Problem 182: Duplicate Emails

Problem Information

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

Problem Description

Table:

Person

+-------------+---------+ | Column Name | Type | +-------------+---------+ | id | int | | email | varchar | +-------------+---------+ id is the primary key (column with unique values) for this table. Each row of this table contains an email. The emails will not contain uppercase letters.

Write a solution to report all the duplicate emails. Note that it's guaranteed that the email field is not NULL.

Return the result table in

any order

.

The result format is in the following example.

Example 1:

Input:

Person table: +----+---------+ | id | email | +----+---------+ | 1 | a@b.com | | 2 | c@d.com | | 3 | a@b.com | +----+---------+

Output:

+---------+ | Email | +---------+ | a@b.com | +---------+

Explanation:

a@b.com is repeated two times.

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 duplicate_emails(person: 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 duplicate_emails(person: pd.DataFrame) -> pd.DataFrame:
```