

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT fname, lname, patronus  
5 FROM characters  
6 WHERE patronus IS NOT null AND NOT patronus = 'Unknown'
```

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

Result
GridForm
EditorField
TypesQuery
Stats

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



Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT lname  
5 FROM characters  
6 WHERE lname LIKE '%e'
```

Result Grid Filter Rows: Export: Wrap Cell Content:

	lname
▶	Crabbe
	Goyle
	Dumbledore
	Snake

Result
GridForm
EditorField
TypesQuery
Stats

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT SUM(age)  
5 FROM characters  
6
```

Result Grid

	SUM(age)
▶	257

Filter Rows: Export: Wrap Cell Content:

Result Grid
Form Editor
Field Types
Query Stats

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT fname, lname, age
5 FROM characters
6 ORDER BY age DESC
7 |
```

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

	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

Result Grid
Form Editor
Field Types
Query Stats

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT fname, age  
5 FROM characters  
6 WHERE age BETWEEN 50 AND 100  
7
```

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

	fname	age
▶	Severus	55

Result Grid
Form Editor
Field Types
Query Stats

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT DISTINCT age
5 FROM characters
6
7
```

Result Grid

	age
▶	11
	111
	14
	55
	NULL

Export: | Wrap Cell Content: |

Result Grid

Form Editor

Field Types

Query Stats

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

Query 1 x



Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT *  
5 FROM characters  
6 WHERE faculty = 'Gryffindor' AND age > 30  
7
```

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

Result Grid

Form Editor

Field Types

Query Stats

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT DISTINCT faculty
5   FROM characters
6  LIMIT 3
7
```

Result Grid

Filter Rows:

Export: Wrap Cell Content: Fetch rows:

	faculty
▶	Gryffindor
	Slytherin
	Ravendaw

Result Grid

Form Editor

Field Types

Query Stats

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT fname
5   FROM characters
6  WHERE fname LIKE 'H____' OR fname LIKE 'L%';
7
```

Result Grid

	fname
▶	Harry
	Luna
	Lord

Form Editor

Field Types

Query Stats

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



```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT AVG (age)  
5 FROM characters  
6  
7
```

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

	AVG (age)
▶	25.7000

Result
GridForm
EditorField
TypesQuery
Stats

10. Посчитайте средний возраст всех персонажей

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • DELETE FROM characters  
5 WHERE char_id = '11'
```

11. Удалите персонажа с ID = 11

Output

Action Output

#	Time	Action	Message
✓ 29	17:35:54	SELECT * FROM characters LIMIT 0, 1000	10 row(s) returned
✓ 30	17:42:47	SELECT * FROM characters LIMIT 0, 1000	11 row(s) returned
✓ 31	17:45:20	SELECT * FROM characters WHERE char_id = '11' LIMIT 0, 1000	1 row(s) returned
✓ 32	17:46:06	DELETE FROM characters WHERE char_id = '11'	1 row(s) affected

Query 1 x



Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT lname  
5 FROM characters  
6 WHERE lname LIKE 'a%'
```

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

	lname
▶	Granger
	Weasley
	Malfoy
	Crabbe
	Snape

Result
Grid

Form
Editor

Field
Types

Query
Stats

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

Query 1 x



Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT fname AS 'Half-Blood Prince', lname, age, faculty, patronus  
5     FROM characters  
6     WHERE char_id = 10
```



Result Grid Filter Rows: Export: Wrap Cell Content:

	Half-Blood Prince	lname	age	faculty	patronus
▶	Severus	Snape	55	Slytherin	Doe

Result Grid

Form Editor

Field Types

Query Stats

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT char_id, patronus
5     FROM characters
6     WHERE patronus IS NOT NULL AND NOT patronus = 'Unknown'
7     ORDER by patronus ASC
```

Result Grid

Filter Rows:

Edit: Export/Import: Wrap Cell Content:

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

Result Grid
Form Editor
Field Types

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

Query 1 x

Limit to 1000 rows

```
2 • SELECT * FROM library;
3
4 • SELECT fname, lname
5     FROM characters
6     WHERE lname IN ('Crabbe', 'Granger', 'Diggory')
7
8
```

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

	fname	lname
▶	Hermione	Granger
	Vincent	Crabbe
	Cedric	Diggory

Result Grid
Form Editor
Field Types
Query Stats

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

Query 1 x

Limit to 1000 rows

```
2 • SELECT * FROM library;  
3  
4 • SELECT MIN(age)  
5 FROM characters  
6  
7  
8
```

Result Grid

	MIN(age)
▶	11

Filter Rows: Export: Wrap Cell Content:

Result Grid
Form Editor
Field Types
Query Stats

16. Выведите минимальный возраст персонажа

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT fname FROM characters  
5 UNION  
6 SELECT book_name FROM library  
7  
8  
9  
10
```

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

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 Dumble...
Fantastic Beasts And Where To Fi...
The Tales Of Beadle The Bard
Advanced Potion-Making
A History Of Magic

Result 48 x

Read Only

Result Grid
Form Editor
Field Types

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT COUNT(char_id), faculty
5     FROM characters
6     GROUP BY faculty
7     HAVING COUNT(char_id) > 1
8
9
```

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

	COUNT(char_id)	faculty
▶	4	Gryffindor
	4	Slytherin

Result Grid
Form Editor
Field Types
Only

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



```
3
4 • SELECT fname, lname,
5 CASE
6     WHEN faculty= 'Gryffindor' THEN 'Godric'
7     WHEN faculty= 'Slytherin' THEN 'Salazar'
8     WHEN faculty= 'Ravenclaw' THEN 'Rowena'
9     WHEN faculty= 'Hufflepuff' THEN 'Helga'
10    ELSE 'Muggle'
11 END AS Founders
12 FROM characters
13
```

Result Grid Filter Rows: Export: Wrap Cell Content:

	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

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT lname
5     FROM characters
6     WHERE lname NOT REGEXP '^[HLS]'
```

Result Grid

lname
Potter
Granger
Weasley
Malfoy
Crabbe
Goyle
Dumbledore
Diggory

Form Editor

Field Types

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



```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT characters.fname, characters.lname, library.book_name
5 FROM characters
6 INNER JOIN library
7 ON characters.char_id = library.char_id
8
9
10
11
```

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

Result Grid

Form Editor

Field Types

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

```

1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT characters.fname, characters.lname, library.book_name
5 FROM characters
6 LEFT JOIN library
7 ON characters.char_id = library.char_id
8
9
10
11
12

```

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

Тут я не совсем
поняла
задание.
По идее
заданию
соответствует
вот этот запрос



Но у всех
персонажей
книги **есть**, зато
у некоторых книг
нет персонажей.
Поэтому я
предполагаю,
что имелось в
виду вот это



```

Query 1 x
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT characters.fname, characters.lname, library.book_name
5 FROM characters
6 RIGHT JOIN library
7 ON characters.char_id = library.char_id
8
9
10
11
12

```

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

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

```
1 • SELECT * FROM characters;  
2 • SELECT * FROM library;  
3  
4 • SELECT characters.patronus, library.book_name  
5 FROM characters  
6 RIGHT JOIN library  
7 ON characters.char_id = library.char_id  
8  
9  
10  
11  
12
```

Result Grid  Filter Rows: Export:  Wrap Cell Content: 

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

The Life And Lies Of Albus Dumbledore

Result
GridForm
Editor

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

Query 1

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT characters.fname, characters.lname, characters.age, library.book_name
5     FROM characters
6     INNER JOIN library
7     ON characters.char_id = library.char_id
8     WHERE age > 15
9
10
11
12
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

Result GridForm EditorField Types

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

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT characters.fname, characters.lname, library.book_name, library.start_date, library.end_date
5 FROM characters
6 INNER JOIN library
7 ON characters.char_id = library.char_id
8 WHERE age < 15 AND patronus = 'Unknown'
9
10
11
```

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

	fname	lname	book_name	start_date	end_date
▶	Cedric	Diggory	The Lockhart Collection	2015-12-20	2030-12-20

Result Grid
Form Editor
Field Types

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

Query 1 x

Limit to 1000 rows

```
2 • SELECT * FROM library;
3
4
5 • SELECT COUNT(end_date)
6     FROM library
7     WHERE end_date > ( SELECT end_date FROM library WHERE char_id = (
8                         SELECT char_id
9                         FROM characters
10                        WHERE fname = 'Hermione' ));
11
```

Result Grid

COUNT(end_date)
2

Export: | Wrap Cell Content: |

Result Grid
Form Editor
Field Types

6. Используя вложенный запрос количество книг, у которых end_date больше, чем end_date у Hermione

Query 1 x

Limit to 1000 rows

```
1 • SELECT * FROM characters;
2 • SELECT * FROM library;
3
4 • SELECT patronus
5     FROM characters
6     WHERE age > ( SELECT age FROM characters WHERE patronus = 'Unknown');
```

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

patronus
Phoenix
Doe

Result Grid
Form Editor
Field Types

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