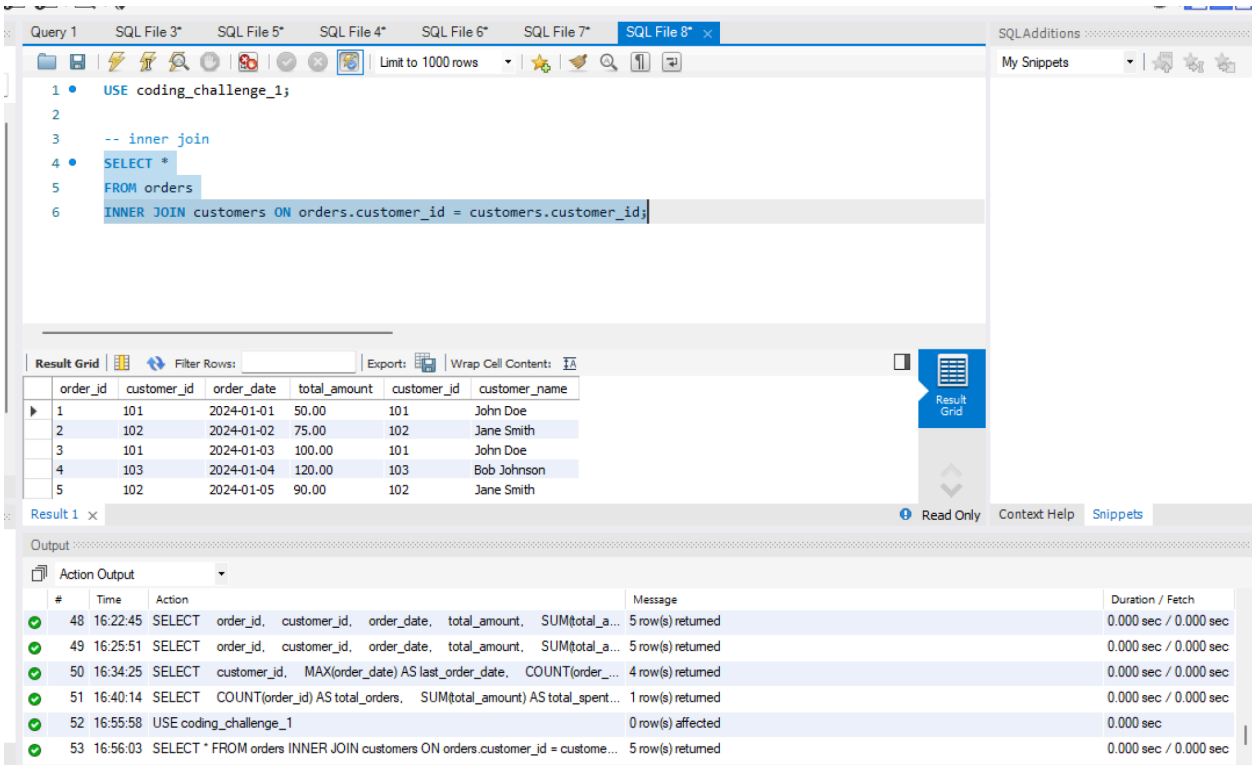
Asmita Porwal
Data Engineering
Batch-1
25/1/24

Coding Challenge -1 Question-2

Execute all the joins with examples.

