

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use howards;
2 SELECT * FROM characters;
3 select fname, lname, patronus from characters where not patronus='unknown' or patronus is not null;
```

The Results window displays the following data:

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	Do

The Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
129	00:02:32	select lname from characters where not lname regexp "[HLS]" LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec
130	00:05:24	select fname, lname, patronus from characters where not patronus='unknown' or patronus is not null LIMIT 0, 1...	7 row(s) returned	0.016 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use howards;
2 SELECT * FROM characters;
3 SELECT lname FROM characters where lname like '%e';
```

The Results window displays the following data:

lname
Crabbe
Goyle
Dumbledore
Snape

The Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
1	20:34:29	use howards	0 row(s) affected	0.000 sec
2	20:34:32	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
3	20:34:38	SELECT fname, lname, patronus FROM characters where patronus is not null LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec
4	20:39:57	SELECT lname FROM characters where lname like '%e' LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use howards;
2 SELECT * FROM characters;
3 SELECT sum(age) FROM characters;
```

The result grid shows the output of the third query:

sum(age)
257

The output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
5	20:43:16	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
6	20:43:38	SELECT sum age FROM characters LIMIT 0, 1000	Error Code: 1054. Unknown column 'sum' in 'field list'	0.000 sec
7	20:44:00	SELECT sum (age) FROM characters LIMIT 0, 1000	Error Code: 1630. FUNCTION howards.sum does not exist. Check the 'Function Name Parsing and Resolutio...	0.000 sec
8	20:44:51	SELECT sum(age) FROM characters LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use howards;
2 SELECT * FROM characters;
3 SELECT fname, lname, age FROM characters order by age desc;
```

The result grid shows the output of the third query:

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	100

The output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
8	20:44:51	SELECT sum(age) FROM characters LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
9	20:46:51	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
10	20:48:16	SELECT fname, lname, age FROM characters order by age LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
11	20:48:26	SELECT fname, lname, age FROM characters order by age desc LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 SELECT fname, age FROM characters where age between 50 and 100;
```

The Results window displays the following data:

fname	age
Severus	55

The Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
13	20:51:23	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
14	20:51:32	use hogswards	0 row(s) affected	0.000 sec
15	20:51:32	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
16	20:51:32	SELECT fname, age FROM characters where age between 50 and 100 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 SELECT distinct age FROM characters;
```

The Results window displays the following data:

age
11
111
14
55
1000

The Output window shows the execution log:

#	Time	Action	Message	Duration / Fetch
16	20:51:32	SELECT fname, age FROM characters where age between 50 and 100 LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
17	20:54:15	use hogswards	0 row(s) affected	0.000 sec
18	20:54:15	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
19	20:54:15	SELECT distinct age FROM characters LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 SELECT * FROM characters where faculty='Gryffindor' and age>'30';
```

The Results grid displays the following data:

char_id	fname	lname	age	faculty	patronus	book_id
7	Albus	Dumbledore	111	Gryffindor	Phoenix	

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
23	21:00:43	SELECT * FROM characters where faculty='Gryffindor', age>'30'	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds to your MySQL se...	0.000 sec
24	21:00:47	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
25	21:01:40	SELECT * FROM characters where faculty='Gryffindor' LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
26	21:02:17	SELECT * FROM characters where faculty='Gryffindor' and age>'30' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 SELECT distinct faculty FROM characters limit 3;
```

The Results grid displays the following data:

faculty
Gryffindor
Slytherin
Ravenclaw

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
27	21:10:54	SELECT distinct faculty FROM characters LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
28	21:11:26	SELECT distinct faculty FROM characters limit 3	3 row(s) returned	0.000 sec / 0.000 sec
29	21:11:35	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
30	21:11:43	SELECT distinct faculty FROM characters limit 3	3 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 SELECT fname FROM characters where fname like 'H____' or fname like 'L____';
```

The Result Grid displays the following data:

fname
Harry
Luna
Lord

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
41	21:18:53	SELECT fname FROM characters where fname like 'H____' or 'L____' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
42	21:19:19	SELECT fname FROM characters where fname like 'L____' or 'H____' LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
43	21:20:36	SELECT fname FROM characters where fname like 'H____' or 'L____' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
44	21:26:10	SELECT fname FROM characters where fname like 'H____' or fname like 'L____' LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

10. Посчитайте средний возраст всех персонажей

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 SELECT avg(age) FROM characters;
```

The Result Grid displays the following data:

avg(age)
25.7000

The Output pane shows the execution log:

#	Time	Action	Message	Duration / Fetch
46	21:41:58	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
47	21:43:12	SELECT avg age FROM characters where age is not null LIMIT 0, 1000	Error Code: 1054. Unknown column 'avg' in 'field list'	0.000 sec
48	21:43:41	SELECT avg(age) FROM characters where age is not null LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
49	21:43:52	SELECT avg(age) FROM characters LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

11. Удалите персонажа с ID = 11

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 • use howards;  
2 • SELECT * FROM characters;  
3 • delete FROM characters where char_id=11;  
4 • SELECT * FROM characters;
```

The 'Result Grid' displays the 'characters' table with 10 rows. The columns are: char_id, fname, lname, age, faculty, patronus, book_id. The data is as follows:

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	None	6
5	Vincent	Crabbe	11	Slytherin	None	6
6	Gregory	Goyle	11	Slytherin	None	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

The 'Output' pane shows the execution results of the queries:

#	Time	Action	Message	Duration / Fetch
49	21:43:52	SELECT avg(age) FROM characters LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
50	21:44:55	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
51	21:45:55	delete FROM characters where char_id=11	1 row(s) affected	0.000 sec
52	21:46:09	SELECT * FROM characters LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 • use howards;  
2 • SELECT * FROM characters;  
3 • select lname from characters where lname like "%a%";
```

The 'Result Grid' displays the 'lname' column for characters whose surnames contain the letter 'a'. The data is as follows:

lname
Granger
Weasley
Malfoy
Crabbe
Snape

The 'Output' pane shows the execution results of the queries:

#	Time	Action	Message	Duration / Fetch
53	21:49:40	select lname from characters where lname like "%a%" LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
54	21:49:47	SELECT * FROM characters LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec
55	21:50:06	select lname from characters where lname like "%p%" LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
56	21:50:15	select lname from characters where lname like "%a%" LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 select char_id, fname as 'Half-Blood Prince', lname, age, faculty, patronus, book_id from characters where fname='Severus';
```

The result grid shows the following data:

char_id	fname	lname	age	faculty	patronus	book_id
81	Half-Blood Prince	Severus	17	DADA	Stag	1

The output pane shows the following messages:

- 81 22:16:16 select char_id, fname as 'Half-Blood Prince', lname, age, faculty, patronus, book_id from characters where fname='Severus'; 1 row(s) returned
- 82 22:18:32 SELECT * FROM characters where fname in (select fname as 'Half-Blood Prince' from characters where fname='Severus'); 1 row(s) returned
- 83 22:20:11 SELECT * FROM characters where fname in (select fname as 'Half-Blood Prince' from characters) LIMIT 0, 1000 10 row(s) returned
- 84 22:20:57 SELECT * FROM characters where fname='Severus' in (select fname as 'Half-Blood Prince' from characters) LIMIT 0, 1000 0 row(s) returned

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 select char_id, patronus from characters where not patronus='unknown' or patronus is not null order by patronus ASC;
```

The result grid shows the following data:

char_id	patronus
10	Doe
8	Hare
3	Jack Russell terrier
2	Otter
7	Phoenix
1	Stag
9	Unknown

The output pane shows the following messages:

- 101 22:56:24 select char_id, patronus from characters where patronus is not null order by patronus LIMIT 0, 1000 7 row(s) returned
- 102 22:57:03 select char_id, patronus from characters where not patronus='unknown' is not null order by patronus LIMIT 0, 1... 3 row(s) returned
- 103 22:57:29 select char_id, patronus from characters where not patronus='unknown' or patronus is not null order by patronu... 7 row(s) returned
- 104 22:59:46 select char_id, patronus from characters where not patronus='unknown' or patronus is not null order by patronu... 7 row(s) returned

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

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 select fname, lname from characters where lname in ('Crabbe', 'Granger', 'Digory');
```

The Result Grid displays the following data:

fname	lname
Hermione	Granger
Vincent	Crabbe
Cedric	Diggory

The Output pane shows the execution log with the following messages:

#	Time	Action	Message	Duration / Fetch
104	22:59:46	select char_id, patronus from characters where not patronus=unknown or patronus is not null order by patronu...	7 row(s) returned	0.000 sec / 0.000 sec
105	23:04:22	SELECT * FROM characters LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec
106	23:08:53	select fname, lname from characters where lname in (select lname='Crabbe' or lname='Granger' or lname='Digg...	7 row(s) returned	0.000 sec / 0.000 sec
107	23:10:27	select fname, lname from characters where lname in ('Crabbe', 'Granger', 'Digory') LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

16.Выведите минимальный возраст персонажа

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following query:

