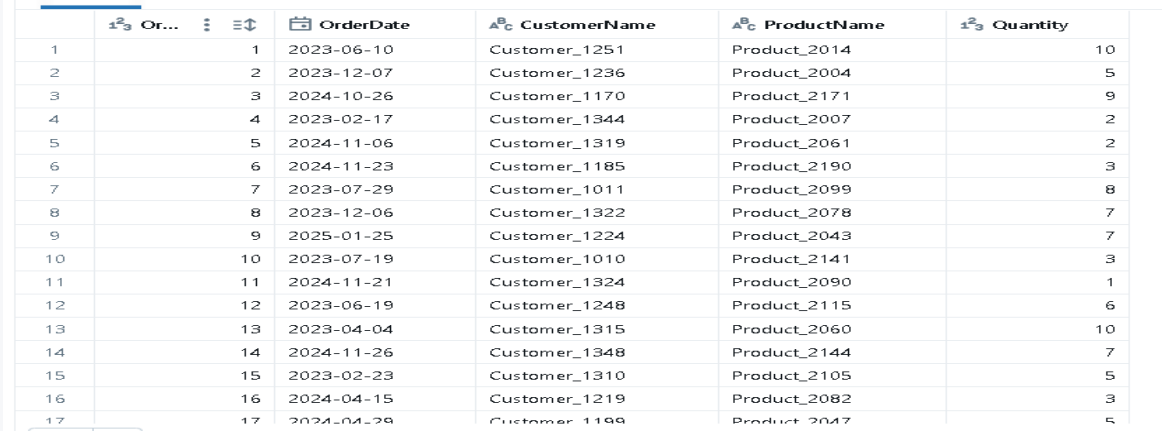
Mavis Mabena : Practical 1- SQL Fundumentals (SQL JOINTS)

```
BrightLight Data Analytics
SQL JOIN Practice Instructions
General Guidelines:
1. Use the provided CSV files:
o orders_large.csv
o products_large.csv
o customers_large.csv
2. Import the datasets into a SQL database system of your choice (e.g., MySQL,
PostgreSQL, SQLite, or Snowflake).
3. Use the appropriate JOINs (INNER JOIN, LEFT JOIN, RIGHT JOIN, FULL OUTER
JOIN) to answer each question.
4. Ensure your query returns exactly the expected columns listed for each
question.
5. Use aliases and format your SQL code for readability.
6. When using FULL OUTER JOIN, make sure your SQL engine supports it (e.g.,
MySQL does not support it natively – use UNION of LEFT JOIN and RIGHT JOIN if
needed).
```

```
1. INNER JOIN: Orders with Customer and Product Names
Question:
List all orders along with the customer name and product name.
Expected Output Columns:
• OrderID, OrderDate, CustomerName, ProductName, Quantity

SELECT
       A. OrderID,
       A. OrderDate,
       C. CustomerName,
       B. ProductName,
       A. Quantity

FROM orders_large_1 AS A
INNER JOIN products_large_1 AS B ON A.ProductID = B.ProductID
INNER JOIN customers_large_1 AS C ON A.CustomerID = C.CustomerID;
```



	Or...	OrderDate	CustomerName	ProductName	Quantity
1	1	2023-06-10	Customer_1251	Product_2014	10
2	2	2023-12-07	Customer_1236	Product_2004	5
3	3	2024-10-26	Customer_1170	Product_2171	9
4	4	2023-02-17	Customer_1344	Product_2007	2
5	5	2024-11-06	Customer_1319	Product_2061	2
6	6	2024-11-23	Customer_1185	Product_2190	3
7	7	2023-07-29	Customer_1011	Product_2099	8
8	8	2023-12-06	Customer_1322	Product_2078	7
9	9	2025-01-25	Customer_1224	Product_2043	7
10	10	2023-07-19	Customer_1010	Product_2141	3
11	11	2024-11-21	Customer_1324	Product_2090	1
12	12	2023-06-19	Customer_1248	Product_2115	6
13	13	2023-04-04	Customer_1315	Product_2060	10
14	14	2024-11-26	Customer_1348	Product_2144	7
15	15	2023-02-23	Customer_1310	Product_2105	5
16	16	2024-04-15	Customer_1219	Product_2082	3
17	17	2024-04-29	Customer_1199	Product_2047	5