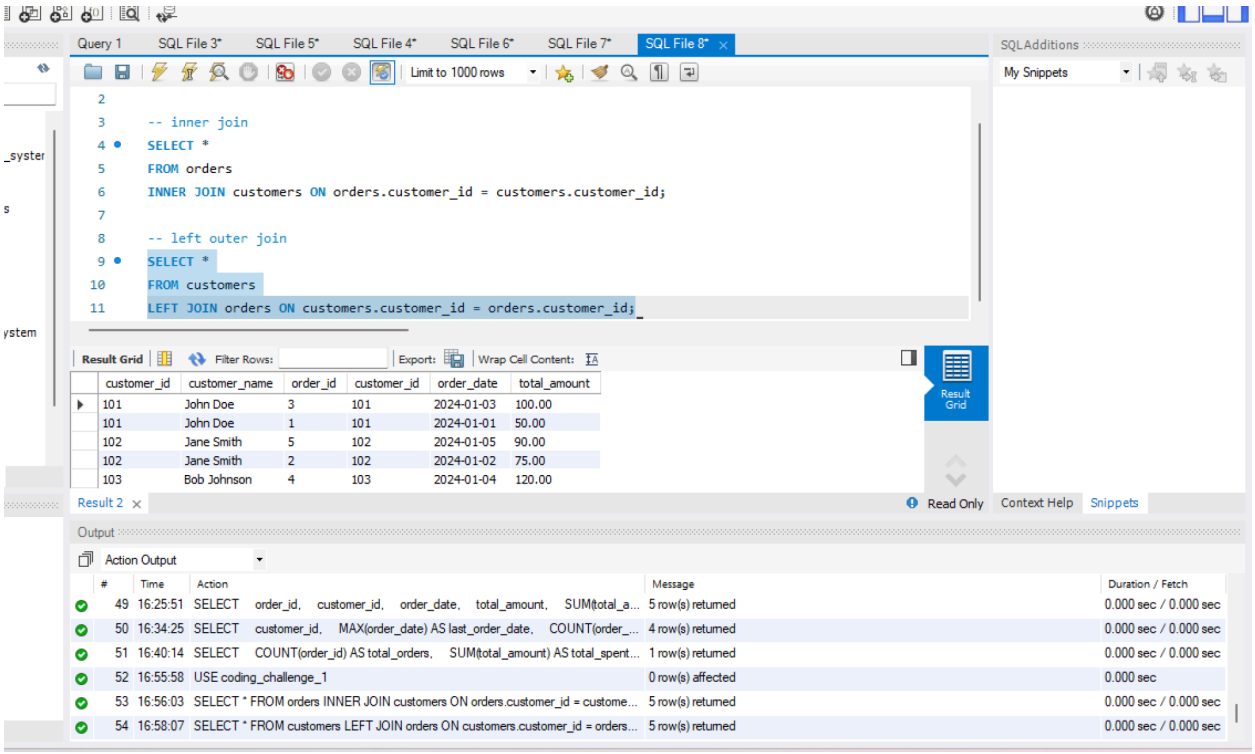
1.Inner Join

An inner join returns only the rows where there is a match in both tables.



2.Left Join (or Left Outer Join):

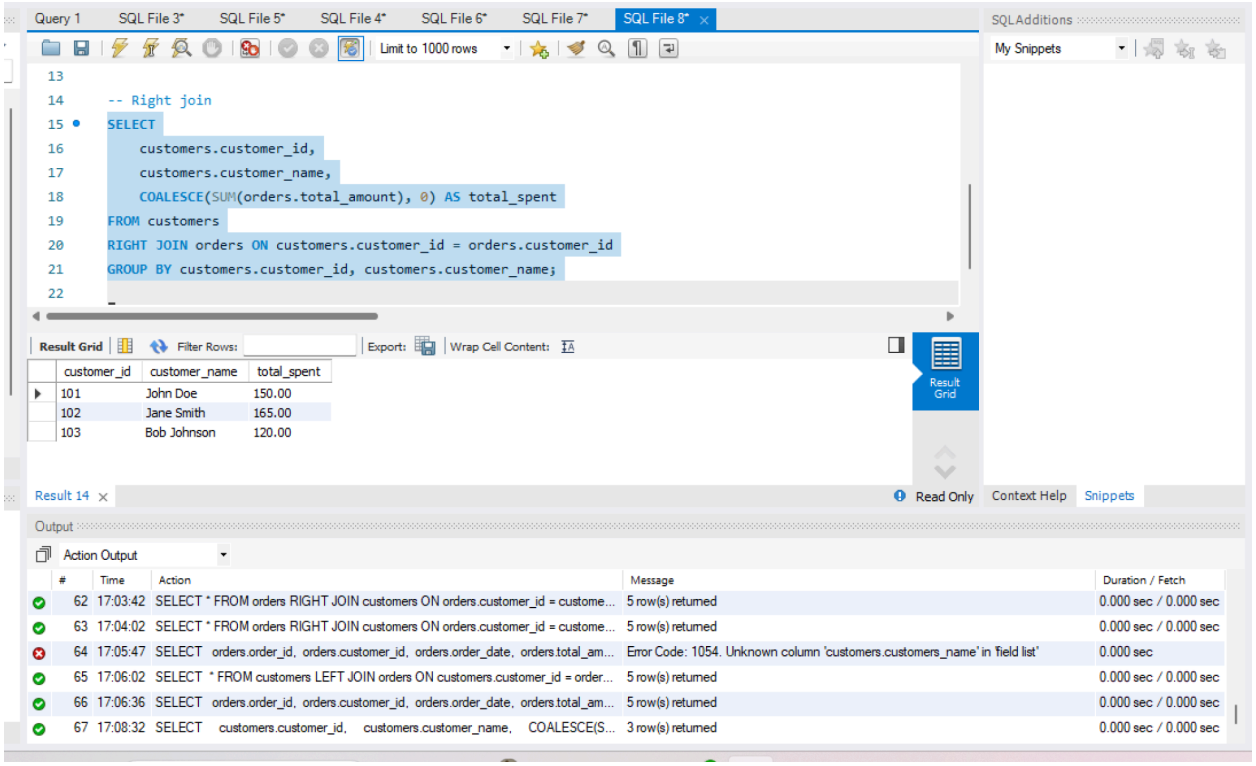
A left join returns all rows from the left table and the matched rows from the right table. If there is no match, NULL values are returned for columns from the right table.



