**Федеральное агентство по образованию**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**Институт информационных технологий и управления**

**Кафедра компьютерных систем и программных технологий**

**ОТЧЕТ**

**по лабораторной работе №3**

# « Язык SQL-DDL »

**Студент гр.43501/1: Русаков Р.В.**

**Преподаватель: Мяснов А.В.**

**Санкт-Петербург**

**2014**

1. **Цели работы**

Познакомить студентов с основами проектирования схемы БД, языком описания сущностей и ограничений БД SQL-DDL.

1. **Создание скрипта БД в соответствии с согласованной схемой**

**CREATE TABLE pub\_houses (**

**pbh\_id INT PRIMARY KEY,**

**name VARCHAR(50),**

**address VARCHAR(50),**

**country VARCHAR(50),**

**phone VARCHAR(30)**

**);**

**CREATE TABLE authors (**

**au\_id INT PRIMARY KEY,**

**first\_name VARCHAR(50),**

**last\_name VARCHAR(50),**

**birth\_date DATE,**

**death\_time DATE,**

**country VARCHAR(50)**

**);**

**CREATE TABLE books (**

**book\_id INT PRIMARY KEY,**

**pbh\_id INT references pub\_houses(pbh\_id),**

**pages INT,**

**price FLOAT,**

**name VARCHAR(50),**

**format VARCHAR(50),**

**au\_id INT references authors(au\_id),**

**udc varchar(50),**

**pub\_year varchar(5)**

**);**

**CREATE TABLE genres (**

**genre\_id INT PRIMARY KEY,**

**name VARCHAR(50)**

**);**

**CREATE TABLE users (**

**user\_id INT PRIMARY KEY,**

**name VARCHAR(50),**

**login VARCHAR(30),**

**password VARCHAR(30),**

**reg\_date DATE,**

**info VARCHAR(100),**

**rating INT**

**);**

**CREATE TABLE books\_genre (**

**book\_id INT references books(book\_id),**

**genre\_id INT references genres(genre\_id)**

**);**

**CREATE TABLE stores (**

**store\_id INT PRIMARY KEY,**

**name VARCHAR(50),**

**address VARCHAR(50),**

**phone VARCHAR(30),**

**work\_time VARCHAR(50)**

**);**

**CREATE TABLE books\_avaliability (**

**book\_id INT references books(book\_id),**

**store\_id INT references stores(store\_id),**

**cnt INT**

**);**

**CREATE TABLE reviews (**

**review\_id INT PRIMARY KEY,**

**book\_id INT references books(book\_id),**

**user\_id INT references users(user\_id),**

**rev\_date DATE,**

**text VARCHAR(300),**

**usefulness INT,**

**rate INT**

**);**

**CREATE TABLE top100 (**

**book\_id INT references books(book\_id),**

**top100\_id INT PRIMARY KEY**

**);**

**CREATE TABLE orders (**

**order\_id INT PRIMARY KEY,**

**user\_id INT references users(user\_id),**

**store\_id INT references stores(store\_id),**

**address VARCHAR(70),**

**order\_date\_create DATE,**

**order\_date\_destination DATE**

**);**

**CREATE TABLE orders\_entre (**

**order\_id INT references orders(order\_id),**

**book\_id INT references books(book\_id),**

**book\_cnt INT,**

**book\_price FLOAT**

**);**

1. **Скрипт, заполняющий все таблицы БД данными**

**insert into authors values(1,'Fyodor',' Dostoyevsky','11-11-1821','09-02-1881','Russia');**

**insert into authors values(2,'Nikolai','Gogol ','03-31-1809','03-04-1852','Russia');**

**insert into authors values(3,'Anton','Chekhov ','01-29-1860','07-15-1904','Russia');**

**insert into pub\_houses values (1,'Eksmo','MSK Lenina St.22','Russia','143-452-54');**

**insert into pub\_houses values (2,'Prosveshenie','SPB Pravdi St.27','Russia','142-452-234');**

**insert into pub\_houses values (3,'Azbuka','MSK Krilova St.54','Russia','145-44457-6234');**

**insert into books values (1,2,120,200,'The Cherry Orchard','Soft',1,'3676-235-12','2007');**

**insert into books values (2,2,267,923,'Nevsky Prospekt','Soft',2,'38726-234-34','2010');**

**insert into books values (3,1,243,545,'Utro v Lesu','Hard',3,'565986-734-23','2001');**

**insert into genres values(1,'Detective');**

**insert into genres values(2,'Drama');**

**insert into genres values(3,'Melodrama');**

**insert into books\_genre values(1,2);**

**insert into books\_genre values(1,1);**

**insert into books\_genre values(2,3);**

**insert into books\_genre values(3,1);**

**insert into stores values(1,'Kizgniy Dom','Knishayz st.21','234-545-35','08:00-22:30 