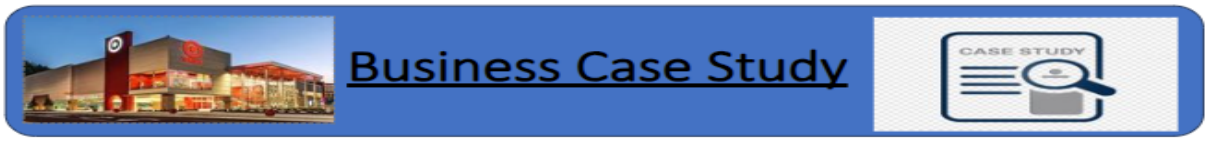


Q1) Fetch the full name and hiring date of all Employees who work as Sales Representatives.

```
SELECT
      CONCAT(firstname,' ', lastname) AS full_name,
      hiredate
 FROM Cochin_Traders.employees
WHERE title='Sales Representative';
```

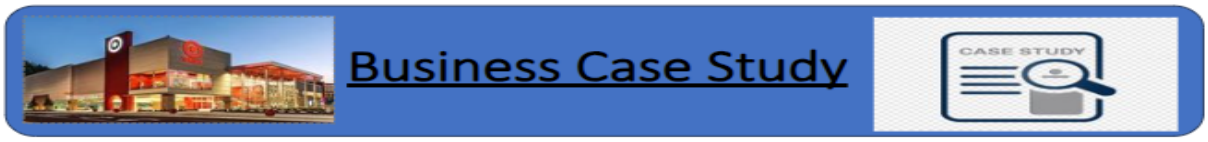
Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION D

Row	full_name	hiredate
1	Michael Suyama	1993-10-17
2	Robert King	1994-01-02
3	Anne Dodsworth	1994-11-15
4	Nancy Davolio	1992-05-01
5	Janet Leverling	1992-04-01
6	Margaret Peacock	1993-05-03

Q2) Which of the products in our inventory need to be reordered?

```
SELECT
      productid,
      productname,
      unitsinstock,
      reorderlevel
 FROM Cochin_Traders.products
 WHERE unitsinstock<=reorderlevel
 ORDER BY 2;
```



Query results

SAVE RESULTS | EXI

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW | EXECUTION GRAPH

Row	productid	productname	unitsinstock	reorderlevel
1	17	Alice Mutton	0	0
2	3	Aniseed Syrup	13	25
3	2	Chang	17	25
4	5	Chef Anton's Gumbo Mix	0	0
5	48	Chocolade	15	25
6	56	Gnocchi di nonna Alice	21	30
7	31	Gorgonzola Telino	0	20
8	37	Gravad lax	11	25

Results per page: 50 1 – 22 of 22

Q3) Find and display the details of customers who have placed more than 5 orders.

```
SELECT
      * FROM Cochin_Traders.customers
  WHERE customerid IN
  (
    SELECT customerid
      FROM Cochin_Traders.orders
    GROUP BY customerid
    HAVING COUNT(*)>5
  );
```

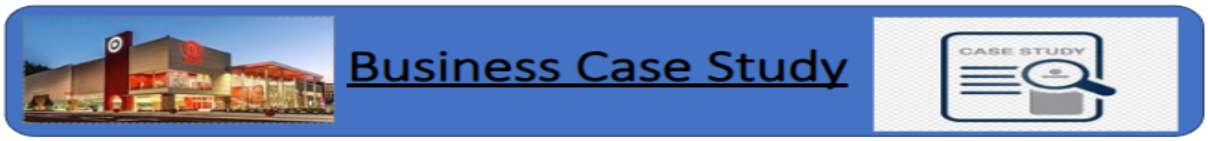
Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW | EXECUTION GRAPH

Row	row_no	customerid	companyname	contactname	contacttitle
1	4	AROUT	Around the Horn	Thomas Hardy	Sales Representative
2	11	BSBEV	B's Beverages	Victoria Ashworth	Sales Representative
3	19	EASTC	Eastern Connection	Ann Devon	Sales Agent
4	72	SEVES	Seven Seas Imports	Hari Kumar	Sales Manager
5	27	FRANS	Franchi S.p.A.	Paolo Accorti	Sales Representative

Results per page: 50 1 – 50 of 63




```
/* Q4) An employee of ours (Margaret Peacock, EmployeeID 4) has the
record of completing most orders. However, there are some customers
who've never placed an order with her. Show such customers.*/

SELECT
      customerid
  FROM Cochin_Traders.customers
WHERE customerid NOT IN
(
  SELECT
      customerid
  FROM Cochin_Traders.orders
WHERE employeeid=4
);
************************************************
 SELECT
       DISTINCT c.customerid
    FROM Cochin_Traders.customers c
      LEFT JOIN Cochin_Traders.orders o
        ON c.customerid = o.customerid
        AND o.employeeid = 4
  WHERE o.customerid IS NULL;
```

