

full code:

```
SELECT hackers.hacker_id, hackers.name
FROM hackers
JOIN submissions ON hackers.hacker_id = submissions.hacker_id
JOIN challenges ON submissions.challenge_id = challenges.challenge_id
JOIN difficulty ON challenges.difficulty_level = difficulty.difficulty_level
where submissions.score = difficulty.score
group by hackers.hacker_id, hackers.name
having count(submissions.hacker_id)>1
order by count(submissions.hacker_id) desc, hackers.hacker_id asc
```

—the idea—

idenya adalah
membuat semua menjadi satu tabel agar lebih mudah di seleksi.

ide pertama dalam 'select'
ingat bahwa yang akan kita pilih adalah hacker id dan nama dari tabel hacker. tabel yang diambil adalah tabel hackers, kenapa, karena nantinya tabel ini akan digabung ke banyak tabel. maka:

```
SELECT hackers.hacker_id, hackers.name
FROM hackers
```

ide kedua kita akan menggabungkan tabel-tabel yang ada

```
JOIN submissions ON hackers.hacker_id = submissions.hacker_id
JOIN challenges ON submissions.challenge_id = challenges.challenge_id
JOIN difficulty ON challenges.difficulty_level = difficulty.difficulty_level
```

yang pertama kita menggabungkan tabel hackers dengan submission, maka perintahnya adalah join tabel submission ke tabel hacker yang hacker idnya sama.
kedua kita menggabungkan tabel challenges dengan tabel submission, maka perintahnya adalah join tabel challenges ke tabel submissions yang challenge id nya sama.
ketiga kita menggabungkan tabel difficulty dengan challenge, maka perintahnya adalah join tabel difficulty ke challenges yang difficulty level nya sama

jika kita memilih semua tabel (bukan **hackers.hacker_id, hackers.name**) maka outputnya akan seperti ini

Your Output (stdout)

```
1 40226 Anna 43954 40226 69855 40 69855 48984 3 3 40
2 85039 Linda 89007 85039 44764 14 44764 14863 2 2 30
3 32172 Jonathan 38171 32172 25419 80 25419 49307 5 5 80
4 95822 Wayne 95655 95822 63530 47 63530 39771 4 4 60
5 61885 Gerald 67667 61885 55235 16 55235 59853 4 4 60
6 72 Rose 608 72 93294 0 93294 59907 4 4 60
7 47641 Patricia 48950 47641 51898 30 51898 5720 2 2 30
8 13762 Gloria 14835 13762 44764 30 44764 14863 2 2 30
9 3768 Earl 5719 3768 44764 20 44764 14863 2 2 30
```

ide ketiga kita masuk ke sistem seleksi. seleksi pertama adalah kita harus tahu hacker mana yang punya score penuh. caranya adalah dengan menyamakan submission score dengan difficulty score.

where submissions.score = difficulty.score

outputnya akan seperti ini:

```
746 49652 Christine
747 49652 Christine
748 49789 Lillian
749 49789 Lillian
750 49789 Lillian
751 49789 Lillian
752 49789 Lillian
753 51410 Bobby
754 51503 Gerald
755 52184 Jeremy
756 55007 Albert
```

hasilnya adalah hacker id dan nama yang telah mengumpulkan challenge. ada yang lebih dari satu challenge ada yang hanya 1 challenge.

ide keempat: maka kita perlu mengelompokkan hacker tersebut berdasarkan id dan namanya.

group by hackers.hacker_id, hackers.name

Your Output (stdout)

```
1 10857 Kevin
2 12539 Paul
3 13122 James
4 13380 Kelly
5 13391 Robin
6 13523 Ralph
7 13762 Gloria
8 13944 Victor
9 14246 David
```

yeay! kita sudah mendapatkan id dan nama hacker yang mengerjakan dengan full score. Selanjutnya kita masuk ke seleksi kedua, yaitu memilih hacker yang mengerjakan lebih dari satu challenge.

idnya adalah kita menghitung id hacker yang lebih dari 1 muncul di tabel tersebut. karena kita tidak dapat menggunakan pengkondisian 'where' dua kali, maka kita menggunakan fungsi 'having'

having count(submissions.hacker_id)>1

hasilnya adalah

Your Output (stdout)

1	39771 Alan
2	36322 Andrew
3	42052 Andrew
4	30721 Ann
5	40226 Anna
6	22196 Anthony
7	25732 Antonio
8	38852 Benjamin
9	16259 Brandon

langkah terakhir adalah mengurutkan hacker id dan nama dengan kondisi:

1. melakukan challenge terbanyak
2. jika lebih dari satu orang maka diurutkan per idnya

order by count(submissions.hacker_id) desc, hackers.hacker_id asc

Your Output (stdout)

1	27232 Phillip
2	28614 Willie
3	15719 Christina
4	43892 Roy
5	14246 David
6	14372 Michelle
7	18330 Lawrence
8	26133 Jacqueline
9	26253 John