3.Right Join (or Right Outer Join):

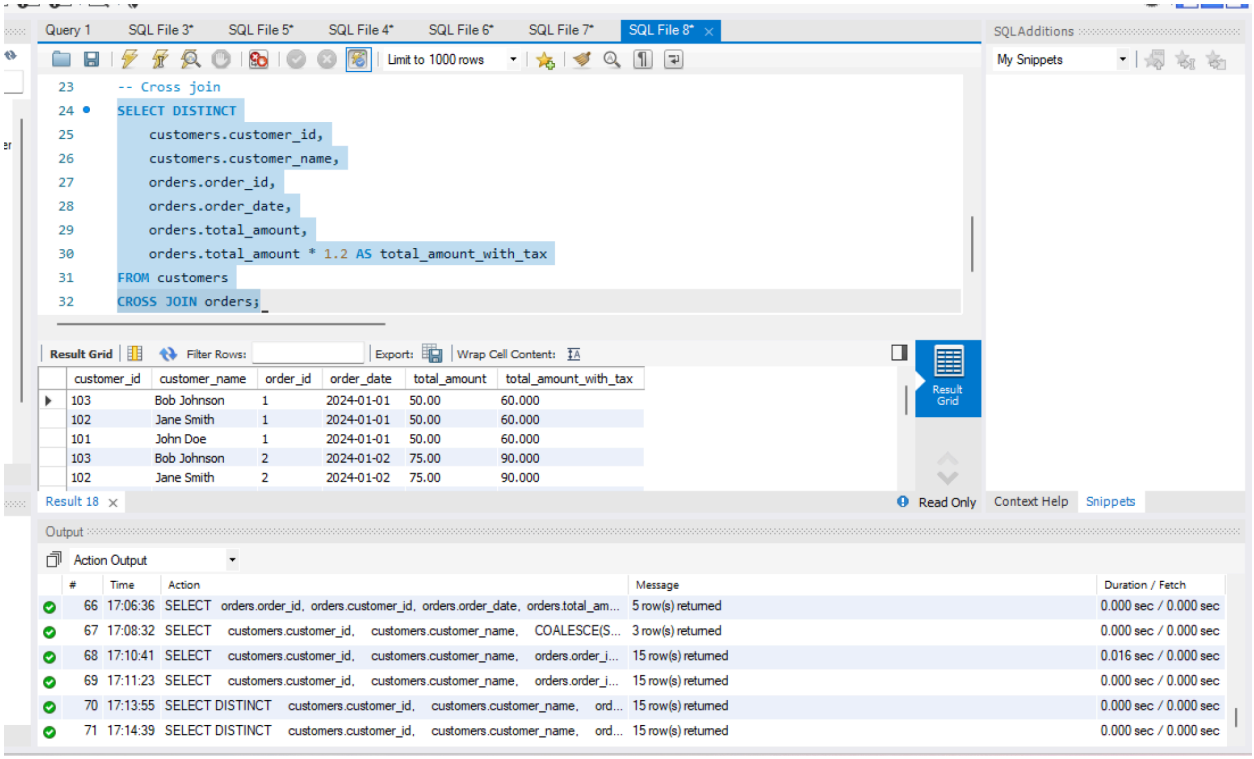
A right join returns all rows from the right table and the matched rows from the left table. If there is no match, NULL values are returned for columns from the left table.