Query results

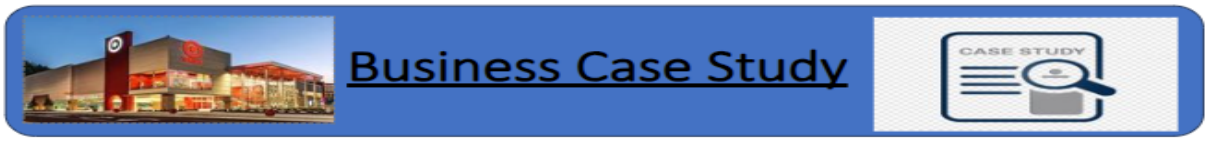
SAVE RESULTS | EXPLORE DA

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW | EXECUTION GRAPH

Row	customerid
1	CONSH
2	NORTS
3	SEVES
4	FISSA
5	DUMON

Results per page: 50 | 1 – 16 of 16

PERSONAL HISTORY | PROJECT HISTORY | RE



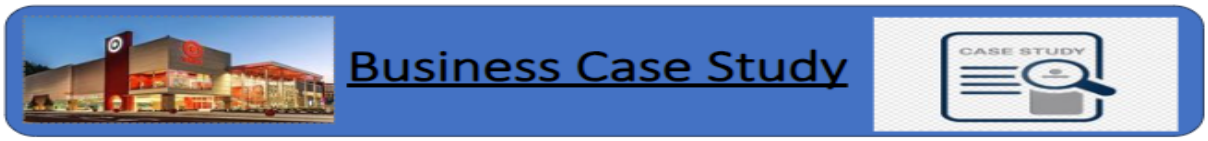
/* Q5) The developers at Cochin Traders are testing an app that the customers will use to show orders. In order to make sure that even the largest orders will show up correctly on the app, they'd like some samples of orders that have lots of individual line items. Display the top 10 orders with the most line items. */

```
SELECT
      orderid,
      COUNT(*) AS No_of_line_items
  FROM Cochin_Traders.orders
GROUP BY orderid
ORDER BY 2 DESC
LIMIT 10;
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS

Row	orderid	No_of_line_items
1	10741	1
2	10793	1
3	10864	1
4	10768	1
5	10383	1
6	10355	1
7	10558	1
8	10743	1
9	10707	1
10	10453	1



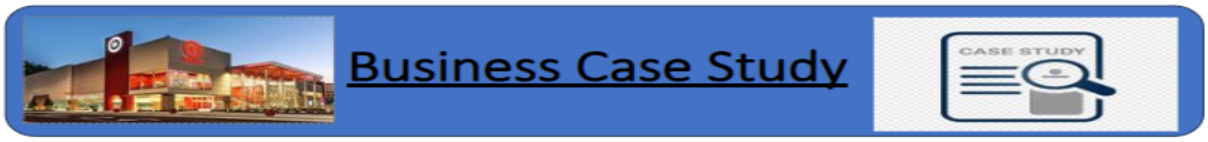
Q6) Retrieve the top 5 best-selling products on the basis of the quantity ordered.

```
SELECT
      p.productid,
      p.productname,
      SUM(d.quantity) AS total_quantity_ordered
  FROM Cochin_Traders.products p
    JOIN Cochin_Traders.orders_details d
      USING(productid)
GROUP BY 1,2
ORDER BY 3 DESC
LIMIT 5;
```

Query results

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW

Row	productid	productname	total_quantity_ordere
1	60	Camembert Pierrot	1577
2	59	Raclette Courdavault	1496
3	31	Gorgonzola Telino	1397
4	56	Gnocchi di nonna Alice	1263
5	16	Pavlova	1158




```
# Q7) Analyze the monthly order count for the year 1997.

SELECT
      EXTRACT(MONTH FROM orderdate ) AS month,
      COUNT(orderid) AS order_count
  FROM Cochin_Traders.orders
WHERE EXTRACT(YEAR FROM orderdate) = 1997
GROUP BY 1
ORDER BY 2 DESC;
```

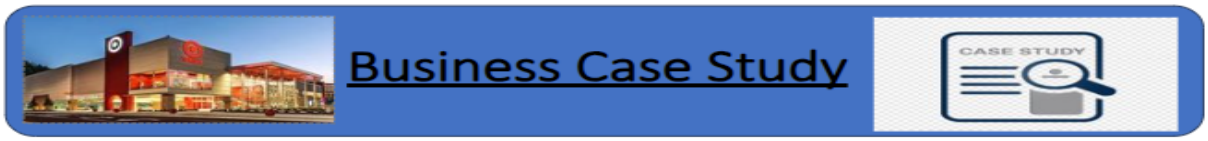
Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW | EXECUTION GRAPH

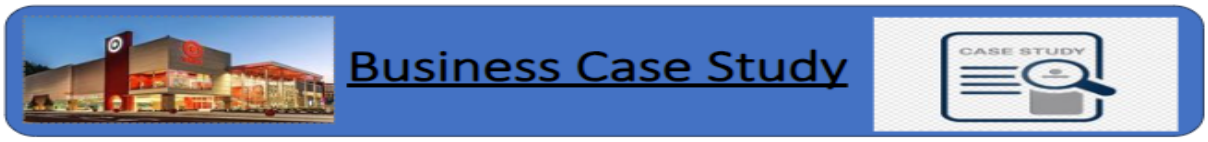
Row	month	order_count
1	12	48
2	10	38
3	9	37
4	11	34
5	7	33
6	1	33
7	8	33
8	5	32
9	4	31
10	6	30

Results per page: 30 1 – 12 of 12



Q8) Calculate the difference in sales revenue for each month compared to the previous month.

