Задание:

Предусловия: установите MySQL Server и MySQL Workbench (версию ниже 8.20)

Наша задача разобраться с простыми и немного усложненными запросами к базам данных, написанных на SQL

Скачайте дампы базы данных Hogwarts отсюда https://drive.google.com/drive/u/3/folders/1MC0AttnmlAmugifFlX3hG6pssYZDqpPB

Осуществите импорт таблиц, используя Workbench

Выполните задание 1 и 2, которые вы найдете здесь https://drive.google.com/drive/u/3/folders/1Lt7CY69nR5awNs\_9q0XJOHRti4vJj3Qa

Сохраните свое решение в виде документа Word, куда необходимо добавить скрины отправленных запросов в CLI или Workbench

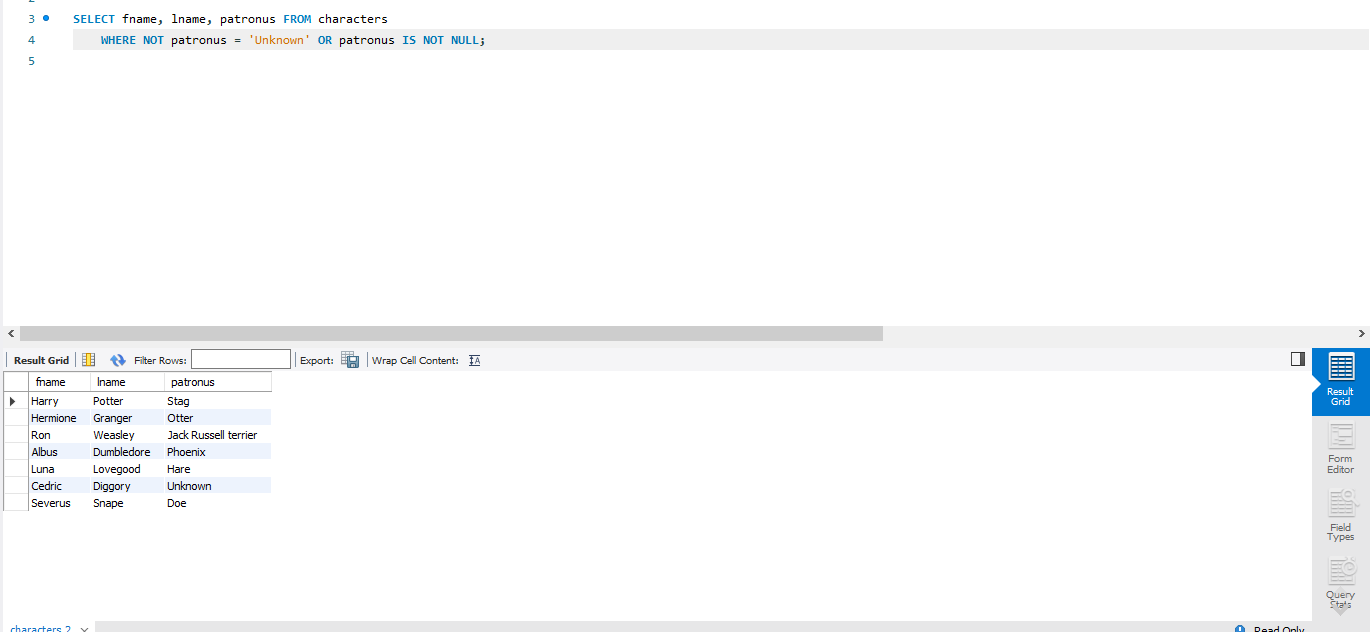
Помните, что большинство ошибок возникает из-за неправильного синтаксиса и использования операторов. Будьте внимательные. Проверяйте результаты, отправленных запросов, чтобы быть уверенным в своем решении