This query is giving the total amount spent by each individual customer.



4. Cross Join

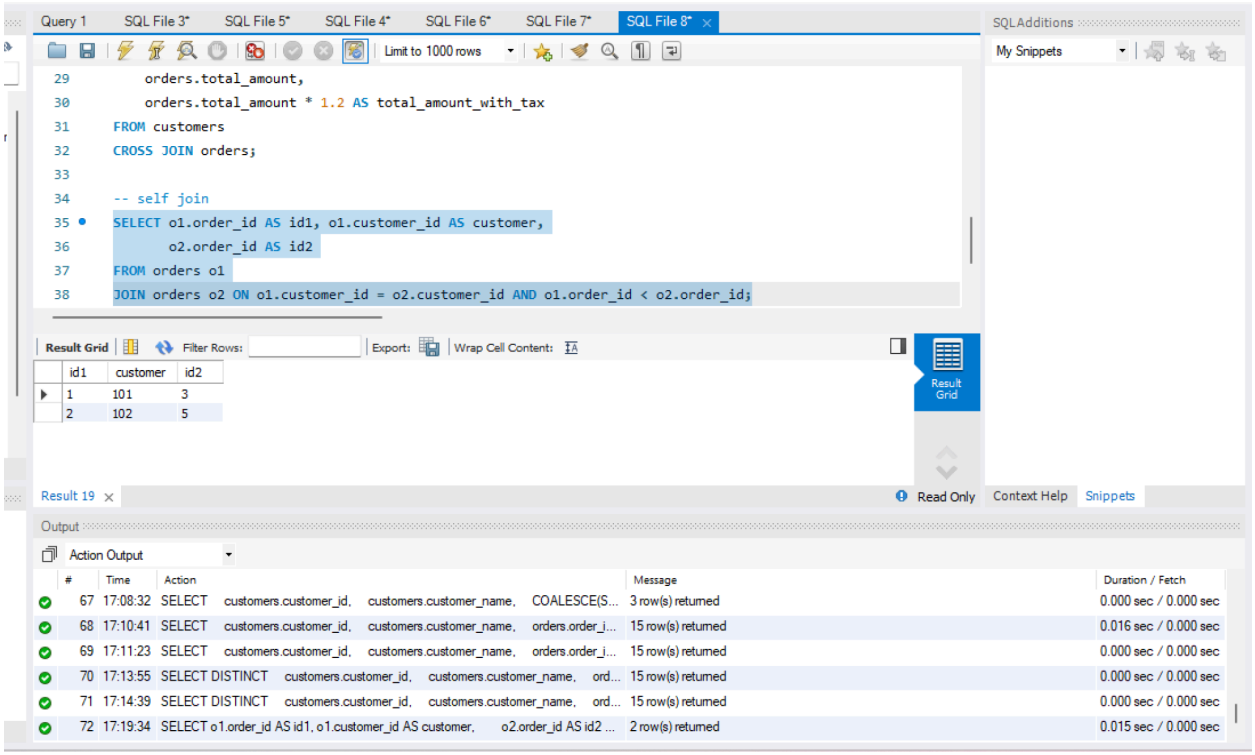
CROSS JOIN returns the Cartesian product of rows from both tables.



5.Self Join:

A self join is a regular join, but the table is joined with itself.

