Problem 1407: Top Travellers

Problem Information

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

Problem Description

Table:

Users

+---------------+---------+ | Column Name | Type | +---------------+---------+ | id | int | | name | varchar | +---------------+---------+ id is the column with unique values for this table. name is the name of the user.

Table:

Rides

+---------------+---------+ | Column Name | Type | +---------------+---------+ | id | int | | user_id | int | | distance | int | +---------------+---------+ id is the column with unique values for this table. user_id is the id of the user who traveled the distance "distance".

Write a solution to report the distance traveled by each user.

Return the result table ordered by

travelled_distance

in

descending order

, if two or more users traveled the same distance, order them by their

name

in

ascending order

.

The result format is in the following example.

Example 1:

Input:

Users table: +------+-----------+ | id | name | +------+-----------+ | 1 | Alice | | 2 | Bob | | 3 | Alex | | 4 | Donald | | 7 | Lee | | 13 | Jonathan | | 19 | Elvis | +------+-----------+ Rides table: +------+----------+----------+ | id | user_id | distance | +------+----------+----------+ | 1 | 1 | 120 | | 2 | 2 | 317 | | 3 | 3 | 222 | | 4 | 7 | 100 | | 5 | 13 | 312 | | 6 | 19 | 50 | | 7 | 7 | 120 | | 8 | 19 | 400 | | 9 | 7 | 230 | +------+----------+----------+

Output:

+----------+--------------------+ | name | travelled_distance | +----------+--------------------+ | Elvis | 450 | | Lee | 450 | | Bob | 317 | | Jonathan | 312 | | Alex | 222 | | Alice | 120 | | Donald | 0 | +----------+--------------------+

Explanation:

Elvis and Lee traveled 450 miles, Elvis is the top traveler as his name is alphabetically smaller than Lee. Bob, Jonathan, Alex, and Alice have only one ride and we just order them by the total distances of the ride. Donald did not have any rides, the distance traveled by him is 0.

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 top_travellers(users: pd.DataFrame, rides: 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 top_travellers(users: pd.DataFrame, rides: pd.DataFrame) -> pd.DataFrame:
```