Bez vihodnih');**

**insert into stores values(2,'Knizgniy Rai','Knishayz st.22','234-545-34','08:00-22:30 Bez vihodnih');**

**insert into stores values(3,'Knizgniy Les','Knishayz st.23','234-545-25','08:00-22:30 Bez vihodnih');**

**insert into users values(1,'Super','Super','Super','04-02-2011','I like books','20');**

**insert into users values(2,'Knigolub','Login','Password','05-02-2011','I love books','30');**

**insert into users values(3,'Harry','Potter','Expeliarmus','07-02-2011','Magic boy','2000');**

**insert into reviews values(1,2,1,'02-05-2011','Enjoy it','30','4');**

**insert into reviews values(2,1,1,'03-05-2011','Like it','40','5');**

**insert into reviews values(3,3,1,'04-05-2011','Norm','100','4');**

**insert into top100 values(1,2);**

**insert into top100 values(2,1);**

**insert into top100 values(3,3);**

**insert into books\_avaliability values(1,1,200);**

**insert into books\_avaliability values(2,2,400);**

**insert into books\_avaliability values(3,3,400);**

**insert into orders values(1,2,1,'Lenina St.1','01-01-2011','02-01-2011');**

**insert into orders values(2,1,1,'0','02-01-2011','03-01-2011');**

**insert into orders values(3,3,2,'0','03-01-2011','04-01-2011');**

**insert into orders\_entre values(1,2,4,54);**

**insert into orders\_entre values(2,1,1,242);**

**insert into orders\_entre values(3,3,3,324);**

1. **Выполнение SQL-запросов, изменяющих схему созданной БД** **по заданию преподавателя**
   1. **Ввести возможность соавторства**

**Введена вспомогательная таблица с соответствием книг и соавторов**

**create table co\_authors(au\_id INT references authors(au\_id),book\_id INT references books(book\_id));**

* 1. **Ввести возможность заказа через интернет. С доставкой. Без привязки к магазину.**

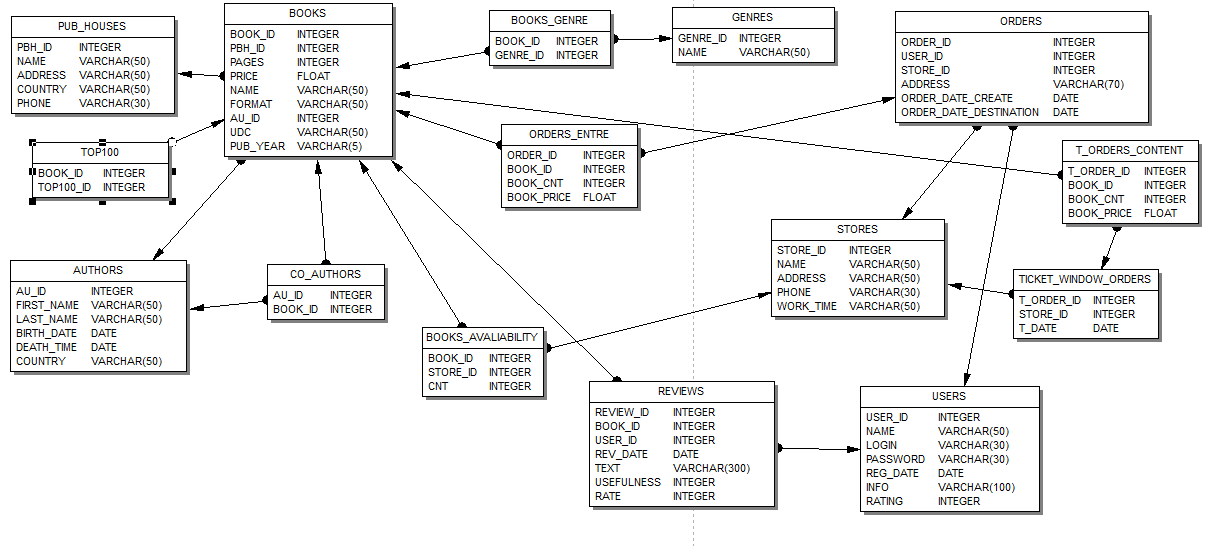
**Таблица orders содержит в себе заказы, сделанные через интернет, и содержит поля store\_id и address. Если в поле store\_id ненулевое значение, это означает , что пользователь решил забрать заказ в магазине (самовывоз). Если значение нулевое, заказ отправляется по адресу в поле address. Для регистрации заказов, сделанных на кассе в магазине, введена таблица ticket\_window\_orders и промежуточная таблица t\_orders\_content, связывающая заказы и книги.**

**create table ticket\_window\_orders (t\_order\_id INT PRIMARY KEY,store\_id INT references stores(store\_id),t\_date DATE);**

**create table t\_orders\_content( t\_order\_id INT references ticket\_window\_orders(t\_order\_id),book\_id INT references books(book\_id),book\_cnt INT,book\_price FLOAT);**

1. **Создание ER- диаграммы в IBExpert**

**После регистрации созданной БД в IBExpert, выполнили Tools/Database designer**

****

**Рис.5.1. ER- диаграмма**

1. **Автоматически сгенерируйте данные при помощи IBExpert**

**При помощи Tools/Test data generator были сгенерированы по 100000 записей в каждой таблице, для первичных индексов была использована автоинкрементация, для строковых данных пункт Get from list, для внешних ключей пункт Get from another table, для численных данных случайные числа с заданным диапазоном.**



**Рис.6.1. Пример заполненной таблицы Reviews**

1. **Вывод**

**В ходе работы были написаны скрипты для создания структуры БД и её наполнения. На практике пришлось столкнуться с ограничениями целостности, так как при неправильном порядке заполнения или создания таблиц возникали ошибки, связанные с отсутствием полей, на которые ссылается внешний ключ. Недостатками DDL является ненаглядность и затруднения при отладке. К достоинствам можно отнести специфические операции работы с реляционными БД, такие как задание внешних ключей, построение индексов для часто используемых атрибутов, явное автоматическое задание первичных ключей, зачастую необходимых при работе с таблицей. К тому же СУБД ведёт контроль этих операций и не даёт совершить заведомо ложных действий. Всё это затруднительно было бы реализовать, используя, допустим, иерархическую модель( например храня базу в файловой системе).**