

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное
учреждение высшего образования

«Московский технический университет связи и информатики»

Факультет «Информационные технологии»

Кафедра «Искусственный интеллект и машинное обучение»

Лабораторная работа №11
Основы SQL

Автор:

Голиков Михаил Вячеславович, БВТ2402

Цель лабораторной работы

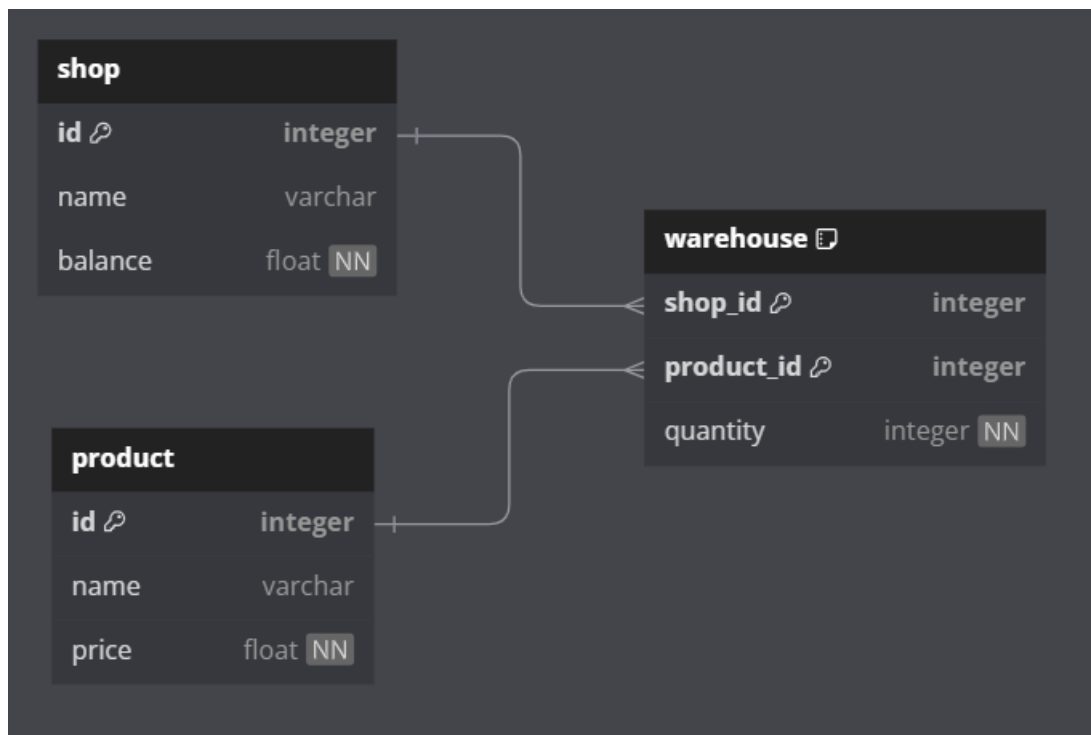
Познакомится с декларативным языком программирования SQL. Изучить основные команды для создания и изменения таблиц, а также научиться писать запросы с фильтрацией.

Ход выполнения лабораторной работы

Задание 1: откроем сайт <https://sunnygoyal.com/jade/> и напишем код для создания базы данных с определённой архитектурой. Воспользуемся декларативным языком программирования SQL в БД (базе данных) типа SQL Lite. Архитектура БД показана на элементе 2.

```
CREATE TABLE "group" (  
  id INTEGER PRIMARY KEY,  
  name VARCHAR(255) UNIQUE,  
  description VARCHAR(1023));  
  
CREATE TABLE student (  
  id INTEGER PRIMARY KEY,  
  name VARCHAR(255),  
  group_id INTEGER REFERENCES "group" (id));  
  
CREATE TABLE student_marks (  
  student_id INTEGER REFERENCES student(id),  
  math_mark_average FLOAT,  
  physics_mark_average FLOAT,  
  python_mark_average FLOAT);
```

Элемент 1 — Код на SQL для создания БД (двойное нажатие для открытия)



Элемент 2 — Архитектура БД

Задание 2: введём новые данные в созданные таблицы. Для этого будем использовать команду “INSERT INTO table VALUES ()”. Сами вводимые данные легко прочитать из SQL кода (элемент 3).

```
INSERT INTO "group" VALUES (1, "PRO", "Best of the best");  
INSERT INTO "group" VALUES (2, "Losers", "NOT Best of the best");
```

```
INSERT INTO student VALUES (1, "Golikov", "1");  
INSERT INTO student VALUES (2, "Volkov", "1");  
INSERT INTO student VALUES (3, "Klopov", "2");  
INSERT INTO student VALUES (4, "Tarakanov", "2");
```

```
INSERT INTO student_marks VALUES (1, 4.9, 4.8, 5.0);  
INSERT INTO student_marks VALUES (2, 4.0, 4.3, 4.2);  
INSERT INTO student_marks VALUES (3, 4.0, 3.1, 3.3);  
INSERT INTO student_marks VALUES (4, 2.0, 2.8, 2.0);
```

Элемент 3 — Код на SQL для добавления данных в БД

Задание 3: создадим пару фильтрующих запросов из БД. Для этого воспользуемся командой “SELECT _ FROM table”, которая позволяет показывать результаты поиска по конкретным таблицам.

```
SELECT * FROM student
```

```
INNER JOIN student_marks ON (student.id = student_marks.student_id)
```

```
WHERE math_mark_average >= 4.0
```

```
ORDER BY group_id DESC, name ASC;
```

```
SELECT "group".name, (math_mark_average + physics_mark_average + python_mark_average)/3 as  
avg_mark FROM "group"
```

```
JOIN student ON ("group".id = student.group_id)
```

```
INNER JOIN student_marks ON (student.id = student_marks.student_id)
```

```
WHERE avg_mark > 3.5;
```

```
SELECT DISTINCT "group".name FROM "group"
```

```
JOIN student ON ("group".id = student.group_id)
```

```
INNER JOIN student_marks ON (student.id = student_marks.student_id)
```

```
WHERE (math_mark_average + physics_mark_average + python_mark_average)/3 > 3.5;
```

Элемент 4 — Код на SQL для фильтрованных запросов

Результаты запросов показаны ниже:

RUN SQL

Элемент 5 — Результат 1-го запроса

RUN SQL

Элемент 6 — Результат 2-го запроса

```
1 SELECT DISTINCT "group".name FROM "group"
2 JOIN student ON ("group".id = student.group_id)
3 INNER JOIN student_marks ON (student.id = student_marks.student_id)
4 WHERE (math_mark_average + physics_mark_average +
        other_mark_average) / 3 > 3.5
```

RUN SQL

TEMPLATES

name
PRO

Элемент 7 — Результат 3-го запроса

Заключение

Был изучен декларативный язык программирования SQL на основе SQL Lite. Все задачи выполнены, результаты соответствуют ожидаемым.