**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**Институт компьютерных наук и технологий**

**Кафедра Компьютерных Систем и Программных Технологий**

**Отчет по дисциплине**

**«Базы данных»**

[**Язык SQL-DML**](http://tiger.ftk.spbstu.ru/trac/edu-db-2016/ticket/33)

**Работу выполнил студент группы №:** 43501/3 Гагаркин И.Ю.

**Работу принял преподаватель:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Мяснов А.В.

**Санкт-Петербург**

**2016 г.**

**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)

**Ход работы:**

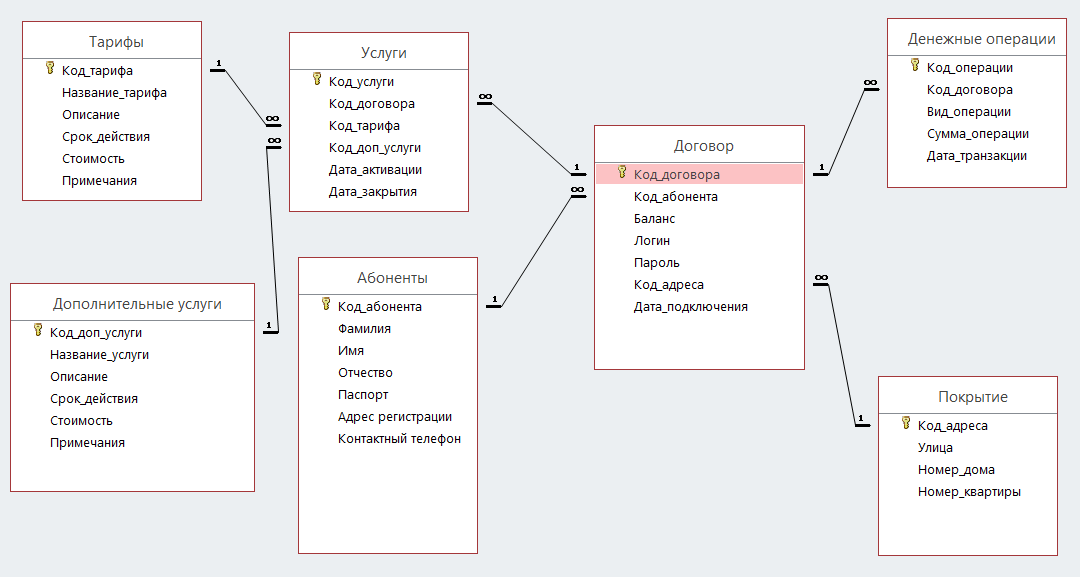


Рисунок 1. SQL-схема БД

На основе данной схемы был написан скрипт, создающий соответствующую БД:

