

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

The screenshot shows a database query editor with a sidebar on the left displaying a schema tree. The main area contains a SQL query with three lines. The first line is a comment. The second line is the query to be executed. The third line is a comment. The results are displayed in a table below the query.

```
1 • select *from hogwarts.characters;
2 • select fname, lname, patronus from hogwarts.characters where patronus is not null or patronus="known";
3 •
```

fname	lname	patronus
Harry	Potter	Stag
Hermione	Granger	Otter
Ron	Weasley	Jack Russell terrier
Albus	Dumbledore	Phoenix
Luna	Lovegood	Hare
Cedric	Diggory	Unknown
Severus	Snape	Doe

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

The screenshot shows a database query editor with a sidebar on the left displaying a schema tree. The main area contains a SQL query with three lines. The first line is a comment. The second line is the query to be executed. The third line is the query to be executed. The results are displayed in a table below the query.

```
1 • select *from hogwarts.characters;
2 • select fname, lname, patronus from hogwarts.characters where patronus is not null or patronus="known";
3 • select lname from hogwarts.characters where lname like "%e";
```

lname
Crabbe
Goyle
Dumbledore
Snape

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

The screenshot shows a SQL query editor with the following queries:

- 1 • `select * from hogwarts.characters;`
- 2 • `select fname, lname, patronus from hogwarts.characters where patronus is not null or patronus="known";`
- 3 • `select lname from hogwarts.characters where lname like "%e";`
- 4 • `select SUM(age) from hogwarts.characters;`
- 5

The result grid shows the following data:

SUM(age)
257

4. Выведите имя, фамилию и возраст персонажей по убыванию их возраста

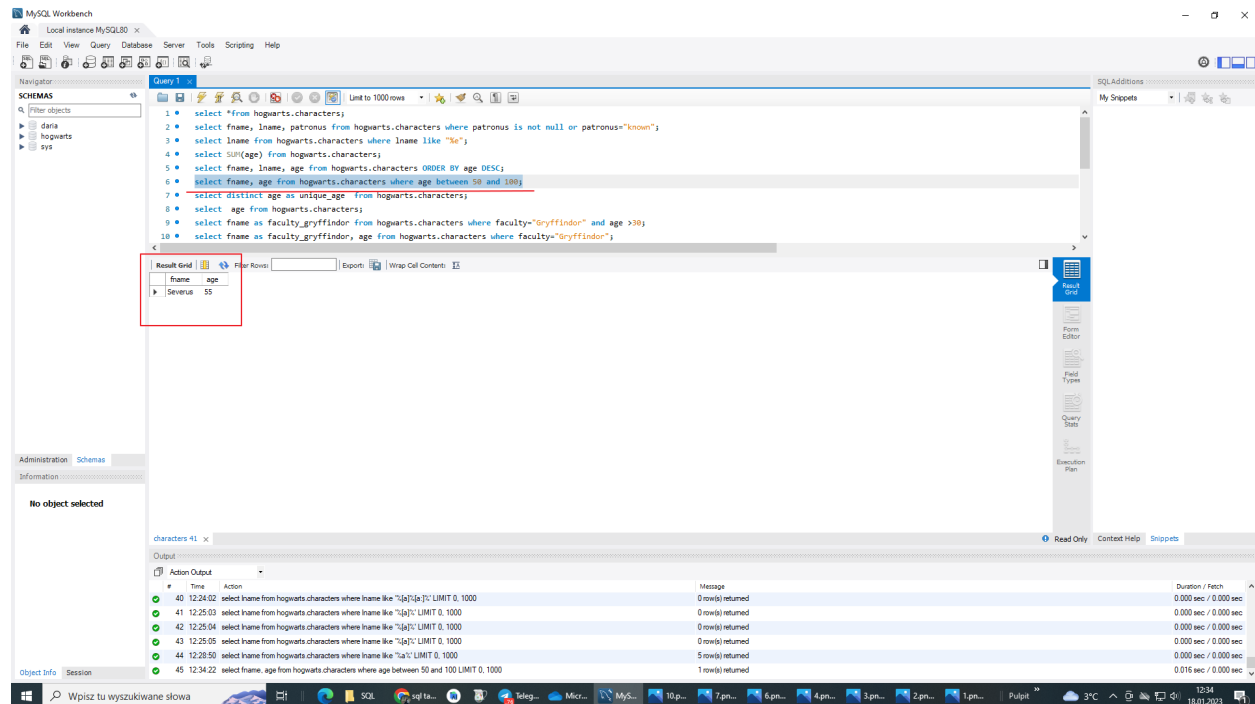
The screenshot shows a SQL query editor with the following queries:

- 1 • `select * from hogwarts.characters;`
- 2 • `select fname, lname, patronus from hogwarts.characters where patronus is not null or patronus="known";`
- 3 • `select lname from hogwarts.characters where lname like "%e";`
- 4 • `select SUM(age) from hogwarts.characters;`
- 5 • `select fname, lname, age from hogwarts.characters ORDER BY age DESC;`

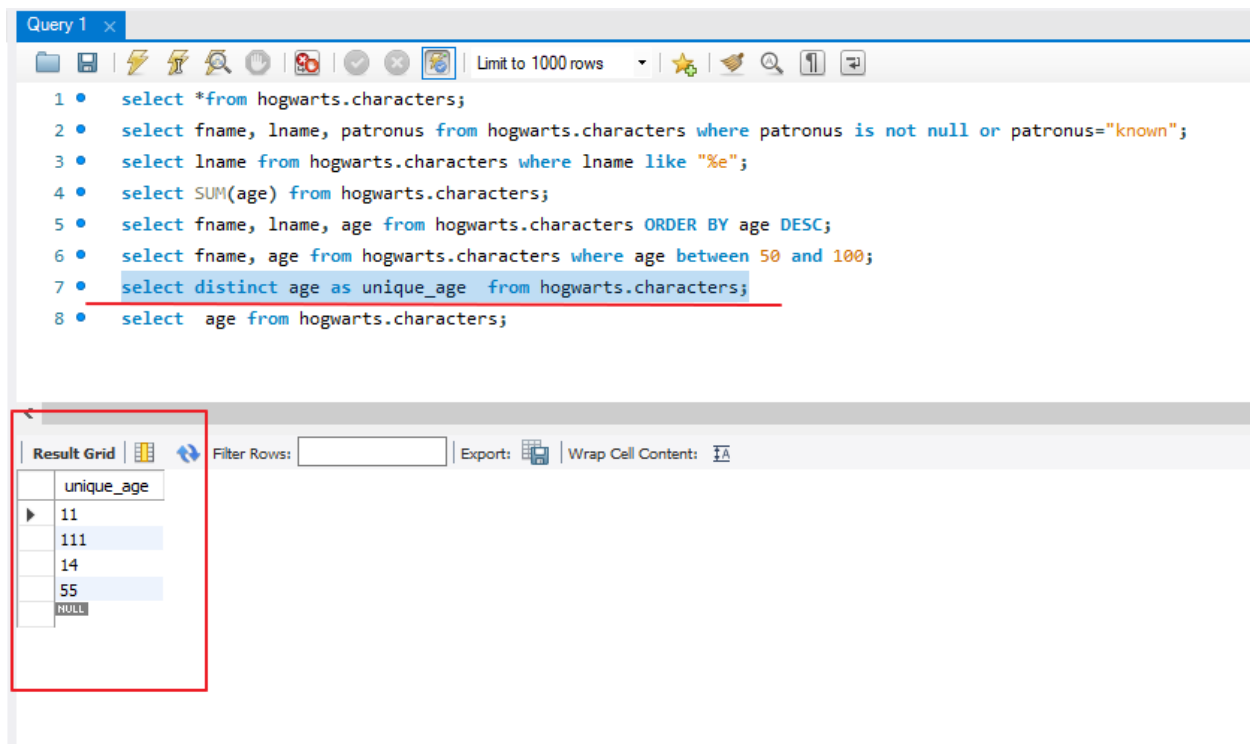
The result grid shows the following data:

fname	lname	age
Albus	Dumbledore	111
Severus	Snape	55
Cedric	Diggory	14
Harry	Potter	11
Hermione	Granger	11
Ron	Weasley	11
Draco	Malfoy	11
Vincent	Crabbe	11
Gregory	Goyle	11
Luna	Lovegood	11
Lord	Voldemort	NULL

