

SQL

CASE PROBLEMS

PART: 2

By Nawazish Khan

Swipe Right



Creating Table

```
-- Creating Schema  
create schema market_star_schema;  
  
-- create table  
use market_star_schema;
```

Swipe Right

Creating Customer Table

```
CREATE TABLE `cust_dimen` (  
  `Cust_id` varchar(12) NOT NULL,  
  `Customer_Name` varchar(25) DEFAULT NULL,  
  `City` varchar(12) DEFAULT NULL,  
  `State` varchar(12) DEFAULT NULL,  
  `Customer_Segment` varchar(25) DEFAULT  
NULL,  
  PRIMARY KEY (`Cust_id`)  
);
```

Swipe Right



Creating Orders Table

```
CREATE TABLE `orders_dimen` (  
  `Ord_id` varchar(12) NOT NULL,  
  `Order_Number` int(11) NOT NULL,  
  `Order_Date` date DEFAULT NULL,  
  `Order_Priority` varchar(25) DEFAULT NULL,  
  PRIMARY KEY (`Ord_id`),  
  KEY `Order_Number_index` (`Order_Number`)  
);
```

Swipe Right



Creating Table - 2

```
CREATE TABLE `market_fact_full` (  
  `Market_fact_id` int(11) NOT NULL,  
  `Ord_id` varchar(12) DEFAULT NULL,  
  `Prod_id` varchar(12) DEFAULT NULL,  
  `Ship_id` varchar(12) DEFAULT NULL,  
  `Cust_id` varchar(12) DEFAULT NULL,  
  `Sales` decimal(65,30) DEFAULT NULL,  
  `Discount` decimal(12,2) DEFAULT NULL,  
  `Order_Quantity` int(11) NOT NULL,  
  `Profit` decimal(12,2) DEFAULT NULL,  
  `Shipping_Cost` decimal(12,2) DEFAULT NULL,  
  `Product_Base_Margin` decimal(12,2) DEFAULT NULL,  
  PRIMARY KEY (`Market_fact_id`),  
  KEY `Order_Quantity_index` (`Order_Quantity`),  
  KEY `Ship_Id_idx` (`Ship_id`),  
  CONSTRAINT `Cust_Id` FOREIGN KEY (`Cust_id`)  
  REFERENCES `cust_dimen` (`Cust_id`)  
);
```

Swipe Right



Inserting records.

```
INSERT INTO `cust_dimen` VALUES  
(`Cust_1`,`MUHAMMED  
MACINTYRE`,`Kolkata`,`West Bengal`,`SMALL  
BUSINESS`),...
```

```
INSERT INTO `market_fact_full` VALUES  
(1,`Ord_5446`,`Prod_16`,`SHP_7609`,  
`Cust_1818`,136.810, 0.01, 23, -30.51, 3.60,  
0.56) ,...
```

```
INSERT INTO `orders_dimen` VALUES  
(`Ord_1`,3,`2010-10-13`,`LOW`),...
```

Swipe Right



Problem: 1

Find the total number of sales made.

```
select count(sales) from  
market_fact_full;
```

	count(sales)
▶	15

Swipe Right

Problem: 2

What are the total numbers of customers from each city?

```
select count(customer_name),  
city from cust_dimen group by  
city ;
```

	count(customer_name)	city
▶	40	Kolkata
	76	Mumbai
	337	Mysore
	54	Hyderabad
	172	Cochin
	45	Coimbatore
	141	Kanyakumari
	130	Trivandrum

Swipe Right

Problem: 3

Find the number of orders which have been sold at a loss.

```
select count(ord_id) from  
market_fact_full where profit < 0 ;
```

	count(ord_id)
▶	5

Swipe Right

Problem: 4

Find the total number of customers from Bihar in each segment.

```
select count(customer_name),  
Customer_Segment  
from cust_dimen  
where state = 'Bihar'  
group by Customer_Segment, state ;
```

	count(customer_name)	Customer_Segment
▶	10	HOME OFFICE
	14	SMALL BUSINESS
	19	CORPORATE
	15	CONSUMER

Swipe Right



Problem: 5

Find the customers who incurred a shipping cost of more than 50.

```
select cust_id, sum(shipping_cost)
from market_fact_full
group by cust_id
having sum(shipping_cost) > 50;
```

	cust_id	sum(shipping_cost)
▶	Cust_1818	53.78
	Cust_839	97.85

Swipe Right



Difference between WHERE and HAVING clause

WHERE Clause	HAVING Clause
WHERE Clause is used to filter the records from the table based on the specified condition.	HAVING Clause is used to filter record from the groups based on the specified condition.
WHERE Clause can be used without GROUP BY Clause	HAVING Clause cannot be used without GROUP BY Clause
WHERE Clause implements in row operations	HAVING Clause implements in column operation
WHERE Clause cannot contain aggregate function	HAVING Clause can contain aggregate function

Source: [geeksforgeeks.org](https://www.geeksforgeeks.org/)

Swipe Right



Problem: 6

List the customer names in alphabetical order.

```
select distinct customer_name from  
cust_dimen order by customer_name ;
```

	customer_name
▶	AARON BERGMAN
	AARON HAWKINS
	AARON SMAYLING
	ADAM BELLAVANCE
	ADAM HART
	ADAM SHILLINGSBURG
	ADRIAN BARTON
	ADRIAN HANE