|  |
| --- |
| /\*\*\* Tables \*\*\*/  CREATE TABLE CONTRACTS (  ID\_CONTRACT INTEGER NOT NULL,  ID\_SUBSCRIBER INTEGER NOT NULL,  "COUNT" FLOAT,  LOGIN VARCHAR(12) NOT NULL,  "PASSWORD" VARCHAR(12) NOT NULL,  ID\_ADRESS INTEGER NOT NULL,  DATE\_CONNECTION DATE NOT NULL  );  CREATE TABLE FINANCE (  ID\_OPERATION INTEGER NOT NULL,  ID\_CONTRACT INTEGER NOT NULL,  TYPE\_OPERATION BOOLEAN NOT NULL,  "VALUE" VARCHAR(10) NOT NULL,  DATE\_OPERATION DATE NOT NULL  );  CREATE TABLE MAP (  ID\_ADRESS INTEGER NOT NULL,  STREET VARCHAR(100) NOT NULL,  HOUSE INTEGER NOT NULL,  FLAT INTEGER NOT NULL,  STATUS BOOLEAN NOT NULL  );  CREATE TABLE OPTIONS (  ID\_OPTIONAL INTEGER NOT NULL,  NAME\_OPT VARCHAR(15) NOT NULL,  DESCRIPTION\_OPT VARCHAR(400) NOT NULL,  TERM\_OPT VARCHAR(10) NOT NULL,  COST DOUBLE PRECISION,  NOTES\_OPT VARCHAR(400)  );  CREATE TABLE SERVICES (  ID\_SERVICE INTEGER NOT NULL,  ID\_CONTRACT INTEGER NOT NULL,  ID\_TARIFF INTEGER NOT NULL,  ID\_OPTIONAL INTEGER,  DATE\_OPEN DATE NOT NULL,  DATE\_CLOSE DATE  );  CREATE TABLE SUBSCRIBERS (  ID\_SUBSCRIBER INTEGER NOT NULL,  LAST\_NAME VARCHAR(30) NOT NULL,  FIRST\_NAME VARCHAR(30) NOT NULL,  PATRONYMIC VARCHAR(30),  PASSPORT VARCHAR(15) NOT NULL,  REGISTRATION VARCHAR(500) NOT NULL,  PHONE\_NUMBER VARCHAR(24) CHARACTER SET NONE NOT NULL  );  CREATE TABLE TARIFFS (  ID\_TARIFF INTEGER NOT NULL,  NAME\_TARIFF VARCHAR(12) NOT NULL,  DESCRIPTION\_TARIFF VARCHAR(400) NOT NULL,  TERM\_TARIFF INTEGER NOT NULL,  COST\_TARIFF DOUBLE PRECISION NOT NULL,  NOTE\_TARIFF VARCHAR(400)  );  /\*\*\* Primary keys \*\*\*/  ALTER TABLE CONTRACTS ADD CONSTRAINT PK\_CONTRACTS PRIMARY KEY (ID\_CONTRACT);  ALTER TABLE MAP ADD CONSTRAINT PK\_MAP PRIMARY KEY (ID\_ADRESS);  ALTER TABLE OPTIONS ADD CONSTRAINT PK\_OPTIONS PRIMARY KEY (ID\_OPTIONAL);  ALTER TABLE SERVICES ADD CONSTRAINT PK\_SERVICES PRIMARY KEY (ID\_SERVICE);  ALTER TABLE SUBSCRIBERS ADD CONSTRAINT PK\_SUBSCRIBERS PRIMARY KEY (ID\_SUBSCRIBER);  ALTER TABLE TARIFFS ADD CONSTRAINT PK\_TARIFFS PRIMARY KEY (ID\_TARIFF);  /\*\*\* Foreign keys \*\*\*/  ALTER TABLE CONTRACTS ADD CONSTRAINT FK\_CONTRACTS\_1 FOREIGN KEY (ID\_SUBSCRIBER) REFERENCES SUBSCRIBERS (ID\_SUBSCRIBER);  ALTER TABLE CONTRACTS ADD CONSTRAINT FK\_CONTRACTS\_2 FOREIGN KEY (ID\_ADRESS) REFERENCES MAP (ID\_ADRESS);  ALTER TABLE FINANCE ADD CONSTRAINT FK\_FINANCE\_1 FOREIGN KEY (ID\_CONTRACT) REFERENCES CONTRACTS (ID\_CONTRACT);  ALTER TABLE SERVICES ADD CONSTRAINT FK\_SERVICES\_1 FOREIGN KEY (ID\_CONTRACT) REFERENCES CONTRACTS (ID\_CONTRACT);  ALTER TABLE SERVICES ADD CONSTRAINT FK\_SERVICES\_2 FOREIGN KEY (ID\_OPTIONAL) REFERENCES OPTIONS (ID\_OPTIONAL);  ALTER TABLE SERVICES ADD CONSTRAINT FK\_SERVICES\_3 FOREIGN KEY (ID\_TARIFF) REFERENCES TARIFFS (ID\_TARIFF); |

