PREDICTIVE ANALYTICS
Case Study

on

Customer Sales Data Transformation & Predictive Analysis Using IBM SPSS Modeler

INDEX

Sr. No	Name of Experiment	Date	Faculty Signature	Remarks
1	**Practical 4:** To define the unit of analysis in a dataset by removing duplicate records using the Distinct node, aggregating customer data using the Aggregate node, and expanding product fields into flag fields using the SetToFlag node to ensure one record per customer.	28-10-25		
2	**Practical 8:** To cleanse and enrich the dataset using SPSS Modeler by applying Derive and Reclassify nodes. These functions are used to transform, categorize, and standardize data values, making the dataset more consistent and ready for analysis.	31-10-25		
3	**Practical 9:** To apply additional field transformations such as Binning, Filler, and Transform nodes to prepare the dataset for modeling. This includes normalizing numeric fields, converting categorical data into numerical format, and ensuring the dataset follows a normal distribution.	01-11-25		
4				
5				

Agenda/Definition:

Practical 4:

Unit of Analysis and Deduplication

Define one record per customer by removing duplicates.

Use Distinct node to remove duplicate entries.

Apply Aggregate node to calculate total sales and number of orders per customer.

Use SetToFlag to create binary flags .

Practical 8:

Data Cleansing and Enrichment

Apply Derive and Reclassify nodes to clean and enrich data fields.

Example: Create a new variable “Sales_Category” → classify as Low, Medium, High.

Standardize Region names (e.g., "South " → "South") to maintain consistency.

Practical 9:

Field Transformations

Use Binning, Filler, and Transform nodes.

Normalize numeric variables such as Profit and Discount.

Prepare categorical variables for modeling.

Outcomes/Learning:

- Learn how to **import Excel data** into SPSS Modeler.
- Understand **data preparation and transformation nodes**.
- Generate a clean dataset ready for **predictive analytics and visualization**.

Required Tool: “IBM SPSS Modeler” (for data preprocessing and transformations)

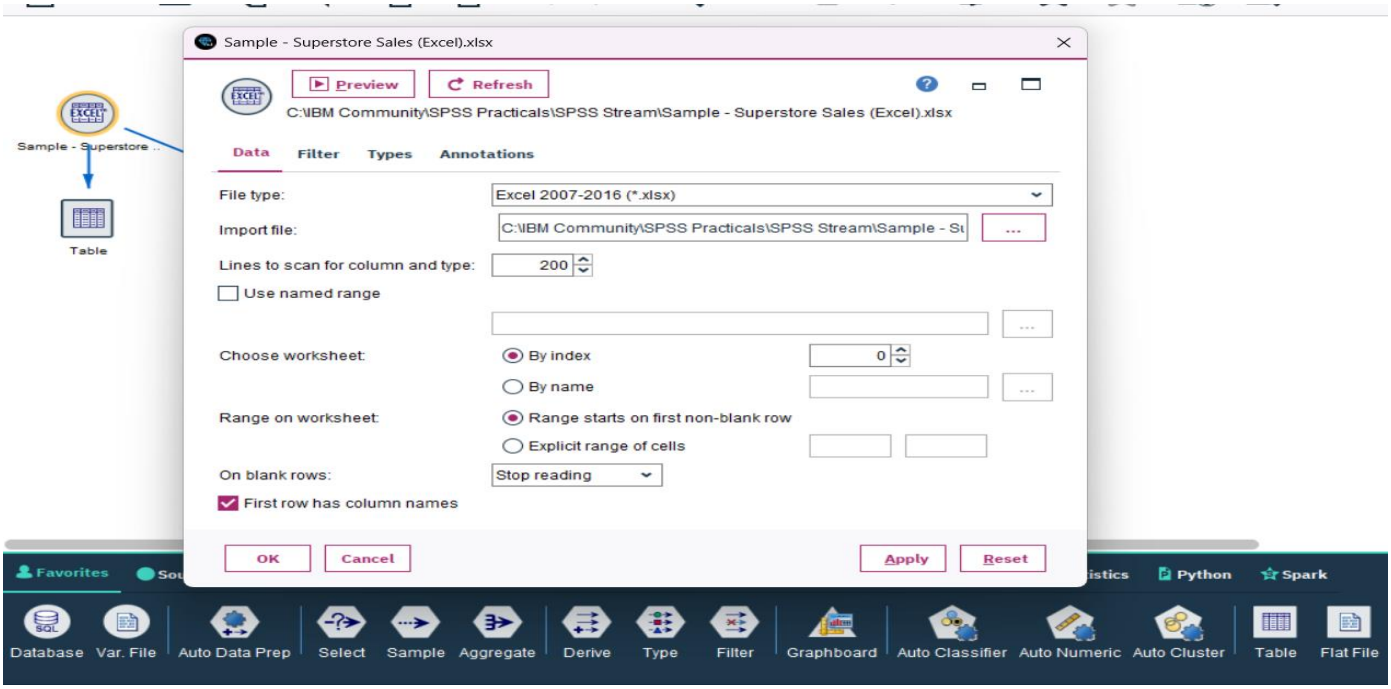
Working:

