



Выборка данных

Агрегатные функции

Агрегатные функции

COUNT — возвращает кол-во значений в таблице

SUM — возвращает сумму значений

AVG — возвращает среднее значение

MIN — возвращает минимальное значение

MAX — возвращает максимальное значение

Запрос с агрегацией

SELECT столбцы

FROM таблица

[**WHERE** условие_фильтрации_строк]

[**GROUP BY** столбцы_для_группировки]

[**HAVING** условие_фильтрации_групп]

[**ORDER BY** столбцы_для_сортировки]

Запрос с группировкой

```
SELECT tp_sputnik, summ(weight) as w  
      FROM sputnik  
GROUP BY tp_sputnik;
```

Условия **HAVING**

```
SELECT s.id_country, sum(weight),  
count(weight), c."name"  
  
FROM sputnik s, country c  
  
WHERE s.id_country = c.id  
group by id_country, c."name"  
  
HAVING sum(weight) > 15  
        OR count(weight) > 1
```

Применение других функций

- Строковые функции
- Математические функции
- Функции работы со временем
- Преобразование типов

Строковые функции

UPPER() - Преобразует в верхний регистр

LOWER() - Возвращает аргумент в нижнем регистре

CONCAT() - Возвращает объединенную строку

TRIM() - Удаляет начальные и конечные пробелы

REPLACE() - Заменяет вхождения указанной строки

SUBSTRING(), SUBSTR() - Возвращает указанную подстроку

<https://www.postgresql.org/docs/current/functions-string.html>

Математические функции

ABS() - возвращает абсолютное значение числового выражения.

POW() - степень числа

ROUND() - округление до целого числа

SQRT() - корень числа

RANDOM() - генерация случайного значения

DIV() - целочисленное деление

LOG() - логарифм числа

<https://www.postgresql.org/docs/current/functions-math.html>

Функции работы со временем

NOW() — возвращает текущую дату и время с точностью до микросекунд;

EXTRACT() — извлекает отдельные компоненты из даты и времени (день, месяц, год, час, минуту или секунду)

DATE_TRUNC() — округление до до заданной единицы измерения, такой как год, месяц, неделя, день, час, минута или секунда.

CURRENT_DATE — возвращает текущую дату без времени;

CURRENT_TIME — возвращает текущее время без даты.

INTERVAL — работа с интервалами

<https://www.postgresql.org/docs/current/functions-datetime.html>

Преобразование типов

CAST (source_type AS target_type)

```
SELECT CAST (WIGHT AS INTEGER)  
FROM SPUTNIK;
```

Создание представлений

```
CREATE [ OR REPLACE ] [ TEMP | TEMPORARY  
] [ RECURSIVE ] VIEW name  
[ ( column_name [, ...] ) ]  
  
    [ WITH ( view_option_name [=  
view_option_value] [, ... ] ) ]  
  
    AS query  
  
    [ WITH [ CASCADED | LOCAL ] CHECK  
OPTION ]
```

Создание представлений

```
CREATE VIEW v_sputnik AS  
SELECT s.id_country, sum(weight),  
count(weight), c."name"  
      FROM sputnik s, country c  
      WHERE s.id_country = c.id  
      group by id_country, c."name"  
      HAVING sum(weight) > 15  
      OR count(weight) > 1
```

Материализованные представления

```
CREATE MATERIALIZED VIEW [ IF NOT EXISTS ]  
table_name  
    [ (column_name [, ...] ) ]  
    [ USING method ]  
    [ WITH ( storage_parameter [= value]  
[, ...] ) ]  
    [ TABLESPACE tablespace_name ]  
AS query  
    [ WITH [ NO ] DATA ]
```

Материализованные представления

```
CREATE MATERIALIZED VIEW v_sputnik AS  
SELECT s.id_country, sum(weight),  
count(weight), c."name"  
  
    FROM sputnik s, country c  
  
    WHERE s.id_country = c.id  
  
group by id_country, c."name"  
  
    HAVING sum(weight) > 15  
           OR count(weight) > 1
```

Обновление представления

После изменения данных в основной таблице необходимо обновлять VIEW

REFRESH MATERIALIZED

VIEW mv_sputnik;

Спасибо за внимание.