**Задание 1**

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

SELECT fname, lname, patronus FROM characters

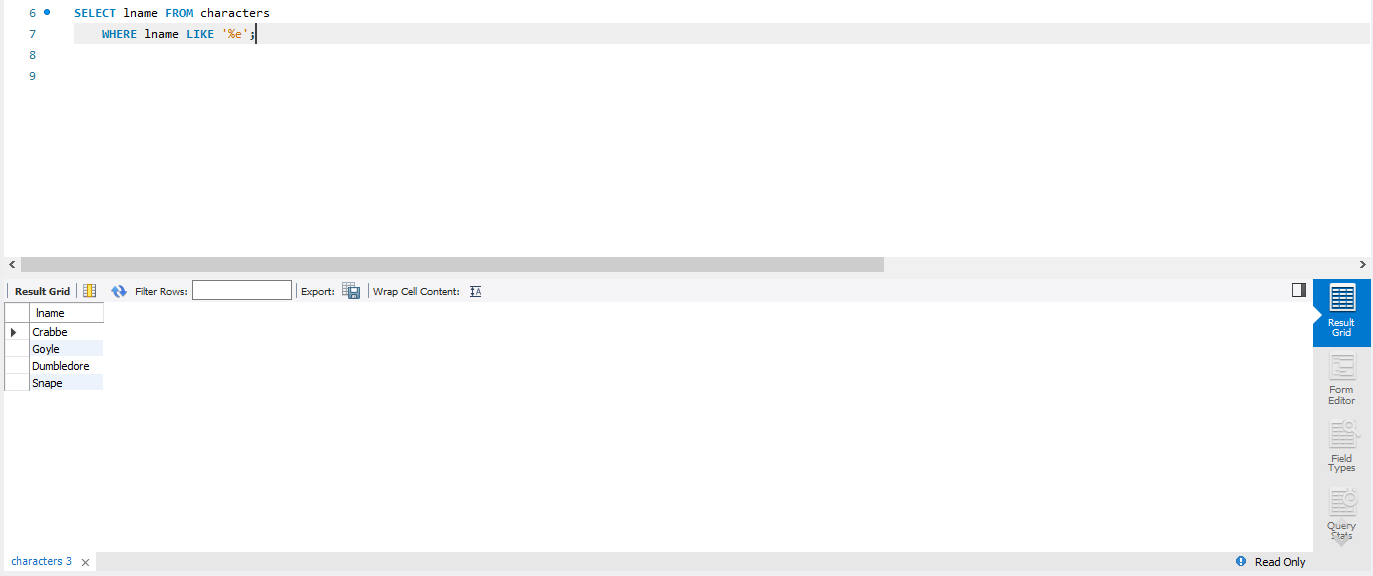
WHERE NOT patronus = 'Unknown' OR patronus IS NOT NULL;



1. Выведите фамилию персонажей, у которых последняя буква в фамилии ‘e’

SELECT lname FROM characters

WHERE lname LIKE '%e';



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

SELECT SUM(age) FROM characters;



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

SELECT fname, lname, age FROM characters

ORDER BY age DESC;



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

SELECT lname, age FROM characters

WHERE age BETWEEN 50 AND 100;



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

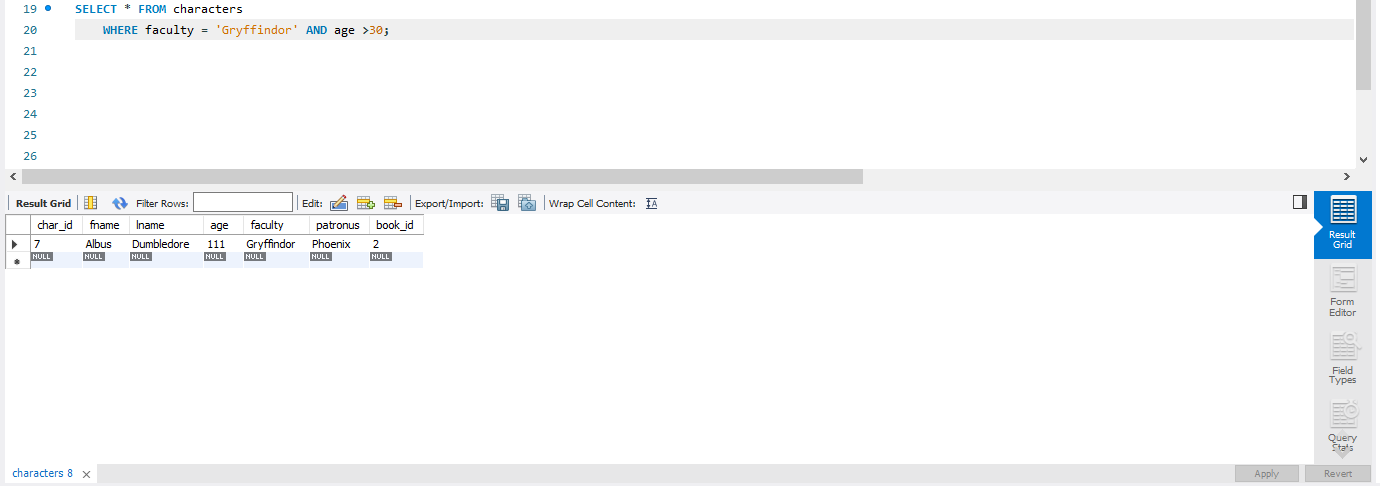
SELECT DISTINCT age FROM characters;



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

SELECT \* FROM characters

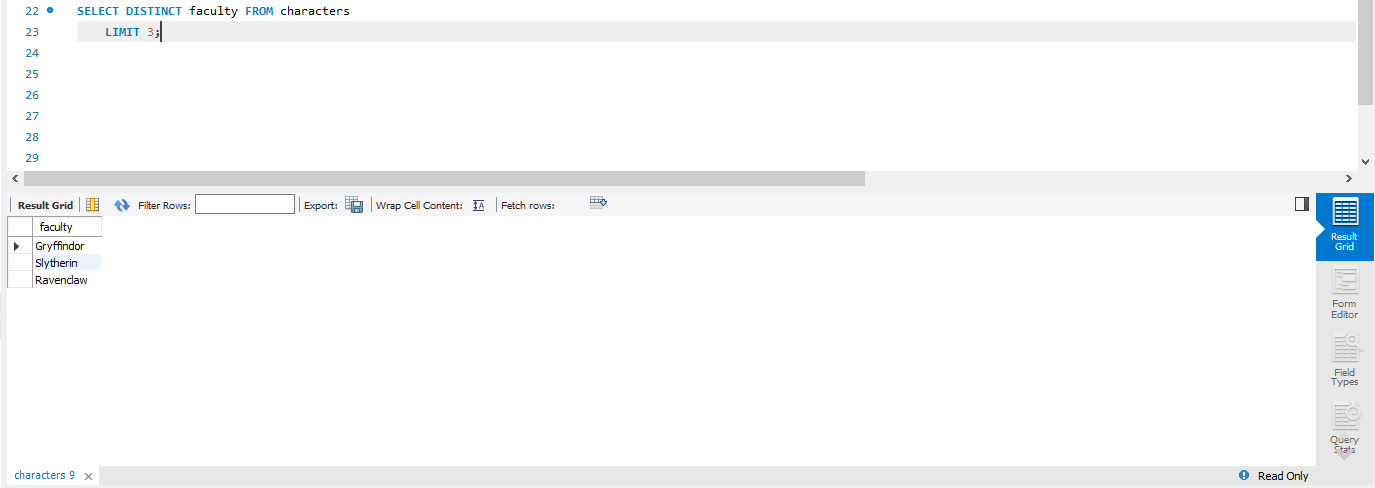
WHERE faculty = 'Gryffindor' AND age >30;



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

SELECT DISTINCT faculty FROM characters

LIMIT 3;



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

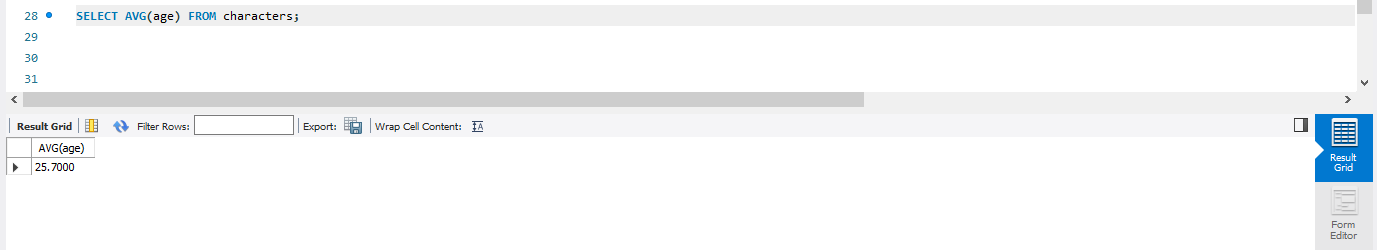
SELECT fname FROM characters

WHERE fname LIKE 'H\_\_\_\_' OR fname LIKE 'L%';



1. Посчитайте средний возраст всех персонажей

SELECT AVG(age) FROM characters;