- Imported the Excel dataset into SPSS Modeler using the Excel Source Node.
- Applied various transformation nodes (Distinct, Aggregate, Type, SetToFlag, Derive, Reclassify, Binning, Transform).
- Exported the transformed data for further visualization and reporting.

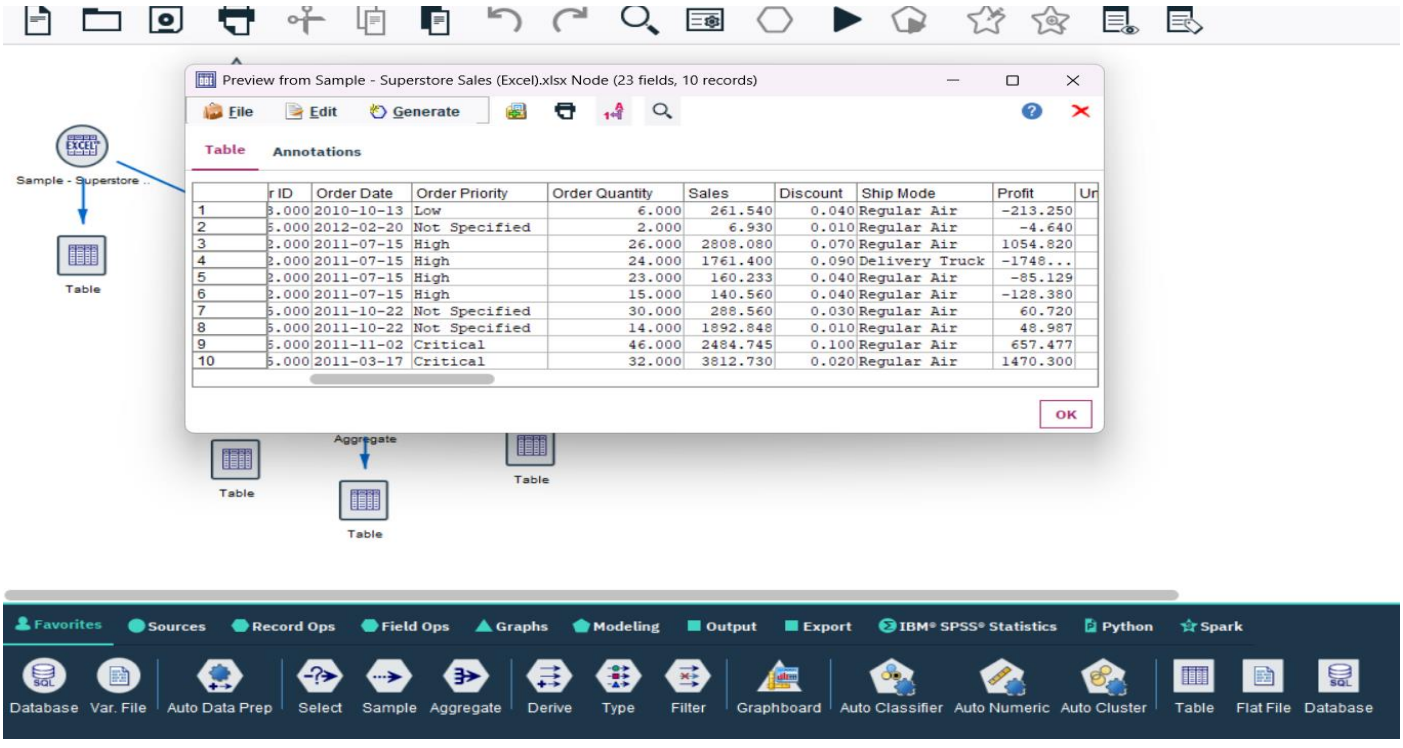
Step 1: " Import Excel Dataset"

Description :

Imported the *Sample – Superstore Sales.xlsx* Excel file into IBM SPSS Modeler using the Excel File node to load the dataset for analysis