Листинг 1. Скрипт для создания БД

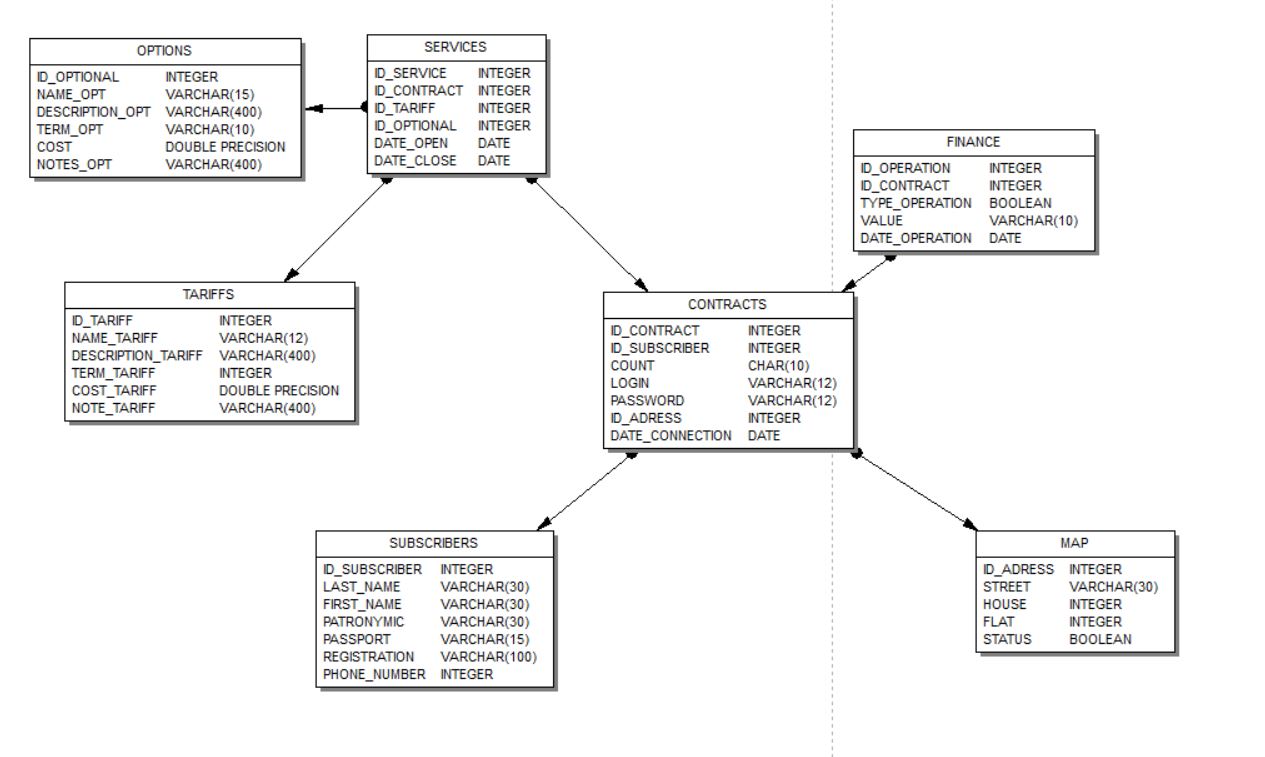


Рисунок 2. ER- диаграмма базы данных.

Далее при помощи инструмента «Генератор тестовых данных» были автоматически сгенерированы данные для таблиц SUBSCRIBERS, CONTRACTS, MAP:

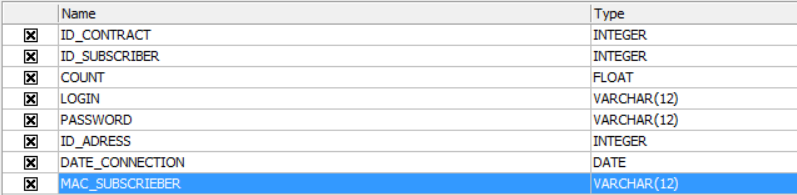


Рисунок 3. Инструмент «Генератор тестовых данных»

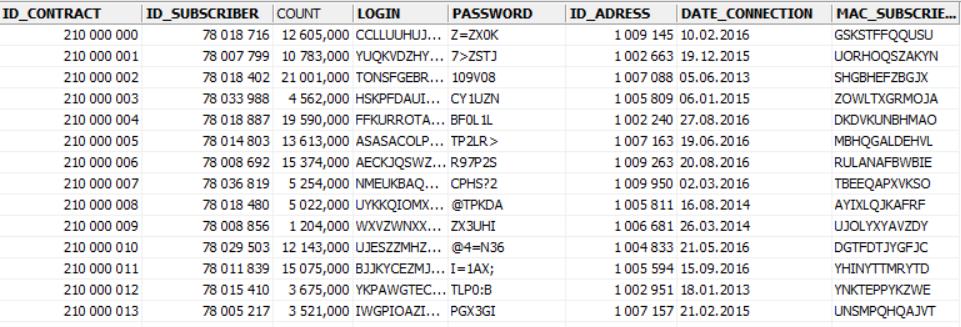


Рисунок 4. Пример данных сгенерированных для таблицы CONTRACTS

**Выполнение индивидуального задания.**

Формулировка задания:

1. Ввести учет оборудования провайдера и MAC-адресов подключения клиентов.
2. Ввести учет акций/скидок.