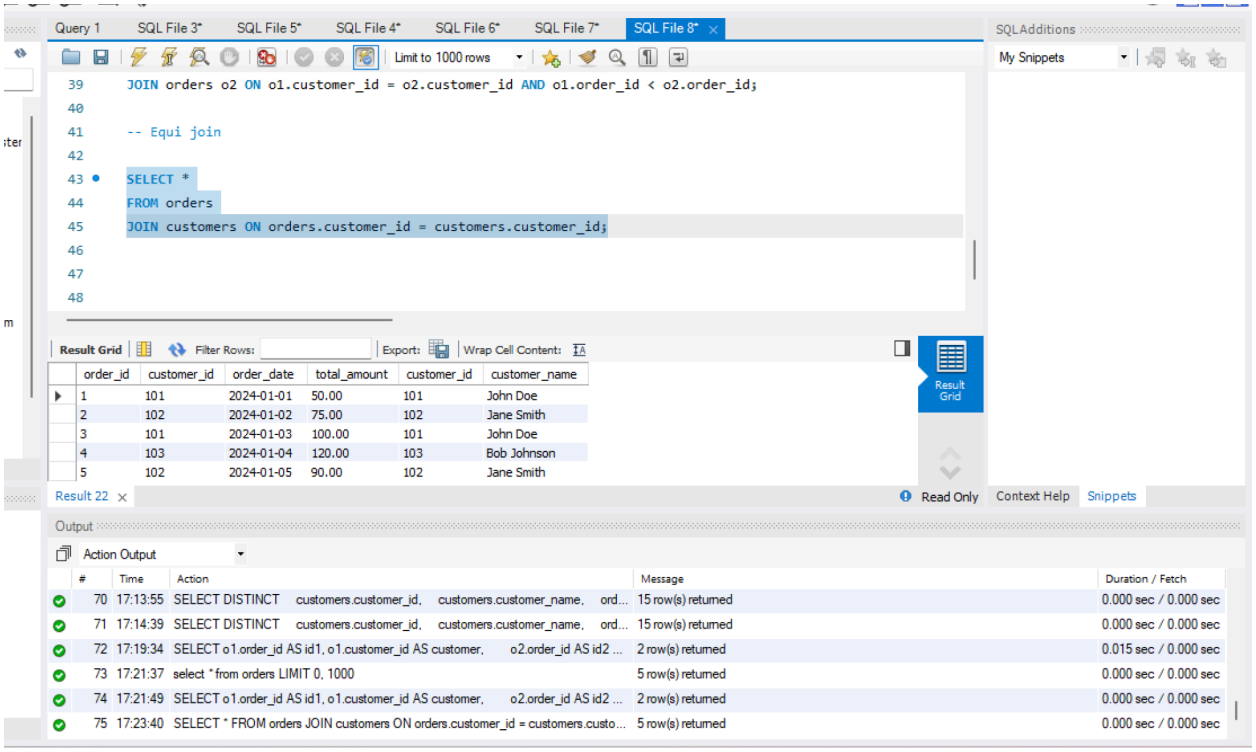
The purpose of this query is to identify pairs of orders made by the same customer where the order with id1 was placed before the order with id2



6.Equi Join:

An Equi join is a join using the equality operator (=) in the join condition.

This query is commonly used when you want to retrieve comprehensive information that involves columns from multiple tables based on a shared key, in this case, the **customer_id**. It provides a consolidated view of orders along with the details of the customers who placed those orders.

