Логический порядок обработки инструкции SELECT:

1. FROM
2. ON
3. JOIN
4. where
5. GROUP BY
6. WITH CUBE или WITH ROLLUP
7. HAVING
8. SELECT
9. DISTINCT
10. ORDER BY
11. TOP

Эквисоединение- на основе равенства.

natural join - эквисоединение, при котором столбцы, которые должны сопоставляться для выполнения соединения, специально не указываются. (столбцы выбираются автоматически)

self join (рефлексивное соединение) - подразумевается соединение таблицы с самой собой за счет использования псевдонимов. Может быть INNER, LEFT, RIGHT

**Оконные функции**

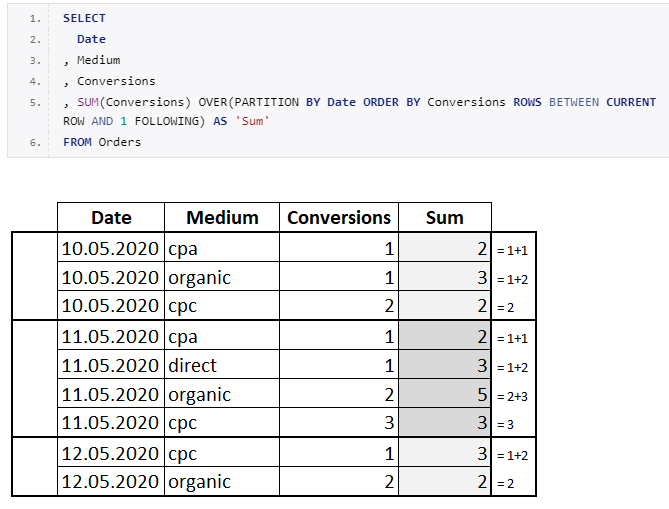
1. **SELECT**
2. Название функции (столбец для вычислений)
3. OVER (
4. PARTITION **BY** столбец для группировки
5. **ORDER** **BY** столбец для сортировки
6. **ROWS** или RANGE выражение для ограничения строк в пределах группы
7. )

Сначала делит на группы с помощью PARTITION **BY** затем внутри группы сортирует с помощью **ORDER** **BY**

Обе инструкции ROWS и RANGE всегда используются вместе с ORDER BY.

* **UNBOUNDED PRECEDING** — указывает, что окно начинается с первой строки группы;
* **UNBOUNDED FOLLOWING** – с помощью данной инструкции можно указать, что окно заканчивается на последней строке группы;
* **CURRENT ROW** – инструкция указывает, что окно начинается или заканчивается на текущей строке;
* **BETWEEN** **«*граница окна*» AND «*граница окна*»** — указывает нижнюю и верхнюю границу окна;
* **«*Значение*»** **PRECEDING** – определяет число строк перед текущей строкой (не допускается в предложении RANGE).;
* **«*Значение*»** **FOLLOWING** — определяет число строк после текущей строки (не допускается в предложении RANGE).

В итоге данные конструкции образуют окно по которому будут применены агрегир функции.