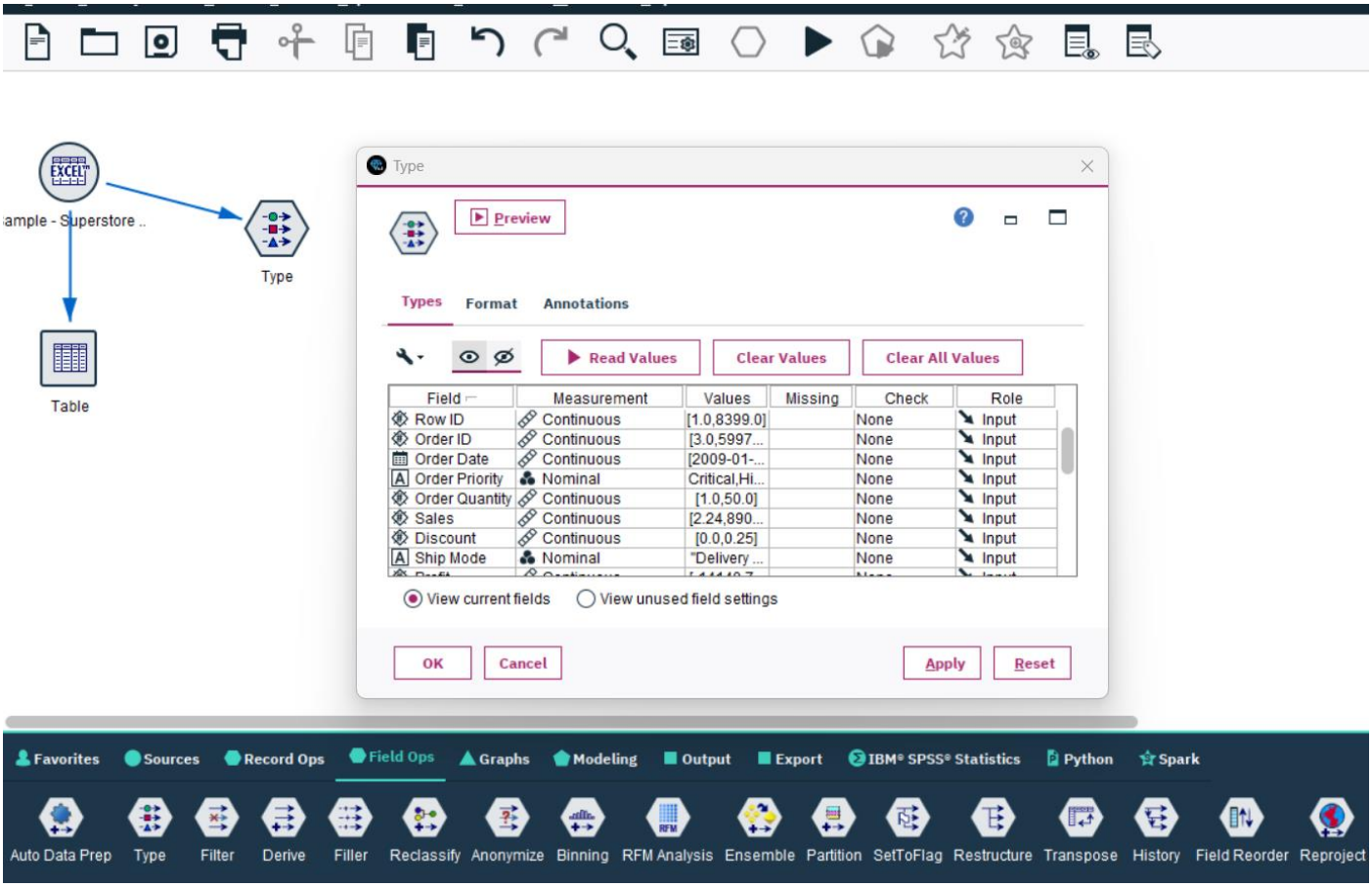
Excel File Node – Import settings showing file path, sheet selection, and column options.



