

Шпаргалки

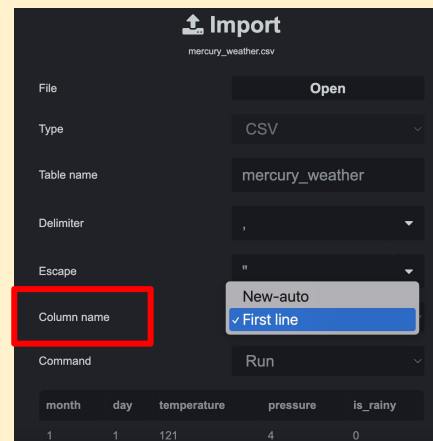
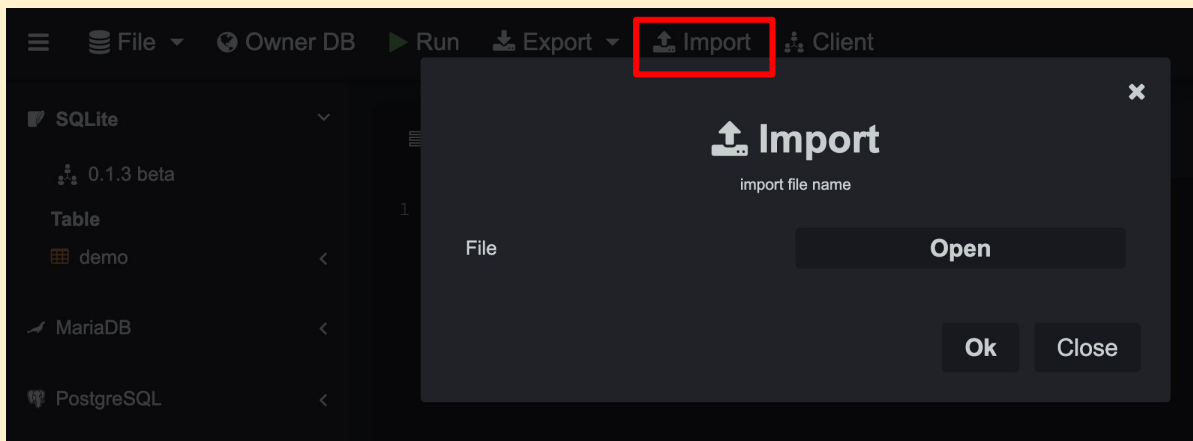
- Интерактивная шпаргалка:
<https://antonz.ru/sql-cheatsheet/>
- Оконные функции:
<https://www.sqlite.org/windowfunctions.html>

```
SELECT
    *,
    SUM(points) OVER w AS score
FROM
    potions
WINDOW w AS (
    PARTITION BY id
    ORDER BY name
    ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING
)
```

```
SELECT
    surname,
    COUNT(*) AS n
FROM
    heroes AS h
JOIN
    boys AS b
ON
    h.name = b.first
WHERE
    b.is_nice = 1
GROUP BY
    h.surname
```

Подготовка

1. Переименуйте этот файл своим именем или ником.
2. Скачайте с платформы **weather.csv** на локальное устройство.
3. На <https://sqliteonline.com/> через “Import” загрузите полученный файл.
4. В появившейся конфигурации поменяйте только
Column name -> First line.



Задание 1

- Изучите структуру таблицы любыми способами
- Какой перепад температур наблюдается на Меркурии?

Скопируйте сюда SQL-запрос:

```
pragma table_info(weather);
```

```
select  
    max(temperature) - min(temperature) as difference  
from  
    weather
```

Ответ: 288

Ревью:

Задание 2

Информация о скольких
месяцах представлена в
таблице?

Скопируйте сюда SQL-запрос:

```
select count(distinct(weather.month))  
from weather
```

Ответ: 4

Ревью:

Задание 3

Какой день был самым жарким?

