

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| one |
| performance_schema |
| sys |
+-----+
5 rows in set (0.21 sec)

mysql> create database countries;
Query OK, 1 row affected (0.09 sec)

mysql> _
```

Просмотр существующих баз данных и создание новой (Show databases; Create database).

```
mysql> drop database one;
Query OK, 2 rows affected (0.20 sec)
```

Удаление базы данных (Drop database).

```
mysql> use countries;
Database changed
mysql> _
```

Использование БД (Use).

```
mysql> create table country
-> (id int(12) not null primary key,
-> city varchar(99) null,
-> number int(10) null);
Query OK, 0 rows affected, 2 warnings (0.09 sec)

mysql> desc country;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int           | NO   | PRI | NULL    |       |
| city  | varchar(99)   | YES  |     | NULL    |       |
| number | int           | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.05 sec)
```

Создание таблицы в БД и столбцов в таблице, а также типов, применяемых к полям, просмотр полей таблицы (Create table, Desc).

```
mysql> insert into country values (1, "Moscow", 1);
Query OK, 1 row affected (0.02 sec)

mysql> insert into country values (2, "Beijing", 2);
Query OK, 1 row affected (0.01 sec)

mysql> insert into country values (3, "Minsk", 3);
Query OK, 1 row affected (0.01 sec)

mysql> insert into country values (4, "Pyongyang", 4);
Query OK, 1 row affected (0.01 sec)

mysql> insert into country values (5, "Delhi", 5);
Query OK, 1 row affected (0.01 sec)
```

Заполнение полей таблицы (Insert).

```
mysql> select * from country;
```

id	city	number
1	Moscow	1
2	Beijing	2
3	Minsk	3
4	Pyongyang	4
5	Delhi	5

```
5 rows in set (0.00 sec)
```

Посмотреть таблицу (Select * from).

```
mysql> delete from country where city= "Delhi";
Query OK, 1 row affected (0.01 sec)
```

```
mysql> select * from country;
```

id	city	number
1	Moscow	1
2	Beijing	2
3	Minsk	3
4	Pyongyang	4

```
4 rows in set (0.00 sec)
```

Удаление значения из столбца (Delete from .. where .. = ..).

```
mysql> update country set city="Ho Chi Minh" where city="Pyongyang";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from country;
+-----+-----+-----+
| id | city          | number |
+-----+-----+-----+
| 1 | Moscow        | 1      |
| 2 | Beijing       | 2      |
| 3 | Minsk         | 3      |
| 4 | Ho Chi Minh   | 4      |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

Замена значения в столбце (Update).

```
mysql> alter table country drop column number;
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from country;
+-----+-----+
| id | city          |
+-----+-----+
| 1 | Moscow        |
| 2 | Beijing       |
| 3 | Minsk         |
| 4 | Ho Chi Minh   |
+-----+-----+
4 rows in set (0.00 sec)
```

Удаление столбца в таблице (Alter table .. drop column ..).

```
mysql> alter table country add column number int after city;
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from country;
+-----+-----+-----+
| id | city          | number |
+-----+-----+-----+
| 1 | Moscow        | NULL   |
| 2 | Beijing       | NULL   |
| 3 | Minsk         | NULL   |
| 4 | Ho Chi Minh   | NULL   |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

Добавление столбца (Alter table .. add column .. int after..).

```
mysql> select distinct number from country;
+-----+
| number |
+-----+
|      NULL      |
+-----+
1 row in set (0.00 sec)

mysql>
```

Посмотреть столбец (Select distinct).

```
mysql> select * from country limit 3;
+----+-----+-----+
| id | city   | number |
+----+-----+-----+
|  1 | Moscow |  NULL  |
|  2 | Beijing |  NULL  |
|  3 | Minsk  |  NULL  |
+----+-----+-----+
3 rows in set (0.00 sec)
```

Вывести лимитированное количество полей из таблицы (Select * from .. limit..).

```
mysql> select number as номер from country;
+-----+
| номер |
+-----+
|      NULL      |
|      NULL      |
|      NULL      |
|      NULL      |
+-----+
4 rows in set (0.00 sec)
```

Временное обозначение столбца (Select .. as..).

```
mysql> select city, id from country where city = "Moscow";
+-----+-----+
| city  | id  |
+-----+-----+
| Moscow | 1  |
+-----+-----+
1 row in set (0.00 sec)
```

Нахождение одного значения в строке используя два других (Select .. from where .. = ..).

```
mysql> select id from country where id > 1;
+-----+
| id  |
+-----+
| 2  |
| 3  |
| 4  |
+-----+
3 rows in set (0.00 sec)
```

Фильтр в столбце по заданному типу (Select .. from .. where .. > ..).

```
mysql> select * from country where id = 3 and city = "Minsk";
+-----+-----+-----+
| id  | city  | number |
+-----+-----+-----+
| 3  | Minsk | NULL   |
+-----+-----+-----+
1 row in set (0.00 sec)
```

Фильтр в таблице по двум значениям (Select * from .. where .. = .. and .. = ..).

```
mysql> select * from country where id = 2 or city = "Minsk";
+-----+-----+-----+
| id  | city  | number |
+-----+-----+-----+
| 2  | Beijing | NULL   |
| 3  | Minsk  | NULL   |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Фильтр в таблице, по двум возможным значениям Select * from .. where .. = .. or .. = ..).

```
mysql> select city from country where city like "%ns_";
+-----+
| city |
+-----+
| Minsk |
+-----+
1 row in set (0.00 sec)
```

Нахождение строки по не полному слову (Select .. from .. where .. like “%.._”).

```
mysql> select * from country where id between 1 and 4;
+----+-----+-----+
| id | city      | number |
+----+-----+-----+
| 1  | Moscow    | NULL   |
| 2  | Beijing   | NULL   |
| 3  | Minsk     | NULL   |
| 4  | Ho Chi Minh | NULL   |
+----+-----+-----+
4 rows in set (0.01 sec)
```

Фильтр диапазона между двумя значениями в столбце (Select * from .. where .. between .. and ..).

```
mysql> select * from country where not id between 1 and 3;
+----+-----+-----+
| id | city      | number |
+----+-----+-----+
| 4  | Ho Chi Minh | NULL   |
+----+-----+-----+
1 row in set (0.00 sec)
```

Фильтр за границами указанного диапазона (Select * from .. where not .. between .. and ..).

```
mysql> select * from country where not id in (2,4);
+----+-----+-----+
| id | city   | number |
+----+-----+-----+
| 1  | Moscow | NULL   |
| 3  | Minsk  | NULL   |
+----+-----+-----+
2 rows in set (0.00 sec)

mysql> select * from country where id in (2,4);
+----+-----+-----+
| id | city       | number |
+----+-----+-----+
| 2  | Beijing   | NULL   |
| 4  | Ho Chi Minh | NULL   |
+----+-----+-----+
2 rows in set (0.00 sec)
```

Фильтр исключающей заданное значение

(Select * from .. where not .. in ..).

```
mysql> select * from country where (city = "Paris" or city = "bangkok") and number =5;
+----+-----+-----+
| id | city   | number |
+----+-----+-----+
| 5  | Paris  | 5       |
| 7  | Bangkok | 5       |
+----+-----+-----+
2 rows in set (0.00 sec)

mysql> _
```

Фильтр с возможным значением и указанным (Select * from .. where ..or.. and ..=..).


```
mysql> select * from country order by city;
+-----+-----+-----+
| id | city          | number |
+-----+-----+-----+
| 7 | Bangkok       | 5      |
| 2 | Beijing       | NULL   |
| 8 | Berlin        | 6      |
| 4 | Ho Chi Minh   | NULL   |
| 3 | Minsk         | NULL   |
| 1 | Moscow        | NULL   |
| 5 | Paris         | 5      |
| 6 | Washington    | 6      |
+-----+-----+-----+
8 rows in set (0.00 sec)

mysql> select * from country order by city desc;
+-----+-----+-----+
| id | city          | number |
+-----+-----+-----+
| 6 | Washington    | 6      |
| 5 | Paris         | 5      |
| 1 | Moscow        | NULL   |
| 3 | Minsk         | NULL   |
| 4 | Ho Chi Minh   | NULL   |
| 8 | Berlin        | 6      |
| 2 | Beijing       | NULL   |
| 7 | Bangkok       | 5      |
+-----+-----+-----+
8 rows in set (0.00 sec)

mysql>
```

Сортировка по алфавиту и в обратном порядке (Select * from .. order by..).

```
mysql> select avg(number) from country;
+-----+
| avg(number) |
+-----+
| 5.5000      |
+-----+
1 row in set (0.02 sec)
```

Вычисление среднего значения в поле (Select avg .. from ..).

```
mysql> select sum(number) from country;
+-----+
| sum(number) |
+-----+
|          22 |
+-----+
1 row in set (0.01 sec)

mysql> _
```

Вычисление суммы чисел (Select sum .. from ..).

```
mysql> select min(number) as min_number from country;
+-----+
| min_number |
+-----+
|          5 |
+-----+
1 row in set (0.01 sec)
```

Фильтр по минимальному числу (Select min .. as .. from ..).

```
mysql> select count(city) from country;
+-----+
| count(city) |
+-----+
|           8 |
+-----+
1 row in set (0.01 sec)

mysql> _
```

Подсчет количества записей в столбце (Select count (..) from ..).

24 • `select country. city, capitals. country, capitals. code from country join capitals on country. id = capitals. id;`

	city	country	code
►	Moscow	Russia	7
	Beijing	China	86
	Minsk	Belarus	37
	Ho Chi Minh	Vietnam	84
	Paris	France	33
	Washington	USA	1
	Bangkok	Thailand	66
	Berlin	Germany	49

Result 141 x Read Only

Поиск соответствия между двумя таблицами по заданному значению (Select .. from .. join .. on ..).

24 • `* from country;`
 25 • `* from capitals;`
 26 • `country. city, capitals. country, capitals. code from country join capitals on country. number = capitals. number;`

	city	country	code
►	Bangkok	France	33
	Paris	France	33
	Berlin	USA	1
	Washington	USA	1

Result Grid Form

Выборка совпадений между двумя таблицами по заданному значению (Select .. from .. join .. on ..).

24 • `select * from country;`
 25 • `select * from capitals;`
 26 • `select country. city, capitals. country, capitals. code from country join capitals on country. number = capitals. number;`
 27 • `select country. number, capitals. code from country left join capitals on country. id = capitals. id;`

	number	code
►	NULL	7
	NULL	86
	NULL	37
	NULL	84
	5	33
	6	1
	5	66
	6	49

Result Grid Form Editor

Выборка совпадений в одном столбце учитывая совпадения из двух других столбцов из разных таблиц.

28 • `country. number, capitals. country, capitals. code, capitals. number, country. city from country right join capitals on country. id = capitals. id;`

29

Result Grid

	number	country	code	number	city
▶	NULL	Russia	7	1	Moscow
	NULL	China	86	2	Beijing
	NULL	Belarus	37	3	Minsk
	NULL	Vietnam	84	4	Ho Chi Minh
	5	France	33	5	Paris
	6	USA	1	6	Washington
	5	Thailand	66	7	Bangkok
	6	Germany	49	8	Berlin

Result 23 x

Read Only

Выборка значений с использованием (Right join).

28 • `select * from capitals;`

29 • `select id, country from capitals where number in (select number from capitals where id>4 and code>=33);`

30

Result Grid

	id	country
▶	5	France
	7	Thailand
	8	Germany
	9	Italy
	10	Vatican
•	NULL	NULL

Result Grid

Form Editor

Выборка значений в указанном диапазоне внутри запроса.