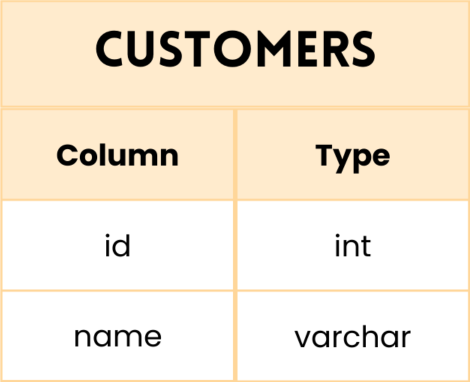
Customers Who Never Order

Question:

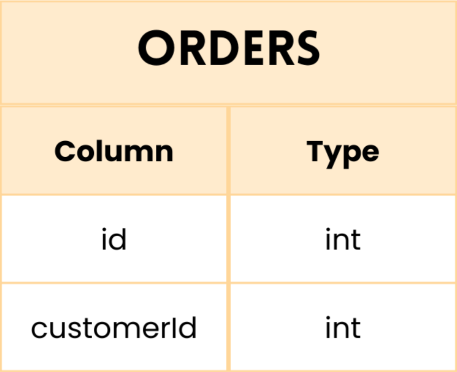
https://leetcode.com/problems/customers-who-never-order/



CUSTOMERS	
Column	**Type**
id	int
name	varchar

id is the primary key column for this table.

Each row of this table indicates the ID and name of a customer.



ORDERS	
Column	**Type**
id	int
customerId	int

id is the primary key column for this table.

customerId is a foreign key of the ID from the Customers table.

Each row of this table indicates the ID of an order and the ID of the customer who ordered it.

Write an SQL query to report all customers who never order anything.

Return the result table in any order.

Query 1

Initially I tried to find all the customers that ordered.
Customers that have **id** in the **Customers DB** as well in the **Orders DB**.
So basically I tried filtering out **rows** with the condition **Customer.id = Orders.customerId**

```
SELECT

    Customers.id as id

FROM Customers

JOIN Orders

ON Customers.id = Orders.customerId))
```

Luckily this became the subquery for Query 2.

Query 2

Using Query 1, we found out the id's of customers that ordered. Now we can just find rows that are not in our subquery and we will find the customer id's that didn't order.

```
SELECT

    name as Customers

FROM Customers

WHERE (id NOT IN (SELECT

                         Customers.id as id

                  FROM Customers

                  JOIN Orders

                  ON Customers.id = customerId))
```

Although this worked well, I personally hate using Subqueries as it takes an overload of scanning the entire database twice.

Query 3

A simple trick I learned while solving SQL is that to optimize subqueries, I use Join. The type of Join depends on the output required.

Since in this case, our focus was the **Customer DB** so I used **LEFT JOIN** and just filtered ID’s that had no customerID in **Order DB**.

```
SELECT

    name as Customers

FROM Customers C

    LEFT JOIN Orders O

    ON C.id = O.customerId

WHERE O.customerId IS NULL
```

As expected , Query 3 was the fastest as JOIN only traverses the Database once compared to the 2 traverses of Subquery.