

# "Characters" table querrys

## Task

1. Print first name, last name, patronus of all characters that have patronus and it is known

## Solution print screen

Query 1 SQL File 3" x

Limit to 1000 rows

```

39
40 • select fname, lname, patronus from characters
41 where patronus is not null and not patronus="Unknown";
42
43

```

Result Grid Filter Rows: Export: Wrap Cell Content: [f1](#)

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

2. Print the last name of the characters whose last letter is 'e'

Query 1 SQL File 3" x

Limit to 1000 rows

```

2
3 • SELECT fname, lname, patronus from characters
4 where patronus IS NOT NULL;
5
6 • SELECT lname from characters
7 where lname like "%e";

```

Result Grid Filter Rows: Export: Wrap Cell Content: [f1](#)

lname
Crabbe
Goyle
Dumbledore
Snape

3. Calculate the total age of all characters and display it on the screen

Query 1 SQL File 3" x

Limit to 1000 rows

```

4 where patronus IS NOT NULL;
5
6 • SELECT lname from characters
7 where lname like "%e";
8
9 • select SUM(Age) from characters;

```

Result Grid Filter Rows: Export: Wrap Cell Content: [f1](#)

SUM(Age)
257

4. Print the first name, last name and age of the characters in descending order of their age

Query 1 SQL File 3" x

Limit to 1000 rows

```

7 where lname like "%e";
8
9 • select SUM(Age) from characters;
10
11 • select fname, lname, Age from characters
12 Order by Age DESC;

```

Result Grid Filter Rows: Export: Wrap Cell Content: [f1](#)

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	1000

5. Print the character's name and age, whose age is between 50 and 100 years old

Query 1 SQL File 3\*

```

10
11 • select fname, lname, Age from characters
12   Order by Age DESC;
13
14 • select fname, Age from characters
15   where Age between 50 and 100;

```

Result Grid

fname	Age
Severus	55

6. Print the age of all the characters so that there are no ones among them who have the same age

Query 1 SQL File 3\*

```

12   Order by Age DESC;
13
14 • select fname, Age from characters
15   where Age between 50 and 100;
16
17 • select distinct age from characters;

```

Result Grid

age
11
111
14
55
HALL

7. Print all information about characters whose faculty = Gryffindor and who are over 30 years old

Query 1 SQL File 3\*

```

15   where Age between 50 and 100;
16
17 • select distinct age from characters;
18
19 • select * from characters
20   where faculty="Gryffindor" and age>30;

```

Result Grid

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

8. Print the names of the first three faculties from the table so that the faculties do not repeat

Query 1 SQL File 3\*

```

18
19 • select * from characters
20   where faculty="Gryffindor" and age>30;
21
22 • select distinct faculty from characters
23   limit 3;

```

Result Grid

faculty
Gryffindor
Slytherin
Ravenclaw

9. Print names of all characters whose name starts with 'H' and consists of 5 letters, or whose name starts with 'L'

Query 1 SQL File 3\*

```

21
22 • select distinct faculty from characters
23 limit 3;
24
25 • select fname from characters
26 where fname like "H____" or fname like "L%";

```

Result Grid

fname
Harry
Luna
Lord

10. Calculate the average age of all characters

Query 1 SQL File 3\*

```

23 limit 3;
24
25 • select fname from characters
26 where fname like "H____" or fname like "L%";
27
28 • select avg(age) from characters;

```

Result Grid

avg(age)
25.7000

11. Delete character with ID = 11

Query 1 SQL File 3\*

```

28 • select avg(age) from characters;
29
30 • delete from characters
31 where char_id=11;
32
33 • select * from characters;

```

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	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
*	None	None	None	None	None	None

12. Print the last name of all characters that contain the letter 'a' in it

Query 1 SQL File 3\*

```

31 where char_id=11;
32
33 • select * from characters;
34
35 • select lname from characters
36 where lname like "%a%";

```

Result Grid

lname
Granger
Weasley
Malfoy
Crabbe
Snape

13. Use an alias to temporarily replace the column name fname to Half-Blood Prince and show real Half-Blood Prince

Query 1 SQL File 3\*

```

34
35 • select lname from characters
36 where lname like "%a%";
37
38 • select char_id, fname as "Half-Blood Prince", lname, age, faculty, patronus, book_id from characters
39 where lname="Snape";

```

Result Grid

	char_id	Half-Blood Prince	lname	age	faculty	patronus	book_id
▶	10	Severus	Snape	55	Slytherin	Doe	4
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL

14. Print the id and names in alphabetical order of whose Patronuses, who exist or are known

Query 1 SQL File 3\*

```

28
29
30 • select char_id, patronus from characters
31 where patronus is not null and not patronus="Unknown"
32 order by patronus ASC;

```

Result Grid

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

15. Using the IN operator, print the first and last names of those characters whose last name is Crabbe, Granger, or Diggory

Query 1 SQL File 3\*

```

42 where patronus is not null
43 order by patronus ASC;
44
45 • select fname, lname from characters
46 where lname in("Crabbe", "Granger", "Diggory");
47

```

Result Grid

	fname	lname
▶	Hermione	Granger
Vincent	Vincent	Crabbe
Cedric	Cedric	Diggory

16. Print the minimum age of the character

Query 1 SQL File 3\*

```

43 order by patronus ASC;
44
45 • select fname, lname from characters
46 where lname in("Crabbe", "Granger", "Diggory");
47
48 • select min(age) from characters;

```

Result Grid

	min(age)
▶	11

17. Using UNION operator select names from the characters table and book titles from library table

Query 1 SQL File 3\*

```

47
48 • select min(age) from characters;
49
50 • select fname from characters
51 union
52 select book_name from library;

```

Result Grid Filter Rows: Export: Wrap Cell Content: 15

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. Using HAVING operator, count the number of characters in each department, keeping only those departments where the number of students is greater than 1

Query 1 SQL File 3\*

```

52 select book_name from library;
53
54 • select count(char_id), faculty
55 from characters
56 group by faculty
57 having count(char_id)>1;

```

Result Grid Filter Rows: Export: Wrap Cell Content: 15

count(char_id)	faculty
4	Gryffindor
4	Slytherin

19. Using the CASE statement, describe the following logic. Print the character's first and last name, as well as the following text message:

- If the faculty is Gryffindor, then Godric should be displayed in the console
- If the faculty is Slytherin, then Salazar should be displayed in the console
- If the faculty is Ravenclaw, then Rowena should be displayed in the console
- If the faculty is Hufflepuff, then Helga should be displayed in the console
- If other information, then Muggle is displayed

Query 1 SQL File 3\*

```

63 group by faculty
64 having count(char_id)>1;
65
66 • select fname, lname,
67 case
68 when faculty="Gryffindor" then "Godric"
69 when faculty="Slytherin" then "Salazar"
70 when faculty="Ravenclaw" then "Rowena"
71 when faculty="Hufflepuff" then "Helga"
72 when faculty not in("Gryffindor", "Slytherin", "Ravenclaw", "Hufflepuff" ) then "Muggle"
73 end as Founders
74 from characters;

```

Result Grid Filter Rows: Export: Wrap Cell Content: 15

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
test_fname	test_lname	Muggle

20. Using a regular expression, find the last names of characters that do not start with the letters H, L or S and print them

Query 1 SQL File 3\*

```

73 end as Founders
74 from characters;
75
76 • select * from characters;
77
78 • select lname from characters
79 where lname not regexp "^H|^L|^S";

```

Result Grid Filter Rows: Export: Wrap Cell Content: [fA](#)

lname
Potter
Granger
Weasley
Malfoy
Crabbe
Goyle
Dumbledore
Diggory

Joins / Library table + Characters table

Task

Solution print screen

1. Print the first and last names of the characters and the title of the book they have

Query 1 SQL File 3\*

```

44
45 • select characters.fname, characters.lname, library.book_name
46 from characters
47 join library
48 on characters.char_id=library.char_id;

```

Result Grid Filter Rows: Export: Wrap Cell Content: [fA](#)

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. Print the first name, last name of the characters, and the title of the book, regardless of whether they have books or not

Query 1 SQL File 3\*

```

78 • select lname from characters
79 where lname not regexp "^H|^L|^S";
80
81 • select characters.fname, characters.lname, library.book_name
82 from characters
83 left join library
84 on characters.book_id=library.book_id;

```

Result Grid Filter Rows: Export: Wrap Cell Content: [fA](#)

fname	lname	book_name
Harry	Potter	Magical Water Plants Of The Highland Rocks
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
Draco	Malfoy	Fantastic Beasts And Where To Find Them
Vincent	Crabbe	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
Albus	Dumbledore	Quidditch Through The Ages
Luna	Lovegood	Quidditch Through The Ages
Luna	Lovegood	Quidditch Through The Ages
Cedric	Diggory	The Lockhart Collection
Severus	Snape	Moste Potente Potions
test_fname	test_lname	

3. Print the title of the book and the name of the patronus, regardless of whether the information about the holder of the book is in the table or not

Query 1 SQL File 3\*

```

44
45 • select library.book_name, characters.patronus
46 from library
47 left join characters
48 on library.char_id=characters.char_id;

```

Result Grid

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. Print the first name, last name, age of the characters and the title of the book they have, provided that all book owners must be over 15 years old (see next page)

Query 1 SQL File 3\*

```

49
50 • select characters.fname, characters.lname, characters.age, library.book_name
51 from characters
52 join library
53 on characters.char_id=library.char_id
54 where age>15;

```

Result Grid

fname	lname	age	book_name
Albus	Dumbledore	111	Quidditch Through The Ages
Severus	Snape	55	Moste Potente Potions

5. Print the character's name, book title, issue date, and completion date, assuming they are under 15 and their patronus is unknown

Query 1 SQL File 3\*

```

48 on library.char_id=characters.char_id;
49
50 • select characters.fname, library.book_name, library.start_date, library.end_date
51 from characters
52 join library
53 on characters.char_id=library.char_id
54 where age<15 and patronus="Unknown";

```

Result Grid

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

6. Using a nested query, print the number of books where end\_date is greater than Hermione's end\_date

Query 1 SQL File 3\*

```

18 on characters.char_id=library.char_id
19 where age<15 and patronus IS NULL;
20
21 • select count(book_id) from library where end_date>
22 (select end_date from library where char_id in
23 (select char_id from characters where fname="Hermione"))
24 );


```

Result Grid




count(book_id)
2

7. Use a subquery to print the names of all Patronuses whose owners are older than the age of the character whose Patronus is Unknown

Query 1 SQL File 3\* x

Limit to 1000 rows

```
22 (select end_date from library where char_id in
23 (select char_id from characters where fname="Hermione")
24 );
25
26 • select patronus from characters where age>
27 (select age from characters where patronus="Unknown");
28
```

Result Grid  Filter Rows:  Export:  Wrap Cell Content: 

patronus
Phoenix
Doe