

Problem 597: Friend Requests I: Overall Acceptance Rate

Problem Information

Difficulty: Easy

Acceptance Rate: 0.00%

Paid Only: No

Problem Description

Table:

FriendRequest

+-----+-----+ | Column Name | Type | +-----+-----+ | sender_id | int | |
send_to_id | int | | request_date | date | +-----+-----+ This table may contain
duplicates (In other words, there is no primary key for this table in SQL). This table contains
the ID of the user who sent the request, the ID of the user who received the request, and the
date of the request.

Table:

RequestAccepted

+-----+-----+ | Column Name | Type | +-----+-----+ | requester_id | int | |
accepter_id | int | | accept_date | date | +-----+-----+ This table may contain
duplicates (In other words, there is no primary key for this table in SQL). This table contains
the ID of the user who sent the request, the ID of the user who received the request, and the
date when the request was accepted.

Find the overall acceptance rate of requests, which is the number of acceptance divided by
the number of requests. Return the answer rounded to 2 decimals places.

Note that:

The accepted requests are not necessarily from the table

friend_request

. In this case, Count the total accepted requests (no matter whether they are in the original requests), and divide it by the number of requests to get the acceptance rate.

It is possible that a sender sends multiple requests to the same receiver, and a request could be accepted more than once. In this case, the 'duplicated' requests or acceptances are only counted once.

If there are no requests at all, you should return 0.00 as the

accept_rate

.

The result format is in the following example.

Example 1:

Input:

FriendRequest table: +-----+-----+-----+ | sender_id | send_to_id |
request_date | +-----+-----+-----+ | 1 | 2 | 2016/06/01 | | 1 | 3 | 2016/06/01 | | 1 |
4 | 2016/06/01 | | 2 | 3 | 2016/06/02 | | 3 | 4 | 2016/06/09 | +-----+-----+-----+
RequestAccepted table: +-----+-----+-----+ | requester_id | acceptor_id |
accept_date | +-----+-----+-----+ | 1 | 2 | 2016/06/03 | | 1 | 3 | 2016/06/08 | | 2
| 3 | 2016/06/08 | | 3 | 4 | 2016/06/09 | | 3 | 4 | 2016/06/10 | +-----+-----+-----+

Output:

+-----+ | accept_rate | +-----+ | 0.8 | +-----+

Explanation:

There are 4 unique accepted requests, and there are 5 requests in total. So the rate is 0.80.

Follow up:

Could you find the acceptance rate for every month?

Could you find the cumulative acceptance rate for every 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 acceptance_rate(friend_request: pd.DataFrame, request_accepted:
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 acceptance_rate(friend_request: pd.DataFrame, request_accepted:
pd.DataFrame) -> pd.DataFrame:
```