

# Problem 1511: Customer Order Frequency

## Problem Information

Difficulty: [Easy](#)

Acceptance Rate: 0.00%

Paid Only: No

## Problem Description

Table:

Customers

+-----+-----+ | Column Name | Type | +-----+-----+ | customer\_id | int || name | varchar | | country | varchar | +-----+-----+ customer\_id is the column with unique values for this table. This table contains information about the customers in the company.

Table:

Product

+-----+-----+ | Column Name | Type | +-----+-----+ | product\_id | int || description | varchar | | price | int | +-----+-----+ product\_id is the column with unique values for this table. This table contains information on the products in the company. price is the product cost.

Table:

Orders

+-----+-----+ | Column Name | Type | +-----+-----+ | order\_id | int || customer\_id | int | | product\_id | int | | order\_date | date | | quantity | int | +-----+-----+ order\_id is the column with unique values for this table. This table contains information on customer orders. customer\_id is the id of the customer who bought "quantity" products with id "product\_id". Order\_date is the date in format ('YYYY-MM-DD') when the order was shipped.

Write a solution to report the

customer\_id

and

customer\_name

of customers who have spent at least

\$100

in each month of

June and July 2020

Return the result table in

any order

The result format is in the following example.

Example 1:

Input:

Customers table: +-----+-----+-----+ | customer\_id | name | country |  
+-----+-----+-----+ | 1 | Winston | USA | | 2 | Jonathan | Peru | | 3 | Moustafa |  
Egypt | +-----+-----+-----+ Product table: +-----+-----+-----+ |  
product\_id | description | price | +-----+-----+-----+ | 10 | LC Phone | 300 | |  
20 | LC T-Shirt | 10 | | 30 | LC Book | 45 | | 40 | LC Keychain | 2 |  
+-----+-----+-----+ Orders table:  
+-----+-----+-----+-----+ | order\_id | customer\_id | product\_id |  
| order\_date | quantity | +-----+-----+-----+-----+ | 1 | 1 | 10 |  
2020-06-10 | 1 | | 2 | 1 | 20 | 2020-07-01 | 1 | | 3 | 1 | 30 | 2020-07-08 | 2 | | 4 | 2 | 10 |

```
2020-06-15 | 2 | | 5 | 2 | 40 | 2020-07-01 | 10 | | 6 | 3 | 20 | 2020-06-24 | 2 | | 7 | 3 | 30 |
2020-06-25 | 2 | | 9 | 3 | 30 | 2020-05-08 | 3 |
+-----+-----+-----+-----+
```

Output:

```
+-----+-----+ | customer_id | name | +-----+-----+ | 1 | Winston |
+-----+-----+
```

Explanation:

Winston spent \$300 ( $300 * 1$ ) in June and \$100 ( $10 * 1 + 45 * 2$ ) in July 2020. Jonathan spent \$600 ( $300 * 2$ ) in June and \$20 ( $2 * 10$ ) in July 2020. Moustafa spent \$110 ( $10 * 2 + 45 * 2$ ) in June and \$0 in July 2020.

## 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 customer_order_frequency(customers: pd.DataFrame, product: pd.DataFrame,
                           orders: 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 customer_order_frequency(customers: pd.DataFrame, product: pd.DataFrame,
orders: pd.DataFrame) -> pd.DataFrame:
```