**2. Язык SQL.**

SQL (Structured Query Language) — это язык для управления данными в системах управления базами данных, таких как MS SQL Server от Microsoft. Основные понятия и операторы SQL:

**1. SELECT: Используется для выборки данных из базы. Пример:**

SELECT имя\_колонки1, имя\_колонки2  
FROM имя\_таблицы;

**2. WHERE: Условие для фильтрации записей. Пример:**

SELECT \*  
FROM имя\_таблицы  
WHERE условие;

**3. INSERT INTO: Добавление новых записей в таблицу. Пример:**

INSERT INTO имя\_таблицы (имя\_колонки1, имя\_колонки2)  
VALUES (значение1, значение2);

**4. UPDATE: Обновление существующих записей. Пример:**

UPDATE имя\_таблицы  
SET имя\_колонки1 = значение1, имя\_колонки2 = значение2  
WHERE условие;

**5. DELETE: Удаление записей. Пример:**

DELETE FROM имя\_таблицы  
WHERE условие;

**6. JOIN: Оператор для соединения строк из двух или более таблиц. Примеры:**

- INNER JOIN: выбирает записи с совпадающими значениями в обеих таблицах.  
- LEFT (OUTER) JOIN: выбирает все записи из левой таблицы и совпадающие из правой.  
- RIGHT (OUTER) JOIN: аналогично LEFT JOIN, но с правой таблицей.  
- FULL (OUTER) JOIN: выбирает все записи, когда есть совпадающее условие в обеих таблицах.

**7. GROUP BY: Группировка результатов по одной или нескольким колонкам. Пример:**

SELECT имя\_колонки, COUNT(\*)  
FROM имя\_таблицы  
GROUP BY имя\_колонки;

**8. HAVING: Условие фильтрации, используется совместно с GROUP BY. Пример:**

SELECT имя\_колонки, COUNT(\*)  
FROM имя\_таблицы  
GROUP BY имя\_колонки  
HAVING COUNT(\*) > 10;

**9. ORDER BY: Сортировка результатов по возрастанию или убыванию. Пример:**

SELECT \*  
FROM имя\_таблицы  
ORDER BY имя\_колонки ASC; -- ASC для возрастания, DESC для убывания

**10. Транзакции: Управление наборами операций, которые либо полностью выполняются, либо не выполняются вовсе. Пример:**

BEGIN TRANSACTION;  
INSERT INTO имя\_таблицы (имя\_колонки1) VALUES (значение1);  
UPDATE имя\_таблицы SET имя\_колонки2 = значение2 WHERE условие;  
COMMIT; -- или ROLLBACK для отмены

**17. Внешние соединения: LEFT(RIGHT) OUTER JOIN, FULL OUTER JOIN.**

В MS SQL, внешние соединения таблиц являются частью SQL-языка, которые позволяют объединять строки из двух и более таблиц, основываясь на связанных колонках, но также включают те строки, которые не имеют соответствия в другой таблице. Основные типы внешних соединений:

**LEFT OUTER JOIN (или просто LEFT JOIN)** возвращает все строки из левой таблицы и совпадающие строки из правой. Если сопоставление отсутствует, результат будет содержать `NULL` на месте колонок правой таблицы.

SELECT \*  
FROM леваяТаблица  
LEFT OUTER JOIN праваяТаблица  
ON леваяТаблица.ключевоеПоле = праваяТаблица.ключевоеПоле;

**RIGHT OUTER JOIN (или просто RIGHT JOIN)** работает аналогично LEFT JOIN, но возвращает все строки из правой таблицы и совпадающие строки из левой. Аналогично, при отсутствии сопоставления будет вставлен `NULL` для колонок левой таблицы.

SELECT \*  
FROM леваяТаблица  
RIGHT OUTER JOIN праваяТаблица  
ON леваяТаблица.ключевоеПоле = праваяТаблица.ключевоеПоле;

**FULL OUTER JOIN** сочетает результаты LEFT OUTER JOIN и RIGHT OUTER JOIN. В итоге в результат попадают все строки из обеих таблиц. Если сопоставление отсутствует с одной стороны, соответствующие поля другой таблицы будут содержать `NULL`.

SELECT \*  
FROM леваяТаблица  
FULL OUTER JOIN праваяТаблица  
ON леваяТаблица.ключевоеПоле = праваяТаблица.ключевоеПоле;

Важно помнить следующее при использовании внешних соединений:  
- ON определяет условие соединения, которое должно быть истинно для объединения записей.  
- Если строки не сочетаются, то в зависимости от типа JOIN заполнение произойдет `NULL` значениями.  
- Внешние соединения часто используются для обработки данных, когда необходимо сохранить информацию о записях, даже если они не имеют связей в соединяемых таблицах.  
- Старайтесь использовать внешние соединения аккуратно, так как они могут быть более ресурсоёмкими, чем внутренние (`INNER JOIN`), из-за необходимости обрабатывать строки без сопоставлений.

Выбор типа JOIN зависит от конкретных требований к данным, которые вы извлекаете.