1. Удалите персонажа с ID = 11

DELETE FROM characters

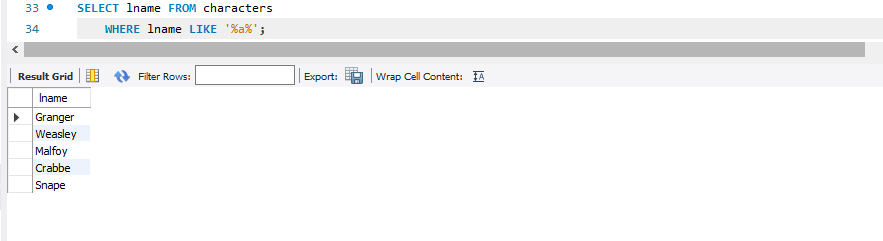
WHERE char\_id = 11;



1. Выведите фамилию всех персонажей, которые содержат в ней букву ‘a’

SELECT lname FROM characters

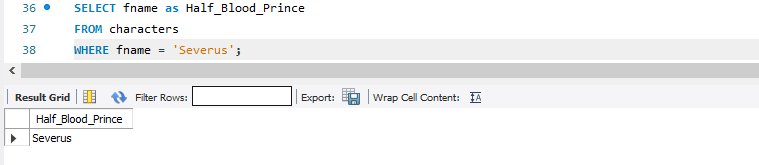
WHERE lname LIKE '%a%';



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

SELECT fname as Half\_Blood\_Prince FROM characters

WHERE fname = 'Severus';

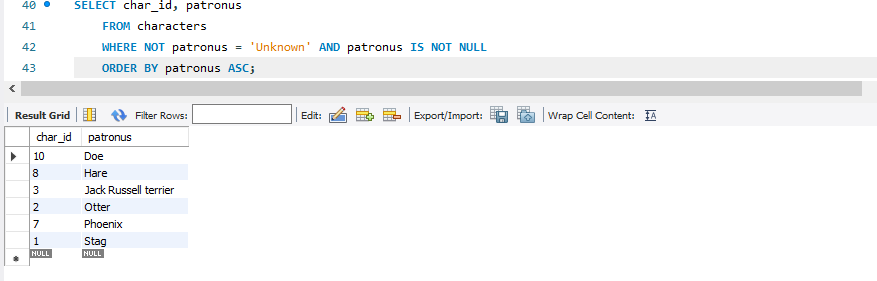


1. Выведите id и имена всех патронусов в алфавитном порядки, при условии, что они есть или известны

SELECT char\_id, patronus FROM characters

WHERE NOT patronus = 'Unknown' AND patronus IS NOT NULL

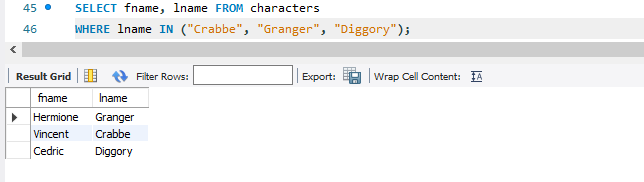
ORDER BY patronus ASC;



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

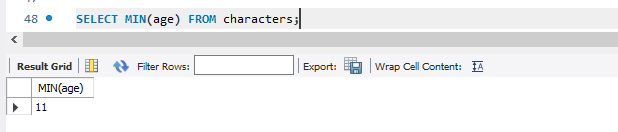
SELECT fname, lname FROM characters

WHERE lname IN ("Crabbe", "Granger", "Diggory");



1. Выведите минимальный возраст персонажа

SELECT MIN(age) FROM characters;

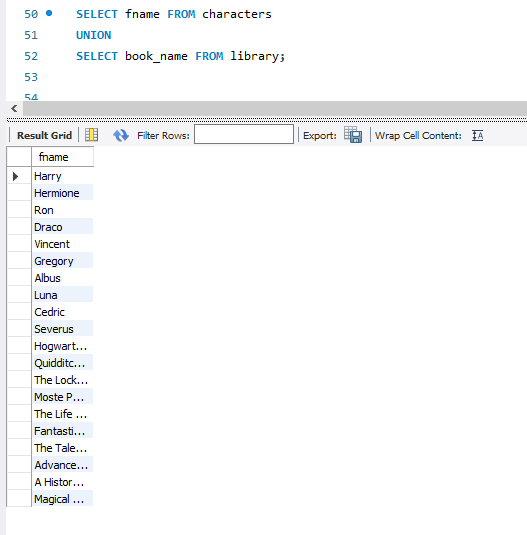


1. Используя оператор [UNION](https://www.w3schools.com/sql/sql_union.asp) выберите имена из таблицы characters и названия книг из таблицы library