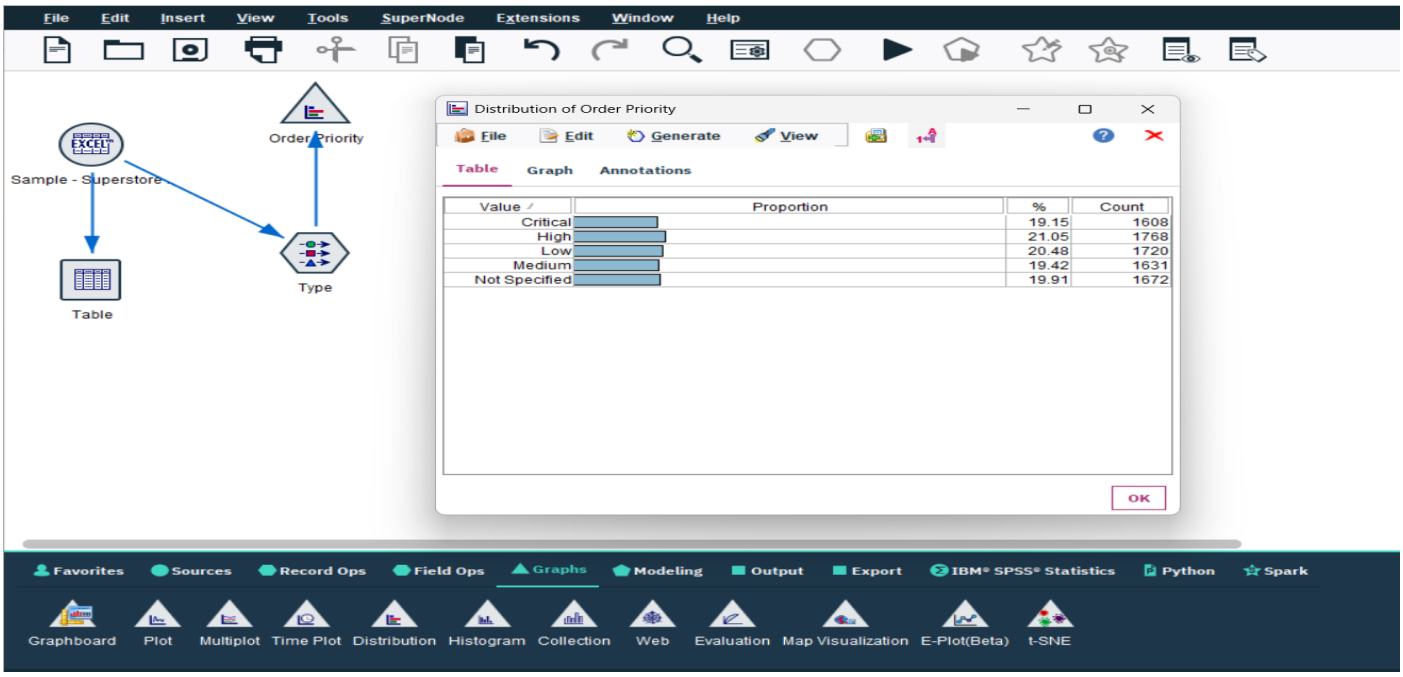
Data Preview – Displays first 10 records of the imported dataset from Excel.

Step 2: "Assign Field Roles and Measurement Levels"

Description :
Used the Type node to assign correct measurement levels and roles (Input, Target, None) for each field to prepare data for transformation and modeling.