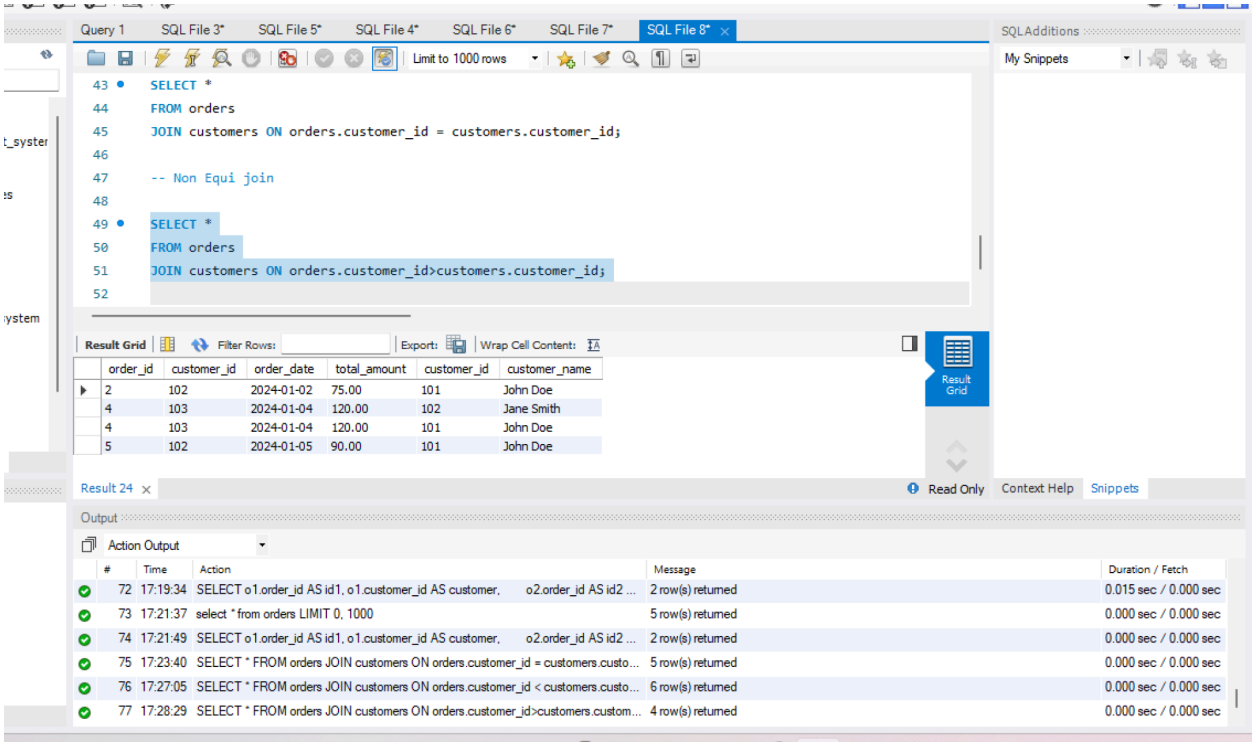


7.Non-Equi Join:

A non-equi join is a join using operators other than equality, such as <, >, <=, >=.

The query will return pairs of rows from the "orders" and "customers" tables where the customer_id in the "orders" table is greater than the customer_id in the "customers" table.

In other words, it will give combinations of orders and customers where the customer who placed the order has a higher customer_id than the customer being joined.



8.Natural join

A NATURAL JOIN is a type of join in SQL that automatically matches columns with the same name in both tables, eliminating the need to explicitly specify the join condition.

Query 1 | SQL File 3* | SQL File 5* | SQL File 4* | SQL File 6* | SQL File 7* | SQL File 8*

Limit to 1000 rows

```sql
-- natural join

SELECT *
FROM orders
NATURAL JOIN customers;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

customer_id	order_id	order_date	total_amount	customer_name
101	1	2024-01-01	50.00	John Doe
102	2	2024-01-02	75.00	Jane Smith
101	3	2024-01-03	100.00	John Doe
103	4	2024-01-04	120.00	Bob Johnson
102	5	2024-01-05	90.00	Jane Smith

Result 25 | Read Only | Context Help | Snippets

SQL Additions | My Snippets

Output | Action Output

#	Time	Action	Message	Duration / Fetch
73	17:21:37	select * from orders LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
74	17:21:49	SELECT o1.order_id AS id1, o1.customer_id AS customer, o2.order_id AS id2 ...	2 row(s) returned	0.000 sec / 0.000 sec
75	17:23:40	SELECT * FROM orders JOIN customers ON orders.customer_id = customers.custo...	5 row(s) returned	0.000 sec / 0.000 sec
76	17:27:05	SELECT * FROM orders JOIN customers ON orders.customer_id < customers.custo...	6 row(s) returned	0.000 sec / 0.000 sec
77	17:28:29	SELECT * FROM orders JOIN customers ON orders.customer_id>customers.custom...	4 row(s) returned	0.000 sec / 0.000 sec
78	17:34:39	SELECT * FROM orders NATURAL JOIN customers LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec