Hello Everyone,
I Used SQL Server to perform in-depth analysis on Superstore data spanning from 2014 to 2016. Below, I showcase my findings and insights derived from this dataset:

Extraction of critical business insights such as:

- Sales trends over the years
- Regional performance analysis
- Customer segmentation based on purchasing behaviour
- Product category profitability
- Inventory management optimization

Through SQL queries and analytical techniques, I've highlighted key metrics and trends that offer valuable insights into business operations and strategies.
Best Regards
Imran Mirza

Data Analysis for Sample Superstore

1. **Write an SQL query to concatenate the Customer Name and Customer_ID for each order.**

```
Select top 100 Order_ID, concat(Customer_Name, ' ', Customer_ID)as Customerdetails
from [dbo].[Sample_Superstore_ORDERS_T1]
```

00 %

Results | Messages

	Order_ID	Customerdetails
1	CA-2016-152156	Claire Gute CG-12520
2	CA-2016-152156	Claire Gute CG-12520
3	CA-2016-138688	Darrin Van Huff DV-13045
4	US-2015-108966	Sean O'Donnell SO-20335
5	US-2015-108966	Sean O'Donnell SO-20335
6	CA-2014-115812	Brosina Hoffman BH-11710
7	CA-2014-115812	Brosina Hoffman BH-11710

2. **Write a query to find the length of the Customer_Name for each order.**

```
Select top 100 Customer_Name, len(Customer_Name) as Char_Length
from [dbo].[Sample_Superstore_ORDERS_T1]
```

.00 %

Results | Messages

	Customer_Name	Char_Length
1	Claire Gute	11
2	Claire Gute	11
3	Darrin Van Huff	15
4	Sean O'Donnell	14
5	Sean O'Donnell	14
6	Brosina Hoffman	15

3. Display the Product_Name in both uppercase and lowercase.

```
Select top 100 [Product_ID], Lower([Product_Name]) as Lower_Prod_Name, Upper([Product_Name]) as Upper_PROD_NAME
from [dbo].[Sample_Superstore_ORDERS_T1]
```

Results | Messages

Product_ID	Lower_Prod_Name	Upper_PROD_NAME
FUR-BO-10001798	bush somerset collection bookcase	BUSH SOMERSET COLLECTION BOOKCASE
FUR-CH-10000454	hon deluxe fabric upholstered stacking chairs, rou...	HON DELUXE FABRIC UPHOLSTERED STACKING CHAIRS, RO...
OFF-LA-10000240	self-adhesive address labels for typewriters by univ...	SELF-ADHESIVE ADDRESS LABELS FOR TYPEWRITERS BY U...
FUR-TA-10000577	bretford cr4500 series slim rectangular table	BRETFORD CR4500 SERIES SLIM RECTANGULAR TABLE
OFF-ST-10000760	eldon fold 'n roll cart system	ELDON FOLD 'N ROLL CART SYSTEM
FUR-FU-10001487	eldon expressions wood and plastic desk accessor...	ELDON EXPRESSIONS WOOD AND PLASTIC DESK ACCESSORI...
OFF-AR-10002833	newell 322	NEWELL 322
TEC-PH-10002275	mitel 5320 ip phone voip phone	MITEL 5320 IP PHONE VOIP PHONE
OFF-BI-10003910	dxl angle-view binders with locking rings by samsill	DXL ANGLE-VIEW BINDERS WITH LOCKING RINGS BY SAMSILL
OFF-AP-10002892	belkin f5c206vtel 6 outlet surge	BELKIN F5C206VTEL 6 OUTLET SURGE
FUR-TA-10001539	chromcraft rectangular conference tables	CHROMCRAFT RECTANGULAR CONFERENCE TABLES
TEC-PH-10002033	konftel 250 conference phone - charcoal black	KONFTEL 250 CONFERENCE PHONE - CHARCOAL BLACK
OFF-PA-10002365	xerox 1967	XEROX 1967

4. Extract the first 5 characters of the Order_ID

```
Select Top 10 Order_ID, SUBSTRING(Order_ID, 1,5) as substr
from [dbo].[Sample_Superstore_ORDERS_T1]
```

100 % | Results | Messages

	Order_ID	substr
1	CA-2016-152156	CA-20
2	CA-2016-152156	CA-20
3	CA-2016-138688	CA-20
4	US-2015-108966	US-20
5	US-2015-108966	US-20
6	CA-2014-115812	CA-20
7	CA-2014-115812	CA-20
8	CA-2014-115812	CA-20
9	CA-2014-115812	CA-20
10	CA-2014-115812	CA-20