Swipe Right

Problem: 7

Print the three most ordered products.

```
select prod_id, sum(order_quantity)
from market_fact_full
group by prod_id
order by sum(order_quantity)
desc limit 3;
```

	prod_id	sum(order_quantity)
▶	Prod_6	112
	Prod_4	74
	Prod_16	50

Swipe Right

Problem: 8

Print the three least ordered products.

```
select prod_id, sum(order_quantity)
from market_fact_full
group by prod_id
order by sum(order_quantity)
limit 3;
```

	prod_id	sum(order_quantity)
▶	Prod_12	5
	Prod_10	10
	Prod_11	15

Swipe Right

Problem: 9

Arrange the order ids in the order of their recency.

```
select ord_id, order_date  
from orders_dimen  
order by order_date  
desc ;
```

	ord_id	order_date
▶	Ord_1468	2012-12-30
	Ord_3440	2012-12-30
	Ord_3686	2012-12-30
	Ord_4758	2012-12-30
	Ord_4206	2012-12-30
	Ord_2436	2012-12-29
	Ord_3773	2012-12-29
	Ord_2548	2012-12-29

Swipe Right



Problem: 10

On which date most orders were placed?

```
select count(ord_id), order_date  
from orders_dimen  
group by order_date  
order by count(ord_id)  
desc limit 1;
```

	count(ord_id)	order_date
▶	11	2010-12-12

Swipe Right

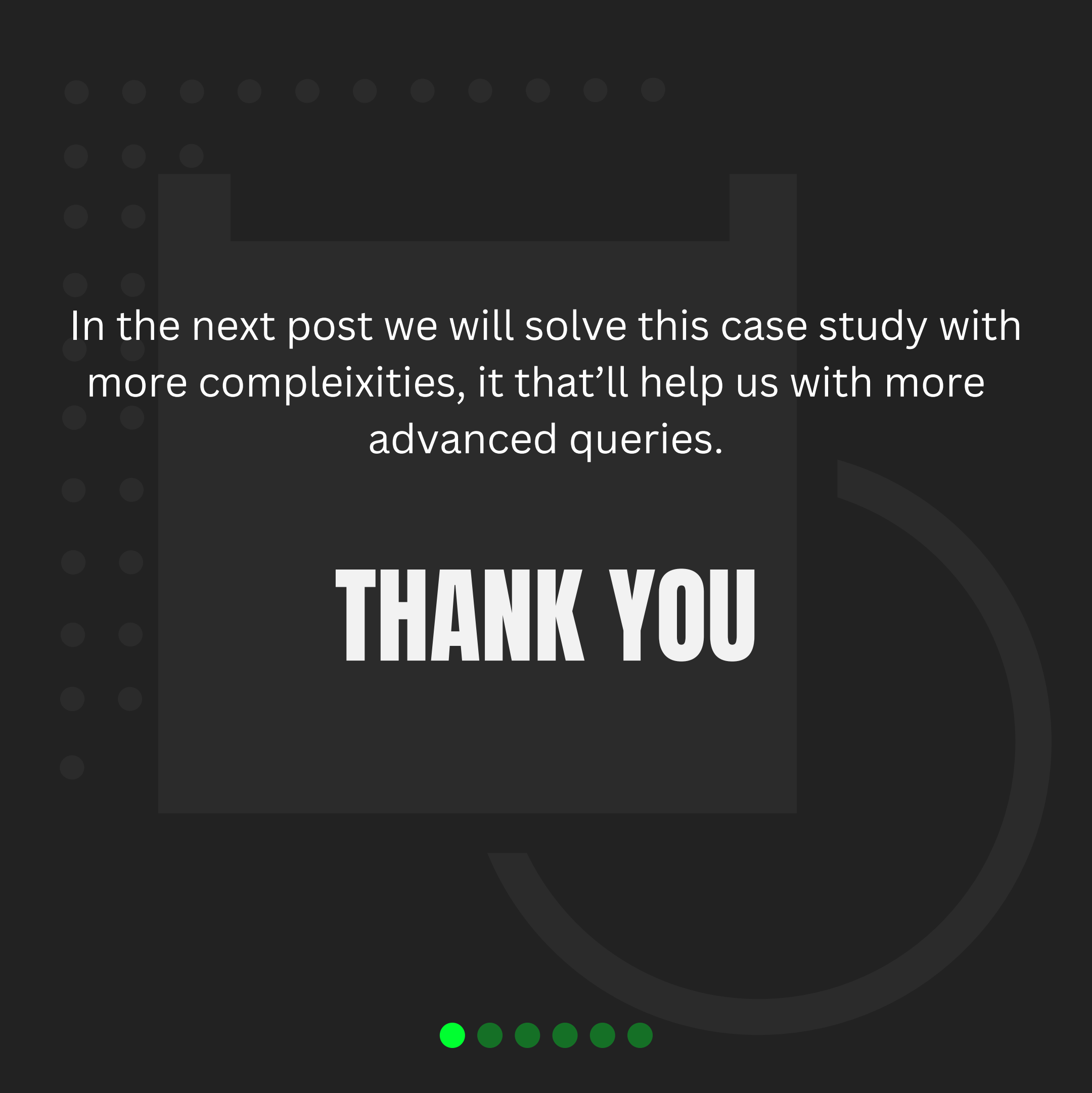
Problem: 11

Arrange all consumers from Coimbatore in alphabetical order.

```
select Customer_Name from cust_dimen  
where city = "Coimbatore" and  
Customer_Segment = "Consumer"  
order by Customer_Name ;
```

	Customer_Name
▶	ANDREW GJERTSEN
	CARLOS SOLTERO
	EDWARD HOOKS
	HAROLD ENGLE
	JENNA CAFFEY
	JOY SMITH
	JULIA WEST
	ROY FRENCH

Swipe Right



In the next post we will solve this case study with more complexities, it that'll help us with more advanced queries.

THANK YOU

