SQL PROJECT

Book Sales Analysis

A Database – SQL Project was created which includes information on Book Sale Customers, Orders made between the 1st of August and 3rd September, 2023 and Shippers used for delivery.

The data collected from the book sales orders were explored and analysed to gain insights into the following Key Performance Indicators (KPI); **Orders made in August 2023**, **Total Order made per Customer, The Shipper with the Highest Delivery** and **Country with the highest Order.**

By analysing this data, informed decisions about Marketing Strategies and Inventory Management can be made on the basis of these;

- Sales trends by month - To identify patterns and adjust marketing campaigns accordingly. This can help to optimize marketing budget and resources.
- Geographic sales analysis - To provide insights into which regions or areas are driving the most sales. By understanding where the customers are located, one can tailor marketing efforts to target those specific regions or areas.
- Shipping and delivery analysis - To help identify any bottlenecks or inefficiencies in supply chain. This can lead to improvements in delivery times, customer satisfaction, and overall operational efficiency.
- Customer purchasing behaviour - To provide insights into customer preferences, buying patterns, and product demand. By understanding customers' behaviour, one can optimize the inventory management strategies, such as stock levels, product assortment, and pricing.

Steps Taken;

Opened my SQL Server, connected to the Database engine by clicking on the "connect" option on the screen

- Creating a Database to store and analyse our Dataset was the first step taken in this analysis.

- Right clicked on Database and select New Database
- Named my new Database “SQL Project” and pressed Ok

After creating the Database, the next step is to create my tables

- The first table created was **BookSalesCustomers**, this contains our customers’ information ranging from CustomerID, CustomerNames, ContactNames, Address, PostalCode and Country with their corresponding Data types mainly `Varchar 255 and int.`
- The next Table is **Orders** which includes the details of orders made by each customer. The table comprises of OrderID, CustomerID, Quantity, OrderDate, and ShipperID.
- **Shippers** Table was the last created which holds details of the logistics services used for the delivery of these products - ShipperID, ShipperName, Contact.
- Click on the “Execute” tab for every query created.

Creating table for BookSalesCustomers

Syntax;

```
CREATE TABLE BookSalesCustomers(CustomerID INT, CustomerNames varchar(255),
 ContactName varchar (255), Address varchar (255), PostalCode int, Country varchar (255)
);
```

To insert new records into BookSalesCustomers Table

Syntax;

```
INSERT INTO BookSalesCustomers (CustomerID, CustomerNames, ContactName, Address,
PostalCode, Country)
VALUES
(1,'Max Legacy', 'Maxwell Ebuka', '01 Wilson Street', '300001', 'Nigeria') ;
(2,'Kochecks', 'Doris Kosi', '40th 1st Avenue', '900108', 'Nigeria');
(3,'Nyfers Holdings', 'Nazom Jennifer', '32 Umudim','435101','Austria'),
(4,'HJJHUB Services','Henry CI','Block 2, Paradise','900180','Turkey'),
(5,'Makkys Touch', 'Amaka Theodora','Street 3, Apo', '900109','United Kingdom'),
(6,'Somkekes Treats','Somky Jen','NO.6 Admiral Street', '100096','United States'),
(7,'Nipress Enterprise','Jon Van','35 Alden Avenue', '304567', 'Nigeria'),
(8,'Books n More', 'Nwadiogo Okoli', '5th Mums Street', '201605', 'Austria'),
```

```
(9,'Milano Clothings','Chika Nancy', 'Abakaliki Street 30th','400086', 'United Kingdom'),
(10,'Estymandy Collections','Amanda Eve','3rd Elon Market','234001','Nigeria'),
(11,'EvansBooks','Evan Doe','7 Church street','108052','Nigeria'),
(12,'Angelstore','Agnes Angel','9 Kadobikio Market','208102','Turkey'),
(13,'Shells Holdings','Alpha John','18th Mpape','8009231','United States'),
(14,'EbySupermarket','Mercy Clems','3rd Avenue','203021','Austria'),
(15, 'Ben International','Benson Sam','5th Allstars Street', '6076054','Germany'
   );
```

To view Table BookSalesCustomers

```
SELECT*FROM BookSalesCustomers
```

Click on Execute,

Here is what our Table looks like;