5. Query to remove leading and trailing spaces if any from the Customer_Name.

```
select top 10 [Customer_Name], trim(Customer_Name) as Trimmed_Cust_name
from [dbo].[Sample_Superstore_ORDERS_T1]
```

100 % | Results | Messages

	Customer_Name	trimmedname
1	Claire Gute	Claire Gute
2	Claire Gute	Claire Gute
3	Darrin Van Huff	Darrin Van Huff
4	Sean O'Donnell	Sean O'Donnell
5	Sean O'Donnell	Sean O'Donnell
6	Brosina Hoffman	Brosina Hoffman
7	Brosina Hoffman	Brosina Hoffman
8	Brosina Hoffman	Brosina Hoffman
9	Brosina Hoffman	Brosina Hoffman
10	Brosina Hoffman	Brosina Hoffman

6. **Write a query to left pad the Customer id field with zeros to ensure a length of 8 characters.**

SQL Statement:

```
SELECT LPAD(Customerid, 8, "0") AS LeftPadCustomerName
FROM Customers;
```

Edit the SQL Statement, and click "Run SQL" to see the result.

Run SQL »

Result:

Number of Records: 91

LeftPadCustomerName
00000001
00000002
00000003
00000004
00000005
00000006
00000007

7. **The query retrieves data from the years 2015**

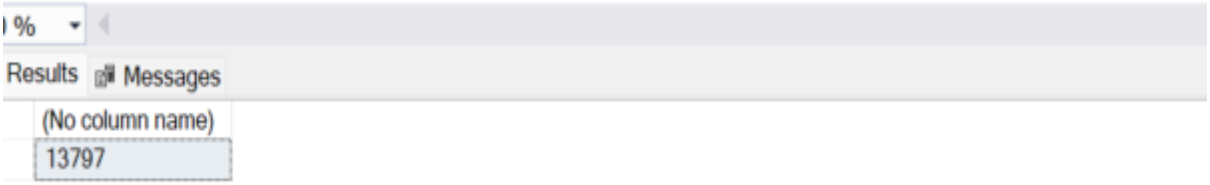
```
Select top 100 *
from [dbo].[Sample_Superstore_ORDERS_T1]
--where year([Order_Date]) between 2015 and 2016 --(BETWEEN 2015 AND 2016 includes data from the years 2015 and 2016)
--where Order_Date >= '01-01-2015' and Order_Date < 01-01-2016 --(This covers data from 1 year: 2015)
where year([Order_Date]) = 2015 --(Extracts the year 2015 from a date or datetime value)
order by Order_Date
```

Results Messages

Row_ID	Order_ID	Order_Date	Ship_Date	Ship_Mode	Customer_ID	Customer_Name	Segment	Country	City	State	Postal_Code	Region
217	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
216	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
215	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
214	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
213	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
2487	CA-2015-104514	2015-01-02	2015-01-04	Second Class	CB-12535	Claudia Bergmann	Corporate	United States	Newark	Delaware	19711	East
2488	CA-2015-104514	2015-01-02	2015-01-04	Second Class	CB-12535	Claudia Bergmann	Corporate	United States	Newark	Delaware	19711	East
2489	CA-2015-104514	2015-01-02	2015-01-04	Second Class	CB-12535	Claudia Bergmann	Corporate	United States	Newark	Delaware	19711	East
213	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
214	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
215	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
216	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
217	CA-2015-146262	2015-01-02	2015-01-09	Standard Class	VW-21775	Victoria Wilson	Corporate	United States	Medina	Ohio	44256	East
2487	CA-2015-104514	2015-01-02	2015-01-04	Second Class	CB-12535	Claudia Bergmann	Corporate	United States	Newark	Delaware	19711	East

8. **Problem statement: Extract no's for registered customers for the year 2015. This analysis excludes any records where the Customer_ID is null**

```
-- Problem statement: Extract no of registered customers for the year 2015
Select distinct count([Customer_ID])
FROM [Supermarket Sales].[dbo].[Sample_Superstore_ORDERS_T1]
where year([Order_Date]) = 2015
and [Customer_ID] is not null
```



(No column name)
13797

9. **Problem Statement: Calculating the frequency of each Order ID for each customer, sorted in descending order based on the frequency count. This analysis aims to provide insight into which Order IDs are most frequently placed by customers, sorted from highest to lowest frequency.**

```
Select s.Customer_Name,s.Order_ID, count(*) as Frequency
FROM [dbo].[Sample_Superstore_ORDERS_T1] s
group by s.Order_ID, s.Customer_Name, s.Order_Date
order by Frequency desc
```