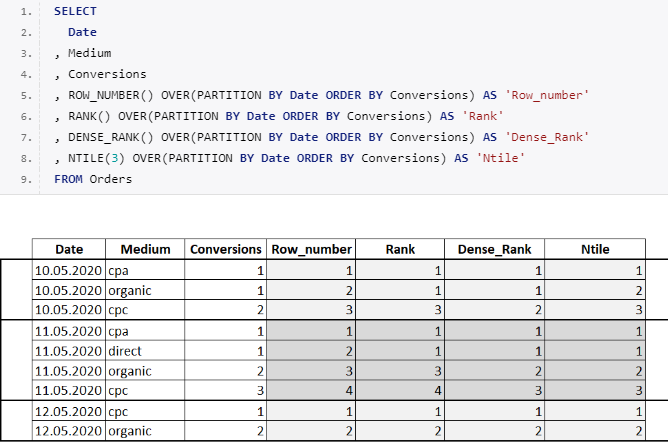
**Ранжирующие функции**

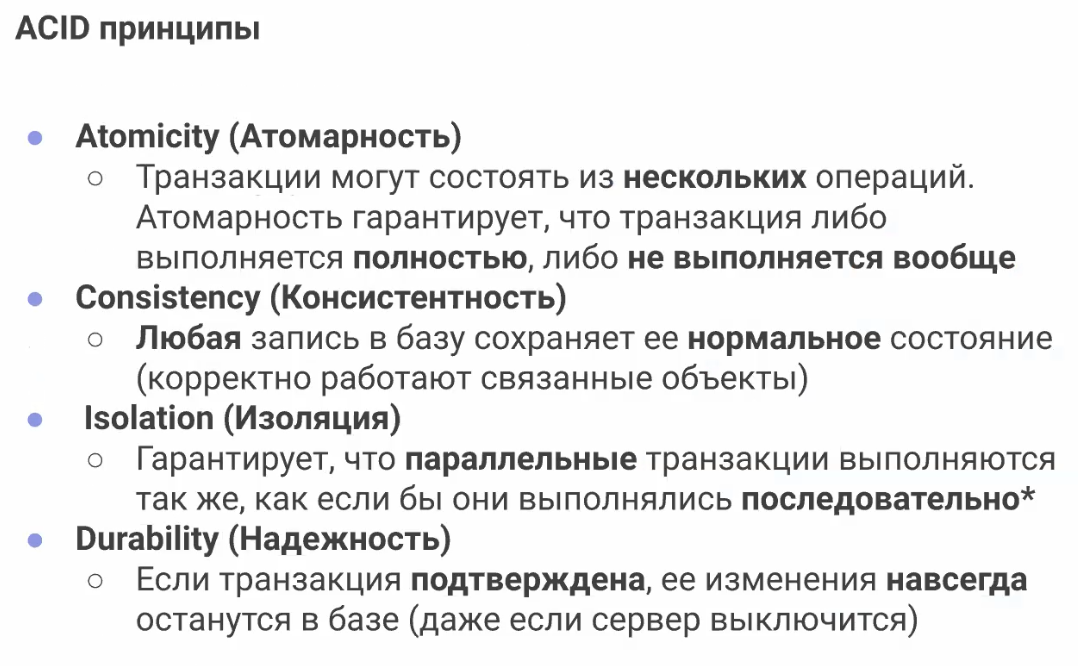
– это функции, которые ранжируют значение для каждой строки в окне

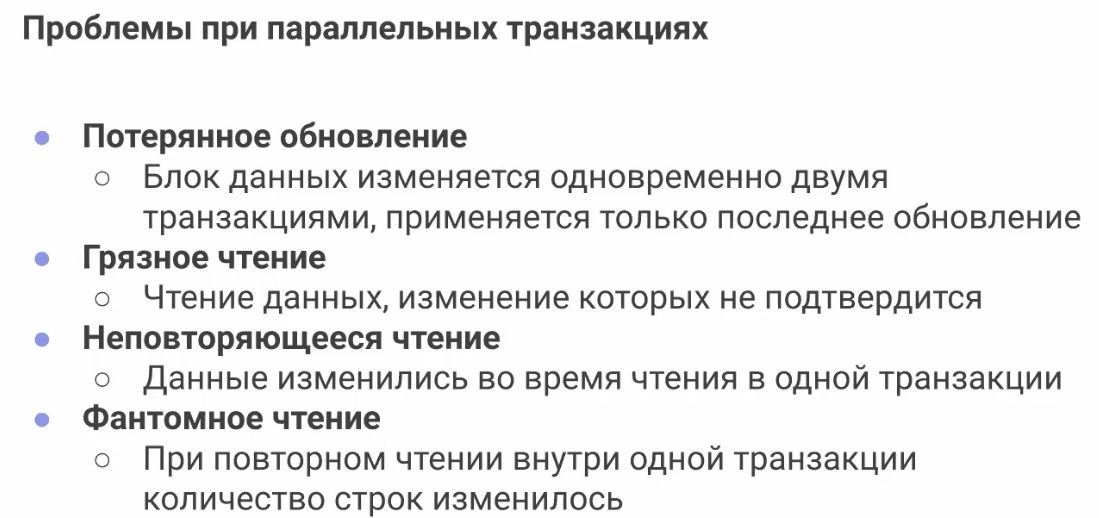
* **ROW\_NUMBER** – функция возвращает номер строки и используется для нумерации;
* **RANK** — функция возвращает ранг каждой строки. В данном случае значения уже анализируются и, в случае нахождения одинаковых, возвращает одинаковый ранг с пропуском следующего значения;
* **DENSE\_RANK** — функция возвращает ранг каждой строки. Но в отличие от функции RANK, она для одинаковых значений возвращает ранг, не пропуская следующий;
* **NTILE** – это функция, которая позволяет определить к какой группе относится текущая строка. Количество групп задается в скобках.

