

# Problem 1811: Find Interview Candidates

## Problem Information

**Difficulty:** Medium

**Acceptance Rate:** 0.00%

**Paid Only:** No

## Problem Description

Table:

Contests

+-----+-----+ | Column Name | Type | +-----+-----+ | contest\_id | int || gold\_medal | int | | silver\_medal | int | | bronze\_medal | int | +-----+-----+ contest\_id is the column with unique values for this table. This table contains the LeetCode contest ID and the user IDs of the gold, silver, and bronze medalists. It is guaranteed that any consecutive contests have consecutive IDs and that no ID is skipped.

Table:

Users

+-----+-----+ | Column Name | Type | +-----+-----+ | user\_id | int | | mail | varchar | | name | varchar | +-----+-----+ user\_id is the column with unique values for this table. This table contains information about the users.

Write a solution to report the

name

and the

mail

of all

interview candidates

. A user is an

interview candidate

if

at least one

of these two conditions is true:

The user won

any

medal in

three or more consecutive

contests.

The user won the

gold

medal in

three or more different

contests (not necessarily consecutive).

Return the result table in

any order

.

The result format is in the following example.

Example 1:

Input:

Contests table: +-----+-----+-----+-----+ | contest\_id | gold\_medal | silver\_medal | bronze\_medal | +-----+-----+-----+-----+ | 190 | 1 | 5 | 2 | | 191 | 2 | 3 | 5 | | 192 | 5 | 2 | 3 | | 193 | 1 | 3 | 5 | | 194 | 4 | 5 | 2 | | 195 | 4 | 2 | 1 | | 196 | 1 | 5 | 2 | +-----+-----+-----+-----+ Users table: +-----+-----+-----+ | user\_id | mail | name | +-----+-----+-----+ | 1 | sarah@leetcode.com | Sarah | | 2 | bob@leetcode.com | Bob | | 3 | alice@leetcode.com | Alice | | 4 | hercy@leetcode.com | Hercy | | 5 | quarz@leetcode.com | Quarz | +-----+-----+-----+

Output:

+-----+-----+-----+-----+ | name | mail | +-----+-----+-----+ | Sarah | sarah@leetcode.com | | Bob | bob@leetcode.com | | Alice | alice@leetcode.com | | Quarz | quarz@leetcode.com | +-----+-----+-----+

Explanation:

Sarah won 3 gold medals (190, 193, and 196), so we include her in the result table. Bob won a medal in 3 consecutive contests (190, 191, and 192), so we include him in the result table. - Note that he also won a medal in 3 other consecutive contests (194, 195, and 196). Alice won a medal in 3 consecutive contests (191, 192, and 193), so we include her in the result table. Quarz won a medal in 5 consecutive contests (190, 191, 192, 193, and 194), so we include them in the result table.

Follow up:

What if the first condition changed to be "any medal in

n

or more

consecutive contests"? How would you change your solution to get the interview candidates? Imagine that

n

is the parameter of a stored procedure.

Some users may not participate in every contest but still perform well in the ones they do. How would you change your solution to only consider contests where the user

was a participant

? Suppose the registered users for each contest are given in another table.

## Code Snippets

### MySQL:

```
# Write your MySQL query statement below
```

### MS SQL Server:

```
/* Write your T-SQL query statement below */
```

### PostgreSQL:

```
-- Write your PostgreSQL query statement below
```

### Oracle:

```
/* Write your PL/SQL query statement below */
```

### Pandas:

```
import pandas as pd

def find_interview_candidates(contests: pd.DataFrame, users: pd.DataFrame) ->
    pd.DataFrame:
```

## Solutions

### MySQL Solution:

```
# Write your MySQL query statement below
```

### MS SQL Server Solution:

```
/* Write your T-SQL query statement below */
```

### PostgreSQL Solution:

```
-- Write your PostgreSQL query statement below
```

### Oracle Solution:

```
/* Write your PL/SQL query statement below */
```

### Pandas Solution:

```
import pandas as pd

def find_interview_candidates(contests: pd.DataFrame, users: pd.DataFrame) ->
    pd.DataFrame:
```