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



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



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

Query 1

Limit to 1000 rows

```
1 • select *from hogwarts.characters;
2 • select fname, lname, patronus from hogwarts.characters where patronus is not null or patronus="known";
3 • select lname from hogwarts.characters where lname like "%e";
4 • select SUM(age) from hogwarts.characters;
5 • select fname, lname, age from hogwarts.characters ORDER BY age DESC;
6 • select fname, age from hogwarts.characters where age between 50 and 100;
7 • select distinct age as unique_age from hogwarts.characters;
8 • select age from hogwarts.characters;
9 • select fname as faculty_gryffindor from hogwarts.characters where faculty="Gryffindor" and age >30;
10 • select fname as faculty_gryffindor, age from hogwarts.characters where faculty="Gryffindor";
```

Result Grid

faculty_gryffindor
Albus

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

Query 1

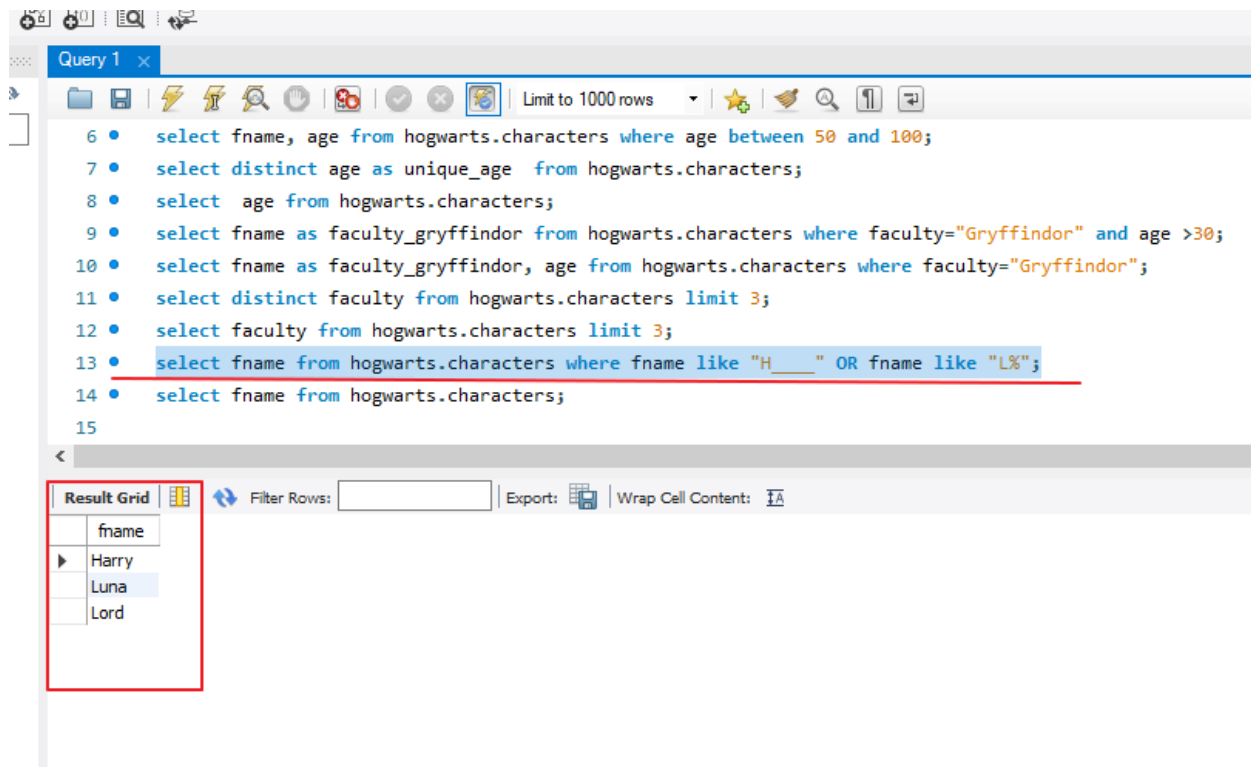
Limit to 1000 rows

```
3 • select lname from hogwarts.characters where lname like "%e";
4 • select SUM(age) from hogwarts.characters;
5 • select fname, lname, age from hogwarts.characters ORDER BY age DESC;
6 • select fname, age from hogwarts.characters where age between 50 and 100;
7 • select distinct age as unique_age from hogwarts.characters;
8 • select age from hogwarts.characters;
9 • select fname as faculty_gryffindor from hogwarts.characters where faculty="Gryffindor" and age >30;
10 • select fname as faculty_gryffindor, age from hogwarts.characters where faculty="Gryffindor";
11 • select distinct faculty from hogwarts.characters limit 3;
12 • select faculty from hogwarts.characters limit 3;
```

Result Grid

faculty
Gryffindor
Slytherin
Ravendaw

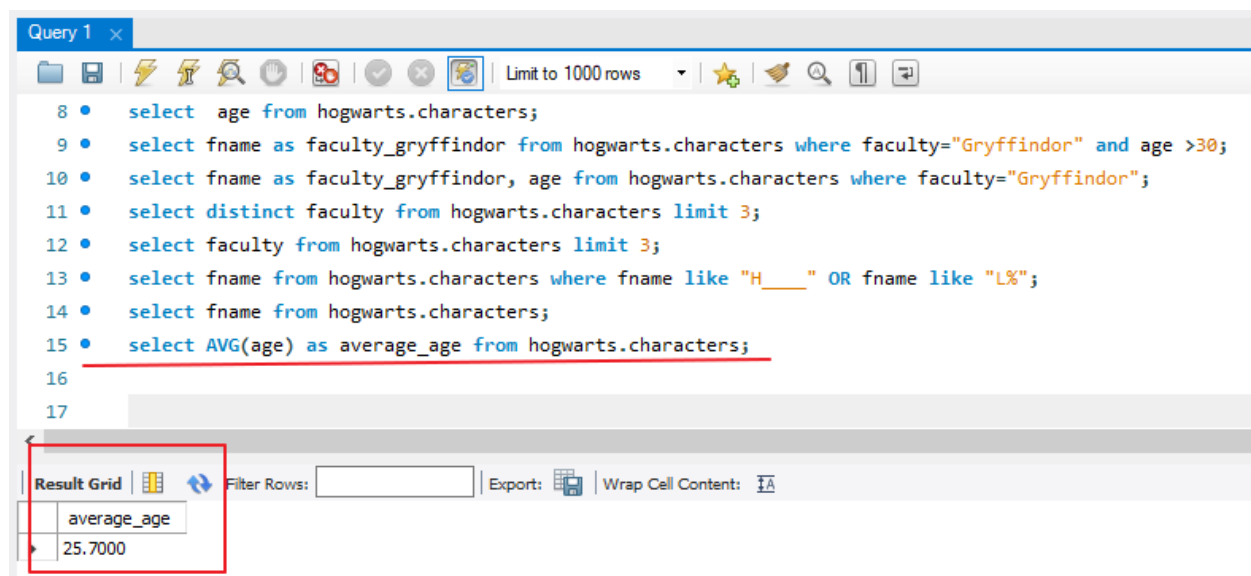
9. Выведите имена всех персонажей, у которых имя начинается с 'H' и состоит из 5 букв, или чье имя начинается с 'L'



```
6 • select fname, age from hogwarts.characters where age between 50 and 100;
7 • select distinct age as unique_age from hogwarts.characters;
8 • select age from hogwarts.characters;
9 • select fname as faculty_gryffindor from hogwarts.characters where faculty="Gryffindor" and age >30;
10 • select fname as faculty_gryffindor, age from hogwarts.characters where faculty="Gryffindor";
11 • select distinct faculty from hogwarts.characters limit 3;
12 • select faculty from hogwarts.characters limit 3;
13 • select fname from hogwarts.characters where fname like "H____" OR fname like "L%";
14 • select fname from hogwarts.characters;
15
```

fname
Harry
Luna
Lord

10. Посчитайте средний возраст всех персонажей



```
8 • select age from hogwarts.characters;
9 • select fname as faculty_gryffindor from hogwarts.characters where faculty="Gryffindor" and age >30;
10 • select fname as faculty_gryffindor, age from hogwarts.characters where faculty="Gryffindor";
11 • select distinct faculty from hogwarts.characters limit 3;
12 • select faculty from hogwarts.characters limit 3;
13 • select fname from hogwarts.characters where fname like "H____" OR fname like "L%";
14 • select fname from hogwarts.characters;
15 • select AVG(age) as average_age from hogwarts.characters;
16
17
```

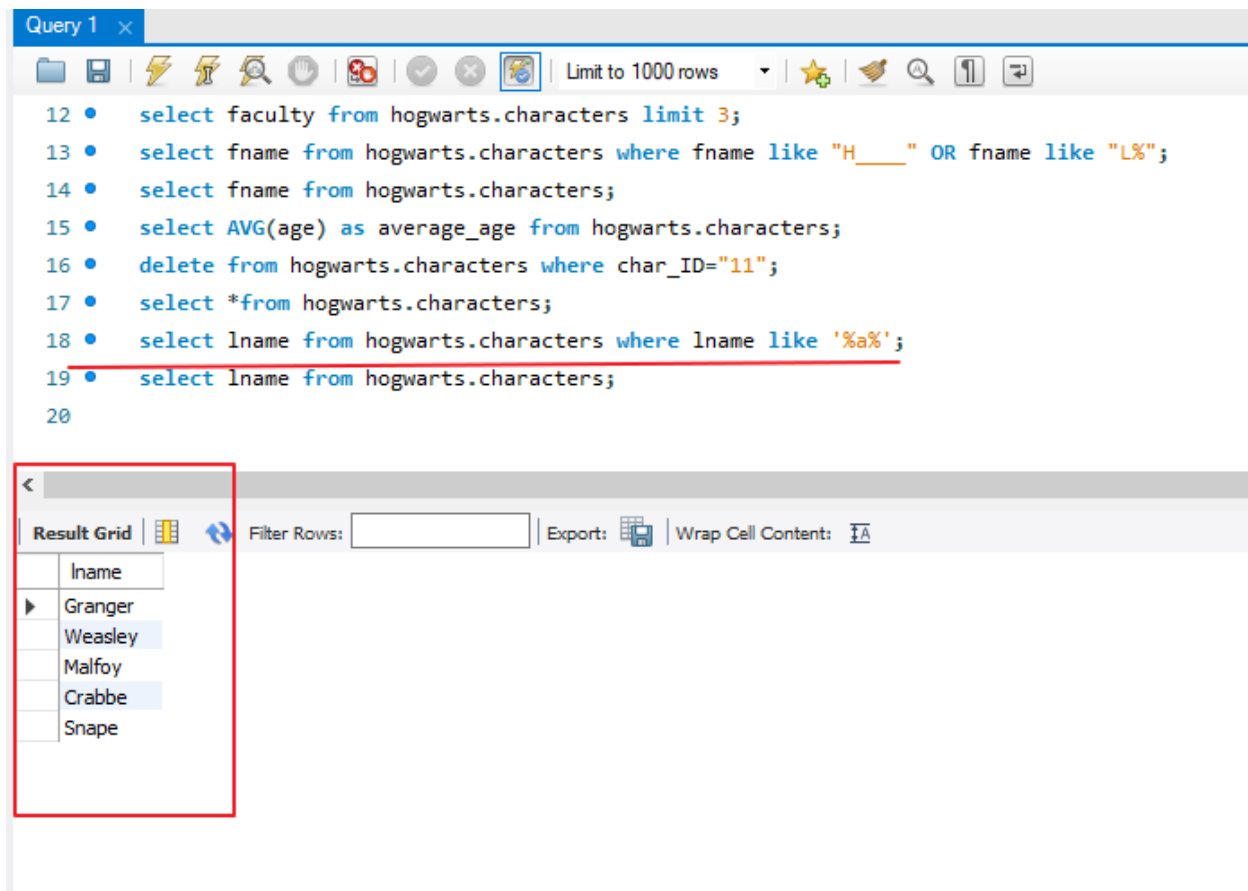
average_age
25.7000

11. Удалите персонажа с ID = 11

```
14 • select fname from hogwarts.characters;
15 • select AVG(age) as average_age from hogwarts.characters;
16 • delete from hogwarts.characters where char_ID="11";
17 • select *from hogwarts.characters;
18
```

Result Grid							
		Filter Rows:		Edit:		Export/Import:	
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	Ravendaw	Hare	2	
9	Cedric	Diggory	14	Hufflepuff	Unknown	3	
10	Severus	Snape	55	Slytherin	Doe	4	
*	NULL	NULL	NULL	NULL	NULL	NULL	