SELECT fname FROM characters

UNION

SELECT book\_name FROM library;

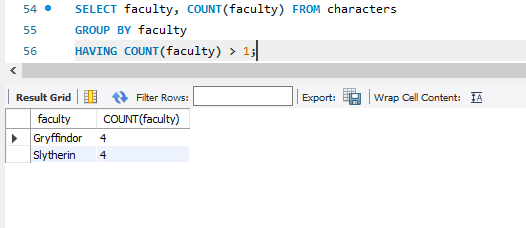


1. Используя оператор [HAVING](https://www.w3schools.com/sql/sql_having.asp) посчитайте количество персонажей на каждом факультете, оставив только те факультеты, где количество студентов больше 1

SELECT faculty, COUNT(faculty) FROM characters

GROUP BY faculty

HAVING COUNT(faculty) > 1;



1. Используя оператор [CASE](https://www.w3schools.com/sql/sql_case.asp) 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

SELECT fname, lname,

CASE

WHEN faculty = "Gryffindor" THEN "Godric"

WHEN faculty = "Slytherin" THEN "Salazar"

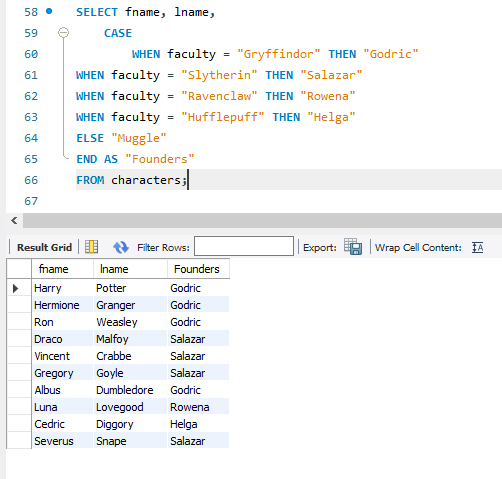
WHEN faculty = "Ravenclaw" THEN "Rowena"

WHEN faculty = "Hufflepuff" THEN "Helga"

ELSE "Muggle"

END AS "Founders"

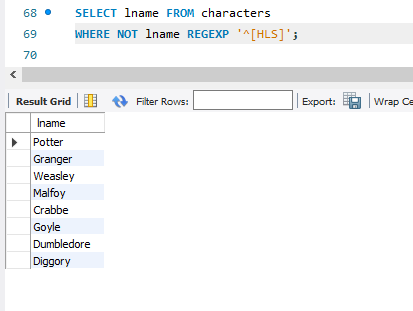
FROM characters;



20. [Используя регулярное выражение](https://dev.mysql.com/doc/refman/5.7/en/pattern-matching.html) найдите фамилии персонажей, которые не начинаются с букв H, L или S и выведите их

SELECT lname FROM characters

WHERE NOT lname REGEXP '^[HLS]';