	Customer_Name	Order_ID	Frequency
1	Seth Vernon	CA-2017-100111	84
2	Ann Chong	CA-2017-157987	84
3	Paul Prost	US-2016-108504	77
4	Naresj Patel	CA-2015-131338	70
5	William Brown	CA-2016-165330	66
6	Paul Prost	CA-2016-145177	63
7	Keith Herrera	CA-2017-117457	63
8	Brian Thompson	CA-2015-132626	63
9	Sandra Glassco	CA-2015-164882	63
10	Greg Guthrie	CA-2014-106439	63
11	Irene Maddox	CA-2015-104346	63
12	Denny Blanton	CA-2017-140949	63
13	Alejandro Grove	CA-2016-105732	60
14	Peter Fuller	US-2015-126977	60
15	Valerie Takahito	CA-2015-167010	56
16	Jim Epp	CA-2015-121720	56
17	Tim Brockman	CA-2014-129364	56
18	Robert Waldorf	CA-2017-110905	56
19	Eric Hoffmann	US-2016-110156	56
20	Dan Reichenbach	CA-2017-161956	56
21	Stefania Perrino	US-2017-118087	56
22	Giulietta Baptist	CA-2015-158421	54
23	Stewart Carmichael	US-2016-114013	54
24	Michael Paige	US-2015-163433	54
25	Fred Harton	CA-2015-165624	49
26	Xylona Preis	CA-2016-109365	49
27	Ted Butterfield	CA-2016-111682	49
28	Edward Hooks	CA-2016-135776	49

10. Problem statement :Total sales for each customer for all orders placed for the year between 2015 and 2016

- This analysis aims to provide a sum of sales attributed to each individual customer.

SQLQuery3.sql - LA...C82GT6\imirz (66))* | SQLQuery2.sql - not connected* | SQLQuery1.sql - not connected

```
--Problem statement :
--Total sales for each customer for all orders placed for the year between 2015 and 2016
--This analysis aims to provide a sum of sales attributed to each individual customer.
Select s.Customer_Name, FORMAT(SUM(s.[Sales]), 'C') AS Tot_sales_for_Customer
--rmat(sum(s.[Sales]), 00)as Tot_sales_for_Customer
FROM [dbo].[Sample_Superstore_ORDERS_T1] s
where  year([Order_Date]) between 2015 and 2016
group by s.Customer_Name
order by Tot_sales_for_Customer
```

100 %

Results | Messages

	Customer_Name	Tot_sales_for_Customer
1	Duane Benoit	$1,012.40
2	Lisa Hazard	$1,023.90
3	Christina VanderZanden	$1,030.82
4	Ann Steele	$1,031.23
5	Andrew Roberts	$1,038.03
6	Robert Barroso	$1,072.88
7	Alex Russell	$1,079.32
8	Shahid Collister	$1,081.46
9	Alejandro Ballentine	$1,093.25
10	Max Ludwig	$1,111.67
11	Michelle Moray	$1,136.35
12	Khloe Miller	$1,136.75
13	Alan Schoenberger	$1,181.20
14	Kristina Nunn	$1,208.56
15	Patrick O'Donnell	$1,239.84
16	Nancy Lomonaco	$1,260.49
17	Todd Boyes	$1,265.19
18	Vivek Gonzalez	$1,267.01
19	Christine Kargatis	$1,295.04
20	Bryan Spruell	$1,320.41

10. Problem Statement: Show the count of customers in each region, sorted in descending order

SQLQuery1.sql - LA...C82GT6\imirz (71))* | ~vs32A3.sql - LAPT...C82GT6\imirz (63)) | ~vsD3C1.sql -