CustomerID	CustomerNames	ContactName	Address	PostalCode	Country
1	Max Legacy	Maxwell Ebuka	01 Wilson Street	300001	Nigeria
2	Kochecks	Doris Kosi	40th 1st Avenue	900108	Nigeria
3	Nyfers Holdings	Nazom Jennifer	32 Umudim	435101	Austria
4	HJJHUB Services	Henry Cl	Block 2, Paradise	900180	Turkey
5	Makkys Touch	Amaka Theodora	Street 3, Apo	900109	United Kingdom
6	Somkekes Treats	Somky Jen	NO.6 Admiral Street	100096	United States
7	Nipress Enterprise	Jon Van	35 Alden Avenue	304567	Nigeria
8	Books n More	Nwadiogo Okoli	5th Mums Street	201605	Austria
9	Milano Clothings	Chika Nancy	Abakaliki Street 30th	400086	United Kingdom
10	Estymandy Collections	Amanda Eve	3rd Elon Market	234001	Nigeria
11	EvansBooks	Evan Doe	7 Church street	108052	Nigeria
12	Angelstore	Agnes Angel	9 Kadobikio Market	208102	Turkey
13	Shells Holdings	Alpha John	18th Mpape	8009231	United States
14	EbySupermarket	Mercy Clems	3rd Avenue	203021	Austria
15	Ben International	Benson Sam	5th Allstars Street	6076054	Germany

To Create Table, Orders

Syntax;

```
CREATE TABLE Orders(OrderID int, CustomerID int, Quantity int, OrderDate Date,
ShipperID int
);
```

Insert Records inside Table Orders

Syntax;

```
INSERT INTO Orders
VALUES
(1001,2,10,'2023-08-01',1), (1002,5,2,'2023-08-01',3), (1003,4,1,'2023-08-02',1),
(1004,1,8,'2023-08-04',2), (1005,3,7,'2023-08-05',2), (1006,10,20,'2023-08-05',3),
 (1007,11,5,'2023-08-06',1)(1008,12,6,'2023-08-07',2), (1009,9,4,'2023-08-07',1),
(1010,8,3,'2023-08-08',3), (1011,15,10,'2023-08-11',1), (1012,14,25,'2023-08-11',2),
(1013,13,16,'2023-08-14',3), (1014,6,2,'2023-08-15',2),(1015,7,7,'2023-08-15',1),
(1016,7,40,'2023-08-16',1), (1017,8,2,'2023-08-16',3), (1018,4,2,'2023-08-16',1),
(1019,4,4,'2023-08-18',1), (1010,3,14,'2023-08-19',2);
```

When adding this values, instead of CustomerID 1020, I mistakenly typed in 1010 and executed, so to correct this I used the “Update” clause and not to affect the original CustomerID 1010, I had to use the “WHERE” clause with a value that is distinct for the former.

Updating OrderID 1020

Syntax;

```
UPDATE Orders
SET ORDERID = 1020
WHERE OrderDate= '2023-08-19';
```

Continuation of the Insert into

```
INSERT INTO Orders
VALUES
(1021,1,12,'2023-08-22',2), (1022,2,10,'2023-08-24',1), (1023,2,2,'2023-08-25',1),
(1024,2,3,'2023-08-25',1), (1025,5,12,'2023-08-26',3), (1026,12,25,'2023-08-26',2),
(1027,15,20,'2023-08-27',1),(1028,15,5,'2023-08-30',1), (1029,5,2,'2023-08-30',3),
```

```
(1030,5,4,'2023-09-01',3), (1031,6,8,'2023-09-02',2), (1032,3,6,'2023-09-02',2),
(1033,9,4,'2023-09-03',1), (1034,14,25,'2023-09-03',2);
```

To view Table

```
SELECT * FROM Orders
```

then Execute,

	OrderID	CustomerID	Quantity	OrderDate	ShipperID
1	1001	2	10	2023-08-01	1
2	1002	5	2	2023-08-01	3
3	1003	4	1	2023-08-02	1
4	1004	1	8	2023-08-04	2
5	1005	3	7	2023-08-05	2
6	1006	10	20	2023-08-05	3
7	1007	11	5	2023-08-06	1
8	1008	12	6	2023-08-07	2
9	1009	9	4	2023-08-07	1
10	1010	8	3	2023-08-08	3
11	1011	15	10	2023-08-11	1
12	1012	14	25	2023-08-11	2
13	1013	13	16	2023-08-14	3
14	1014	6	2	2023-08-15	2
15	1015	7	7	2023-08-15	1
16	1016	7	40	2023-08-16	1
17	1017	8	2	2023-08-16	3
18	1018	4	2	2023-08-16	1
19	1019	4	4	2023-08-18	1
20	1020	3	14	2023-08-19	2

21	1021	1	12	2023-08-22	2
22	1022	2	10	2023-08-24	1
23	1023	2	2	2023-08-25	1
24	1024	2	3	2023-08-25	1
25	1025	5	12	2023-08-26	3
26	1026	12	25	2023-08-26	2
27	1027	15	20	2023-08-27	1
28	1028	15	5	2023-08-30	1
29	1029	5	2	2023-08-30	3
30	1030	5	4	2023-09-01	3
31	1031	6	8	2023-09-02	2
32	1032	3	6	2023-09-02	2
33	1033	9	4	2023-09-03	1
34	1034	14	25	2023-09-03	2

Creating Table for Shippers

Syntax;

```
CREATE TABLE Shippers(
ShipperID int, ShipperName varchar(255), Contact int);
```

Inserting Record for Table Shippers

Syntax;

```
INSERT INTO Shippers
VALUES
(1, 'DHL','(234)701-529-3314 ) (2, 'FedEx', '(234)817-224-7139');
(3,'UX Express','(234)905-763-1375');
```

Execute,

Inserting values for contacts was proving difficult, discovered I used the wrong data type `int` instead of `varchar` while creating the Table. To change that I have to use the “ALTER” clause.

