

**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 №12
ОСНОВЫ ВЫБОРКИ 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

Первым делом создадим все необходимые таблицы. Для этого выполним следующий код:

```
CREATE TABLE shop (  
  id INTEGER PRIMARY KEY,  
  name VARCHAR(255) UNIQUE,  
  balance FLOAT NOT NULL);  
  
CREATE TABLE product (  
  id INTEGER PRIMARY KEY,  
  name VARCHAR(255) UNIQUE,  
  price FLOAT NOT NULL);  
  
CREATE TABLE warehouse (  
  shop_id INTEGER REFERENCES shop (id),  
  product_id INTEGER REFERENCES product (id),  
  quantity INTEGER NOT NULL,  
  PRIMARY KEY (shop_id, product_id));  
  
CREATE TABLE worker (  
  worker_id INTEGER PRIMARY KEY,  
  shop_id INTEGER REFERENCES product (id),  
  name VARCHAR(255),  
  salary INTEGER NOT NULL,  
  position VARCHAR(255));
```

Элемент 1 — код для создания таблиц

```
INSERT INTO shop  
VALUES (1, "Пятёрочка", 1000);  
INSERT INTO shop  
VALUES (2, "Магнит", 100);  
INSERT INTO shop  
VALUES (3, "Перекрёсток", 10000);  
  
INSERT INTO product  
VALUES (1, "Хлеб", 100);  
INSERT INTO product  
VALUES (2, "Масло", 1000);
```

```

INSERT INTO product
VALUES (3, "Соль", 10);

INSERT INTO warehouse
VALUES (1, 1, 10);
INSERT INTO warehouse
VALUES (1, 3, 100);
INSERT INTO warehouse
VALUES (2, 2, 1000);
INSERT INTO warehouse
VALUES (3, 1, 100);
INSERT INTO warehouse
VALUES (3, 2, 100);
INSERT INTO warehouse
VALUES (3, 3, 100);

INSERT INTO worker
VALUES (1, 1, "Костя", 1000, "Директор");
INSERT INTO worker
VALUES (2, 1, "Маша", 100, "Кассир");

INSERT INTO worker
VALUES (3, 2, "Паша", 100, "Директор");
INSERT INTO worker
VALUES (4, 2, "Галя", 10, "Кассир");

INSERT INTO worker
VALUES (5, 3, "Миша", 10000, "Директор");
INSERT INTO worker
VALUES (6, 3, "Гоша", 2000, "Кассир");
INSERT INTO worker
VALUES (7, 3, "Толик", 1000, "Кассир");

```

Элемент 2 — код для вставки значений

worker_id	shop_id	name	salary	position
1	1	Костя	1000	Директор
2	1	Маша	100	Кассир
3	2	Паша	100	Директор
4	2	Галя	10	Кассир
5	3	Миша	10000	Директор
6	3	Гоша	2000	Кассир
7	3	Толик	1000	Кассир

Элемент 3 — итоговая таблица worker

Теперь напишем запросы с использованием ORDER BY и GROUP BY.

```
SELECT * FROM worker
ORDER BY salary DESC;
```

Элемент 4 — код с использованием ORDER BY

```
1 SELECT * FROM worker
2 ORDER BY salary DESC;
3
4
```

RUN SQL

worker_id	shop_id	name	salary	position
5	3	Миша	10000	Директор
6	3	Гоша	2000	Кассир
1	1	Костя	1000	Директор
7	3	Толик	1000	Кассир
2	1	Маша	100	Кассир
3	2	Паша	100	Директор
4	2	Галя	10	Кассир

Элемент 5 — результат запроса

```
SELECT AVG (salary), shop.name FROM worker
JOIN shop ON shop.id = worker.shop_id
GROUP BY shop id;
```

Элемент 6 — код с использованием GROUP BY

1	SELECT AVG (salary), shop.name FROM worker
2	JOIN shop ON shop.id = worker.shop_id
3	GROUP BY shop_id;
4	
RUN SQL	
AVG (salary)	name
550	Пятёрочка
55	Магнит
4333.333333333333	Перекрёсток

Элемент 7 — результат запроса

Заключение

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