

Problem 1919: Leetcodify Similar Friends

Problem Information

Difficulty: Hard

Acceptance Rate: 0.00%

Paid Only: No

Problem Description

Table:

Listens

+-----+-----+ | Column Name | Type | +-----+-----+ | user_id | int | | song_id | int | | day | date | +-----+-----+ This table may contain duplicate rows. Each row of this table indicates that the user user_id listened to the song song_id on the day day.

Table:

Friendship

+-----+-----+ | Column Name | Type | +-----+-----+ | user1_id | int | | user2_id | int | +-----+-----+ (user1_id, user2_id) is the primary key (combination of columns with unique values) for this table. Each row of this table indicates that the users user1_id and user2_id are friends. Note that user1_id < user2_id.

Write a solution to report the similar friends of Leetcodify users. A user

x

and user

y

are similar friends if:

Users

x

and

y

are friends, and

Users

x

and

y

listened to the same three or more different songs

on the same day

.

Return the result table in

any order

. Note that you must return the similar pairs of friends the same way they were represented in the input (i.e., always

user1_id < user2_id

).

The result format is in the following example.

Example 1:

Input:

Listens table: +-----+-----+-----+ | user_id | song_id | day |
+-----+-----+-----+ | 1 | 10 | 2021-03-15 || 1 | 11 | 2021-03-15 || 1 | 12 | 2021-03-15 |
| 2 | 10 | 2021-03-15 || 2 | 11 | 2021-03-15 || 2 | 12 | 2021-03-15 || 3 | 10 | 2021-03-15 || 3 |
11 | 2021-03-15 || 3 | 12 | 2021-03-15 || 4 | 10 | 2021-03-15 || 4 | 11 | 2021-03-15 || 4 | 13 |
2021-03-15 || 5 | 10 | 2021-03-16 || 5 | 11 | 2021-03-16 || 5 | 12 | 2021-03-16 |
+-----+-----+-----+ Friendship table: +-----+-----+ | user1_id | user2_id |
+-----+-----+ | 1 | 2 || 2 | 4 || 2 | 5 | +-----+

Output:

+-----+-----+ | user1_id | user2_id | +-----+-----+ | 1 | 2 | +-----+-----+

Explanation:

Users 1 and 2 are friends, and they listened to songs 10, 11, and 12 on the same day. They are similar friends. Users 1 and 3 listened to songs 10, 11, and 12 on the same day, but they are not friends. Users 2 and 4 are friends, but they did not listen to the same three different songs. Users 2 and 5 are friends and listened to songs 10, 11, and 12, but they did not listen to them on the same day.

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 leetcodify_similar_friends(listens: pd.DataFrame, friendship:
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 leetcodify_similar_friends(listens: pd.DataFrame, friendship:
pd.DataFrame) -> pd.DataFrame:
```