To change the Data type of column Contact from `int` to `varchar`

```
ALTER TABLE Shippers
ALTER COLUMN Contact varchar(255)
```

Execute,

Given the above error in Data type, the record in Row 1 gave a wrong value because I executed it first before adding the rest. To correct that, I used the “UPDATE” clause.

Updating Contact for Row 1

```
UPDATE Shippers
SET Contact = ('(234)701-529-3314')
WHERE ShipperName = 'DHL';
```

Execute,

To view the table Shippers

```
SELECT*FROM Shippers
```

Execute,

ShipperID	ShipperName	Contact
1	DHL	(234)701-529-3314
2	FedEx	(234)817-224-7139
3	UX Express	(234)905-763-1375

To view the three Tables together

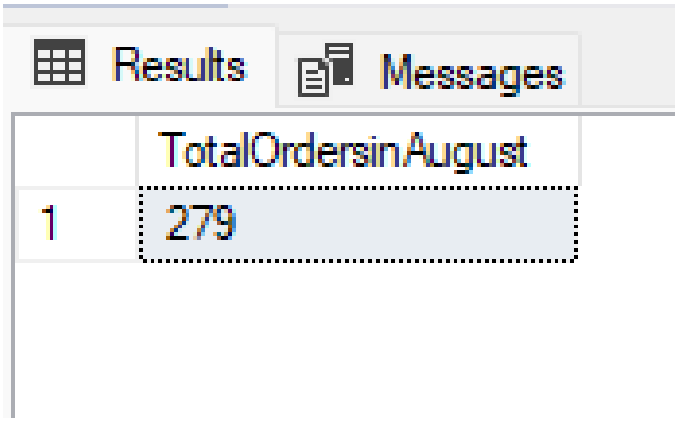
```
SELECT*FROM [SQL Project]..BookSalesCustomers
SELECT * FROM [SQL Project]..Orders
SELECT * FROM [SQL Project]..Shippers;
```

TECHNICAL ANALYSIS

- **TO Calculate the total Orders placed in August**

```
SELECT SUM(Quantity) as TotalOrdersinAugust
FROM Orders
WHERE MONTH(OrderDate) = 8 AND YEAR(OrderDate) = 2023;
```

Execute,



- **To calculate total order made per customer**

```
SELECT CustomerNames, SUM(Quantity) AS Totalpurchases
FROM Orders
JOIN BookSalesCustomers ON Orders.CustomerID = BookSalesCustomers.CustomerID
GROUP BY CustomerNames ORDER BY SUM(Quantity) DESC ;
```

Execute ,

	CustomerNames	Totalpurchases
1	EbySupermarket	50
2	Nipress Enterprise	47
3	Ben International	35
4	Angelstore	31
5	Nyfers Holdings	27
6	Kochecks	25
7	Makkys Touch	20
8	Max Legacy	20
9	Estymandy Collections	20
10	Shells Holdings	16
11	Somkekes Treats	10
12	Milano Clothings	8
13	HJJHUB Services	7
14	EvansBooks	5
15	Books n More	5

- **Which Shipper made the most Delivery**

```
SELECT ShipperName, SUM(Quantity) As TotalDelivery
FROM Shippers
JOIN Orders ON Shippers.ShipperID = Orders.ShipperID
GROUP BY ShipperName ORDER BY SUM(Quantity) DESC;
```

Execute,

	ShipperName	TotalDelivery
1	FedEx	138
2	DHL	127
3	UX Express	61

- **To calculate the total quantity of products purchased per country**

```
SELECT BookSalesCustomers.Country, SUM(Quantity) AS TotalOrderPerCountry
FROM orders
JOIN BookSalesCustomers ON Orders.CustomerID = BookSalesCustomers.CustomerID
GROUP BY BookSalesCustomers.Country ORDER BY SUM(Quantity) DESC;
```

	Country	TotalOrderPerCountry
1	Nigeria	117
2	Austria	82
3	Turkey	38
4	Germany	35
5	United Kingdom	28
6	United States	26

Conclusion

From the above analysis, we can deduce that;

A Total of 326 Orders were made within this period of which 279 were made in August.

Eby Supermarket made 50 purchases which was the highest, followed by Nipress Enterprises with 47 purchases. EvansBooks and Books ‘n’ More made the least with 5

purchases each within this period of time.

Books bought per country, Nigeria made the highest purchase with 117 books ordered, next is Austria 82 purchases while United States made the least purchase with 26 books ordered.

FedEx made the most delivery, followed by DHL.