2. INNER JOIN: Customers Who Placed Orders
Question:
Which customers have placed at least one order?
Expected Output Columns:
• CustomerID, CustomerName, Country, OrderID, OrderDate

```
SELECT C. CustomerID,
       C. CustomerName,
       C. Country,
       A. OrderID,
       A. OrderDate

FROM orders_large_1 AS A
INNER JOIN customers_large_1 AS C ON A.CustomerID = C.CustomerID
WHERE A. Quantity >=1;
```

Table +

	CustomerID	Customer...	Country	OrderID	OrderDate
1	1251	Customer_1251	Germany	1	2023-06-10
2	1236	Customer_1236	Australia	2	2023-12-07
3	1170	Customer_1170	Germany	3	2024-10-26
4	1344	Customer_1344	Canada	4	2023-02-17
5	1319	Customer_1319	USA	5	2024-11-06
6	1185	Customer_1185	Australia	6	2024-11-23
7	1011	Customer_1011	Germany	7	2023-07-29
8	1322	Customer_1322	Australia	8	2023-12-06
9	1224	Customer_1224	Australia	9	2025-01-25
10	1010	Customer_1010	UK	10	2023-07-19
11	1324	Customer_1324	India	11	2024-11-21
12	1248	Customer_1248	Germany	12	2023-06-19
13	1315	Customer_1315	Canada	13	2023-04-04
14	1348	Customer_1348	USA	14	2024-11-26
15	1310	Customer_1310	Germany	15	2023-02-23
16	1219	Customer_1219	Germany	16	2024-04-15
17	1199	Customer_1199	USA	17	2024-04-29

3. LEFT JOIN: All Customers and Their Orders
Question:
List all customers and any orders they might have placed. Include customers who have not placed any orders.
Expected Output Columns:
• CustomerID, CustomerName, Country, OrderID, OrderDate, ProductID, Quantity

```
SELECT C. CustomerID,
       C. CustomerName,
       C. Country,
       A. OrderID,
       A. OrderDate,
       B. ProductID,
       A. Quantity

       FROM orders_large_1 AS A
       LEFT JOIN products_large_1 AS B ON A. ProductID = B.ProductID
       LEFT JOIN customers_large_1 AS C ON A.CustomerID = C.CustomerID;
```

Table +

	CustomerID	CustomerName	Country	OrderID	OrderDate	ProductID	Quantity
1	1251	Customer_1251	Germany	1	2023-06-10	2014	10
2	1236	Customer_1236	Australia	2	2023-12-07	2004	5
3	1170	Customer_1170	Germany	3	2024-10-26	2171	9
4	1344	Customer_1344	Canada	4	2023-02-17	2007	2
5	1319	Customer_1319	USA	5	2024-11-06	2061	2
6	1185	Customer_1185	Australia	6	2024-11-23	2190	3
7	1011	Customer_1011	Germany	7	2023-07-29	2099	8
8	1322	Customer_1322	Australia	8	2023-12-06	2078	7
9	1224	Customer_1224	Australia	9	2025-01-25	2043	7
10	1010	Customer_1010	UK	10	2023-07-19	2141	3
11	1324	Customer_1324	India	11	2024-11-21	2090	1
12	1248	Customer_1248	Germany	12	2023-06-19	2115	6
13	1315	Customer_1315	Canada	13	2023-04-04	2060	10
14	1348	Customer_1348	USA	14	2024-11-26	2144	7
15	1310	Customer_1310	Germany	15	2023-02-23	2105	5
16	1219	Customer_1219	Germany	16	2024-04-15	2082	3
17	1199	Customer_1199	USA	17	2024-04-29	2047	5