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

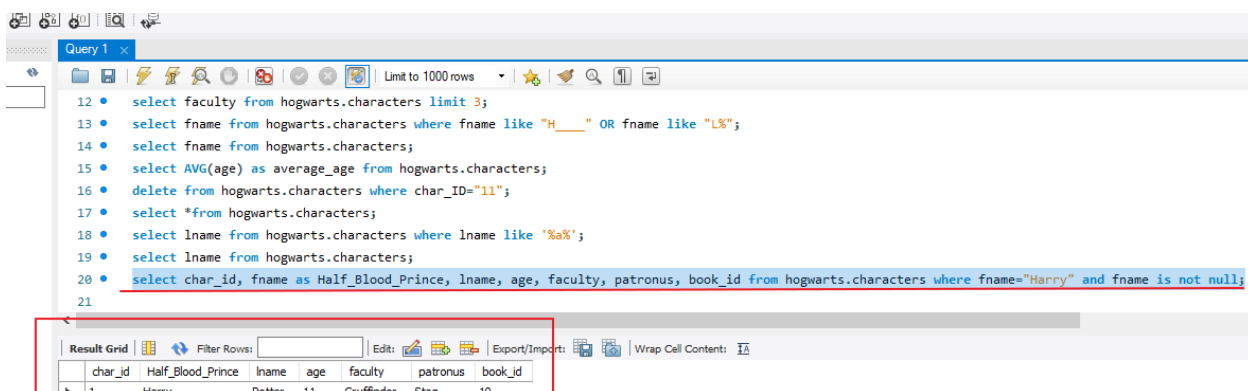


```
12 • select faculty from hogwarts.characters limit 3;
13 • select fname from hogwarts.characters where fname like "H____" OR fname like "L%";
14 • select fname from hogwarts.characters;
15 • select AVG(age) as average_age from hogwarts.characters;
16 • delete from hogwarts.characters where char_ID="11";
17 • select *from hogwarts.characters;
18 • select lname from hogwarts.characters where lname like '%a%';
19 • select lname from hogwarts.characters;
20
```

Result Grid

	lname
▶	Granger
	Weasley
	Malfoy
	Crabbe
	Snape

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



```
12 • select faculty from hogwarts.characters limit 3;
13 • select fname from hogwarts.characters where fname like "H____" OR fname like "L%";
14 • select fname from hogwarts.characters;
15 • select AVG(age) as average_age from hogwarts.characters;
16 • delete from hogwarts.characters where char_ID="11";
17 • select *from hogwarts.characters;
18 • select lname from hogwarts.characters where lname like '%a%';
19 • select lname from hogwarts.characters;
20 • select char_id, fname as Half_Blood_Prince, lname, age, faculty, patronus, book_id from hogwarts.characters where fname="Harry" and fname is not null;
21
```

Result Grid

	char_id	Half_Blood_Prince	lname	age	faculty	patronus	book_id
▶	1	Harry	Potter	11	Gryffindor	Stag	10

14. Выведите id и имена всех патронусов в алфавитном порядке, при условии что они есть или известны

The screenshot shows a SQL query editor with a list of 21 queries. The 21st query is highlighted: `select char_id as ID, patronus as name_of_potronus from hogwarts.characters where patronus is not null and patronus not like "Unknown";`. Below the queries, a result grid is displayed with the following data:

ID	name_of_potronus
1	Stag
2	Otter
3	Jack Russell terrier
7	Phoenix
8	Hare
10	Doe

15. Используя оператор IN, выведите имя и фамилию тех персонажей, у которых фамилия Crabbe, Granger или Diggory

The screenshot shows a SQL query editor with a list of 25 queries. The 22nd query is highlighted: `select lname, fname from hogwarts.characters where lname IN("Crabbe", "Granger", "Diggory");`. Below the queries, a result grid is displayed with the following data:

lname	fname
Granger	Hermione
Crabbe	Vincent
Diggory	Cedric



## 16. Выведите минимальный возраст персонажа

Query 1

```
15 • select AVG(age) as average_age from hogwarts.characters;  
16 • delete from hogwarts.characters where char_ID="11";  
17 • select *from hogwarts.characters;  
18 • select lname from hogwarts.characters where lname like '%a%';  
19 • select lname from hogwarts.characters;  
20 • select char_id, fname as Half_Blood_Prince, lname, age, faculty, patronus, book_id from hogwarts.characters where fname="Harry" and fname is not null;  
21 • select char_id as ID, patronus as name_of_potronus from hogwarts.characters where patronus is not null and patronus not like "Unknown";  
22 • select lname, fname from hogwarts.characters where lname IN("Crabbe", "Granger", "Diggory");  
23 • select MIN(age) as min_age from hogwarts.characters;  
24
```

Result Grid

min_age
11

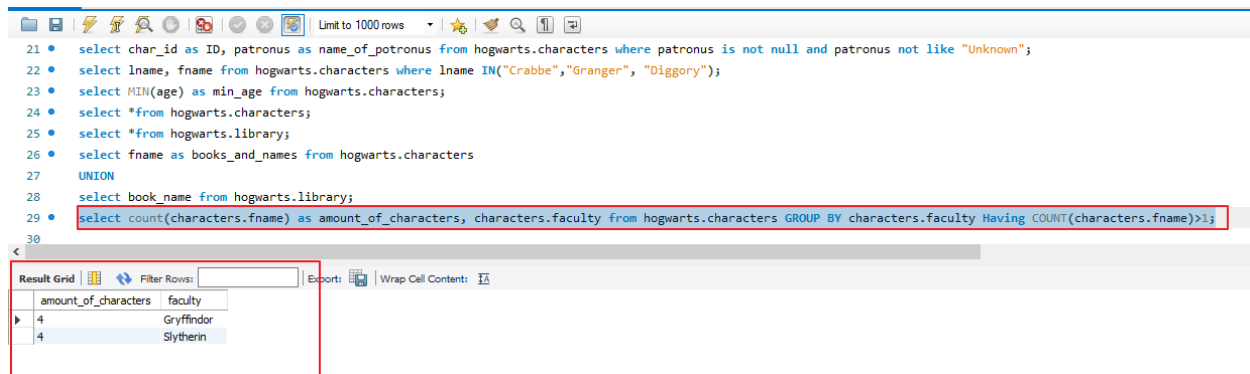
## 17. Используя оператор UNION выберите имена из таблицы characters и названия книг из таблицы library

```
24 • select *from hogwarts.characters;  
25 • select *from hogwarts.library;  
26 • select fname as books_and_names from hogwarts.characters  
27 UNION  
28 select book_name from hogwarts.library;  
29
```

Result Grid

books_and_names
Harry
Hermione
Ron
Draco
Vincent
Gregory
Albus
Luna
Cedric
Severus
Hogwarts: A History
Quidditch Throug...
The Lockhart Coll...
Moste Potente Po...
The Life And Lies ...
Fantastic Beasts ...
The Tales Of Bea...
Advanced Potion-...
A History Of Magic
Magical Water Pla...

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



```
21 • select char_id as ID, patronus as name_of_potronus from hogwarts.characters where patronus is not null and patronus not like "Unknown";
22 • select lname, fname from hogwarts.characters where lname IN("Crabbe", "Granger", "Diggory");
23 • select MIN(age) as min_age from hogwarts.characters;
24 • select *from hogwarts.characters;
25 • select *from hogwarts.library;
26 • select fname as books_and_names from hogwarts.characters
27 UNION
28 select book_name from hogwarts.library;
29 • select count(characters.fname) as amount_of_characters, characters.faculty from hogwarts.characters GROUP BY characters.faculty Having COUNT(characters.fname)>1;
30
```