На рисунке 4 представлена измененная SQL-схема базы данных:

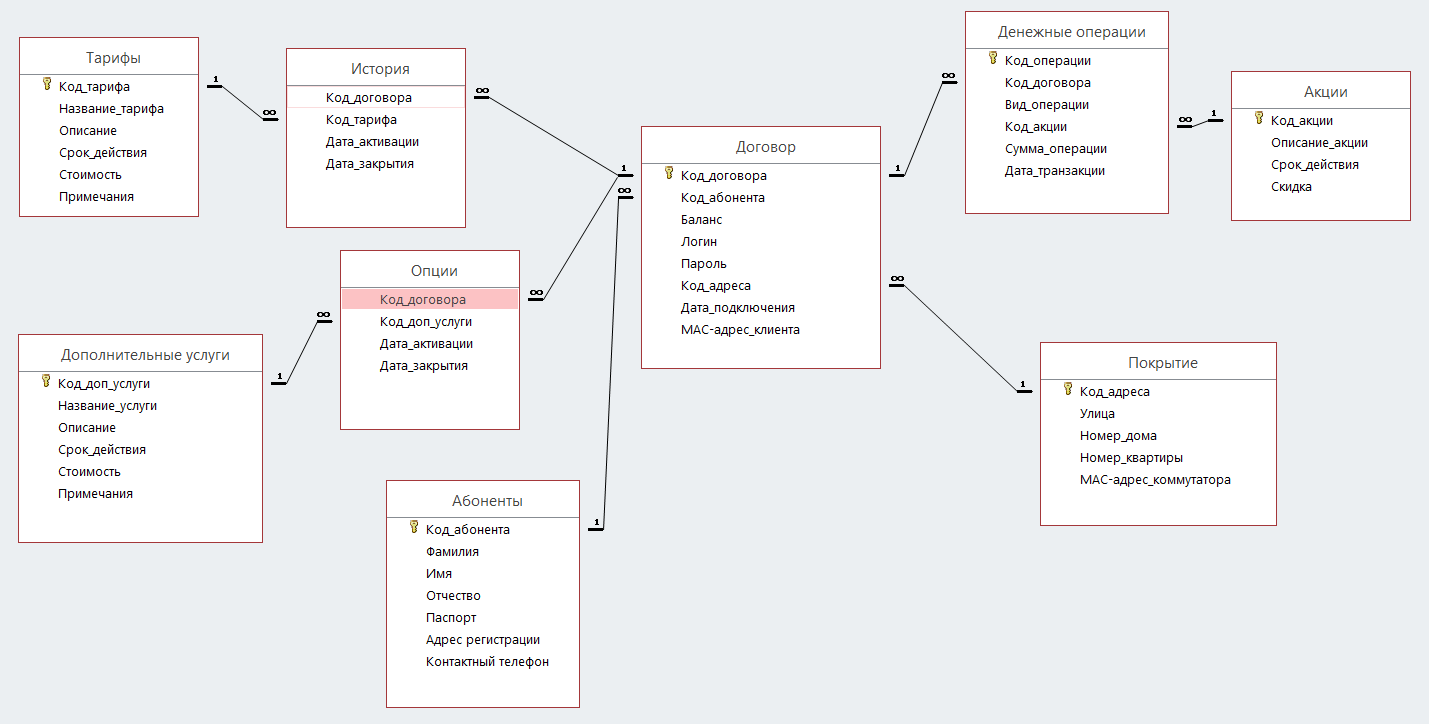


Рисунок 5. Измененная SQL-схема БД

**Список изменений:**

1. Таблица услуги была переименована в историю
2. Из таблицы «История» было удалено поле «код\_доп\_услуги»
3. Была создана таблица «Опции» для корректного учета дополнительных услуг каждого абонента
4. Была создана таблица «Акции» и добавлено поле «код\_акции» в таблицу «Денежные операции»
5. В таблицу «Договор» было добавлено поле «МАС-адрес\_клиента»
6. В таблицу «Покрытие» было добавлено поле «МАС-адрес\_коммутатора»

Скрипт, изменяющий БД в соответствии с новой sql - схемой:

|  |
| --- |
| CREATE TABLE DISCOUNTS (  ID\_DISCOUNT INTEGER NOT NULL,  DESCRIPTION VARCHAR(500) NOT NULL,  "TERM" INTEGER NOT NULL,  "VALUE" FLOAT NOT NULL  );  ALTER TABLE DISCOUNTS ADD CONSTRAINT PK\_DISCOUNTS PRIMARY KEY (ID\_DISCOUNT);  CREATE TABLE HISTORY\_OPT (  ID\_CONTRACT INTEGER NOT NULL,  ID\_OPTION INTEGER NOT NULL,  DATE\_OPEN DATE NOT NULL,  DATE\_CLOSE DATE  );  ALTER TABLE HISTORY\_OPT ADD CONSTRAINT FK\_HISTORY\_OPT\_1 FOREIGN KEY (ID\_CONTRACT) REFERENCES CONTRACTS (ID\_CONTRACT);  ALTER TABLE HISTORY\_OPT ADD CONSTRAINT FK\_HISTORY\_OPT\_2 FOREIGN KEY (ID\_OPTION) REFERENCES OPTIONS (ID\_OPTION);  ALTER TABLE HISTORY\_TARIFF DROP COLUMN ID\_OPTION;  ALTER TABLE CONTRACTS ADD MAC\_SUBSCRIEBER VARCHAR(12) NOT NULL;  ALTER TABLE MAP ADD MAC\_SWITCH VARCHAR(12) NOT NULL;  ALTER TABLE CONTRACTS ADD MAC\_SUBSCRIEBER VARCHAR(12) NOT NULL;  ALTER TABLE FINANCE ADD ID\_DISCOUNT INTEGER NULL; |

