ID\_GENRE,ID\_BOOK) values(6,2,5);  insert into RESELLER\_EDITION(ID\_RESELLER,ID\_EDITION,number\_partcopy) values(1,1, 10000);  insert into RESELLER\_EDITION(ID\_RESELLER,ID\_EDITION,number\_partcopy) values(2,3, 15000);  insert into RESELLER\_EDITION(ID\_RESELLER,ID\_EDITION,number\_partcopy) values(3,2, 20000);  insert into RESELLER\_EDITION(ID\_RESELLER,ID\_EDITION,number\_partcopy) values(2,4, 8000);  insert into RESELLER\_EDITION(ID\_RESELLER,ID\_EDITION,number\_partcopy) values(3,2, 15000); |

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

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.

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.Со строками такая группа операторов не работает вовсе.

* CREATE — создание объекта (например таблицы);
* ALTER— изменение объекта (например добавление/изменение полей);
* DROP — удаление объекта;

После выполнения данных команд автоматически выполняется команда <commit>, т.е. выполненные изменения уже нельзя отменить, поэтому следует быть предельно внимательным при работе с данным языком. Также можно отметить что тексты SQL-запросов, написанные на SQL-DDL могут быть достаточно легко перенесены из одной СУБД в другую.

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

При выполнении работы проблем не возникло.