Санкт-Петербургский политехнический университет

Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

Отчет по лабораторной работе

Дисциплина: «Базы данных»

Тема: «Язык SQL-DDL»

Выполнил студент гр. 43501/3 М.Ю. Попсуйко

(подпись)

Преподаватель А.В. Мяснов

(подпись)

“\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_ 2016 г.

Санкт-Петербург

2016

Оглавление

[Цель работы: 3](#_Toc465781390)

[Выполнение работы: 3](#_Toc465781391)

[Создание первичной схемы БД 3](#_Toc465781392)

[Модернизация БД 6](#_Toc465781393)

[Автоматическая генерация данных для заполнения БД с помощью IBExpert 8](#_Toc465781394)

[Выводы: 10](#_Toc465781395)

# Цель работы:

1. Самостоятельное изучение SQL-DDL;
2. Создание скрипта БД в соответствии с согласованной схемой (первичные и внешние ключи, ограничения на диапазоны значений);
3. Создать скрипт, заполняющий все таблицы БД данными;
4. Выполнение SQL-запросов, изменяющих схему созданной БД;
5. Изучение основных возможностей IBExpert. Получение ER-диаграммы, созданной БД с помощью Database Designer;
6. Автоматически сгенерировать данные при помощи IBExpert (не менее 100000 записей в каждой из выбранных таблиц).

# Выполнение работы:

В качестве индивидуального задания была выбрана тема «Создание базы данных автосалона». Автосалон будет продавать разные модели машин одной марки. В базе данных должна храниться информация о клиенте, о выбранном им автомобиле, о комплектации авто, установленных опциях, заводе изготовителе.