4. LEFT JOIN: Product Order Count
Question:
List all products and how many times each was ordered (if any).
Expected Output Columns:
• ProductID, ProductName, TotalOrders
(TotalOrders is the count of how many times the product appears in orders)

```
SELECT A. ProductID,
       A. ProductName,
       COUNT(DISTINCT(B.OrderID)) AS Total_orders
FROM products_large_1 AS A
left JOIN orders_large_1 AS B ON A.ProductID = B.ProductID
GROUP BY A.ProductID, A.ProductName;
```

Table +

	ProductID	ProductName	Total_orders
1	2075	Product_2075	21
2	2173	Product_2173	16
3	2167	Product_2167	27
4	2172	Product_2172	18
5	2174	Product_2174	13
6	2125	Product_2125	19
7	2057	Product_2057	13
8	2035	Product_2035	18
9	2141	Product_2141	24
10	2126	Product_2126	21
11	2024	Product_2024	17
12	2136	Product_2136	27

5. RIGHT JOIN: Orders with Product Info (Include Products Not Ordered)
Question:
Find all orders along with product details, including any products that might not have
been ordered.
Expected Output Columns:
• OrderID, OrderDate, ProductID, ProductName, Price, Quantity

```
SELECT A. OrderID,
       A. OrderDate,
       B. ProductID,
       B. ProductName,
       B. Price,
       A. Quantity

FROM orders_large_1 AS A
RIGHT JOIN Products_large_1 AS B ON A.ProductID = B.ProductID;
```

Table +

	OrderID	OrderDate	ProductID	ProductName	Price	Quantity
1	3788	2023-11-15	2001	Product_2001	833	7
2	3767	2023-02-11	2002	Product_2002	1558	3
3	3925	2023-02-28	2003	Product_2003	1398	4
4	3974	2025-01-30	2004	Product_2004	1996	3
5	3786	2023-06-18	2005	Product_2005	1146	3
6	3719	2024-06-04	2006	Product_2006	694	5
7	2896	2023-10-11	2007	Product_2007	156	9
8	3971	2023-10-07	2008	Product_2008	811	8
9	3972	2025-01-21	2009	Product_2009	342	8
10	3936	2023-10-12	2010	Product_2010	1864	7
11	3853	2024-11-22	2011	Product_2011	881	1
12	3949	2024-11-30	2012	Product_2012	129	2

6. RIGHT JOIN: Customer Info with Orders (Include All Customers)
Question:
Which customers have made orders, and include customers even if they have never placed an order.
Expected Output Columns:
• CustomerID, CustomerName, Country, OrderID, OrderDate, ProductID, Quantity

```
SELECT A. CustomerID,
       A. CustomerName,
       A. Country,
       B. OrderID,
       B. OrderDate,
       B. ProductID,
       B. Quantity

FROM customers_large_1 AS A
LEFT JOIN orders_large_1 AS B ON A.CustomerID = B.CustomerID;
```

Table +

	CustomerID	CustomerName	Country	OrderID	OrderDate	ProductID	Quantity
1	1001	Customer_1001	India	3408	2024-04-15	2100	5
2	1002	Customer_1002	Germany	3269	2024-06-07	2058	9
3	1003	Customer_1003	USA	3318	2024-10-17	2144	10
4	1004	Customer_1004	Germany	3910	2024-04-20	2145	8
5	1005	Customer_1005	USA	3986	2024-03-27	2111	2
6	1006	Customer_1006	India	3875	2024-08-03	2132	10
7	1007	Customer_1007	Germany	3803	2023-11-22	2053	4
8	1008	Customer_1008	UK	3814	2023-04-27	2195	9
9	1009	Customer_1009	Canada	3985	2024-10-19	2164	4
10	1010	Customer_1010	UK	3316	2023-11-21	2162	5
11	1011	Customer_1011	Germany	3942	2025-02-08	2175	10

6. RIGHT JOIN: Customer Info with Orders (Include All Customers)

Question:

Which customers have made orders, and include customers even if they have never placed an order.

