Problem 3586: Find COVID Recovery Patients

Problem Information

Difficulty: Medium
Acceptance Rate: 0.00%
Paid Only: No

Problem Description

Table:

patients

+-------------+---------+ | Column Name | Type | +-------------+---------+ | patient_id | int | | patient_name| varchar | | age | int | +-------------+---------+ patient_id is the unique identifier for this table. Each row contains information about a patient.

Table:

covid_tests

+-------------+---------+ | Column Name | Type | +-------------+---------+ | test_id | int | | patient_id | int | | test_date | date | | result | varchar | +-------------+---------+ test_id is the unique identifier for this table. Each row represents a COVID test result. The result can be Positive, Negative, or Inconclusive.

Write a solution to find patients who have

recovered from COVID

- patients who tested positive but later tested negative.

A patient is considered recovered if they have

at least one

Positive

test followed by at least one

Negative

test on a

later date

Calculate the

recovery time

in days as the

difference

between the

first positive test

and the

first negative test

after that

positive test

Only include

patients who have both positive and negative test results

Return

the result table ordered by

recovery_time

in

ascending

order, then by

patient_name

in

ascending

order

.

The result format is in the following example.

Example:

Input:

patients table:

+------------+--------------+-----+ | patient_id | patient_name | age | +------------+--------------+-----+ | 1 | Alice Smith | 28 | | 2 | Bob Johnson | 35 | | 3 | Carol Davis | 42 | | 4 | David Wilson | 31 | | 5 | Emma Brown | 29 | +------------+--------------+-----+

covid_tests table:

+---------+------------+------------+--------------+ | test_id | patient_id | test_date | result | +---------+------------+------------+--------------+ | 1 | 1 | 2023-01-15 | Positive | | 2 | 1 | 2023-01-25 | Negative | | 3 | 2 | 2023-02-01 | Positive | | 4 | 2 | 2023-02-05 | Inconclusive | | 5 | 2 | 2023-02-12 | Negative | | 6 | 3 | 2023-01-20 | Negative | | 7 | 3 | 2023-02-10 | Positive | | 8 | 3 | 2023-02-20 | Negative | | 9 | 4 | 2023-01-10 | Positive | | 10 | 4 | 2023-01-18 | Positive | | 11 | 5 | 2023-02-15 | Negative | | 12 | 5 | 2023-02-20 | Negative | +---------+------------+------------+--------------+

Output:

+------------+--------------+-----+---------------+ | patient_id | patient_name | age | recovery_time | +------------+--------------+-----+---------------+ | 1 | Alice Smith | 28 | 10 | | 3 | Carol Davis | 42 | 10 | | 2 | Bob Johnson | 35 | 11 | +------------+--------------+-----+---------------+

Explanation:

Alice Smith (patient_id = 1):

First positive test: 2023-01-15

First negative test after positive: 2023-01-25

Recovery time: 25 - 15 = 10 days

Bob Johnson (patient_id = 2):

First positive test: 2023-02-01

Inconclusive test on 2023-02-05 (ignored for recovery calculation)

First negative test after positive: 2023-02-12

Recovery time: 12 - 1 = 11 days

Carol Davis (patient_id = 3):

Had negative test on 2023-01-20 (before positive test)

First positive test: 2023-02-10

First negative test after positive: 2023-02-20

Recovery time: 20 - 10 = 10 days

Patients not included:

David Wilson (patient_id = 4): Only has positive tests, no negative test after positive

Emma Brown (patient_id = 5): Only has negative tests, never tested positive

Output table is ordered by recovery_time in ascending order, and then by patient_name in ascending order.

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_covid_recovery_patients(patients: pd.DataFrame, covid_tests:
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_covid_recovery_patients(patients: pd.DataFrame, covid_tests:
pd.DataFrame) -> pd.DataFrame:
```