

Памятка

```
SELECT *
FROM heroes
LIMIT 5
```

```
SELECT DISTINCT (name)
FROM heroes
LIMIT 5
```

```
SELECT MAX (founder)
FROM school
WHERE
    faculty != 'Гриффиндор' AND
    faculty != 'Слизерин'
```

```
SELECT
    MAX(students),
    MIN(students),
    SUM(students),
    AVG(students)
FROM school
```

```
SELECT
    COUNT(DISTINCT name)
FROM heroes
```

Задание 1.

1. Загрузите БД себе на компьютер.
2. Зайдите на <https://sqliteonline.com/>.
3. Загрузите базу данных: File → Open DB

Задание 1.

В таблице **map** собраны данные о расположении различных объектов в городе.

Напишите запрос, который позволит вам познакомиться с этой таблицей: выведете на экран названия столбцов и первые 5 строк содержимого.

Скопируйте сюда SQL-запрос:

```
SELECT *  
FROM map  
LIMIT 5;
```

```
PRAGMA table_info(map);
```

```
select sql from sqlite_master  
where type='table' and tbl_name='map'
```

Сколько столбцов в таблице?

- 5

 Получите всю метаинформацию о столбцах

 Получите запрос, которым можно создать таблицу с той же схемой

Задание 2.

Какие объекты есть на карте города Найденовска?

Выведите их все по одному разу, догадавшись, в каком столбце расположены эти данные?

Скопируйте сюда SQL-запрос:

```
select distinct (object)  
from map
```

```
select count(distinct (object))  
from map
```

```
select *  
from map  
order by object  
limit 1
```

Сколько всего различных объектов?

- 19

Какой объект первый в алфавитном порядке?

- Bench

Задание 3.

Найденовск имеет форму прямоугольника. Для удобства он разбит на небольшие квадраты, имеющие свои уникальные целые положительные координаты по **x** (горизонталь) и по **y** (вертикаль).

В таблице **map** хранятся данные о каждом из этих квадратов.

Например, в столбце **height** записаны высоты участков относительно уровня моря в метрах.

Найдите:

1. Среднюю высоту участков.
2. Высоту самой высокой точки

Скопируйте сюда SQL-запрос:

```
select avg(height), max(height)  
from map
```

Ответы:

1. Округлите до двух знаков после запятой: 26,19
2. 1515

Задание 4.

Город Сочи известен тем, что перепад высот в нем – самый большой в России – 1569 метров.

Что можно сказать о том, в какой стране находится Найденовск по данным из столбца `height`?

Скопируйте сюда SQL-запрос:

```
select max(height) - min(height)  
from map
```

Какой перепад высот?

1515

Что можно сказать?

Задание 5.

Какова площадь города
Найденовска?

Скопируйте сюда SQL-запрос:

```
select count(number)  
from map
```

Площадь: 12345

Задание 6.

Определите ширину
(горизонтальный размер) и
длину (вертикальный размер)
Найденовска?

Скопируйте сюда SQL-запрос:

```
select
    max(x) as width,
    max(y) as heigth
from map
```

Ширина: 14

Высота: 822

Задание 7.

Линия хребта Горизонтальпы делит Найденовск ровно пополам и строго параллелен горизонтали. Определите, какие объекты расположены на линии этого хребта.

- а) Для начала вы можете использовать два запроса.
- б) Попробуйте самостоятельно догадаться, как оформить решение в один запрос.

Скопируйте сюда SQL-запрос:

```
select object  
from map  
where y = (select max(y)/2 from map)
```

Объекты:

Гора, скала

Задание 8.

Найденовский парк имеет форму квадрата 5 на 5 и называется “Английский”. Расположен он в левом верхнем углу города – в начале координат. Постройте карту парка. Саму лужайку отмечать не надо.



Скопируйте сюда

```
select * from map  
where x < 5 and y < 5
```

0 x

0

y

Задание 9.

Какую часть площади занимают горы и озера?

Можно использовать два запроса

Скопируйте сюда SQL-запрос:

```
select
    count(case when object = 'Гора' or object = 'Озеро' then 1 end) * 1.0 /
    count(*) * 100
from map
```

Ответ:

1.09 %

Задание 10.

Таблица **residents** содержит информацию о жителях Найденовска.

- а) Выясните, сколько человек в семьях с фамилиями, начинающимися на “Р”?
- б) Сколько всего таких различных фамилий?

Оба пункта необходимо выполнить одним запросом.

⚠ Для решения вы можете прочитать:
<https://sql-academy.org/ru/guide/operator-like>

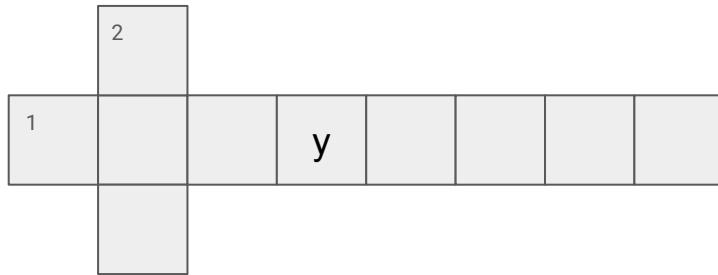
Скопируйте сюда SQL-запрос:

```
select  
    count(*), count(distinct family)  
from residents  
where family like 'P%'
```

Ответ: 5, 1

Задание 11.

Разгадайте кроссворд:



1. Семья из города Найденовска
2. Имя члена семьи Хомяковых

Скопируйте сюда SQL-запрос:

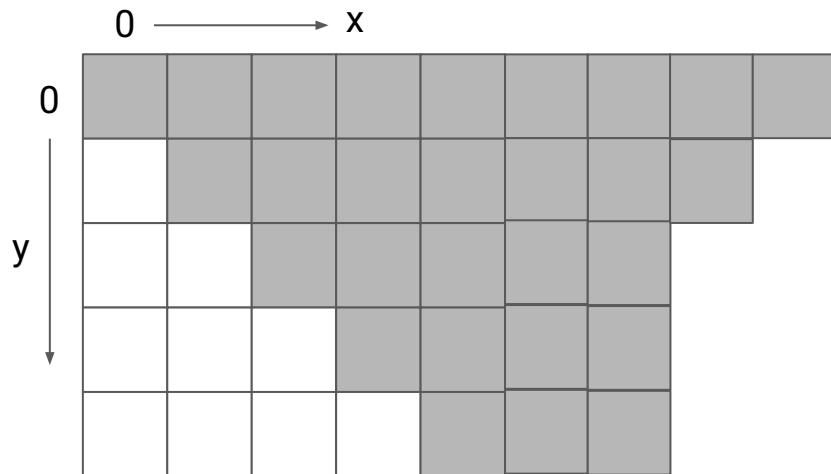
```
select distinct family, name  
from residents  
where  
    family like '___у___' or  
(family = 'Хомяковы' and  
name like '___')
```

2 - Лев
1 - Лекуновы

Задание 12.

В таблице **found** находятся данные о найденных вещах в городе Найденовске в марте.

Исследуйте эту таблицу и посчитайте, сколько всего вещей было найдено в марте в области, обозначенной серым (выше диагонали)?



Скопируйте сюда SQL-запрос:

```
select count(*)  
from found  
where where_x >= where_y
```

Ответ: 1

Часть 2

Памятка

```
CREATE TABLE nutrients (
    id int,
    name varchar,
    calories int
);
```

: id	name	calories
3	tomato	100
4	egg	120
5	biscuit	240

```
INSERT INTO nutrients(id, name, calories) VALUES (3, 'tomato', 100);
INSERT INTO nutrients(id, name, calories) VALUES (4, 'egg', 120);
INSERT INTO nutrients(id, name, calories) VALUES (5, 'biscuit', 240);
```

Задание 13.

- Выполните запросы из памятки
- Еще самостоятельно создайте такую таблицу:

my_meals

dish	weight
3	100
3	1000
id блюда, которое съели бы на ужин (4 или 5)	10000

Скопируйте сюда SQL-запрос:

```
create table my_meals (
    dish int,
    weight int
);
```

```
insert into my_meals (dish, weight) values (3, 100);
insert into my_meals (dish, weight) values (3, 1000);
insert into my_meals (dish, weight) values (4, 10000);
```

Задание 14.

- Выполните запрос
- Проанализируйте результаты

```
SELECT
    my_meals.*,
    nutrients.name as name
FROM
    my_meals
LEFT JOIN
    nutrients
ON
    my_meals.dish == nutrients.id
```

id	name	calories
3	tomato	100
4	egg	120
5	biscuit	240



dish	weight
3	100
3	1000
5	10000



Задание 15.

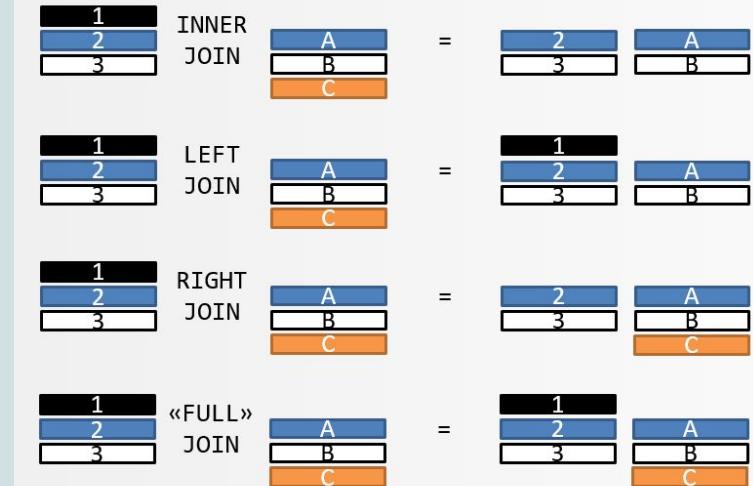
1. Модифицируйте предыдущий запрос так, чтобы на экран вывелаась таблица `my_meals` с указанием количества калорий в каждом блюде
2. Посчитайте суммарное количество съеденных калорий

Скопируйте сюда SQL-запрос:

```
select  
sum(nutrients.calories) as calories  
from my_meals  
left join nutrients  
on my_meals.dish == nutrients.id
```

Памятка

```
SELECT heroes.* , school.students
FROM
    heroes
LEFT JOIN
    school
ON
    heroes.decision = school.faculty
```



Задание 16.

Что когда-либо теряли
Растеряевы?

Скопируйте сюда SQL-запрос:

```
select
found.what
from residents
join found
on residents.id == found.who
where residents.family == 'Растеряевы'
```

Ответ:

Перчатка, Телефон, Нашники,
Пульт от телевизора, Ключи, Наушники,
Хомячок, Рюкзак

Задание 17.

Сколько всего перчаток потеряли члены семьи Растеряевых?

Скопируйте сюда SQL-запрос:

```
select
count(found.what)
from residents
join found
on residents.id == found.who
where residents.family == 'Растеряевы' and found.what == 'Перчатка'
```

Ответ: 5

Задание 18.

Сколько всего перчаток потеряла
Мария Растеряева?

Скопируйте сюда SQL-запрос:

```
select count(*)  
from residents  
join found  
on residents.id == found.who and  
residents.family == 'Растеряевы'  
where residents.name == 'Мария' and  
found.what == 'Перчатка'
```

Ответ: 5

Задание 19.

Выведите фамилии тех, кто когда либо терял хомячка?

Скопируйте сюда SQL-запрос:

```
select residents.family  
from residents  
join found  
on residents.id == found.who  
where found.what == 'Хомячок'
```

Ответ: Растеряевы,
Хомяковы

Задание 20.

Сколько человек потеряли что-то
в школе за всю историю?

Скопируйте сюда SQL-запрос:

```
select count(distinct found.who) as number
from map
join found
on map.x == found.where_x and map.y == found.where_y
where map.object == 'Школа'
```

Ответ: 3

Задание 21.

Назовите фамилии людей,
потерявших что-то у склепа?

Скопируйте сюда SQL-запрос:

```
select family  
from map  
join found  
on map.x == found.where_x and map.y == found.where_y  
join residents  
on residents.id == found.who  
where map.object == 'Склеп'
```

Ответ:

Растеряевы,
Дуров

Задание 22.

Как вы думаете, где живет Мария Растеряева, если ее дом расположен где-то вдоль трассы, а ходит она только в школу и только по прямой?

Необходимо назвать координаты.

Скопируйте сюда SQL-запрос:

```
select *  
from found  
join residents on residents.id == found.who  
join map on found.where_x = map.x and found.where_y = map.y  
where residents.id = 0
```

Ответ:

x = 6, y = 10

Часть 3

Памятка

```
SELECT
    decision,
    COUNT(id) as number
FROM
    heroes
GROUP BY
    decision
```

Задание 23.

Выведите, сколько человек в каждой из семей.

Скопируйте сюда SQL-запрос:

```
select  
family, count(id)  
from residents  
group by  
family
```

Ответ:

Банаховы, 2
Дуров, 2
Кнутовы, 2
Кустовы, 2
Лекуновы, 2
Многоликовы, 1
Найденкины, 3
Растеряевы, 5
Трамповы, 2
Хомяковы, 3

Задание 24.

Найдите самую большую высоту для каждой из широт.

Скопируйте сюда SQL-запрос:

```
select
x, max(height) as max_height
from map
group by x
```

Задание 25.

Выведите семью, в которой всего лишь один человек.

Скопируйте сюда SQL-запрос:

```
select
    family, count(id) as number
from
    residents
group by
    family
having
    number = 1
```

Ответ:

Многоликовы, 1

Задание 26.

Найдите одним запросом координаты самой высокой точки (известно, что она одна). Что там расположено?

Скопируйте сюда SQL-запрос:

```
select x, y, object  
from map  
where height = (select max(height) from map);
```

Ответ:

x=9, y=15,
Fitness-клуб

Задание 27.

Найдите одним запросом координаты второй по высоте точки (известно, что она одна). Что там расположено?

Скопируйте сюда SQL-запрос:

```
select x, y, object  
from map  
order by height desc  
limit 1 offset 1
```

Ответ:

x = 14, y = 4, Трасса

Задание 28.

Найдите имя и фамилию человека, который ничего никогда не терял.

Стефан,Банаховы
Лев,Дуров
Курт,Кнутовы
Дональд,Кнутовы
Ив,Кустовы
Ян,Лекуновы
Яна,Лекуновы
Петр,Найденкины
Глафира,Растеряевы
Геннадий,Хомяковы
Ирина,Хомяковы

Скопируйте сюда SQL-запрос:

```
select distinct residents.name, residents.family  
from residents  
left join found on residents.id = found.who  
where found.who is null  
order by family
```

Ответ:

Задание 29.

Найдите имена и фамилии людей, кто терял в одном и том же месте одну и ту же вещь несколько раз (т.е. не меньше 2 раз).

Скопируйте сюда SQL-запрос:

```
select r.name, r.family  
from residents as r  
join found as f on r.id = f.who  
group by r.id, r.name, r.family, f.where_x, f.where_y  
having count(*) >= 2;
```

Ответ:

Мария,Растеряевы
Жак,Кустовы

Задание 30.

Найдите все фамилии, начинающиеся с буквы “Д”.

Выведите их в алфавитном порядке.

Скорее всего, вам понадобится помочь Интернета.

Скопируйте сюда SQL-запрос:

```
select family  
from residents  
where family like 'Д%'  
order by family;
```

Ответ:

Дуров
Дуров

Задание 31.

Один человек умудрился
потерять некий объект трижды
на одной долготе (с одним у).

Кто это и что он потерял.

Вам понадобится
http://www.sql-tutorial.ru/ru/book_lag_and_lead_functions.html

Скопируйте сюда SQL-запрос:

```
select r.name, r.family, f.what
from residents as r
join found as f on r.id = f.who
group by r.id, r.name, r.family, f.what, f.where_y
having count(*) = 3;
```

Ответ:

Павел, Дуров, Совесть

Задание 32.

1. Зарегистрируйтесь на <https://sql-ex.ru>
2. Войдите под своим логином и паролем
3. Перейдите на https://sql-ex.ru/learn_exercises.php
4. Выполняйте упражнения по очереди, а результаты скидывайте на следующие слайды

Nº1

```
select model, speed, hd from PC  
where price < 500
```

Nº2

```
select distinct maker, Printer.model  
from Product  
join Printer on Product.model = Printer.model
```

Nº3

```
select model, ram, screen  
from Laptop  
where price > 1000
```

Nº4

```
select *  
from Printer  
where color = 'y'
```

Nº5

```
select distinct model, speed, hd from PC  
where cd in ('12x', '24x') and price < 600
```

Nº 6

```
select p.maker, l.speed
from Product p
join Laptop l on p.model = l.model
where p.type = 'Laptop' and l.hd >= 10
order by l.speed
```

Nº 7

```
select distinct p.model, coalesce(pc.price, l.price, pr.price) as price from Product p
left join PC pc on p.model = pc.model
left join Laptop l on p.model = l.model
left join Printer pr on p.model = pr.model
where p.maker = 'B'
```

Nº 8

```
select distinct maker from Product p join PC pc on p.model = pc.model where p.type in ('PC')
except
select distinct maker from Product p join Laptop l on p.model = l.model
```

Nº 9

```
select distinct maker from Product as p
join PC on PC.model = p.model
where PC.speed >= 450
```

Nº 10

```
select model, price from Printer as p
where p.price = (select max(price) from Printer)
order by model
```