

## Задание 1

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

```
characters* library characters x
1 • select fname, lname, patronus from characters where patronus like "%" and patronus != "Unknown";
2 /* 1)Выведите имя, фамилию, патронуса всех персонажей, у которых есть patronus или он известен
```

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	

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

```
characters* characters x
1 select lname from characters where lname like "%e";
2 /* 2)Выведите фамилию персонажей, у которых последняя буква в фамилии 'е' */
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
lname			
Crabbe			
Goyle			
Dumbledore			
Snape			

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

```
characters* characters x
1 • select sum(age) from characters;
2 /* 3)Посчитайте общий возраст всех персонажей и выведите это на экран */
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
sum(age)			
257			

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

characters\* characters x

Limit to 1000 rows

```
1 • select fname, lname, age from characters order by age desc
2 /* 4)Выведите имя, фамилию и возраст персонажей по убыванию их возраста */
```

Result Grid

	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

characters\* characters x

Limit to 1000 rows

```
1 • select fname, age from characters where age between 50 and 100;
2 /* 5)Выведите имя персонажа и возраст, у которых последний находится в диапазоне от 50 до 100 лет
```

Result Grid

	fname	age
▶	Severus	55

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

characters\* characters x

Limit to 1000 rows

```
1 • select distinct age from characters where age is not null;
2 /* 6)Выведите возраст всех персонажей так, чтобы среди них не было тех, у кого он одинаковый */
```

Result Grid

	age
▶	11
	111
	14
	55

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

characters\* library x

Limit to 1000 rows

```
1 select * from characters where faculty = "Gryffindor" and age > 30;
2 /* 7)Выведите всю информацию о персонажах, у которых faculty = Gryffindor и чей возраст больше 30 лет */
```

Result Grid

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

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

characters\* characters x

Limit to 1000 rows

```
1 select distinct faculty from characters limit 3;
2 /* 8)Выведите имена первых трех факультетов из таблицы, так чтобы факультеты не повторялись */
```

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'

characters\* characters x

Limit to 1000 rows

```
1 select fname from characters where (fname like "H%" and char_length(fname) =5) or fname like "L%";
2 /* 9)Выведите имена всех персонажей, у которых имя начинается с 'H' и состоит из 5 букв, или чье имя начинается с 'L' */
```

Result Grid

fname
Harry
Luna
Lord

10) Посчитайте средний возраст всех персонажей

characters\* characters x

Limit to 1000 rows

```
1 • select avg(age) from characters;  
2 /* 10)Посчитайте средний возраст всех персонажей */
```

Result Grid

	avg(age)
▶	25.7000

11) Удалите персонажа с ID = 11

characters characters x

Limit to 1000 rows

```
1 • select * from hogwarts.characters;  
2 • delete from characters where char_id="11";  
3 /* 11)Удалите персонажа с ID = 11 */
```

Result Grid

	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 2 x

Output

Action Output

#	Time	Action	Message
✓ 1	14:54:33	SELECT * FROM hogwarts.characters LIMIT 0, 1000	11 row(s) returned
✓ 2	14:54:50	delete from characters where char_id="11"	1 row(s) affected
✓ 3	14:54:55	SELECT * FROM hogwarts.characters LIMIT 0, 1000	10 row(s) returned

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

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

```
1 select lname from characters where lname like "%a%";
2 /* 12)Выведите фамилию всех персонажей, которые содержат в ней букву 'a' */
```

Below the query editor, the "Result Grid" is displayed with the following data:

	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

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

```
1 select fname as Half_Blood_Prince from characters where fname="Severus";
2 /* 13)Используйте псевдоним для того, чтобы временно заменить название столбца fname на Half-Blood Prince для реального принца-полукровки */
3
```

Below the query editor, the "Result Grid" is displayed with the following data:

	Half_Blood_Prince
▶	Severus

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

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

```
1 • select char_id, fname from characters where (patronus like "%" and patronus != "Unknown") order by fname;
2 /* 14)Выведите id и имена всех патронусов в алфавитном порядке, при условии что они есть или известны */
```

Below the query editor, the "Result Grid" is displayed with the following data:

	char_id	fname
▶	7	Albus
	1	Harry
	2	Hermione
	8	Luna
	3	Ron
	10	Severus
*	NULL	NULL

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

characters\* characters x

Limit to 1000 rows

```
1 • select fname, lname from characters where lname in ("Crabbe","Granger","Diggory");
2 /* 15)Используя оператор IN, выведите имя и фамилию тех персонажей, у которых фамилия Crabbe, Granger или Diggory */
```

Result Grid

fname	lname
Hermione	Granger
Vincent	Crabbe
Cedric	Diggory

16) Выведите минимальный возраст персонажа

characters\* characters x

Limit to 1000 rows

```
1 • select min(age) from characters;
2 /* 16)Выведите минимальный возраст персонажа */
```

Result Grid

min(age)
11

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

characters\* library x

Limit to 1000 rows

```
1 • select fname from characters union select book_name from library;
2 /* 17)Используя оператор UNION выберите имена из таблицы characters и названия книг из таблицы library */
```

Result Grid

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

characters\* characters1 characters characters x

Limit to 1000 rows

```
1 select distinct count(faculty), faculty from characters group by faculty having count(faculty) > 1;
2 /* 18)Используя оператор HAVING посчитайте количество персонажей на каждом факультете,
3 оставив только те факультеты, где количество студентов больше 1 */
```

Result Grid

count(faculty)	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 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

```
characters characters1 characters x
Limit to 1000 rows

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 info
9   from characters;

10 /* 19)Используя оператор CASE опишите следующую логику:
11   Выведите имя и фамилию персонажа, а также следующий текстовое сообщение:
12   Если факультет Gryffindor, то в консоли должно выводиться Godric
13   Если факультет Slytherin, то в консоли должно выводиться Salazar
14   Если факультет Ravenclaw, то в консоли должно выводиться Rowena
15   Если факультет Hufflepuff, то в консоли должно выводиться Helga
16   Если другая информация, то выводится Muggle
17   Для сообщения используйте псевдоним Founders*/
```

Result Grid			
Filter Rows: <input type="text"/>			
Export:  Wrap Cell Content:			
	fname	lname	info
▶	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
	Lord	Voldemort	Salazar



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

```
characters*  characters1  characters x
Limit to 1000 rows
1 • select lname from characters where not lname regexp "^(H|L|S)";
2 /* 20)Используя регулярное выражение найдите фамилии персонажей,
3 которые не начинаются с букв H, L или S и выведите их */
```

Result Grid	
	lname
▶	Potter
	Granger
	Weasley
	Malfoy
	Crabbe
	Goyle
	Dumbledore
	Diggory
	Voldemort

## Задание 2

1) Выведите имя, фамилию персонажей и название книги, которая на них числится

characters\* characters1\* characters x

Limit to 200 rows

```
1 • select characters.fname, characters.lname, library.book_name from library join characters on library.char_id = characters.char_id;
2 /* 1)Выведите имя, фамилию персонажей и название книги, которая на них числится */
```

Result Grid Filter Rows: Export: Wrap Cell Content:

	fname	lname	book_name
▶	Harry	Potter	Magical Water Plants Of The Highland Rocks
	Hermione	Granger	A History Of Magic
	Ron	Weasley	Advanced Potion-Making
	Draco	Malfoy	Fantastic Beasts And Where To Find Them
	Vincent	Crabbe	Fantastic Beasts And Where To Find Them
	Gregory	Goyle	Hogwarts: A History
	Albus	Dumbledore	Quidditch Through The Ages
	Luna	Lovegood	Quidditch Through The Ages
	Cedric	Diggory	The Lockhart Collection
	Severus	Snape	Moste Potente Potions

2) Выведите имя, фамилию персонажей и название книги, вне зависимости от того, есть ли у них книги или нет

characters\* characters1\* characters x

Limit to 200 rows

```
1 • select characters.fname, characters.lname, library.book_name from library right join characters on library.char_id = characters.char_id;
2 /* 2)Выведите имя, фамилию персонажей и название книги, вне зависимости от того, есть ли у них книги или нет */
```

