

Thatiso Sethema

Practical 2 – SQL JOINS

Question 1

The screenshot shows the BrightLight Tutorials interface with a SQL worksheet titled 'practical_2.sql'. The code is a three-table join:

```
SELECT orderid,
       orderdate,
       customername,
       productname,
       quantity
  FROM practical2.joins.orders AS A
 INNER JOIN practical2.joins.customers AS B
 ON A.customerid = B.customerid
 INNER JOIN practical2.joins.products AS C
 ON A.productid = C.productid;
```

The results table shows 4,000 rows in 237ms. The columns are ORDERID, ORDERDATE, CUSTOMERNAME, PRODUCTNAME, and QUANTITY.

#	ORDERID	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_2081	2
6	6	2024-11-23	Customer_1185	Product_2180	3
7	7	2023-07-29	Customer_1011	Product_2099	8

Question 2

The screenshot shows the BrightLight Tutorials interface with a SQL worksheet titled 'practical_2.sql'. The code is a three-table join:

```
-- 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 A.customerid,
       A.customername,
       A.country,
       B.orderid,
       B.orderdate
  FROM practical2.joins.customers AS A
 INNER JOIN practical2.joins.orders AS B
 ON A.customerid = B.customerid;
| Ctrl+I to generate
```

The results table shows 4,000 rows in 34ms. The columns are CUSTOMERID, CUSTOMERNAME, COUNTRY, ORDERID, and ORDERDATE.

#	CUSTOMERID	CUSTOMERNAME	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

Question 3

The screenshot shows the Snowflake SQL interface. The code in the editor is:

```
-- 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 A.customerid,
       A.customername,
       A.country,
       B.orderid,
       B.orderdate,
       B.productid,
       B.quantity
FROM practical2.joins.customers AS A
LEFT JOIN practical2.joins.orders AS B
ON A.customerid = B.customerid;
```

The results table shows the following data:

# CUSTOMERID	A CUSTOMERNAME	A COUNTRY	# ORDERID	ORDERDATE	# PRODUCTID	# QUANTITY
1	Customer_1251	Germany	1	2023-06-10	2014	10
2	Customer_1236	Australia	2	2023-12-07	2004	5
3	Customer_1170	Germany	3	2024-10-26	2171	9
4	Customer_1344	Canada	4	2023-02-17	2007	2
5	Customer_1319	USA	5	2024-11-06	2061	2

Question 4

The screenshot shows the Snowflake SQL interface. The code in the editor is:

```
-- 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(B.orderid) AS totalorders
FROM practical2.joins.products AS A
LEFT JOIN practical2.joins.orders AS B
ON A.productid = B.productid
GROUP BY A.productid, A.productname;
```

The results table shows the following data:

# PRODUCTID	A PRODUCTNAME	# TOTALORDERS
1	Product_2171	15
2	Product_2177	20
3	Product_2073	19
4	Product_2089	20
5	Product_2054	24
6	Product_2019	17
7	Product_2190	20

Question 5

The screenshot shows the Snowflake web interface. The query editor window displays a SQL script named 'practical_2.sql' with the following code:

```
-- 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 practical2.joins.orders AS A
RIGHT JOIN practical2.joins.products AS B
ON A.productid = B.productid;
```

The results table shows the following data:

# ORDERID	ORDERDATE	# PRODUCTID	PRODUCTNAME	# PRICE	# QUANTITY
1	2023-06-10	2014	Product_2014	522	10
2	2023-12-07	2004	Product_2004	1996	5
3	2024-10-26	2171	Product_2171	76	9
4	2023-02-17	2007	Product_2007	156	2
5	2024-11-06	2061	Product_2061	1595	2
6	2024-11-23	2190	Product_2190	1755	3

Question 6

The screenshot shows the Snowflake web interface. The query editor window displays a SQL script named 'practical_2.sql' with the following code:

```
-- 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,
       C.productid,
       B.quantity
FROM practical2.joins.customers AS A
RIGHT JOIN practical2.joins.orders AS B
ON A.customerid = B.customerid
RIGHT JOIN practical2.joins.products AS C
ON B.productid = C.productid;
```

The results table shows the following data:

# CUSTOMERID	CUSTOMERNAME	COUNTRY	# ORDERID	ORDERDATE	# PRODUCTID	# QUANTITY
1251	Customer_1251	Germany	1	2023-06-10	2014	10
1236	Customer_1236	Australia	2	2023-12-07	2004	5
1170	Customer_1170	Germany	3	2024-10-26	2171	9
1344	Customer_1344	Canada	4	2023-02-17	2007	2

Question 7

The screenshot shows the Snowflake SQL interface with a query editor window open. The query is a full outer join between three tables: customers, orders, and products. The results show all customers and their corresponding orders and products, including rows where no orders or products exist.

```
-- 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,
       C.productid,
       B.quantity
  FROM practical2.joins.customers AS A
  FULL OUTER JOIN practical2.joins.orders AS B
  ON A.customerid = B.customerid
  FULL OUTER JOIN practical2.joins.products AS C
  ON B.productid = C.productid;
```

CUSTOMERID	CUSTOMERNAME	COUNTRY	ORDERID	PRODUCTID	QUANTITY
1251	Customer_1251	Germany	1	2014	10
1236	Customer_1236	Australia	2	2004	5
1170	Customer_1170	Germany	3	2171	9

Question 8

The screenshot shows the Snowflake SQL interface with a query editor window open. The query is a full outer join between three tables: products, orders, and customers. The results show all products and their corresponding orders and customers, including rows where no orders or customers exist.

```
-- 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 A.productid,
       A.productname,
       A.price,
       B.orderid,
       B.orderdate,
       C.customerid,
       B.quantity
  FROM practical2.joins.products AS A
  FULL OUTER JOIN practical2.joins.orders AS B
  ON A.productid = B.productid
  FULL OUTER JOIN practical2.joins.customers AS C
  ON B.customerid = C.customerid;
```

PRODUCTID	PRODUCTNAME	PRICE	ORDERID	ORDERDATE	CUSTOMERID	QUANTITY
2014	Product_2014	522	1	2023-06-10	1251	10
2004	Product_2004	1996	2	2023-12-07	1236	5
2171	Product_2171	76	3	2024-10-26	1170	9
2007	Product_2007	156	4	2023-02-17	1344	2