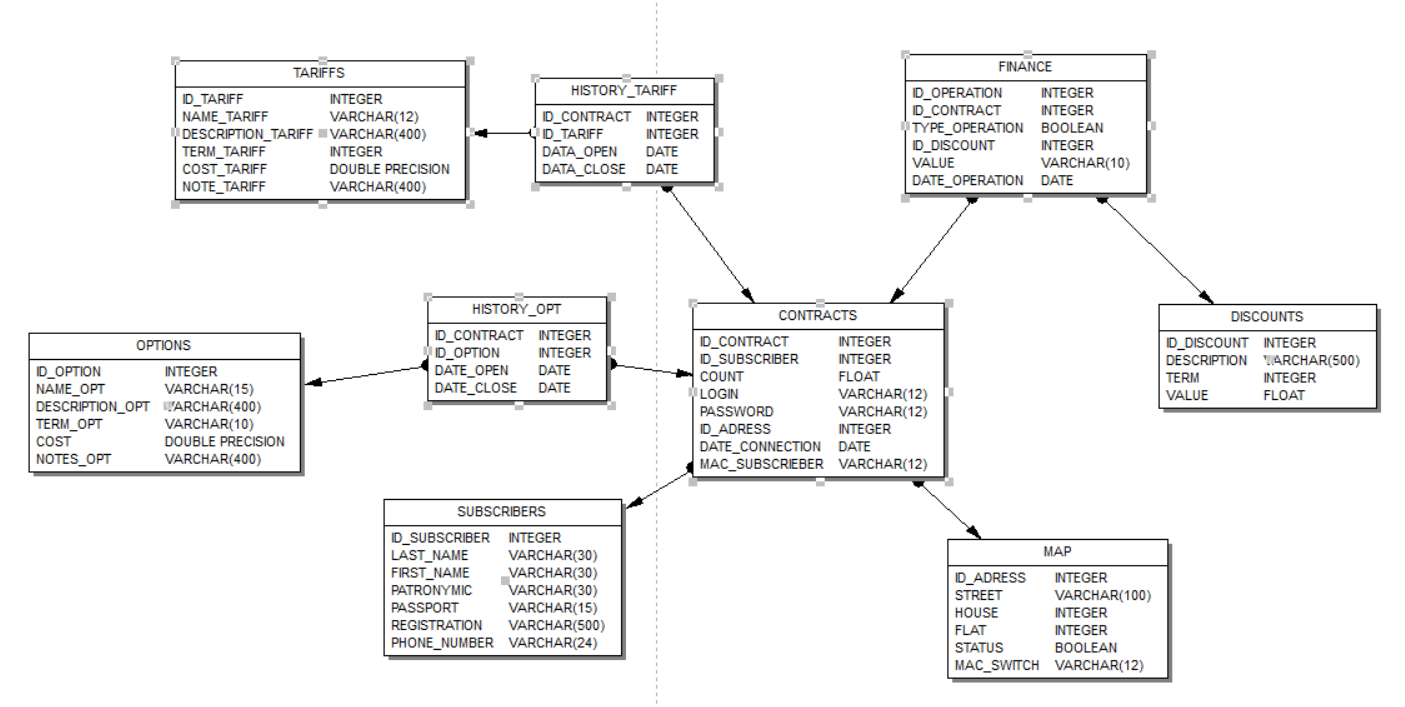


Рисунок 6. ER-диаграмма измененной БД

**Вывод:**

В ходе работы я применил на практике навыки работы с языком SQL-DDL. 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.

Также был получен опыт работы с IBExpert. IBExpert - GUI-оболочка, предназначенная для разработки и администрирования баз данных InterBase и Firebird, а также для выбора и изменения данных, хранящихся в базах. Наиболее полезными инструментами являются: дизайнер БД, позволяющий создавать ER-диаграммы для проверки результатов работы и генератор тестовых данных, позволяющий заполнить базу достаточно большим количеством значений.