Необходимо обойтись лишь одним запросом (возможно, с подзапросами)

1,3,169
1,4,169
2,1,169
2,4,169
2,7,169
4,2,169
4,4,169

Скопируйте сюда SQL-запрос:

```
select month, day
from weather
where temperature = (
    select max(temperature) from weather
)
```

Ответ:

Ревью:

Задание 4

Когда **“туристический сезон”**
на Меркурии?

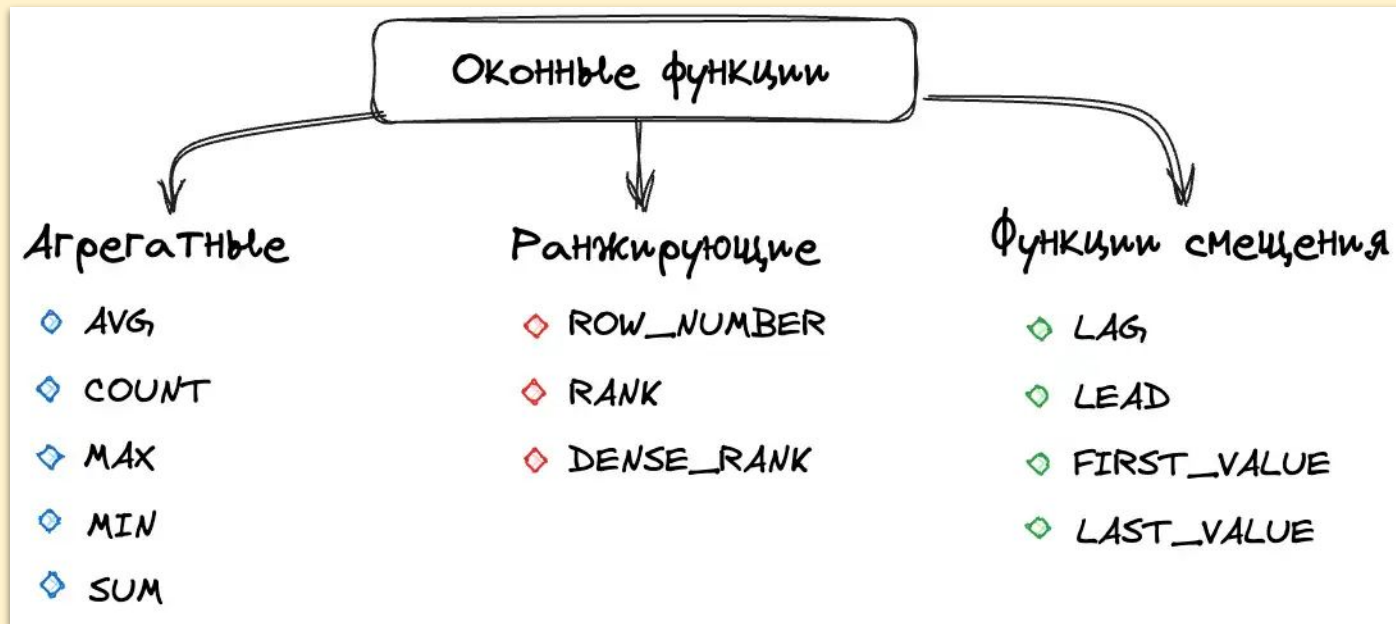
Выведите среднюю
температуру в каждом
месяце.

Скопируйте сюда SQL-запрос:

```
select distinct  
  month,  
  avg(temperature)  
over w as at  
from weather  
window w as (partition by month)
```

Ревью:

Шпаргалки



`<Window_function> (arguments) OVER ([Partitioning] [Ordering] [Frame])`

Задание 5



Для каждой даты посчитайте **отклонение** от средней температуры (арифм.) в этом месяце.

Необходимо учесть знак:
когда температура меньше ср.
ар., она должна быть
отрицательной.

Скопируйте сюда SQL-запрос:

```
select
  month,
  day,
  temperature - avg(temperature) over w as diff
from weather
window w as (partition by month)
```

Ревью:

Задание 6



Для каждой даты посчитайте количество дней с такой же температурой.