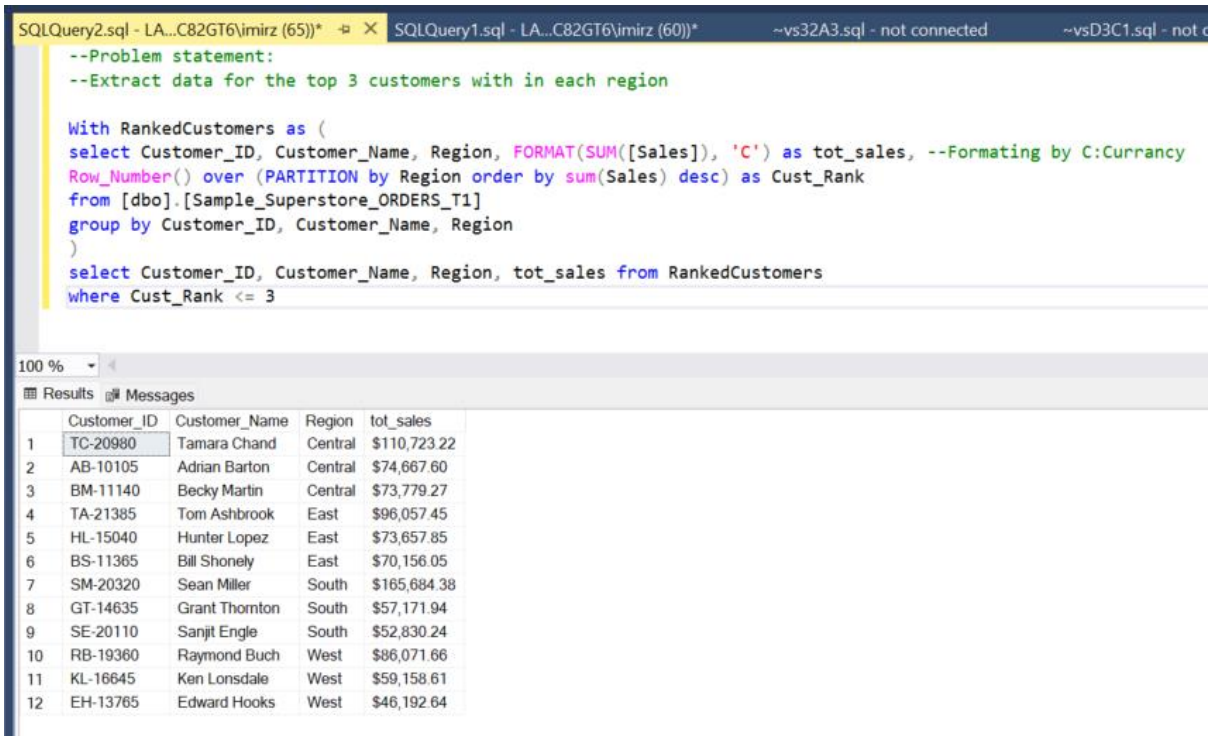
```
--Problem Statement:
-- Show the count of customers in each region, sorted in descending order
select [Region], count([Customer_ID]) as Cust_Count_by_Region
from [dbo].[Sample_Superstore_ORDERS_T1]
where [Customer_ID] is not null
group by [Region]
order by Cust_Count_by_Region desc
```

100 %

Results | Messages

	Region	Cust_Count_by_Region
1	West	21015
2	East	18668
3	Central	15248
4	South	10604

11. Problem statement: Extract data for the top 3 customers with in each region



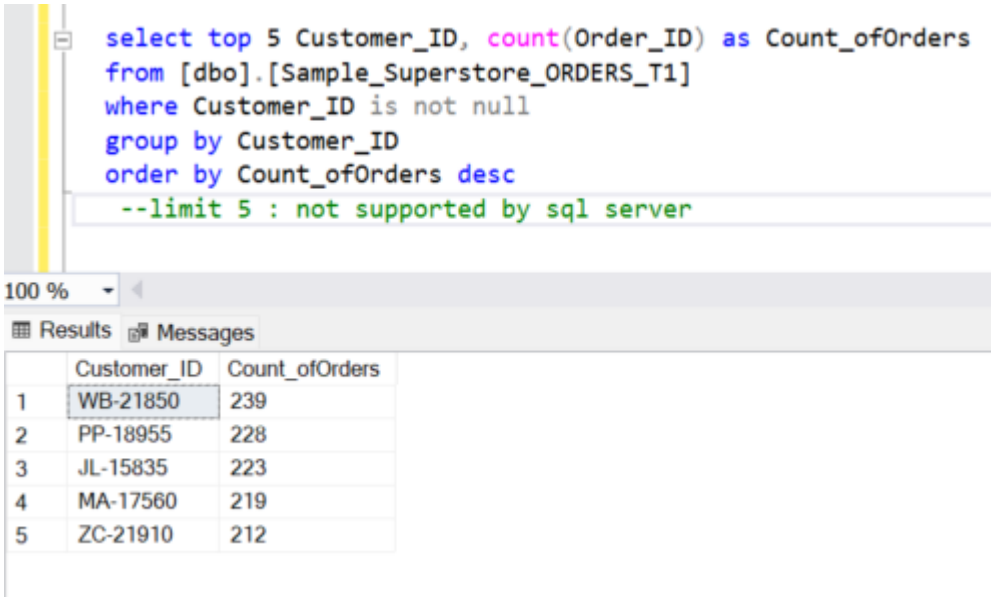
```
--Problem statement:
--Extract data for the top 3 customers with in each region

With RankedCustomers as (
select Customer_ID, Customer_Name, Region, FORMAT(SUM([Sales]), 'C') as tot_sales, --Formating by C:Currancy
Row_Number() over (PARTITION by Region order by sum(Sales) desc) as Cust_Rank
from [dbo].[Sample_Superstore_ORDERS_T1]
group by Customer_ID, Customer_Name, Region
)
select Customer_ID, Customer_Name, Region, tot_sales from RankedCustomers
where Cust_Rank <= 3
```