Создание первичной схемы БД

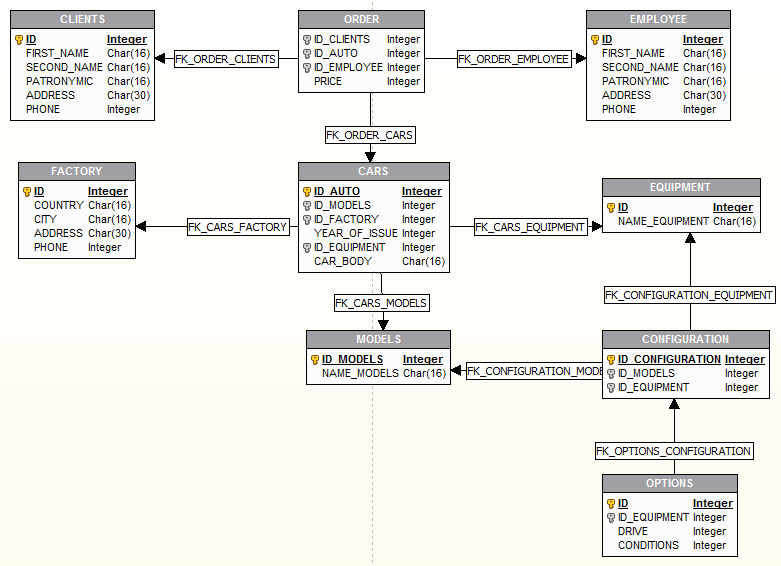


Рис. 1 SQL-схема БД

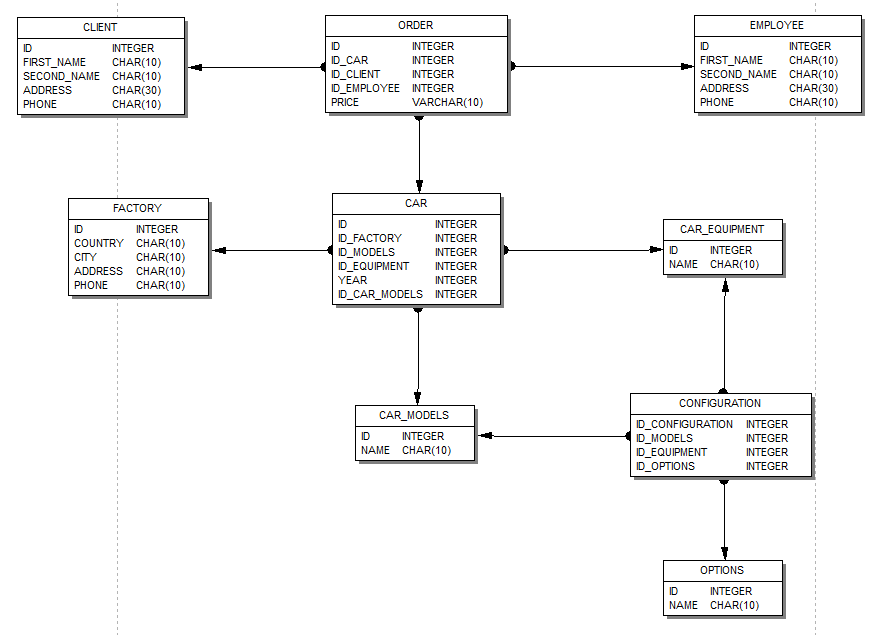


Рис. 2 ER-диаграмма БД

Согласно данной схеме был написан скрипт по созданию необходимых таблицы с ограничениями на диапазоны значений:

CREATE TABLE CAR (

ID INTEGER NOT NULL,

ID\_FACTORY INTEGER,

ID\_MODELS INTEGER,

ID\_EQUIPMENT INTEGER,

"YEAR" INTEGER,

ID\_CAR\_MODELS INTEGER

);

ALTER TABLE CAR ADD PRIMARY KEY (ID);

CREATE TABLE "ORDER" (

ID INTEGER NOT NULL,

ID\_CAR INTEGER,

ID\_CLIENT INTEGER,

ID\_EMPLOYEE INTEGER,

PRICE VARCHAR(10)

);

ALTER TABLE "ORDER" ADD CONSTRAINT PK\_ORDER PRIMARY KEY (ID);

CREATE TABLE CLIENT (

ID INTEGER NOT NULL,

FIRST\_NAME CHAR(10),

SECOND\_NAME CHAR(10),

ADDRESS CHAR(30),

PHONE CHAR(10)

);

ALTER TABLE CLIENT ADD CONSTRAINT PK\_CLIENT PRIMARY KEY (ID);

CREATE TABLE EMPLOYEE (

ID INTEGER NOT NULL,

FIRST\_NAME CHAR(10),

SECOND\_NAME CHAR(10),

ADDRESS CHAR(30),

PHONE CHAR(10)

);

ALTER TABLE EMPLOYEE ADD CONSTRAINT PK\_EMPLOYEE PRIMARY KEY (ID);

CREATE TABLE FACTORY (

ID INTEGER NOT NULL,

COUNTRY CHAR(10),

CITY CHAR(10),

ADDRESS CHAR(10),

PHONE CHAR(10)

);

ALTER TABLE FACTORY ADD CONSTRAINT PK\_FACTORY PRIMARY KEY (ID);

CREATE TABLE CAR\_MODELS (

ID INTEGER NOT NULL,

NAME CHAR(10)

);

ALTER TABLE CAR\_MODELS ADD CONSTRAINT PK\_CAR\_MODELS PRIMARY KEY (ID);

CREATE TABLE CAR\_BODY\_TYPE (

ID INTEGER NOT NULL,

NAME CHAR(10)

);

ALTER TABLE CAR\_BODY\_TYPE ADD CONSTRAINT PK\_CAR\_BODY\_TYPE PRIMARY KEY (ID);

CREATE TABLE CAR\_EQUIPMENT (

ID INTEGER NOT NULL,

NAME CHAR(10)

);

ALTER TABLE CAR\_EQUIPMENT ADD CONSTRAINT PK\_CAR\_EQUIPMENT PRIMARY KEY (ID);

CREATE TABLE CONFIGURATION

(

ID\_CONFIGURATION INTEGER NOT NULL,

ID\_MODELS INTEGER,

ID\_EQUIPMENT INTEGER,

ID\_OPTIONS INTEGER

);

ALTER TABLE CONFIGURATION ADD CONSTRAINT PK\_CONFIGURATION PRIMARY KEY (ID\_CONFIGURATION);

CREATE TABLE OPTIONS (

ID INTEGER NOT NULL,

NAME CHAR(10)

);

ALTER TABLE OPTIONS ADD CONSTRAINT PK\_OPTIONS PRIMARY KEY (ID);

Добавим foreign key в созданные таблицы:

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Foreign keys\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE "ORDER" ADD CONSTRAINT FK\_ORDER\_1 FOREIGN KEY (ID\_CLIENT) REFERENCES CLIENT(ID);

