



Limit to 1000 rows

```
1 • SELECT fname, lname, patronus FROM hogwarts.characters where patronus is not null and patronus <> 'Unknown'
```

2

3

&lt;

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	fname	lname	patronus
--	-------	-------	----------

▶	Harry	Potter	Stag
	Hermione	Granger	Otter
	Ron	Weasley	Jack Russell terrier
	Albus	Dumbledore	Phoenix
	Luna	Lovegood	Hare
	Severus	Snape	Doe

1. Выведите имя, фамилию, патронуса всех персонажей, у которых есть patronus и он известен

characters 4 x

```
1 • SELECT lname FROM hogwarts.characters where lname like '%e'
```

2

3

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	lname
▶	Crabbe
	Goyle
	Dumbledore
	Snape

2. Выведите фамилию персонажей, у которых последняя буква в фамилии 'е'

characters 7 x



Limit to 1000 rows

```
1 • SELECT SUM(Age) FROM hogwarts.characters
```

2

3



Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	SUM(Age)
▶	257

3. Посчитайте общий возраст всех персонажей и выведите это на экран



SQL File 2\* characters characters characters x

Limit to 1000 rows

```
1 • SELECT fname, age FROM hogwarts.characters where age between 50 and 100
2
3
```

Result Grid

	fname	age
▶	Severus	55

Filter Rows: Export: Wrap Cell Content:

characters 17 x

5. Выведите имя персонажа и возраст, у которых последний находится в диапазоне от 50 до 100 лет



```
1 • SELECT distinct age FROM hogwarts.characters
```

```
2
```

```
3
```



Result Grid | Filter Rows:  | Export: | Wrap Cell Content:

	age
▶	11
	111
	14
	55

6. Выведите возраст всех персонажей так, чтобы среди них не было тех, у кого он одинаковый



```
1 • SELECT * FROM characters where faculty = 'Gryffindor' and age > 30
```

```
2
```

```
3
```



Result Grid Filter Rows:  Edit: Export/Import: Wrap Cell Content:

	char_id	fname	lname	age	faculty	patronus	book_id
▶	7	Albus	Dumbledore	111	Gryffindor	Phoenix	2
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL

7. Выведите всю информацию о персонажах, у которых faculty = Gryffindor и чей возраст больше 30 лет

```
1 • SELECT distinct faculty FROM hogwarts.characters order by faculty limit 3;
```

< Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

	faculty
▶	Gryffindor
	Hufflepuff
	Ravendaw

8. Выведите имена первых трех факультетов из таблицы, так чтобы факультеты не повторялись



1 • `SELECT fname FROM hogwarts.characters where fname like 'h____' or fname like 'l%'`

<

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	fname
▶	Harry
	Luna
	Lord

9. Выведите имена всех персонажей, у которых имя начинается с 'Н' и состоит из 5 букв, или чье имя начинается с 'L'

characters 4 x

Output

1 • `SELECT avg(age) FROM hogwarts.characters`

Result Grid



Filter Rows:

Export:



Wrap Cell Content:



	avg(age)
▶	25.7000

10. Посчитайте средний возраст всех персонажей

1 • `delete FROM hogwarts.characters where char_id = 11;`  
2 • `select * from hogwarts.characters`

11. Удалите персонажа с ID = 11

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

	char_id	fname	lname	age	faculty	patronus	book_id
▶	1	Harry	Potter	11	Gryffindor	Stag	10
	2	Hermione	Granger	11	Gryffindor	Otter	9
	3	Ron	Weasley	11	Gryffindor	Jack Russell terrier	8
	4	Draco	Malfoy	11	Slytherin	NULL	6
	5	Vincent	Crabbe	11	Slytherin	NULL	6
	6	Gregory	Goyle	11	Slytherin	NULL	1
	7	Albus	Dumbledore	111	Gryffindor	Phoenix	2
	8	Luna	Lovegood	11	Ravenclaw	Hare	2
	9	Cedric	Diggory	14	Hufflepuff	Unknown	3
	10	Severus	Snape	55	Slytherin	Doe	4
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

characters 7 x

Output

Action Output

#	Time	Action	Message
✓	6 22:54:44	SELECT distinct faculty FROM hogwarts.characters LIMIT 0, 1000	4 row(s) returned
✓	7 22:55:32	SELECT distinct faculty FROM hogwarts.characters order by faculty limit 3	3 row(s) returned
✓	8 22:57:54	SELECT fname FROM hogwarts.characters where fname like 'h____' or fname like '1%' LIMIT 0, 1000	3 row(s) returned
✓	9 22:58:37	SELECT avg(age) FROM hogwarts.characters LIMIT 0, 1000	1 row(s) returned
✗	10 23:00:47	delete FROM hogwarts.characters where ID=11	Error Code: 1054. Unknown column 'ID' in 'where clause'
✓	11 23:01:15	select * FROM hogwarts.characters LIMIT 0, 1000	11 row(s) returned
✓	12 23:01:36	delete FROM hogwarts.characters where char_id = 11	1 row(s) affected
✓	13 23:02:51	delete FROM hogwarts.characters where char_id = 11	0 row(s) affected
✓	14 23:02:51	select * from hogwarts.characters LIMIT 0, 1000	10 row(s) returned

characters - Table

characters

characters x

Limit to 1000 rows

1 • `SELECT lname FROM hogwarts.characters where lname like 'a%'`

<

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	lname
▶	Granger
	Weasley
	Malfoy
	Crabbe
	Snape

12. Выведите фамилию всех персонажей, которые содержат в ней букву 'а'

characters 5 x

Read Only

characters - Table   characters   **characters** x

Limit to 1000 rows

```
1 select lname as "Half-blooded prince" from characters where lname = 'Snape'
2
```

Result Grid   Filter Rows:   Export:   Wrap Cell Content: [⌕](#)

	Half-blooded prince
▶	Snape

13. Используйте псевдоним для того, чтобы временно заменить название столбца lname на Half-Blood Prince для реального принца-полукровки



3

3

### Result Grid



Filter Rows:

Export:



Wrap Cell Content:

 $\overline{IA}$ min(Age)

11

16. Выведите минимальный возраст персонажа





```
1 select fname from characters UNION select book_name from library
```

```
2
```

```
3
```



Result Grid Filter Rows:  Export: Wrap Cell Content:

	fname
	Vincent
	Gregory
	Albus
	Luna
	Cedric
	Severus
	Hogwarts: A History
	Quidditch Through The Ages

17. Используя оператор UNION  
выберите имена из таблицы  
characters и названия книг из  
таблицы library

Result 17 x

Output

```
1 select count(fname), faculty from hogwarts.characters group by faculty having count(fname)>1
```

2

3

◀

Result Grid   Filter Rows:  Export:  Wrap Cell Content: 

	count(fname)	faculty
►	4	Gryffindor
	4	Slytherin

18. Используя оператор HAVING посчитайте количество персонажей на каждом факультете, оставив только те факультеты, где количество студентов больше 1



```

1  select fname, lname,
2  case
3  when faculty = 'Gryffindor' then 'Godric'
4  when faculty = 'Slytherin' then 'Salazar'
5  when faculty = 'Ravenclaw' then 'Rowena'
6  when faculty = 'Hufflepuff' then 'Helga'
7  else 'muggle'
8  END AS Founders
9  from characters
10

```

19. Используя оператор CASE опишите следующую логику:  
Выведите имя и фамилию персонажа, а также следующий текстовое сообщение:

Если факультет Gryffindor, то в консоли должно вывестись Godric

Если факультет Slytherin, то в консоли должно вывестись Salazar

Если факультет Ravenclaw, то в консоли должно вывестись Rowena

Если факультет Hufflepuff, то в консоли должно вывестись Helga

Если другая информация, то выводится Muggle

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	fname	lname	Founders
▶	Harry	Potter	Godric
	Hermione	Granger	Godric
	Ron	Weasley	Godric
	Draco	Malfoy	Salazar
	Vincent	Crabbe	Salazar

Для сообщения используйте псевдоним Founders



```
1 • select lname from characters where lname NOT REGEXP '^(H|L|S)'
```



Result Grid Filter Rows:  Export: Wrap Cell Content:

	lname
▶	Potter
	Granger
	Weasley
	Malfoy
	Crabbe
	Goyle
	Dumbledore
	Diggory

20. Используя регулярное выражение найдите фамилии персонажей, которые не начинаются с букв H, L или S и выведите их