Result Grid Filter Rows: Export: Wrap Cell Content:

	fname	lname	book_name
▶	Harry	Potter	Magical Water Plants Of The Highland Rocks
	Hermione	Granger	A History Of Magic
	Ron	Weasley	Advanced Potion-Making
	Draco	Malfoy	Fantastic Beasts And Where To Find Them
	Vincent	Crabbe	Fantastic Beasts And Where To Find Them
	Gregory	Goyle	Hogwarts: A History
	Albus	Dumbledore	Quidditch Through The Ages
	Luna	Lovegood	Quidditch Through The Ages
	Cedric	Diggory	The Lockhart Collection
	Severus	Snape	Moste Potente Potions

3) Выведите название книги и имя патронуса, вне зависимости от того, есть ли информация о держателе книги в таблице или нет

characters\* characters1\* characters x

Limit to 200 rows

```
1 • select library.book_name, characters.patronus from library left join characters on library.char_id = characters.char_id;
2 /* 3)Выведите название книги и имя патронуса, вне зависимости от того, есть ли информация о держателе книги в таблице или нет */
```

Result Grid Filter Rows: Export: Wrap Cell Content:

	book_name	patronus
▶	Hogwarts: A History	NULL
	Quidditch Through The Ages	Phoenix
	The Lockhart Collection	Unknown
	Moste Potente Potions	Doe
	The Life And Lies Of Albus Dumbledore	NULL
	Fantastic Beasts And Where To Find Them	NULL
	The Tales Of Beadle The Bard	NULL
	Advanced Potion-Making	Jack Russell terrier
	A History Of Magic	Otter
	Magical Water Plants Of The Highland Rocks	Stag
	Quidditch Through The Ages	Hare
	Magical Water Plants Of The Highland Rocks	NULL
	Fantastic Beasts And Where To Find Them	NULL

- 4) Выведите имя, фамилию, возраст персонажей и название книги, которая на них числится, при условии, что все владельцы книг должны быть старше 15 лет

characters\* characters1\* characters x

Limit to 200 rows

```
1 • select characters.lname, characters.lname, characters.age, library.book_name
2   from library join characters on library.char_id = characters.char_id where age > 15;
3   /* 4)Выведите имя, фамилию, возраст персонажей и название книги,
4   которая на них числится, при условии, что все владельцы книг должны быть старше 15 лет */
```

Result Grid

	lname	lname	age	book_name
▶	Dumbledore	Dumbledore	111	Quidditch Through The Ages
	Snape	Snape	55	Moste Potente Potions

- 5) Выведите имя персонажа, название книги, дату выдачи и дату завершения, при условии, что он младше 15 лет и его патронус неизвестен

characters\* characters1\* characters x

Limit to 200 rows

```
1 • select characters.fname, library.book_name, library.start_date, library.end_date
2   from library join characters on library.char_id = characters.char_id
3   where age < 15 and patronus is null or patronus = "Unknown";
4   /* 5)Выведите имя персонажа, название книги, дату выдачи и дату завершения, при условии,
5   что он младше 15 лет и его патронус неизвестен */
```

Result Grid

	fname	book_name	start_date	end_date
▶	Draco	Fantastic Beasts And Where To Find Them	2010-10-20	2020-10-20
	Vincent	Fantastic Beasts And Where To Find Them	2006-06-20	2010-00-20
	Gregory	Hogwarts: A History	2010-10-20	2020-10-20
	Cedric	The Lockhart Collection	2015-12-20	2030-12-20

- 6) Используя вложенный запрос количество книг, у которых end\_date больше, чем end\_date у Hermione

characters\* characters1\* characters x

Limit to 1000 rows









```
1 • select count(book_name) from library where date(end_date) >
2   (select library.end_date from characters join library on library.char_id = characters.char_id where fname = "Hermione");
3   /* 6)Используя вложенный запрос количество книг, у которых end_date больше, чем end_date у Hermione */
4
5
```

Result Grid






	count(book_name)
▶	2

- 7) С помощью вложенного запроса выведите имена всех патронусов, у которых владельцы старше возраста персонажа, у которого патронус Unknown

characterscharacters1characters x



Limit to 1000 rows



1 •



```
select patronus from characters where age > (select age from characters where patronus = "Unknown");
```


2 /\* 7)С помощью вложенного запроса выведите имена всех патронусов,


3 у которых владельцы старше возраста персонажа, у которого патронус Unknown \*/

<

Result Grid

 Filter Rows:

Export: 

Wrap Cell Content: 

	patronus
▶	Phoenix
	Doe