ALTER TABLE "ORDER" ADD CONSTRAINT FK\_ORDER\_2 FOREIGN KEY (ID\_EMPLOYEE) REFERENCES EMPLOYEE (ID);

ALTER TABLE "ORDER" ADD CONSTRAINT FK\_ORDER\_3 FOREIGN KEY (ID\_CAR) REFERENCES CAR (ID);

ALTER TABLE "CAR" ADD CONSTRAINT FK\_CAR\_1 FOREIGN KEY (ID\_FACTORY) REFERENCES FACTORY (ID);

ALTER TABLE "CAR" ADD CONSTRAINT FK\_CAR\_2 FOREIGN KEY (ID\_CAR\_MODELS) REFERENCES CAR\_MODELS (ID);

ALTER TABLE "CAR" ADD CONSTRAINT FK\_CAR\_3 FOREIGN KEY (ID\_EQUIPMENT) REFERENCES CAR\_EQUIPMENT (ID);

ALTER TABLE CONFIGURATION ADD CONSTRAINT FK\_CONFIGURATION\_1 FOREIGN KEY (ID\_EQUIPMENT) REFERENCES CAR\_EQUIPMENT(ID);

ALTER TABLE CONFIGURATION ADD CONSTRAINT FK\_CONFIGURATION\_2 FOREIGN KEY (ID\_MODELS) REFERENCES CAR\_MODELS(ID);

ALTER TABLE CONFIGURATION ADD CONSTRAINT FK\_CONFIGURATION\_3 FOREIGN KEY (ID\_OPTIONS) REFERENCES OPTIONS(ID);

Модернизация БД

Дополнительное задание по модернизации исходной таблицы:

* Реализовать множественную связь между опциями и комплектациями;
* Добавить в заказ дополнительные опции;
* Добавить учет типов кузовов для моделей, двигателей и трансмиссий для комплектаций.

Для реализации требуемых задач требуется видоизменить предыдущую схему БД. Добавим новую таблицу ADDITIONAL\_OPTIONS, которая будет связывать таблицу с

заказом и различными опциями. Тем самым реализуем возможность докупать доп. опции. Создадим таблицу CAR\_MODIFICATION, которая будет связывать выбор моделей, типов кузовов, автомобилей, двигателей, трансмиссий и приводов. Через таблицу CAR будет происходить связь CAR\_MODIFICATION с комплектациями, опциями, а также с заказчиком.

Были добавлены следующие таблицы:

* ADDITIONAL\_OPTIONS;
* CAR\_MODIFICATION;
* CAR\_MODELS;
* CAR\_BODY\_TYPE;
* ENGINE\_TYPE;
* GEAR\_BOX;
* DRIVE\_GEAR.

Изменим схему БД с учетом дополнительного задания:

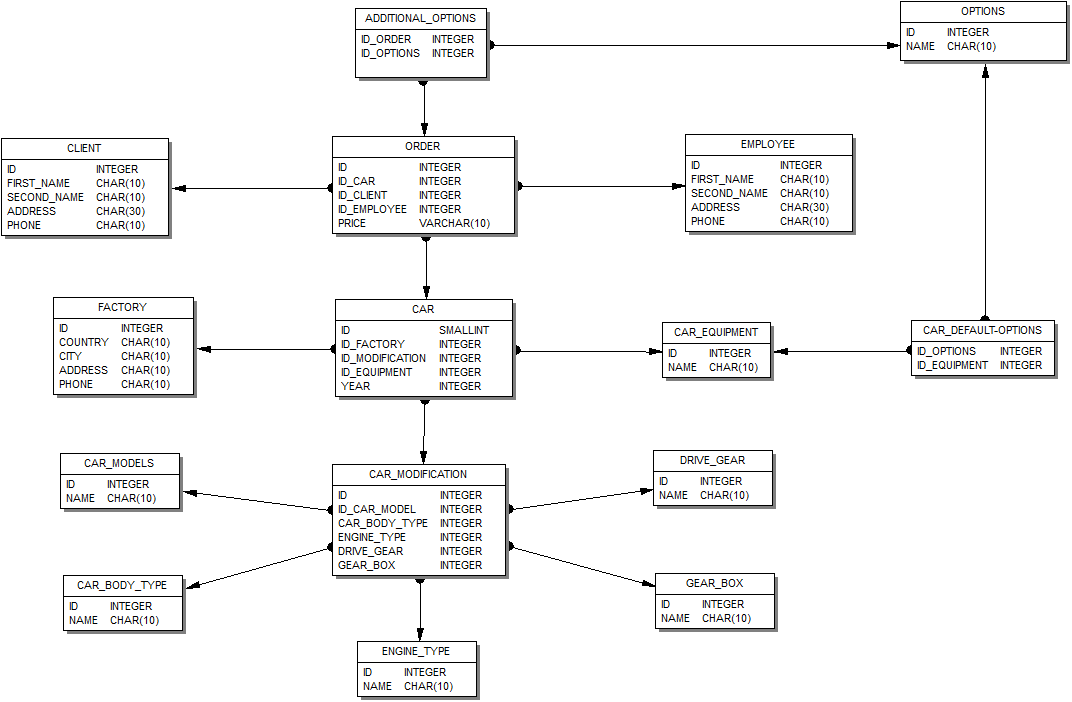


Рис. 3 ER-диаграмма БД

Скрипт для изменения:

/\* Modification table \*/

CREATE TABLE ADDITIONAL\_OPTIONS (

ID\_ORDER INTEGER,

ID\_OPTIONS INTEGER

);

CREATE TABLE "CAR\_DEFAULT-OPTIONS" (

ID\_OPTIONS INTEGER,

ID\_EQUIPMENT INTEGER

);

CREATE TABLE CAR\_MODIFICATION (

ID INTEGER NOT NULL,

ID\_CAR\_MODEL INTEGER,

CAR\_BODY\_TYPE INTEGER,

ENGINE\_TYPE INTEGER,

DRIVE\_GEAR INTEGER,

GEAR\_BOX INTEGER

);

CREATE TABLE DRIVE\_GEAR (

ID INTEGER NOT NULL,

NAME CHAR(10)

);

CREATE TABLE ENGINE\_TYPE (

ID INTEGER NOT NULL,

NAME CHAR(10)

);

CREATE TABLE GEAR\_BOX (

ID INTEGER NOT NULL,

NAME CHAR(10)

);

DROP TABLE CONFIGURATION;

ALTER TABLE CAR DROP ID\_CAR\_MODELS;

ALTER TABLE CAR ALTER ID\_MODELS TO ID\_MODIFICATION;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*PRIMARY KEY\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*ALTER TABLE CAR\_BODY\_TYPE ADD CONSTRAINT PK\_CAR\_BODY\_TYPE PRIMARY KEY (ID);\*/

/\*ALTER TABLE CAR\_MODELS ADD CONSTRAINT PK\_CAR\_MODELS PRIMARY KEY (ID);\*/

ALTER TABLE CAR\_MODIFICATION ADD CONSTRAINT PK\_CAR\_MODIFICATION PRIMARY KEY (ID);

ALTER TABLE DRIVE\_GEAR ADD CONSTRAINT PK\_DRIVE\_GEAR PRIMARY KEY (ID);

ALTER TABLE ENGINE\_TYPE ADD CONSTRAINT PK\_ENGINE\_TYPE PRIMARY KEY (ID);

ALTER TABLE GEAR\_BOX ADD CONSTRAINT PK\_GEAR\_BOX PRIMARY KEY (ID);

/\*ALTER TABLE OPTIONS ADD CONSTRAINT PK\_OPTIONS PRIMARY KEY (ID);\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*FOREIGN KEY\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE ADDITIONAL\_OPTIONS ADD CONSTRAINT FK\_ADDITIONAL\_OPTIONS\_1 FOREIGN KEY (ID\_OPTIONS) REFERENCES OPTIONS (ID);

ALTER TABLE ADDITIONAL\_OPTIONS ADD CONSTRAINT FK\_ADDITIONAL\_OPTIONS\_2 FOREIGN KEY (ID\_ORDER) REFERENCES "ORDER" (ID);

ALTER TABLE CAR ADD CONSTRAINT FK\_CAR\_2 FOREIGN KEY (ID\_MODIFICATION) REFERENCES CAR\_MODIFICATION (ID);

ALTER TABLE "CAR\_DEFAULT-OPTIONS" ADD CONSTRAINT "FK\_CAR\_DEFAULT-OPTIONS\_1" FOREIGN KEY (ID\_OPTIONS) REFERENCES OPTIONS (ID);

ALTER TABLE "CAR\_DEFAULT-OPTIONS" ADD CONSTRAINT "FK\_CAR\_DEFAULT-OPTIONS\_2" FOREIGN KEY (ID\_EQUIPMENT) REFERENCES CAR\_EQUIPMENT (ID);

ALTER TABLE CAR\_MODIFICATION ADD CONSTRAINT FK\_CAR\_MODIFICATION\_1 FOREIGN KEY (ID\_CAR\_MODEL) REFERENCES CAR\_MODELS (ID);

ALTER TABLE CAR\_MODIFICATION ADD CONSTRAINT FK\_CAR\_MODIFICATION\_2 FOREIGN KEY (ENGINE\_TYPE) REFERENCES ENGINE\_TYPE (ID);

ALTER TABLE CAR\_MODIFICATION ADD CONSTRAINT FK\_CAR\_MODIFICATION\_3 FOREIGN KEY (DRIVE\_GEAR) REFERENCES DRIVE\_GEAR (ID);

ALTER TABLE CAR\_MODIFICATION ADD CONSTRAINT FK\_CAR\_MODIFICATION\_4 FOREIGN KEY (GEAR\_BOX) REFERENCES GEAR\_BOX (ID);

ALTER TABLE CAR\_MODIFICATION ADD CONSTRAINT FK\_CAR\_MODIFICATION\_5 FOREIGN KEY (CAR\_BODY\_TYPE) REFERENCES CAR\_BODY\_TYPE (ID);

Автоматическая генерация данных для заполнения БД с помощью IBExpert

Для заполнения воспользуемся встроенной в IBExpert программой, запустим ее: tools/test data generator, и выполним настройку:

* В поле «Table» выбираем название заполняемой таблицы;
* Выбираем нужные колонки для заполнения;
* Выставляем для каждой выбранной колонки параметры.

В каждую таблицу внесем по 100.000 рандомных значений.

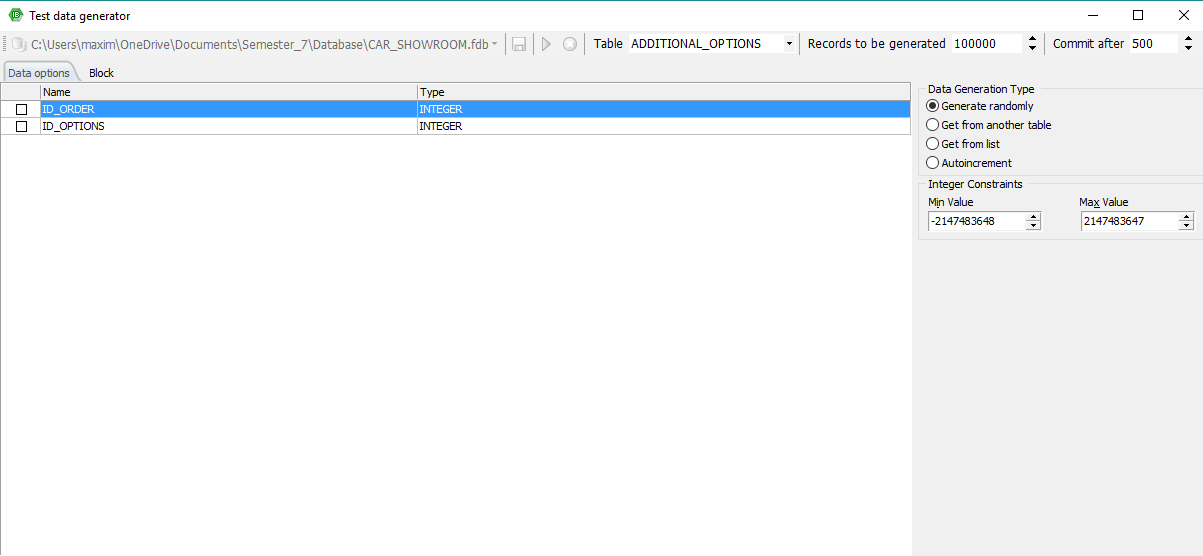


Рис. 4 Рабочее окно IBExpert test data generator



Рис. 5 Пример заполнения таблицы

# Выводы: