

## Solution 1 :

```
5 List all products in categories related to golf. Display the Product_Id, Product_Name in the output. Sort the output in the order of prod
6 Hint: You can identify a Golf category by the name of the category that contains golf.
7
8 */
9 > Execute
10 SELECT "Product_Id", "Product_Name"
11 FROM product_info
12 WHERE "category_Id" IN (
13     SELECT "Id"
14     FROM category
15     WHERE "Name" LIKE '%Golf%'
16 )
17 ORDER BY "Product_Id"; 7ms
```

Result

Search results

Cost: 7ms < 1 > Total 10

	Product_Id	Product_Name
1	703	Top Flite Womens 2014 XL Hybrid
2	728	LIJA Womens Eyelet Sleeveless Golf Polo
3	771	Clicgear Rovic Cooler Bag
4	775	Clicgear 8.0 Shoe Brush
5	778	Bag Boy Beverage Holder
6	792	Hirzl Mens Hybrid Golf Glove
7	793	Hirzl Womens Hybrid Golf Glove
8	797	Hirzl Womens Soffft Flex Golf Glove
9	804	Glove It Womens Imperial Golf Glove
10	810	Glove It Womens Mod Oval Golf Glove

## Solution 2 :

```
C: > Users > Admin > .dbclient > query > 1689443183275@@127.0.0.1@5432@postgres@public > New_Query_1689443809683.sql > ...
26 HINT:
27 Use orders, ordered_items, product_info, and category tables from the Supply chain dataset.
28 */
29
30 > Execute
31 SELECT p."Product_Name", SUM(oi."Sales") AS Total_Sales
32 FROM product_info p
33 JOIN ordered_items oi ON p."Product_Id" = oi."Item_Id"
34 JOIN category c ON p."Category_Id" = c."Id"
35 WHERE c."Name" LIKE '%Golf%'
36 GROUP BY p."Product_Name"
37 ORDER BY Total_Sales DESC
38 LIMIT 10;
```

Result

Search results

Cost: 19ms < 1 > Total 10

	Product_Name	total_sales
1	LIJA Womens Eyelet Sleeveless Golf Polo	3510.00
2	Bag Boy Beverage Holder	1074.57
3	Clicgear Rovic Cooler Bag	959.76
4	Glove It Womens Imperial Golf Glove	859.57
5	Glove It Womens Mod Oval Golf Glove	739.63
6	Hirzl Womens Soffft Flex Golf Glove	467.74
7	Top Flite Womens 2014 XL Hybrid	419.79
8	Hirzl Mens Hybrid Golf Glove	314.79
9	Clicgear 8.0 Shoe Brush	199.80
10	Hirzl Womens Hybrid Golf Glove	119.92

### Solution 3 :

Assignment+Part+II+(1).sql • Assignment+Part+I.sql • New\_Query\_1689443809683.sql • Untitled-1 • supply\_db schema.sql

C: > Users > Admin > .dbclient > query > 1689443183275@@@127.0.0.1@5432@postgres@public > New\_Query\_1689443809683.sql > ...

```
43  /*
44  Question: Segment wise orders
45  Find the number of orders by each customer segment for orders. Sort the result from the highest to the lowest
46  number of orders.The output table should have the following information:
47  -Customer_segment
48  -Orders
49  */
50
51  ▶ Execute
52  SELECT c."Segment" AS Customer_segment, COUNT(*) AS Orders
53  FROM customer_info c
54  JOIN orders o ON c."Id" = o."Customer_Id"
55  GROUP BY c."Segment"
56  ORDER BY Orders DESC;
```

Result

Search results

Free 1

Cost: 11ms < 1 > Total 3

	customer_segment	orders
1	Consumer	1104
2	Corporate	649
3	Home Office	399

### Solution 4 :

```
60  Description: Find the percentage of split of orders by each customer segment for orders that took six days
61  to ship (based on Real_Shipping_Days). Sort the result from the highest to the lowest percentage of split orders,
62  rounding off to one decimal place. The output table should have the following information:
63  -Customer_segment
64  -Percentage_order_split
65
66  HINT:
67  Use the orders and customer_info tables from the Supply chain dataset.
68  */
69
70  ▶ Execute
71  SELECT c."Segment" AS Customer_segment,
72  ROUND((COUNT(*) * 100.0 / (SELECT COUNT(*) FROM orders WHERE "Real_Shipping_Days" = 6)), 1) AS Percentage_order_split
73  FROM customer_info c
74  JOIN orders o ON c."Id" = o."Customer_Id"
75  WHERE o."Real_Shipping_Days" = 6
76  GROUP BY c."Segment"
77  ORDER BY Percentage_order_split DESC;
```

Result

Search results

Free 1

Cost: 23ms < 1 > Total 3

	customer_segment	percentage_order_split
1	Consumer	55.6
2	Corporate	28.2
3	Home Office	16.2