**LAG или LEAD**– функция LAG обращается к данным из предыдущей строки окна, а LEAD к данным из следующей строки. Функцию можно использовать для того, чтобы сравнивать текущее значение строки с предыдущим или следующим. Имеет три параметра: столбец, значение которого необходимо вернуть, количество строк для смещения (*по умолчанию 1*), значение, которое необходимо вернуть если после смещения возвращается значение NULL;

**FIRST\_VALUE** или **LAST\_VALUE**— с помощью функции можно получить первое и последнее значение в окне. В качестве параметра принимает столбец, значение которого необходимо вернуть.

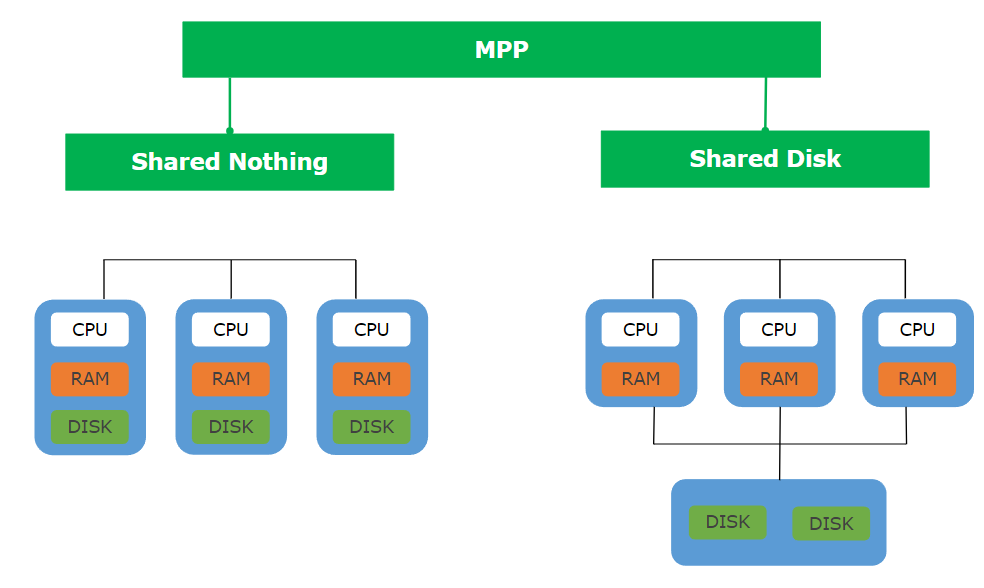






Для решения этих проблем, используются уровни изоляции коннектов:

1. Read uncommitted
2. Read committed
3. Repeatable read
4. Serializable



**Общие замечания**

1. При Left join в условии если стоит одна таблица, то это не идентично с where. Будет влиять на учет второй таблицы

2. Алиасы таблиц внутри from недоступны извне

**Transact-SQL (T-SQL)**

Процедурное расширение для ms-SQL server

@-переменные

Атрибут **IDENTITY(seed, increment)** позволяет сделать столбец идентификатором(в него нельзя записывать)

Select INTO – создает таблицу по запросу