# **SQL интервью**

## ЗАДАНИЕ 1

## ВОПРОСЫ

1. Что такое реляционная база данных и для чего она используется?
2. Что такое первичный КЛЮЧ?
3. Что такое DML и DDL и в чем между ними разница?
4. Что делает условие DISTINCT?
5. Напишите пример кода для следующих команд: SELECT, INSERT, UPDATE, DELETE, CREATE
6. DATABASE, CREATE TABLE.
7. В чем разница между LEFT JOIN, INNER JOIN, RIGHT JOIN, OUTER JOIN?
8. Напишите пример кода с использованием LEFT JOIN.
9. В чем разница между Запросом и подзапросом?
10. Напишите пример кода с использованием подзапроса.

## ОТВЕТЫ

1. **Реляционная база данных** — система организации и управления данными, хранящая данные в виде набора связанных между собою таблиц. Используется (в приложениях или сайтах) для хранения, извлечения, обновления и удаления данных.
2. **Первичный ключ** — атрибут или группа атрибутов (уникальные индексы, не имеющие дубликатов), которые однозначно идентифицируют запись в таблице.
3. **DML vs DDL:**
   1. **DML** (*Data Manipulation Language*) - язык манипуляции данными (управление базами данных путём выбора, удаления и добавления данных командами *SELECT, INSERT, UPDATE и DELETE*).
   2. **DDL** (*Data Definition Language*) - язык определения и изменения структур баз данных операторами *CREATE, ALTER, DROP*.

Таким образом, **разница между *DML* и *DDL***в том, что DML используется для операций с данными, а DDL - для определения структур данных.

1. Условие **DISTINCT** используется в команде SELECT для исключения дубликатов записей из результата запроса.
2. **Примеры кода для команд**:
   1. **SELECT**:

SELECT \* FROM table\_name;

* 1. **INSERT**:

INSERT INTO table\_name (column1, column2) VALUES (value1, value2);

* 1. **UPDATE**:

UPDATE table\_name SET column1 = value1 WHERE condition;

* 1. **DELETE**:

DELETE FROM table\_name WHERE condition;

1. ***CREATE DATABASE, CREATE TABLE*:**
   1. **CREATE DATABASE**:

CREATE DATABASE database\_name;

* 1. **CREATE TABLE**:

CREATE TABLE table\_name(column1 datatype, column2 datatype, ...);

1. Разница между **JOIN**-ами заключается в том, как они соединяют таблицы.
   1. **LEFT JOIN** возвращает все записи из левой таблицы и соответствующие записи из правой таблицы, даже если в правой таблице нет связанной записи.
   2. **INNER JOIN** возвращает только те записи, которые имеют связанные записи в обеих таблицах.
   3. **RIGHT JOIN** работает аналогично *LEFT JOIN*, но меняет левую и правую таблицы местами.
   4. **OUTER JOIN** объединяет свойства *LEFT JOIN* и *INNER JOIN*.
2. Пример кода с использованием **LEFT JOIN**:

SELECT t1.\*, t2.\* FROM table1 AS t1 LEFT JOIN table2 AS t2 ON t1.id = t2.id;

1. **Разница между Запросом и Подзапросом**:

**Запрос** — операция, возвращающая набор данных на основе заданных условий.

**Подзапрос** — запрос, который выполняется внутри другого запроса, и его результат является частью основного запроса.

1. **Пример кода с подзапросом**:

SELECT \* FROM t1 WHERE column\_name IN (SELECT column\_name FROM t2);