 **Структура команды SELECT**

sql

SELECT [столбцы] FROM [таблица]

[WHERE условие]

[GROUP BY столбцы]

[HAVING условие]

[ORDER BY столбцы];

Основные части: выбор столбцов, указание таблицы, фильтрация, группировка, сортировка.

 **Приоритеты логических операций в WHERE**

1. Скобки ()
2. NOT
3. AND
4. OR Пример: WHERE (a > 5 AND b = 10) OR c < 3 — сначала выполняются условия в скобках, затем AND, потом OR.

 **Правила использования выражений в SELECT**

* В SELECT можно использовать столбцы, константы, арифметические операции, функции (например, SUM, UPPER).
* Выражения можно комбинировать: SELECT column1 + column2 AS result.
* Псевдонимы (AS) задают имя результата.
* Выражения не должны нарушать тип данных.

 **Устранение дубликатов в SELECT** Используется ключевое слово DISTINCT:

sql

SELECT DISTINCT column\_name FROM table\_name;

Убирает повторяющиеся строки из результата.

 **Формирование списка вывода команды SELECT**

* Указываются столбцы или выражения после SELECT.
* Можно использовать \* для выбора всех столбцов.
* Псевдонимы (AS) задают имена для вывода: SELECT column AS new\_name.
* Результат формируется в виде таблицы.

 **Можно ли менять порядок конструкций в SELECT?** Нет, порядок фиксирован: SELECT → FROM → WHERE → GROUP BY → HAVING → ORDER BY. Нарушение порядка приведет к ошибке.

 **Конструкция для сортировки результата запроса** Используется ORDER BY:

sql

SELECT column FROM table ORDER BY column;

 **Ключевое слово для сортировки по убыванию** DESC:

sql

SELECT column FROM table ORDER BY column DESC;

(Для возрастания — ASC, используется по умолчанию).

 **Типы условий поиска (предикаты) и ключевые слова**

* Сравнение: =, <>, >, <, >=, <=.
* IN — проверка на вхождение в список.
* BETWEEN — проверка диапазона.
* LIKE — поиск по шаблону.
* IS NULL / IS NOT NULL — проверка на NULL.
* Логические: AND, OR, NOT.

 **Назначение и правила предиката IN**

* Проверяет, входит ли значение в список.
* Синтаксис: WHERE column IN (value1, value2, ...).
* Пример: SELECT \* FROM table WHERE id IN (1, 2, 3);.
* Можно использовать подзапрос: WHERE column IN (SELECT ...).

 **Назначение и правила предиката BETWEEN**

* Проверяет, находится ли значение в диапазоне (включительно).
* Синтаксис: WHERE column BETWEEN value1 AND value2.
* Пример: SELECT \* FROM table WHERE price BETWEEN 10 AND 20;.
* Работает с числами, датами, строками.

 **Назначение и правила предиката LIKE**

* Проверяет соответствие строки шаблону.
* Синтаксис: WHERE column LIKE 'шаблон'.
* Шаблоны:
  + % — любое количество символов.
  + \_ — один символ.
* Пример: SELECT \* FROM table WHERE name LIKE 'A%'; (имена, начинающиеся на A).

 **Соединение таблиц**

* Используется JOIN для объединения данных из нескольких таблиц.
* Синтаксис:

sql

SELECT columns FROM table1 JOIN table2 ON table1.column = table2.column;

* Типы:
  + INNER JOIN — только совпадающие строки.
  + LEFT JOIN — все строки из левой таблицы.
  + RIGHT JOIN — все строки из правой таблицы.
  + FULL JOIN — все строки из обеих таблиц.

 **Виды внешних соединений и примеры**

* LEFT OUTER JOIN — все строки из левой таблицы, NULL для непарных из правой:

sql

SELECT t1.name, t2.order\_id FROM customers t1 LEFT JOIN orders t2 ON t1.id = t2.customer\_id;

* RIGHT OUTER JOIN — все строки из правой таблицы, NULL для непарных из левой:

sql

SELECT t1.name, t2.order\_id FROM customers t1 RIGHT JOIN orders t2 ON t1.id = t2.customer\_id;

* FULL OUTER JOIN — все строки из обеих таблиц, NULL для непарных:

sql

SELECT t1.name, t2.order\_id FROM customers t1 FULL JOIN orders t2 ON t1.id = t2.customer\_id;

 **Назначение UNION, INTERSECT, EXCEPT**

* UNION — объединяет результаты двух запросов, убирая дубликаты:

sql

SELECT column FROM table1 UNION SELECT column FROM table2;

* INTERSECT — возвращает только общие строки двух запросов:

sql

SELECT column FROM table1 INTERSECT SELECT column FROM table2;

* EXCEPT — возвращает строки из первого запроса, которых нет во втором:

sql

SELECT column FROM table1 EXCEPT SELECT column FROM table2;

* Условие: запросы должны иметь одинаковое количество столбцов и совместимые типы данных.