Result Grid

amount_of_characters	faculty
4	Gryffindor
4	Slytherin

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

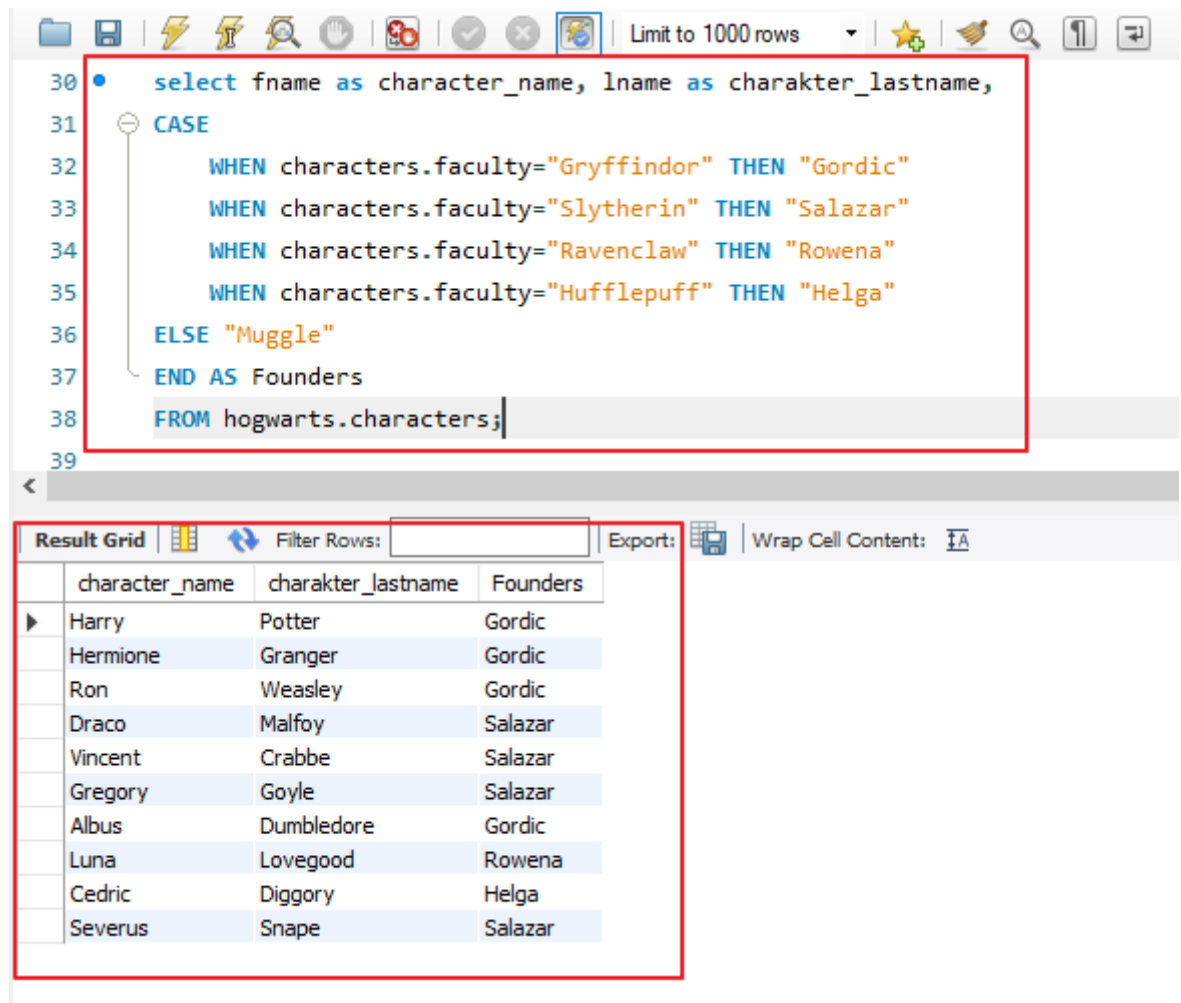
Если факультет Slytherin, то в консоли должно вывестись Salazar

Если факультет Ravenclaw, то в консоли должно вывестись Rowena

Если факультет Hufflepuff, то в консоли должно вывестись Helga

Если другая информация, то выводится Muggle

Для сообщения используйте псевдоним Founders



The screenshot shows a SQL query editor with a query that selects character names and last names from a table named 'characters', and assigns a founder name based on the house. The query is as follows:

```
30 • select fname as character_name, lname as charakter_lastname,  
31 CASE  
32     WHEN characters.faculty="Gryffindor" THEN "Gordic"  
33     WHEN characters.faculty="Slytherin" THEN "Salazar"  
34     WHEN characters.faculty="Ravenclaw" THEN "Rowena"  
35     WHEN characters.faculty="Hufflepuff" THEN "Helga"  
36     ELSE "Muggle"  
37 END AS Founders  
38 FROM hogwarts.characters;  
39
```

Below the query editor is a 'Result Grid' showing the results of the query. The grid has four columns: 'character\_name', 'charakter\_lastname', and 'Founders'. The results are as follows:

	character_name	charakter_lastname	Founders
▶	Harry	Potter	Gordic
	Hermione	Granger	Gordic
	Ron	Weasley	Gordic
	Draco	Malfoy	Salazar
	Vincent	Crabbe	Salazar
	Gregory	Goyle	Salazar
	Albus	Dumbledore	Gordic
	Luna	Lovegood	Rowena
	Cedric	Diggory	Helga
	Severus	Snape	Salazar

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

Выведите их

Query 1

Limit to 1000 rows

```
34     WHEN characters.faculty="Ravenclaw" THEN "Rowena"
35     WHEN characters.faculty="Hufflepuff" THEN "Helga"
36     ELSE "Muggle"
37   END AS Founders
38   FROM hogwarts.characters;
39 • select lname from hogwarts.characters where lname not regexp "^[L,P,S]";
40
41
42
43
```

Result Grid

	lname
▶	Granger
	Weasley
	Malfoy
	Crabbe
	Goyle
	Dumbledore
	Diggory