```
1 use hogswards;
2 SELECT * FROM characters;
3 select min(age) from characters;
```

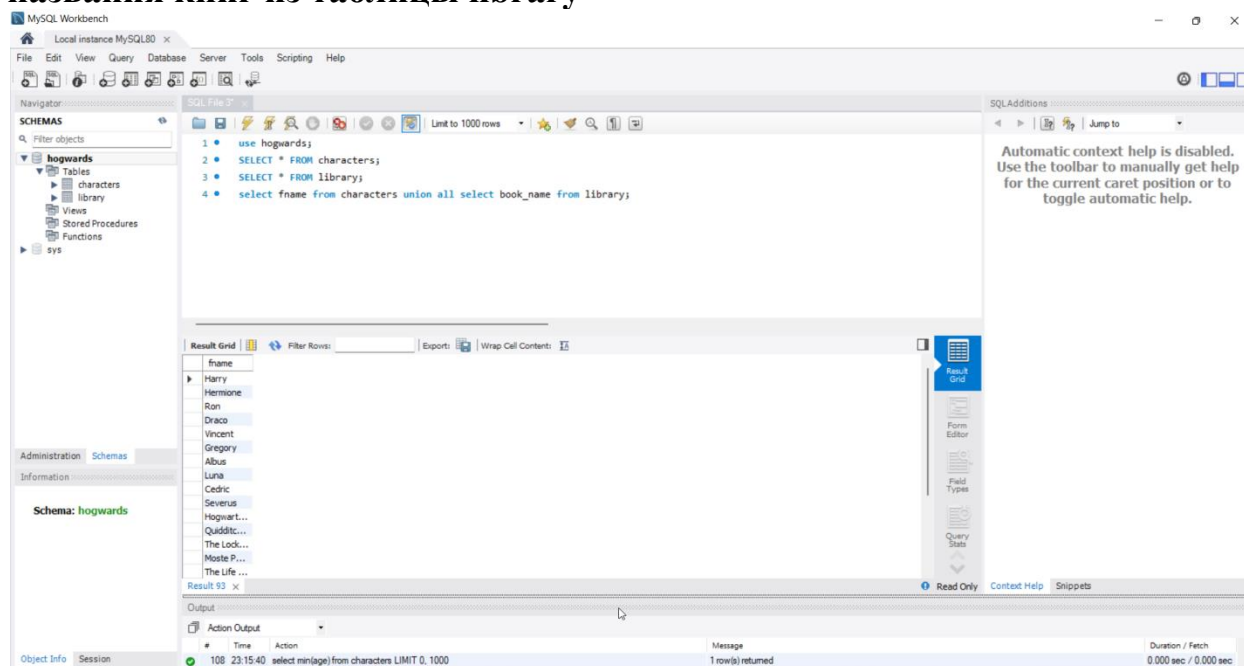
The Result Grid displays the following data:

min(age)
11

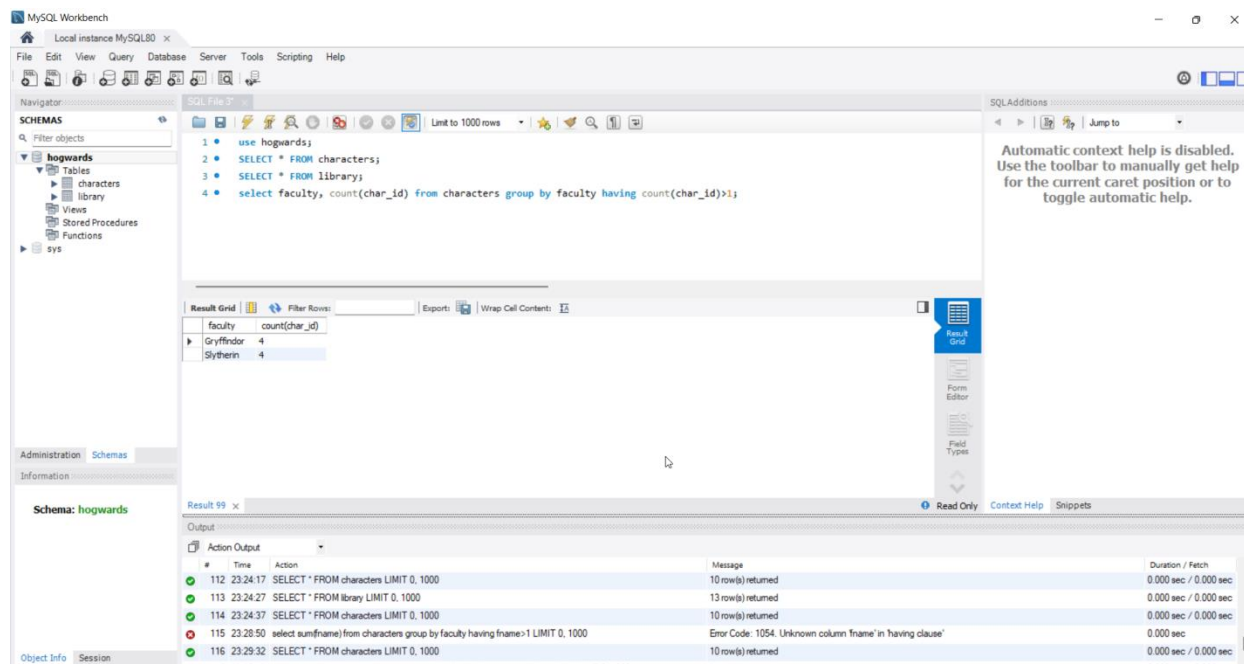
The Output pane shows the execution log with the following messages:

#	Time	Action	Message	Duration / Fetch
105	23:04:22	SELECT * FROM characters LIMIT 0, 1000	10 row(s) returned	0.000 sec / 0.000 sec
106	23:08:53	select fname, lname from characters where lname in (select lname='Crabbe' or lname='Granger' or lname='Digg...	7 row(s) returned	0.000 sec / 0.000 sec
107	23:10:27	select fname, lname from characters where lname in ('Crabbe', 'Granger', 'Digory') LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
108	23:15:40	select min(age) from characters LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

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



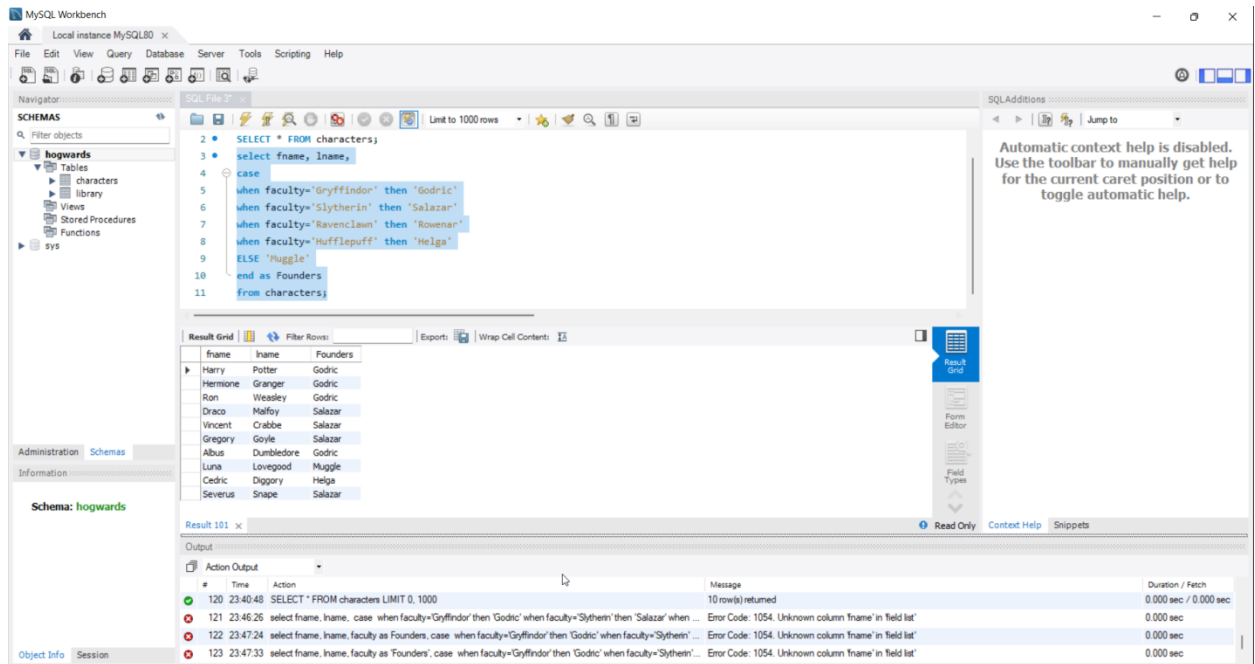
18. Используя оператор **HAVING** посчитайте количество персонажей на каждом факультете, оставив только те факультеты, где количество студентов больше 1



19. Используя оператор CASE опишите следующую логику:

Выведите имя и фамилию персонажа, а также следующий текстовое сообщение:

1. Если факультет Gryffindor, то в консоли должно выводиться Godric
 2. Если факультет Slytherin, то в консоли должно выводиться Salazar
 3. Если факультет Ravenclaw, то в консоли должно выводиться Rowena
 4. Если факультет Hufflepuff, то в консоли должно выводиться Helga
 5. Если другая информация, то выводится Muggle
- Для сообщения используйте псевдоним Founders



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