Скопируйте сюда SQL-запрос:

```
select
  *,
  count(*) over w as num
from weather
window w as (partition by temperature)
```

Ревью:

Задание 7



Для каждой даты посчитайте количество дождливых дней с начала года до текущего момента.

Скопируйте сюда SQL-запрос:

```
select
  month,
  day,
  is_rainy,
  sum(is_rainy) over w as nums
from weather
window w as (order by month, day)
```

Ревью:

Задание 8



Для каждой даты посчитайте количество дождливых дней с начала года до текущего момента, **не учитывая текущий день.**

Скопируйте сюда SQL-запрос:

```
select
  month,
  day,
  is_rainy,
  - is_rainy + sum(is_rainy) over w as nums
from weather
window w as (order by month, day)
```

Ревью:

Задание 9



Для каждой даты посчитайте разницу в температуре с предыдущим днем.

Скопируйте сюда SQL-запрос:

```
select *,  
       temperature - lag(temperature) over w as prev_temp  
from weather  
window w as (order by month)
```

Ревью:

Задание 10



Какое давление было в дни
перед дождем?

Скопируйте сюда SQL-запрос:

```
select *  
from (  
  select  
    month,  
    day,  
    is_rainy,  
    lag(pressure) over (order by month, day) as prev_pressure  
  from weather )  
where is_rainy = 1
```

Ревью:

Задание 11



- Выполните запрос:

```
SELECT
    *,
    RANK() OVER w AS score
FROM
    weather
OVER WINDOW w AS (
    PARTITION BY month
    ORDER BY temperature
)
```

- Попробуйте без документации понять назначение функции `RANK()`.
- В чем ее отличие от `ROW_NUMBER()`?
- А от `DENSE_RANK()`?

В чем отличие от `ROW_NUMBER()`?

`RANK` присваивает одинаковый ранг, на основе `ORDER BY`. Если несколько строк имеют одинаковые значения, они получают одинаковый ранг, а следующий ранг пропускается. В отличие от `ROW_NUMBER`, `RANK` имеет дубликаты

В чем отличие от `DENSE_RANK()`?

`RANK` присваивает одинаковый ранг, на основе `ORDER BY`. Если несколько строк имеют одинаковые значения, они получают одинаковый ранг, а следующий ранг пропускается, в то время как в `DENSE_RANK()` следующий ранг не пропускается

Задание 12

Решите любую из предыдущих задач на оконные функции, не используя оконные функции.

Скопируйте сюда SQL-запрос:

— Задание 4

```
select distinct  
  month,  
  avg(temperature)  
over w as at  
from weather  
window w as (partition by month);
```

— без оконных функций:

```
select month, avg(temperature)  
from weather  
group by month  
order by month;
```

Ревью:

Задание 13



Сколько дней длится **сезон дождей** на Меркурии?

P.S. **Сезон дождей** = самый длинный период из подряд идущих дождливых дней.

P.P.S. Имеются в виду метеоритные дожди.

P.P.S. Можно использовать то, что в месяце – 7 дней.

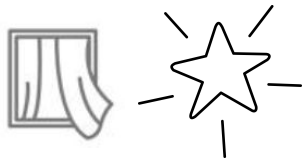
Скопируйте сюда SQL-запрос:

```
select max(rainy_days_count)
from (
  select month, count(*) as
rainy_days_count
  from weather
  where is_rainy = 1
  group by month
  order by month
)
```

Ответ: 3

Ревью:

Задание 14



Используя `RANK()`, посчитайте **медиану** давления в каждый из месяцев.

С четностью управьтесь на свое усмотрение.



Скопируйте сюда SQL-запрос:

```
select month, avg(pressure) as median_pressure
from (
  select
    month,
    pressure,
    row_number() over (partition by month order by pressure) as rn,
    count(*) over (partition by month) as tc
  from weather)
where rn in (
  (tc + 1) / 2,
  (tc + 2) / 2)
group by month
order by month
```

Ревью: