Project Title : Sales Store Data Analysis Project

YouTube Link: https://youtu.be/GfMEiYjYZ-Y?si=mIdxI7sOMuU1HTNq

---

❖ Project Overview

The Sales Store Data Analysis Project is an end-to-end SQL Server–based analytics project designed to clean, process, and analyze online retail sales data.
The goal is to transform raw transactional data into actionable business insights by performing data import, cleaning, validation, and analysis using SQL.

This project demonstrates practical SQL skills including:

- Data modeling and schema design
- Bulk data loading
- Data cleaning using SQL
- Exploratory and business analytics using SQL queries

❖ Objective of the Project

- Import raw sales data into SQL Server using BULK INSERT.
- Clean and validate data by removing duplicates, handling NULLs, and correcting inconsistent fields.
- Standardize data formats (gender, payment mode, etc.).
- Analyze customer trends, sales performance, and business insights.
- Generate reports that help management make data-driven decisions.

❖ Tools and Technologies Used

- **Microsoft SQL Server Management Studio (SSMS)**
- **T-SQL** (DDL, DML, Aggregate Functions, CTEs)
- **BULK INSERT** for dataset import
- **Microsoft Excel / CSV Dataset** sales. csv.csv
- **Data Source:** Retail e-commerce sales data

❖ Database Design Overview

1. Create Table

Table: Sales_store

Each record in the table represents one sales transaction, including customer, product, and payment details.

```
CREATE TABLE Sales_store (
transaction_id VARCHAR(15),
customer_id VARCHAR(15),
customer_name VARCHAR(30),
customer_age INT,
gender VARCHAR(15),
product_id VARCHAR(15),
product_name VARCHAR(15),
product_category VARCHAR(15),
quantiy INT,
prce FLOAT,
payment_mode VARCHAR(15),
purchase_date DATE,
time_of_purchase TIME,
status VARCHAR(15)
);
```

---

2. Data Import (Using BULK INSERT)

The dataset is imported into SQL Server using the BULK INSERT method from a CSV file.

```
SET DATEFORMAT dmy
BULK INSERT Sales_store
FROM 'C:\Users\Monika pawar\Desktop\Sales_store\sales.csv'
WITH (
FIRSTROW = 2,
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
);
```

---

3. Data Cleaning Process

➢ Step 1: Identify and Remove Duplicates

- **Identify Duplicates**

```
SELECT transaction_id, COUNT(*)
FROM sales_store
GROUP BY transaction_id
HAVING COUNT(transaction_id)>1
```

	transaction_id	(No column name)
1	TXN240646	2
2	TXN342128	2
3	TXN855235	2
4	TXN981773	2

```
WITH CTE AS(

SELECT*,

ROW_NUMBER() OVER (PARTITION BY transaction_id ORDER BY transaction_id) AS ROW_NUM

FROM sales_store

)

SELECT * FROM CTE

WHERE ROW_NUM >1
```

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantiy	prce	payment_mode	purchase_date	time_of_purchase	status	ROW_NUM
1	TXN240646	CUST5356	Renee Kant	37	Female	P4193	Textbook	Books	2	9770	UPI	2023-09-23	11:43:16.0000000	cancelled	2
2	TXN342128	CUST1214	Nehmat Batta	39	Male	P1286	Sweater	Clothing	3	1284	Cash	2023-08-24	20:05:11.0000000	returned	2
3	TXN855235	CUST5938	Yashvi Sachar	44	Male	P6308	Smartwatch	Electronics	3	11214	EMI	2023-08-26	16:59:09.0000000	delivered	2
4	TXN981773	CUST2365	Tushar Chakrabarti	50	M	P5760	Milk	Groceries	4	4120	Cash	2023-11-19	11:01:31.0000000	cancelled	2

- **Remove Duplicates**

```
WITH CTE AS(

SELECT*,

ROW_NUMBER() OVER (PARTITION BY transaction_id ORDER BY transaction_id) AS ROW_NUM

FROM sales_store

)

--Delete FROM CTE

--WHERE ROW_NUM =2

SELECT * FROM CTE
```

WHERE transaction_id IN ('TXN240646', 'TXN342128', 'TXN855235', 'TXN981773')

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantiy	prce	payment_mode	purchase_date	time_of_purchase	status	ROW_NUM
1	TXN240646	CUST5356	Renee Kant	37	Female	P4193	Textbook	Books	2	9770	UPI	2023-09-23	11:43:16.0000000	cancelled	1
2	TXN342128	CUST1214	Nehmat Batta	39	Male	P1286	Sweater	Clothing	3	1284	Cash	2023-08-24	20:05:11.0000000	returned	1
3	TXN855235	CUST5938	Yashvi Sachar	44	Male	P6308	Smartwatch	Electronics	3	11214	EMI	2023-08-26	16:59:09.0000000	delivered	1
4	TXN981773	CUST2365	Tushar Chakrabarti	50	M	P5760	Milk	Groceries	4	4120	Cash	2023-11-19	11:01:31.0000000	cancelled	1

➢ Step 2: Correct Header Names

EXEC sp_rename 'Sales_store.quantiy','quantity', 'Column'
EXEC sp_rename 'Sales_store.prce','price', 'Column'

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status

➢ Step 3: Check Data Types

SELECT COLUMN_NAME, DATA_TYPE
FROM INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_NAME='Sales_store'

	COLUMN_NAME	DATA_TYPE
1	transaction_id	varchar
2	customer_id	varchar
3	customer_name	varchar
4	customer_age	int
5	gender	varchar
6	product_id	varchar
7	product_name	varchar
8	product_category	varchar
9	quantity	int
10	price	float
11	payment_mode	varchar
12	purchase_date	date
13	time_of_purchase	time
14	status	varchar

➢ Step 4: Check for NULL Values

SELECT *
FROM Sales_store
WHERE
transaction_id IS NULL OR
customer_id IS NULL OR
customer_name IS NULL OR
customer_age IS NULL OR
gender IS NULL OR
product_id IS NULL OR
product_name IS NULL OR

product_category IS NULL OR
quantity IS NULL OR
price IS NULL OR
payment_mode IS NULL OR
purchase_date IS NULL OR

time_of_purchase IS NULL OR

status IS NULL;

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	NULL	NULL	NULL	NULL	NULL	P4524	T-Shirt	Clothing	4	14788	NULL	NULL	NULL	NULL
2	TXN432798	CUST1003	NULL	NULL	NULL	P5717	Dining Table	Furniture	3	2346	EMI	2023-04-30	05:46:48.0000000	cancelled
3	TXN977900	NULL	Ehsaan Ram	25	M	P8212	Milk	Groceries	3	12342	Cash	2023-08-13	07:46:18.0000000	returned
4	TXN985663	NULL	Damini Raju	49	Female	P3367	Notebook	Books	5	8130	EMI	2023-01-24	01:40:38.0000000	cancelled

- **Step 5: Deleting the Outliner**

DELETE FROM Sales_store

WHERE transaction_id IS NULL

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	TXN432798	CUST1003	NULL	NULL	NULL	P5717	Dining Table	Furniture	3	2346	EMI	2023-04-30	05:46:48.0000000	cancelled
2	TXN977900	NULL	Ehsaan Ram	25	M	P8212	Milk	Groceries	3	12342	Cash	2023-08-13	07:46:18.0000000	returned
3	TXN985663	NULL	Damini Raju	49	Female	P3367	Notebook	Books	5	8130	EMI	2023-01-24	01:40:38.0000000	cancelled

- **Step 6: Fetch all records from your table and replace the null with previous record**

SELECT * FROM Sales_store

WHERE customer_name='Ehsaan Ram'

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	TXN553735	CUST9494	Ehsaan Ram	25	M	P3115	Handbag	Accessories	2	4220	Debit Card	2023-11-22	11:21:36.0000000	cancelled
2	TXN977900	NULL	Ehsaan Ram	25	M	P8212	Milk	Groceries	3	12342	Cash	2023-08-13	07:46:18.0000000	returned
3	TXN495746	CUST9494	Ehsaan Ram	25	M	P7385	Wardrobe	Furniture	3	7164	CC	2023-04-28	23:40:54.0000000	cancelled
4	TXN586594	CUST9494	Ehsaan Ram	25	M	P4479	Shirt	Clothing	4	11832	Cash	2023-11-26	22:05:45.0000000	pending

UPDATE Sales_store

SET customer_id='CUST9494'

WHERE transaction_id='TXN977900'

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	TXN553735	CUST9494	Ehsaan Ram	25	M	P3115	Handbag	Accessories	2	4220	Debit Card	2023-11-22	11:21:36.0000000	cancelled
2	TXN977900	CUST9494	Ehsaan Ram	25	M	P8212	Milk	Groceries	3	12342	Cash	2023-08-13	07:46:18.0000000	returned
3	TXN495746	CUST9494	Ehsaan Ram	25	M	P7385	Wardrobe	Furniture	3	7164	CC	2023-04-28	23:40:54.0000000	cancelled
4	TXN586594	CUST9494	Ehsaan Ram	25	M	P4479	Shirt	Clothing	4	11832	Cash	2023-11-26	22:05:45.0000000	pending

SELECT * FROM Sales_store

WHERE customer_name='Damini Raju'

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	TXN749812	CUST1401	Damini Raju	49	Female	P2348	Dining Table	Furniture	5	2455	CC	2023-04-29	02:25:23.0000000	returned
2	TXN985663	NULL	Damini Raju	49	Female	P3367	Notebook	Books	5	8130	EMI	2023-01-24	01:40:38.0000000	cancelled
3	TXN227567	CUST1401	Damini Raju	49	Female	P9862	Handbag	Accessories	1	4496	Debit Card	2023-07-25	19:30:41.0000000	pending
4	TXN287174	CUST1401	Damini Raju	49	Female	P4575	Jacket	Clothing	3	11532	Cash	2023-08-19	00:36:11.0000000	pending

UPDATE Sales_store

SET customer_id='CUST1401'

WHERE transaction_id='TXN985663'

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	TXN749812	CUST1401	Damini Raju	49	Female	P2348	Dining Table	Furniture	5	2455	CC	2023-04-29	02:25:23.0000000	returned
2	TXN985663	CUST1401	Damini Raju	49	Female	P3367	Notebook	Books	5	8130	EMI	2023-01-24	01:40:38.0000000	cancelled
3	TXN227567	CUST1401	Damini Raju	49	Female	P9862	Handbag	Accessories	1	4496	Debit Card	2023-07-25	19:30:41.0000000	pending
4	TXN287174	CUST1401	Damini Raju	49	Female	P4575	Jacket	Clothing	3	11532	Cash	2023-08-19	00:36:11.0000000	pending

SELECT * FROM Sales_store

WHERE customer_id='CUST1003'

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	TXN737152	CUST1003	Mahika Saini	35	Male	P3228	Bed	Furniture	4	13332	CC	2023-10-14	20:59:58.0000000	pending
2	TXN432798	CUST1003	NULL	NULL	NULL	P5717	Dining Table	Furniture	3	2346	EMI	2023-04-30	05:46:48.0000000	cancelled
3	TXN553240	CUST1003	Mahika Saini	35	Male	P5769	Wallet	Accessories	4	1412	CC	2023-03-12	08:45:44.0000000	cancelled
4	TXN834755	CUST1003	Mahika Saini	35	Male	P1071	Jacket	Clothing	1	655	Credit Card	2023-06-18	00:06:38.0000000	pending

UPDATE Sales_store

SET customer_name='Mahika Saini',customer_age= 35,gender='Male'

WHERE transaction_id='TXN432798'

	transaction_id	customer_id	customer_name	customer_age	gender	product_id	product_name	product_category	quantity	price	payment_mode	purchase_date	time_of_purchase	status
1	TXN737152	CUST1003	Mahika Saini	35	Male	P3228	Bed	Furniture	4	13332	CC	2023-10-14	20:59:58.0000000	pending
2	TXN432798	CUST1003	Mahika Saini	35	Male	P5717	Dining Table	Furniture	3	2346	EMI	2023-04-30	05:46:48.0000000	cancelled
3	TXN553240	CUST1003	Mahika Saini	35	Male	P5769	Wallet	Accessories	4	1412	CC	2023-03-12	08:45:44.0000000	cancelled
4	TXN834755	CUST1003	Mahika Saini	35	Male	P1071	Jacket	Clothing	1	655	Credit Card	2023-06-18	00:06:38.0000000	pending

--Now there is no NULL

- **Step 7: Standardize Gender and Payment Data**

UPDATE Sales_store SET gender = 'M' WHERE gender = 'Male';

```
UPDATE Sales_store SET gender = 'F' WHERE gender = 'Female';
UPDATE Sales_store SET payment_mode = 'Credit Card' WHERE payment_mode = 'CC';
```

❖ Business Insight Questions

Below are key business questions answered using SQL queries.

---

1. What are the top 5 most selling products by quantity?

```
SELECT TOP 5 product_name, SUM(quantity) AS total_quantity_sold
FROM Sales_store
WHERE status = 'delivered'
GROUP BY product_name
ORDER BY total_quantity_sold DESC;
```

	product_name	total_quantity_sold
1	Wardrobe	70
2	Vegetables	69
3	Sofa	66
4	Dining Table	65
5	Fruits	60

Business Problem: Unclear which products customers buy the most.

Business Impact: Helps keep popular items in stock and increase sales with focused promotions.

---

2. Which products are most frequently cancelled?

```
SELECT TOP 5 product_name, COUNT(*) AS total_cancelled
FROM Sales_store
WHERE status = 'cancelled'
GROUP BY product_name
ORDER BY total_cancelled DESC;
```

	product_name	total_cancelled
1	Comics	24
2	Sweater	23
3	Chair	21
4	Vegetables	21
5	Smartphone	20

Business Problem: Frequent order cancellations reduce sales and customer trust.

Business Impact: Helps spot low-quality products to fix or remove from the catalog.

---

3. What time of the day has the highest number of purchases?

```
SELECT
CASE
WHEN DATEPART(HOUR, time_of_purchase) BETWEEN 0 AND 5 THEN 'Night'
WHEN DATEPART(HOUR, time_of_purchase) BETWEEN 6 AND 11 THEN 'Morning'
WHEN DATEPART(HOUR, time_of_purchase) BETWEEN 12 AND 17 THEN 'Afternoon'
ELSE 'Evening' END AS time_of_day,
COUNT(*) AS total_orders
FROM Sales_store
GROUP BY CASE
WHEN DATEPART(HOUR, time_of_purchase) BETWEEN 0 AND 5 THEN 'Night'
WHEN DATEPART(HOUR, time_of_purchase) BETWEEN 6 AND 11 THEN 'Morning'
WHEN DATEPART(HOUR, time_of_purchase) BETWEEN 12 AND 17 THEN 'Afternoon'
ELSE 'Evening' END
ORDER BY total_orders DESC;
```

	time_of_day	total_orders
1	EVENING	515
2	MORNING	514
3	NIGHT	496
4	AFTERNOON	475

Business Problem Solved: Discover when sales are highest.

Business Impact: Helps manage staff, discounts, and system usage more effectively.

---

4. Who are the top 5 highest spending customers?

```
SELECT TOP 5 customer_name,
FORMAT(SUM(price * quantity), 'C0', 'en-IN') AS total_spend
FROM Sales_store
GROUP BY customer_name
ORDER BY SUM(price * quantity) DESC;
```

	customer_name	total_spend
1	Darshit Mann	₹ 5,07,530
2	Anahita Shenoy	₹ 4,55,637
3	Saira Ahluwalia	₹ 4,47,933
4	Gatik Khare	₹ 3,86,156
5	Samaira Subramaniam	₹ 3,57,388

Business Problem Solved: Find the most valuable and loyal customers.

Business Impact: Helps give special offers, rewards, and keep them coming back.

---

5. Which product categories generate the highest revenue?

```
SELECT product_category,
FORMAT(SUM(price * quantity), 'C0', 'en-IN') AS Revenue
FROM Sales_store
GROUP BY product_category
ORDER BY SUM(price * quantity) DESC;
```

	product_category	Revenue
1	Accessories	₹ 1,03,65,306
2	Clothing	₹ 1,01,95,727
3	Books	₹ 99,12,929
4	Furniture	₹ 96,59,478
5	Electronics	₹ 95,04,028
6	Groceries	₹ 94,64,153

Business Problem Solved: Find out which product categories sell the most.

Business Impact: Helps improve product planning, stock management, and offers.

---

6. What is the return/cancellation rate per product category?

Cancellation:

```
SELECT product_category,
FORMAT(COUNT(CASE WHEN status='cancelled' THEN 1 END)*100.0/COUNT(*),'N3')+' %' AS cancelled_percent
FROM Sales_store
GROUP BY product_category
ORDER BY cancelled_percent DESC
```

	product_category	cancelled_percent
1	Books	26.205 %
2	Clothing	25.634 %
3	Electronics	24.675 %
4	Accessories	23.547 %
5	Furniture	22.832 %
6	Groceries	22.289 %

Return:

SELECT product_category,

FORMAT(COUNT(CASE WHEN status='returned' THEN 1 END)*100.0/COUNT(*),'N3')+' %' AS returned_percent

FROM Sales_store

GROUP BY product_category

ORDER BY returned_percent DESC

	product_category	returned_percent
1	Accessories	31.498 %
2	Books	25.602 %
3	Clothing	24.789 %
4	Groceries	23.494 %
5	Furniture	23.410 %
6	Electronics	20.779 %

Business Problem Solved: Track which product categories have more unhappy customers.

Business Impact: Helps lower returns and make product details clearer.

---

7. What is the most preferred payment mode?

SELECT payment_mode, COUNT(*) AS total_count

FROM Sales_store

GROUP BY payment_mode

ORDER BY total_count DESC;

	payment_mode	total_count
1	Credit Card	648
2	EMI	350
3	Debit Card	344
4	Cash	332
5	UPI	326

Business Problem Solved: Find out which payment methods customers like the most.

Business Impact: Make payments faster and focus on the most used payment methods.

---

8. How does age group affect purchasing behavior?

```
SELECT
CASE
WHEN customer_age BETWEEN 18 AND 25 THEN '18-25'
WHEN customer_age BETWEEN 26 AND 35 THEN '26-35'
WHEN customer_age BETWEEN 36 AND 50 THEN '36-50'
ELSE '51+'
END AS Age_Group,
FORMAT(SUM(price * quantity), 'C0', 'en-IN') AS Total_Purchase
FROM Sales_store
GROUP BY CASE
WHEN customer_age BETWEEN 18 AND 25 THEN '18-25'
WHEN customer_age BETWEEN 26 AND 35 THEN '26-35'
WHEN customer_age BETWEEN 36 AND 50 THEN '36-50'
ELSE '51+'
END
ORDER BY SUM(price * quantity) DESC;
```

	customer_age	total_purchase
1	36-50	₹ 1,94,60,276
2	51+	₹ 1,43,86,538
3	26-35	₹ 1,36,96,027
4	18-25	₹ 1,15,58,780

Business Problem Solved: Identify and understand customer age groups and preferences.

Business Impact: Helps suggest the right products and ads for each age group.

---

9. What's the monthly sales trend?

```
SELECT FORMAT(purchase_date, 'yyyy-MM') AS Month_Year,
FORMAT(SUM(price * quantity), 'C0', 'en-IN') AS total_sales,
SUM(quantity) AS total_quantity
FROM Sales_store
GROUP BY FORMAT(purchase_date, 'yyyy-MM')
ORDER BY Month_Year;
```

	Month_Year	total_sales	total_quantity
1	2023-01	₹ 46,28,608	478
2	2023-02	₹ 46,98,929	529
3	2023-03	₹ 52,41,364	471
4	2023-04	₹ 49,89,315	505
5	2023-05	₹ 39,02,263	418
6	2023-06	₹ 41,00,112	478
7	2023-07	₹ 51,29,904	577
8	2023-08	₹ 47,88,207	497
9	2023-09	₹ 50,37,847	512
10	2023-10	₹ 58,86,414	547
11	2023-11	₹ 51,09,229	523
12	2023-12	₹ 52,49,987	521
13	2024-01	₹ 3,39,442	31

Business Problem: Sales fluctuations go unnoticed.

Business Impact: Plan inventory and marketing according to seasonal trends.

❖ Conclusion

This SQL project covers the full data analysis process — from importing and cleaning sales data to generating useful insights.

It highlights key areas like customer behavior, product performance, peak sales time, and payment trends.

These insights help improve inventory management, marketing strategies, and overall business growth.