```
WITH present_revenue AS
(
  SELECT  FORMAT_DATE('%Y-%m',orderdate) AS year_monthwise,
          ROUND(SUM(d.unitprice*d.quantity),2) AS revenue
    FROM  Cochin_Traders.orders o
      JOIN Cochin_Traders.orders_details d
        USING(orderid)
  GROUP BY 1
),
previous_revenue AS
(
  SELECT
        year_monthwise,
        revenue,
        lag(revenue) OVER(ORDER BY year_monthwise) AS
previous_month_revenue
    FROM present_revenue
  ORDER BY 1
)
SELECT
      year_monthwise,
      revenue,
      previous_month_revenue,
      ROUND((revenue-previous_month_revenue),2) AS
difference_in_revenue
  FROM previous_revenue
  ORDER BY 1;
```



Query results

SAVE RESULTS | EXPLORE D

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW | EXECUTION GRAPH

Row	year_monthwise	revenue	previous_month_reve	difference_in_revenu
1	1996-07	30192.1	null	null
2	1996-08	26609.4	30192.1	-3582.7
3	1996-09	27636.0	26609.4	1026.6
4	1996-10	41203.6	27636.0	13567.6

Results per page: 50 | 1 – 23 of 23

Q9) Calculate the percentage of total sales revenue for each product.

```
SELECT
      productid,
      productname,
      ROUND((SUM(d.unitprice * d.quantity))/(SUM(SUM(d.unitprice * d.quantity)) OVER() ),2)*100 AS total_percent_sales
  FROM Cochin_Traders.products p
    JOIN Cochin_Traders.orders_details d
      USING(productid)
GROUP BY 1,2
ORDER BY 3 DESC;
```

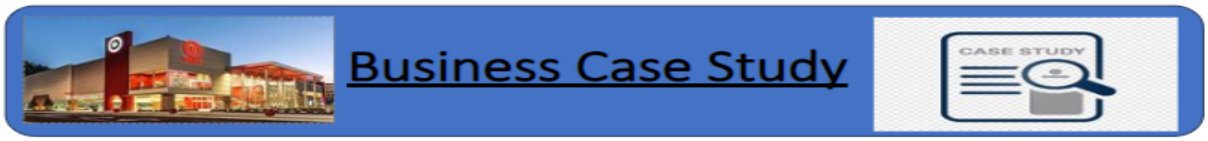
Query results

SAVE RESULTS | E

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW | EXECUTION GRAPH

Row	productid	productname	total_percent_sales
1	38	Côte de Blaye	11.0
2	59	Raclette Courdavault	6.0
3	29	Thüringer Rostbratwurst	6.0
4	60	Camembert Pierrot	4.0
5	62	Tarte au sucre	4.0
6	17	Alice Mutton	3.0
7	56	Gnocchi di nonna Alice	3.0
8	51	Manjimup Dried Apples	3.0

Results per page: 50 | 1 – 50 of 77




```
# Q10) Determine the cumulative percentage of total sales revenue for
each month.
WITH total_sales AS
(
  SELECT  FORMAT_DATE('%Y-%m',orderdate) AS year_monthwise,
          ROUND(SUM(d.unitprice*d.quantity),2) AS revenue
    FROM  Cochin_Traders.orders o
      JOIN Cochin_Traders.orders_details d
        USING(orderid)
  GROUP BY 1
)
SELECT
      year_monthwise,
      revenue,
      ROUND(SUM(revenue) OVER(ORDER BY year_monthwise)/SUM(revenue)
OVER(),2) * 100 AS cumulative_percentage
  FROM total_sales
  ORDER BY 1 DESC;
```

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | JSON | EXECUTION DETAILS | CHART PREVIEW | EXECUTION GRAPH

Row	year_monthwise_revenue	revenue	CumulativePercentag
1	1998-05	19898.66	100.0000000000...
2	1998-04	134630.56	98.53087719721...
3	1998-03	109825.45	88.59107091638...
4	1998-02	104561.95	80.48263180936...
5	1998-01	100854.72	72.76279815981...
6	1997-12	77476.26	65.31667018332...

Results per page: 50 1 – 23 of 23