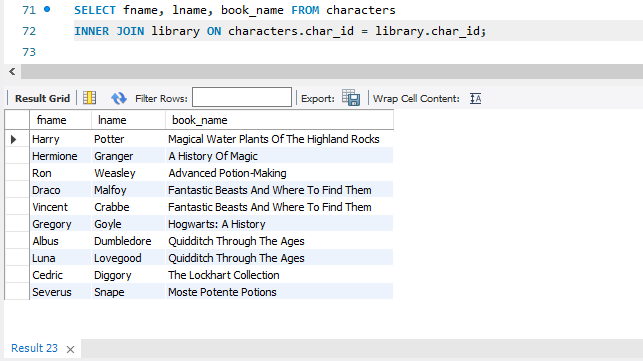


**Задание 2.**

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

SELECT fname, lname, book\_name FROM characters

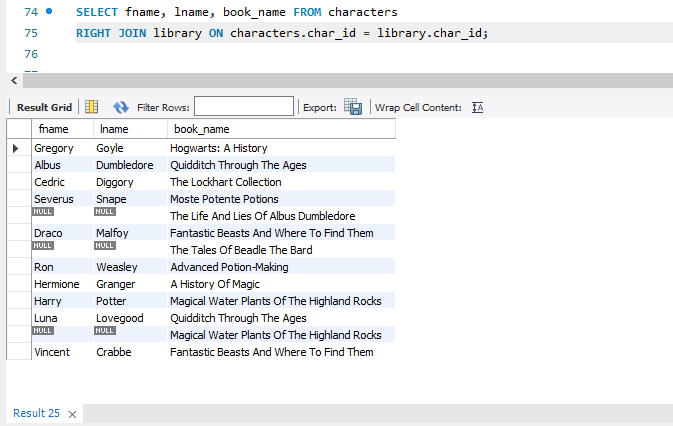
INNER JOIN library ON characters.char\_id = library.char\_id;



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

SELECT fname, lname, book\_name FROM characters

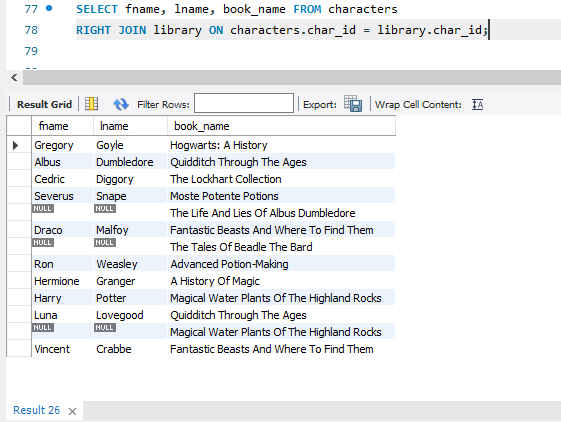
RIGHT JOIN library ON characters.char\_id = library.char\_id;



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

SELECT fname, lname, book\_name FROM characters

RIGHT JOIN library ON characters.char\_id = library.char\_id;

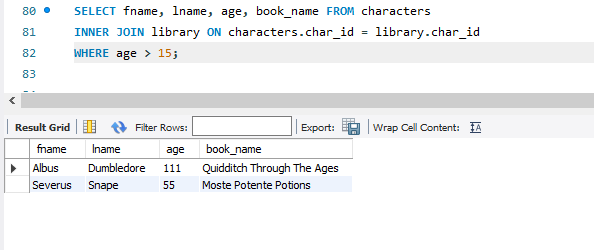


1. Выведите имя, фамилию, возраст персонажей и название книги, которая на них числится, при условии, что все владельцы книг должны быть старше 15 лет

SELECT fname, lname, age, book\_name FROM characters

INNER JOIN library ON characters.char\_id = library.char\_id

WHERE age > 15;

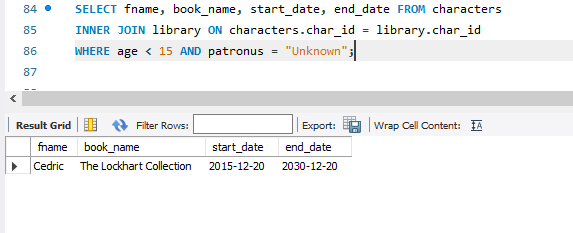


1. Выведите имя персонажа, название книги, дату выдачи и дату завершения, при условии, что он младше 15 лет и его патронус неизвестен

SELECT fname, book\_name, start\_date, end\_date FROM characters

INNER JOIN library ON characters.char\_id = library.char\_id

WHERE age < 15 AND patronus = "Unknown";

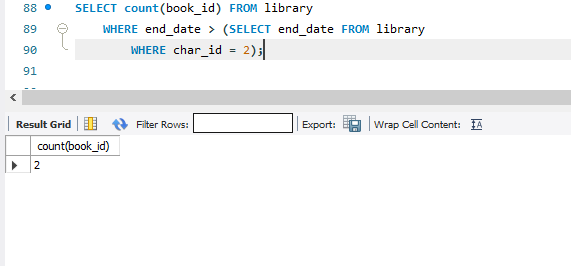


6. Используя вложенный запрос количество книг, у которых end\_date больше, чем end\_date у Hermione

SELECT count(book\_id) FROM library

WHERE end\_date > (SELECT end\_date FROM library

WHERE char\_id = 2);



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

SELECT patronus FROM characters

WHERE age > (SELECT age FROM characters

WHERE patronus = 'Unknown');

