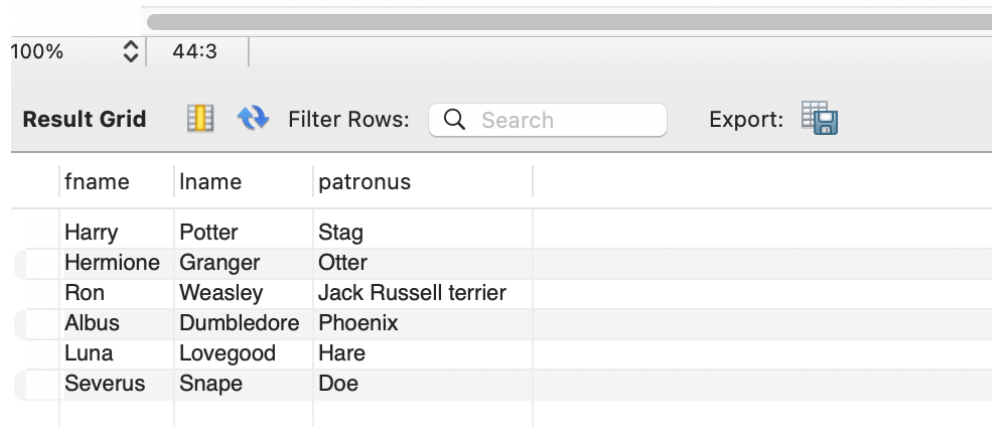


1. Выведите имя, фамилию, патронуса всех персонажей, у которых есть patronus и он известен

```
2 • SELECT fname, lname, patronus FROM characters
3 WHERE patronus IS NOT NULL AND patronus <> 'Unknown';
```



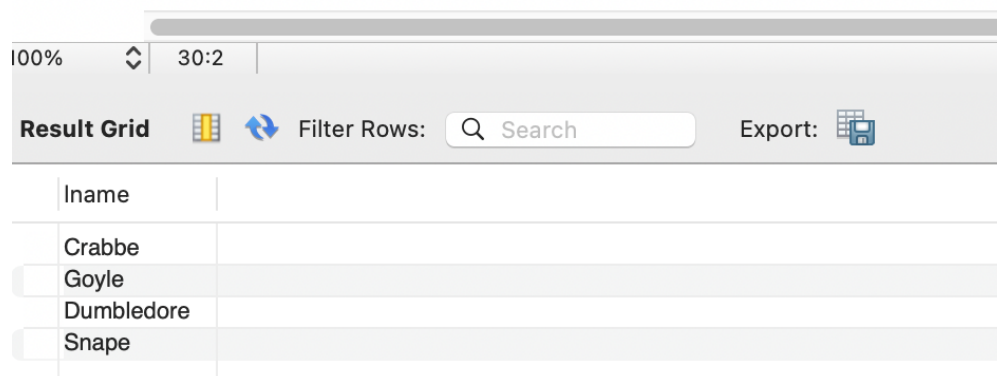
100% 44:3

Result Grid Filter Rows: Search Export:

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

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

```
2 • SELECT lname FROM characters WHERE lname LIKE '%e';
```



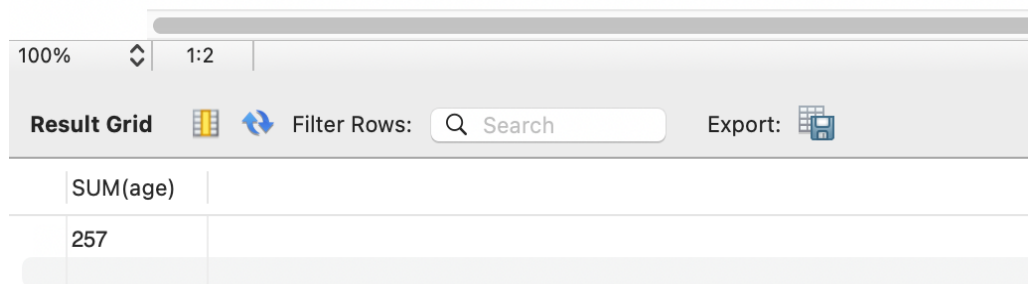
100% 30:2

Result Grid Filter Rows: Search Export:

lname
Crabbe
Goyle
Dumbledore
Snape

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

```
3 • SELECT SUM(age) FROM characters;
```



100% 1:2

Result Grid Filter Rows: Search Export:

SUM(age)
257

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

```
3 • SELECT fname, lname, age FROM characters ORDER BY age DESC;
```

100%	42:3	Result Grid	Filter Rows: Search	Export:
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

```
3 • SELECT fname, age FROM characters WHERE age BETWEEN 50 AND 100;
```

100%	63:3	Result Grid	Filter Rows: Search	Export:
fname	age			
Severus	55			

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

```
3 • SELECT DISTINCT age FROM characters;
```

100%	36:3	Result Grid	Filter Rows: Search	Export:
age				
11				
111				
14				
55				
NULL				

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

3 • **SELECT** * **FROM** characters **WHERE** faculty = 'Gryffindor' **AND** age > 30;

100%

68:3

Result Grid

Filter Rows:

Search

Edit:

Export/Import:

char_id	fname	lname	age	faculty	patronus	book_id	
7	Albus	Dumbledore	111	Gryffindor	Phoenix	2	
<div>NULL</div>	<div>NULL</div>	<div>NULL</div>	<div>NULL</div>	<div>NULL</div>	<div>NULL</div>	<div>NULL</div>	

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

3 • **SELECT DISTINCT** faculty **FROM** characters **LIMIT** 3;

100%

49:3

Result Grid

Filter Rows:

Search

Export:

Fetch rows:

faculty	
Gryffindor	
Slytherin	
Ravenclaw	

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

3 • **SELECT** fname **FROM** characters **WHERE** fname **LIKE** 'H____' **OR** fname **LIKE** 'L%';

100% 53:3

Result Grid

Filter Rows:

Search


Export:

fname	
Harry	
Luna	
Lord	

10. Посчитайте средний возраст всех персонажей


```
3 • SELECT AVG(age) FROM characters;
```


100%




16:3

Result Grid






Filter Rows:



Search

Export:



AVG(age)
25.7000

11. Удалите персонажа с ID = 11

```
1 • SELECT * FROM characters;  
2  
3 • DELETE FROM characters WHERE char_id = 11;
```

100%

43:3

Result Grid

Filter Rows:

Search

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	Ravenclaw	Hare	2
9	Cedric	Diggory	14	Hufflepuff	Unknown	3
10	Severus	Snape	55	Slytherin	Doe	4
NULL	NULL	NULL	NULL	NULL	NULL	NULL

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


```
3 • SELECT lname FROM characters WHERE lname LIKE '%a%';
```


100%




53:3

Result Grid






Filter Rows:



Search

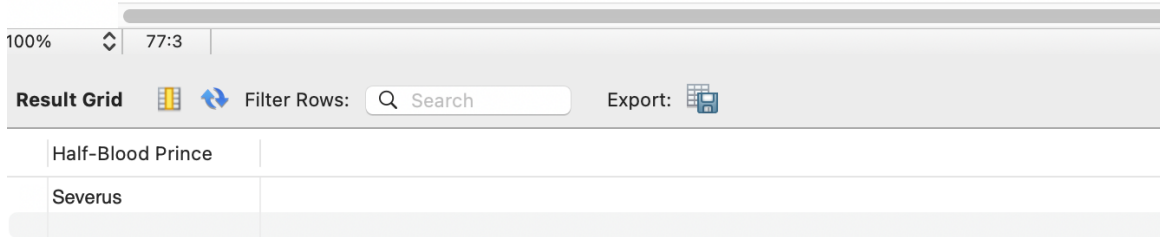
Export:



	Iname	
	Granger	
<input type="checkbox"/>	Weasley	
	Malfoy	
<input type="checkbox"/>	Crabbe	
	Snape	

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

```
3 • SELECT fname AS 'Half-Blood Prince' FROM characters WHERE fname = 'Severus';
```



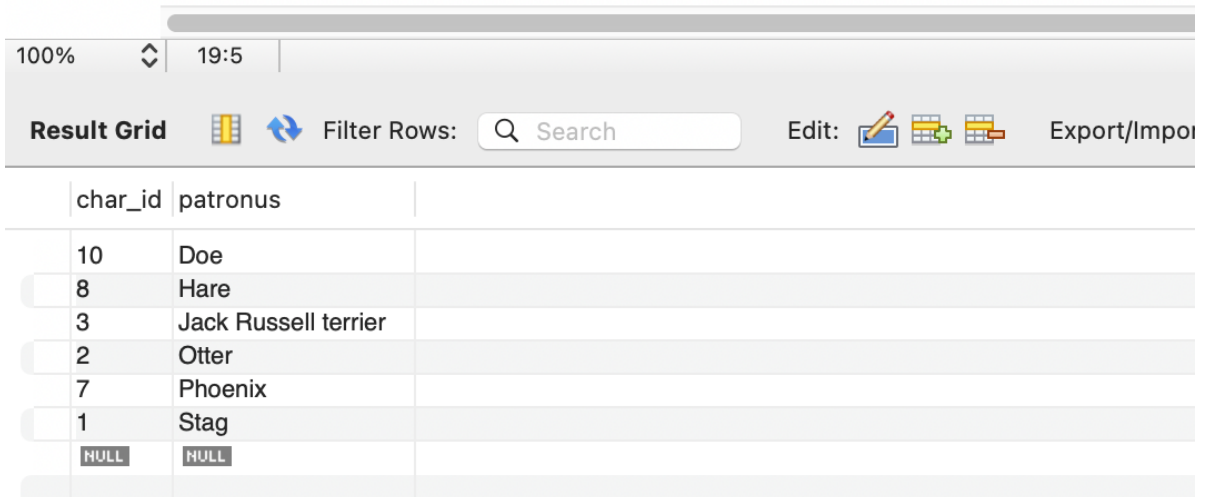
100% 77:3

Result Grid Filter Rows: Search Export:

Half-Blood Prince

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

```
3 • SELECT char_id, patronus FROM characters
4 WHERE patronus IS NOT NULL AND patronus <> 'Unknown'
5 ORDER BY patronus;
```



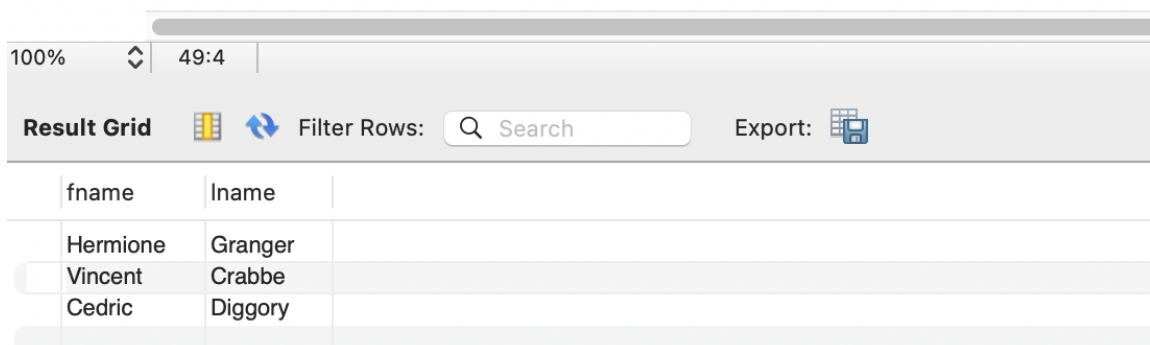
100% 19:5

Result Grid Filter Rows: Search Edit: Export/Import

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

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

```
3 • SELECT fname, lname FROM characters
4 WHERE lname IN ('Crabbe', 'Granger', 'Diggory');
```



100% 49:4

Result Grid Filter Rows: Search Export:

fname	lname
Hermione	Granger
Vincent	Crabbe
Cedric	Diggory

16. Выведите минимальный возраст персонажа

```
3 • SELECT MIN(age) FROM characters;
```

100% 33:3

Result Grid Filter Rows: Search Export:

MIN(age)
11

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

```
3 • SELECT fname FROM characters UNION SELECT book_name FROM library;
```

100% 23:1

Result Grid Filter Rows: Search Export:

fname
Harry
Hermione
Ron
Draco
Vincent
Gregory
Albus
Luna
Cedric
Severus
Hogwarts: A History
Quidditch Through The Ages
The Lockhart Collection
Moste Potente Potions
The Life And Lies Of Albus Dumbledore
Fantastic Beasts And Where To Find Them
The Tales Of Beadle The Bard
Advanced Potion-Making
A History Of Magic
Magical Water Plants Of The Highland Rocks

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

```
3 • SELECT faculty, COUNT(faculty) FROM characters GROUP BY faculty HAVING COUNT(faculty) > 1;
```

100% 91:3

Result Grid Filter Rows: Search Export:

faculty	COUNT(faculty)
Gryffindor	4
Slytherin	4

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

- Если факультет Gryffindor, то в консоли должно выводиться Godric
- Если факультет Slytherin, то в консоли должно выводиться Salazar
- Если факультет Ravenclaw, то в консоли должно выводиться Rowena
- Если факультет Hufflepuff, то в консоли должно выводиться Helga
- Если другая информация, то выводится Muggle

Для сообщения используйте псевдоним Founders

```
3 • SELECT fname, lname,
4 CASE WHEN faculty = 'Gryffindor' THEN 'Godric'
5 WHEN faculty = 'Slytherin' THEN 'Salazar'
6 WHEN faculty = 'Ravenclaw' THEN 'Rowena'
7 WHEN faculty = 'Hufflepuff' THEN 'Helga'
8 ELSE 'Muggle'
9 END AS Founders
10 FROM characters;
```

Result Grid			
Filter Rows: Search			
Export:			
fname	lname	Founders	
Harry	Potter	Godric	
Hermione	Granger	Godric	
Ron	Weasley	Godric	
Draco	Malfoy	Salazar	
Vincent	Crabbe	Salazar	
Gregory	Goyle	Salazar	
Albus	Dumbledore	Godric	
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

```
3 • SELECT lname FROM characters WHERE lname NOT LIKE '[^HSL%]';
```

Result Grid	
Filter Rows: Search	
Export:	
lname	
Potter	
Granger	
Weasley	
Malfoy	
Crabbe	
Goyle	
Dumbledore	
Lovegood	
Diggory	
Snape	