Julia just finished conducting a coding contest, and she needs your help assembling the leaderboard! Write a query to print the respective hacker\_id and name of hackers who achieved full scores for more than one challenge. Order your output in descending order by the total number of challenges in which the hacker earned a full score. If more than one hacker received full scores in same number of challenges, then sort them by ascending hacker\_id.

select s.hacker\_id, count(s.score)

from Submissions s

inner join Challenges c on c.challenge\_id = s.challenge\_id

inner join Difficulty d on c.difficulty\_level = d.difficulty\_level

where s.score = d.score and c.difficulty\_level = d.difficulty\_level

group by s.hacker\_id

having count(s.score) > 1

order by count(s.score) desc, s.hacker\_id asc

select h.hacker\_id, h.name, count(s.hacker\_id)

from Submissions s

inner join Challenges c on c.challenge\_id = s.challenge\_id

inner join Difficulty d on c.difficulty\_level = d.difficulty\_level

inner join hackers h on s.hacker\_id = h.hacker\_id

where s.score = d.score and c.difficulty\_level = d.difficulty\_level

group by h.hacker\_id, h.name

having count(s.hacker\_id) > 1

order by count(s.hacker\_id) desc, h.hacker\_id asc

select h.hacker\_id, h.name

from submissions s

inner join challenges c on s.challenge\_id = c.challenge\_id

inner join difficulty d on c.difficulty\_level = d.difficulty\_level

inner join hackers h on s.hacker\_id = h.hacker\_id

where s.score = d.score and c.difficulty\_level = d.difficulty\_level

group by h.hacker\_id, h.name

having count(s.hacker\_id) > 1

order by count(s.hacker\_id) desc, s.hacker\_id asc

Write a query to print the id, age, coins\_needed, and power of the wands that Ron's interested in, sorted in order of descending power. If more than one wand has same power, sort the result in order of descending age.

Select w.id, wp.age, w.coins\_needed, w.power

From wands w

inner join Wands\_Property wp on w.code = wp.code and is\_evil !=1

where w.coins\_needed=(select min(coins\_needed)

from wands

inner join wands\_property on wands.code = wands\_property.code

where wands\_property.age=wp.age and wands.power = w.power)

order by w.power desc, wp.age desc

select a.id, b.age, a.coins\_needed, a.power

from Wands a

inner join Wands\_Property b on a.code=b.code

where b.is\_evil!=1

and a.coins\_needed=(select min(Wands.coins\_needed)

from Wands

inner join Wands\_Property on Wands.code=Wands\_Property.code

where Wands\_Property.age=b.age and Wands.power=a.power)

order by a.power desc,b.age desc

* **1038 496 4789 10**
* **1130 494 9439 10**
* **1315 492 4126 10**
* **9 491 7345 10**
* **858 483 4352 10**
* **1164 481 9831 10**
* **1288 464 4952 10**
* **861 462 8302 10**
* **412 455 5625 10**
* **996 451 8884 10**
* **1608 446 8351 10**
* **1376 443 1735 10**
* **1330 430 5182 10**
* **1633 425 2206 10**

Write a query to print the hacker\_id, name, and the total number of challenges created by each student. Sort your results by the total number of challenges in descending order. If more than one student created the same number of challenges, then sort the result by hacker\_id. If more than one student created the same number of challenges and the count is less than the maximum number of challenges created, then exclude those students from the result.

Select h.hacker\_id, h.name, count(c.hacker\_id)

from hackers h

inner join challenges c on c.hacker\_id = h.hacker\_id

group by h.hacker\_id, h.name

having count(c.hacker\_id) =

(Select max(a.cnt) from

(Select count(hacker\_id) as cnt

from challenges

group by hacker\_id)a

)

or count(c.hacker\_id) in

(select count(b.cnt) from

(Select count(hacker\_id) as cnt

from challenges

group by hacker\_id)b

group by b.cnt

having count(b.cnt) = 1)

order by count(c.hacker\_id) desc, h.hacker\_id asc

Select max(a.cnt) from

(Select count(hacker\_id) as cnt

from challenges

group by hacker\_id)a

select count(b.cnt) from

(Select count(hacker\_id) as cnt

from challenges

group by hacker\_id)b

group by b.cnt

having count(b.cnt) = 1

/\* these are the columns we want to output \*/

select c.hacker\_id, h.name ,count(c.hacker\_id) as c\_count

/\* this is the join we want to output them from \*/

from Hackers as h

inner join Challenges as c on c.hacker\_id = h.hacker\_id

/\* after they have been grouped by hacker \*/

group by c.hacker\_id

/\* but we want to be selective about which hackers we output \*/

/\* having is required (instead of where) for filtering on groups \*/

having

/\* output anyone with a count that is equal to... \*/

c\_count =

/\* the max count that anyone has \*/

(SELECT MAX(temp1.cnt)

from (SELECT COUNT(hacker\_id) as cnt

from Challenges

group by hacker\_id

order by hacker\_id) temp1)

/\* or anyone who's count is in... \*/

or c\_count in

/\* the set of counts... \*/

(select t.cnt

from (select count(\*) as cnt

from challenges

group by hacker\_id) t

/\* who's group of counts... \*/

group by t.cnt

/\* has only one element \*/

having count(t.cnt) = 1)

/\* finally, the order the rows should be output \*/

order by c\_count DESC, c.hacker\_id