100 %

Results Messages

	Customer_ID	Customer_Name	Region	tot_sales
1	TC-20980	Tamara Chand	Central	$110,723.22
2	AB-10105	Adrian Barton	Central	$74,667.60
3	BM-11140	Becky Martin	Central	$73,779.27
4	TA-21385	Tom Ashbrook	East	$96,057.45
5	HL-15040	Hunter Lopez	East	$73,657.85
6	BS-11365	Bill Shonely	East	$70,156.05
7	SM-20320	Sean Miller	South	$165,684.38
8	GT-14635	Grant Thornton	South	$57,171.94
9	SE-20110	Sanjit Engle	South	$52,830.24
10	RB-19360	Raymond Buch	West	$86,071.66
11	KL-16645	Ken Lonsdale	West	$59,158.61
12	EH-13765	Edward Hooks	West	$46,192.64

12. Problem Statement: Extract the top 5 customers based on the total number of orders they have placed.



```
select top 5 Customer_ID, count(Order_ID) as Count_ofOrders
from [dbo].[Sample_Superstore_ORDERS_T1]
where Customer_ID is not null
group by Customer_ID
order by Count_ofOrders desc
 --limit 5 : not supported by sql server
```

100 %

Results Messages

	Customer_ID	Count_ofOrders
1	WB-21850	239
2	PP-18955	228
3	JL-15835	223
4	MA-17560	219
5	ZC-21910	212

13. Problem statement: Calculate the total sales for each quarter of the year 2016.

```
--Problem statement:
--Calculate the total sales for each quarter of the year 2016.
With Quarterly_sales As(
select datepart(Quarter, Order_Date) as Quarterly,
FORMAT(SUM([Sales]), 'C') as tot_sales
from [dbo].[Sample_Superstore_ORDERS_T1]
where year(Order_Date) = 2016
group by datepart(Quarter, Order_Date)
)
select Quarterly, tot_sales
from Quarterly_sales
order by Quarterly
```

Results Messages

Quarterly	tot_sales
1	$600,461.65
2	$898,253.64
3	$938,418.78
4	$1,513,337.16

14. Problem statement:Find the average number of days it takes for orders to ship after being placed for each shipping mode

```
--Problem statement:
--Find the average number of days it takes for orders to ship after being placed for each shipping mode
    Select Ship_Mode, Avg(DATEDIFF(day, Order_Date,Ship_Date)) as avg_days_to_ship
    from [dbo].[Sample_Superstore_ORDERS_T1]
    group by Ship_Mode
    order by avg_days_to_ship
```

Results Messages

	Ship_Mode	avg_days_to_ship
1	Same Day	0
2	First Class	2
3	Second Class	3
4	Standard Class	5

15. Problem statement: Identify orders where the discount rate is greater than 50%.

```
--Problem statement
--Identify orders where the discount rate is greater than 50%.
SELECT TOP (1000)
      [Order_ID]
      ,Discount
    --,cast([Discount] * 100.0 as Varchar) + '%' As Discount_Percantage
  FROM [Supermarket Sales].[dbo].[Sample_Superstore_ORDERS_T1]
  where [Discount] > 0.50
  order by Discount desc
```

Results Messages

	Order_ID	Discount
1	US-2015-118983	0.800000011920929
2	US-2015-118983	0.800000011920929
3	US-2017-118038	0.800000011920929
4	CA-2016-158568	0.800000011920929
5	CA-2014-139892	0.800000011920929
6	US-2017-152366	0.800000011920929
7	US-2014-100853	0.800000011920929
8	US-2017-116701	0.800000011920929
9	US-2017-155299	0.800000011920929
10	US-2015-161991	0.800000011920929