Expected Output Columns:

• CustomerID, CustomerName, Country, OrderID, OrderDate, ProductID, Quantity

```
SELECT A. CustomerID,
       A. CustomerName,
       A. Country,
       B. OrderID,
       B. OrderDate,
       B. ProductID,
       B. Quantity

FROM customers_large_1 AS A
LEFT JOIN orders_large_1 AS B ON A.CustomerID = B.CustomerID;
```

Table +

	CustomerID	CustomerName	Country	OrderID	OrderDate	ProductID	Quantity
1	1001	Customer_1001	India	3408	2024-04-15	2100	5
2	1002	Customer_1002	Germany	3269	2024-06-07	2058	9
3	1003	Customer_1003	USA	3318	2024-10-17	2144	10
4	1004	Customer_1004	Germany	3910	2024-04-20	2145	8
5	1005	Customer_1005	USA	3986	2024-03-27	2111	2
6	1006	Customer_1006	India	3875	2024-08-03	2132	10
7	1007	Customer_1007	Germany	3803	2023-11-22	2053	4
8	1008	Customer_1008	UK	3814	2023-04-27	2195	9
9	1009	Customer_1009	Canada	3985	2024-10-19	2164	4
10	1010	Customer_1010	UK	3316	2023-11-21	2162	5
11	1011	Customer_1011	Germany	3942	2025-02-08	2175	10
12	1012	Customer_1012	Canada	3996	2024-06-10	2038	10

7. FULL OUTER JOIN: All Customers and All Orders

Question:

List all customers and orders, showing NULLs where customers have not ordered or where orders have no customer info.

Expected Output Columns:

• CustomerID, CustomerName, Country, OrderID, OrderDate, ProductID, Quantity

```
SELECT A. CustomerID,
       A. CustomerName,
       A. Country,
       B. OrderID,
       B. OrderDate,
       B. ProductID,
       B. Quantity

FROM customers_large_1 AS A
FULL OUTER JOIN orders_large_1 AS B ON A.CustomerID = B.CustomerID;
```

Table +

	CustomerID	CustomerName	Country	OrderID	OrderDate	ProductID	Quantity
1	1315	Customer_1315	Canada	13	2023-04-04	2060	10
2	1219	Customer_1219	Germany	16	2024-04-15	2082	3
3	1044	Customer_1044	Canada	116	2023-08-31	2114	4
4	1283	Customer_1283	Canada	194	2024-11-12	2153	7
5	1044	Customer_1044	Canada	196	2025-02-08	2165	3
6	1168	Customer_1168	India	197	2023-10-26	2122	8
7	1150	Customer_1150	Australia	236	2023-03-17	2079	5
8	1226	Customer_1226	Australia	255	2023-12-05	2195	7
9	1177	Customer_1177	USA	256	2025-01-14	2006	6
10	1238	Customer_1238	Australia	266	2023-03-07	2010	4

8. FULL OUTER JOIN: All Products and Orders

Question:

List all products and orders, showing NULLs where products were never ordered or orders are missing product info.

Expected Output Columns:

• ProductID, ProductName, Price, OrderID, OrderDate, CustomerID, Quantity

```
SELECT B. CustomerID,
       A. ProductName,
       A. Price,
       B. OrderID,
       B. OrderDate,
       B. ProductID,
```

```
        B. Quantity

FROM products_large_1 AS A
FULL OUTER JOIN orders_large_1 AS B ON A.ProductID = B.ProductID
FULL OUTER JOIN customers_large_1 AS C  ON C.CustomerID = B.CustomerID;
```

Table +

	CustomerID	ProductName	Price	OrderID	OrderDate	ProductID	Quantity
1	1133	Product_2085	574	361	2023-04-09	2085	2
2	1130	Product_2188	861	370	2024-02-08	2188	5
3	1025	Product_2054	620	582	2024-08-31	2054	9
4	1315	Product_2154	1195	695	2023-12-27	2154	3
5	1122	Product_2075	667	1074	2024-11-19	2075	10
6	1238	Product_2185	791	1105	2024-04-03	2185	4
7	1117	Product_2023	1465	1289	2023-03-18	2023	7
8	1126	Product_2025	941	1438	2023-01-31	2025	1
9	1004	Product_2054	620	1460	2023-10-29	2054	8
10	1180	Product_2068	1395	1628	2024-06-20	2068	6