Problem 1517: Find Users With Valid E-Mails

Problem Information

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

Problem Description

Table:

Users

+---------------+---------+ | Column Name | Type | +---------------+---------+ | user_id | int | | name | varchar | | mail | varchar | +---------------+---------+ user_id is the primary key (column with unique values) for this table. This table contains information of the users signed up in a website. Some e-mails are invalid.

Write a solution to find the users who have

valid emails

.

A valid e-mail has a prefix name and a domain where:

The prefix name

is a string that may contain letters (upper or lower case), digits, underscore

'_'

, period

'.'

, and/or dash

'-'

. The prefix name

must

start with a letter.

The domain

is

'@leetcode.com'

.

Return the result table in

any order

.

The result format is in the following example.

Example 1:

Input:

Users table: +---------+-----------+-------------------------+ | user_id | name | mail | +---------+-----------+-------------------------+ | 1 | Winston | winston@leetcode.com | | 2 | Jonathan | jonathanisgreat | | 3 | Annabelle | bella-@leetcode.com | | 4 | Sally | sally.come@leetcode.com | | 5 | Marwan | quarz#2020@leetcode.com | | 6 | David | david69@gmail.com | | 7 | Shapiro | .shapo@leetcode.com | +---------+-----------+-------------------------+

Output:

+---------+-----------+-----------------------+ | user_id | name | mail | +---------+-----------+-----------------------+ | 1 | Winston | winston@leetcode.com | | 3 | Annabelle | bella-@leetcode.com | | 4 | Sally | sally.come@leetcode.com | +---------+-----------+-----------------------+

Explanation:

The mail of user 2 does not have a domain. The mail of user 5 has the # sign which is not allowed. The mail of user 6 does not have the leetcode domain. The mail of user 7 starts with a period.

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 valid_emails(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 valid_emails(users: pd.DataFrame) -> pd.DataFrame:
```