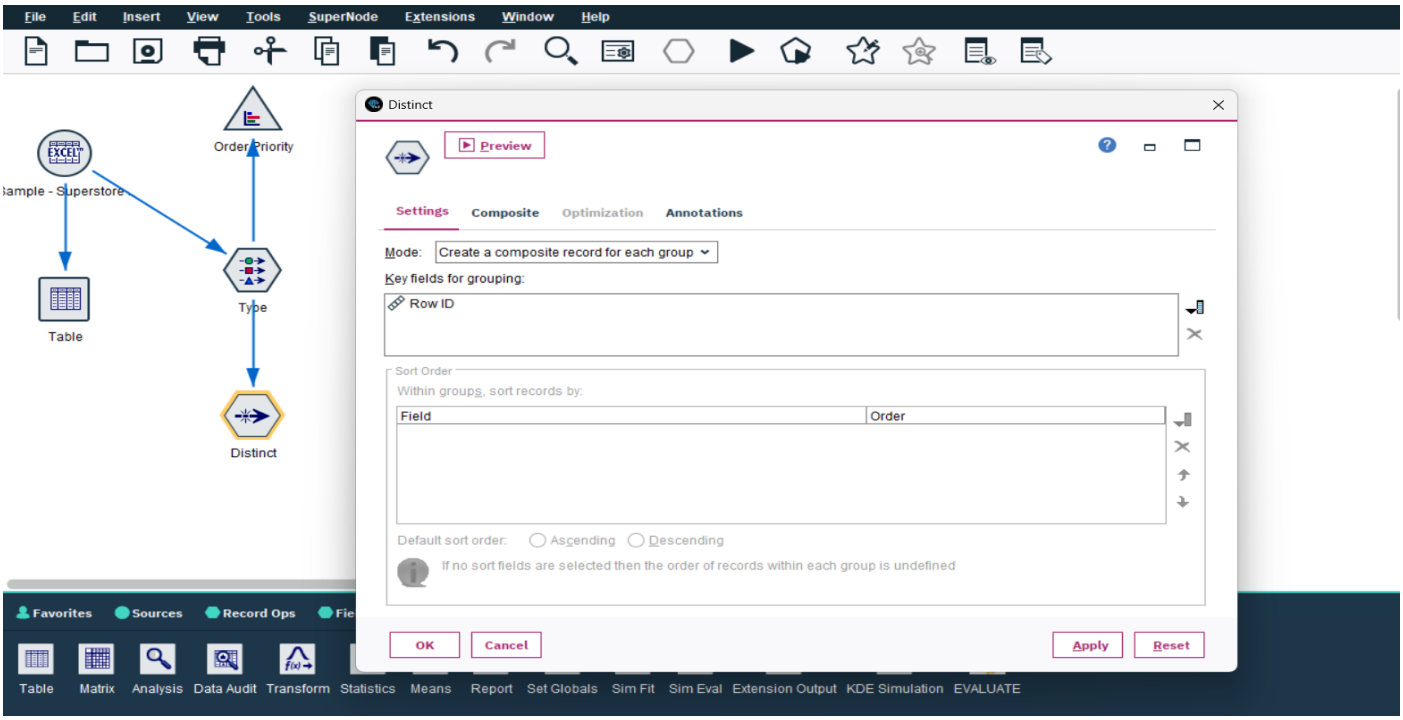


Type Node – Defined field types (numeric, categorical, string) and set their roles to ensure accurate data processing in SPSS Modeler



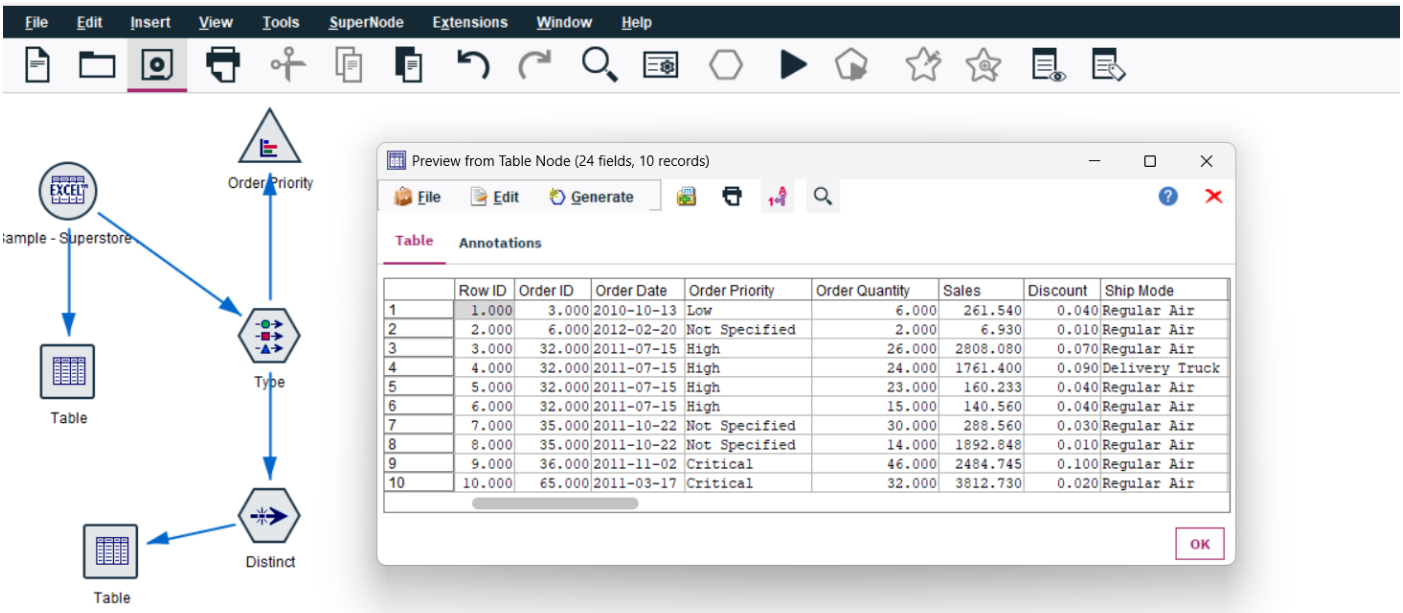
Type Node Distribution Graph – Displays the data distribution of numeric and categorical fields, helping identify data spread and missing values**.**

Step 3: "Remove Duplicate Records"



Description :

Applied the **Distinct node** using **Row ID** as the key field to remove duplicate records and ensure each entry in the dataset is unique.

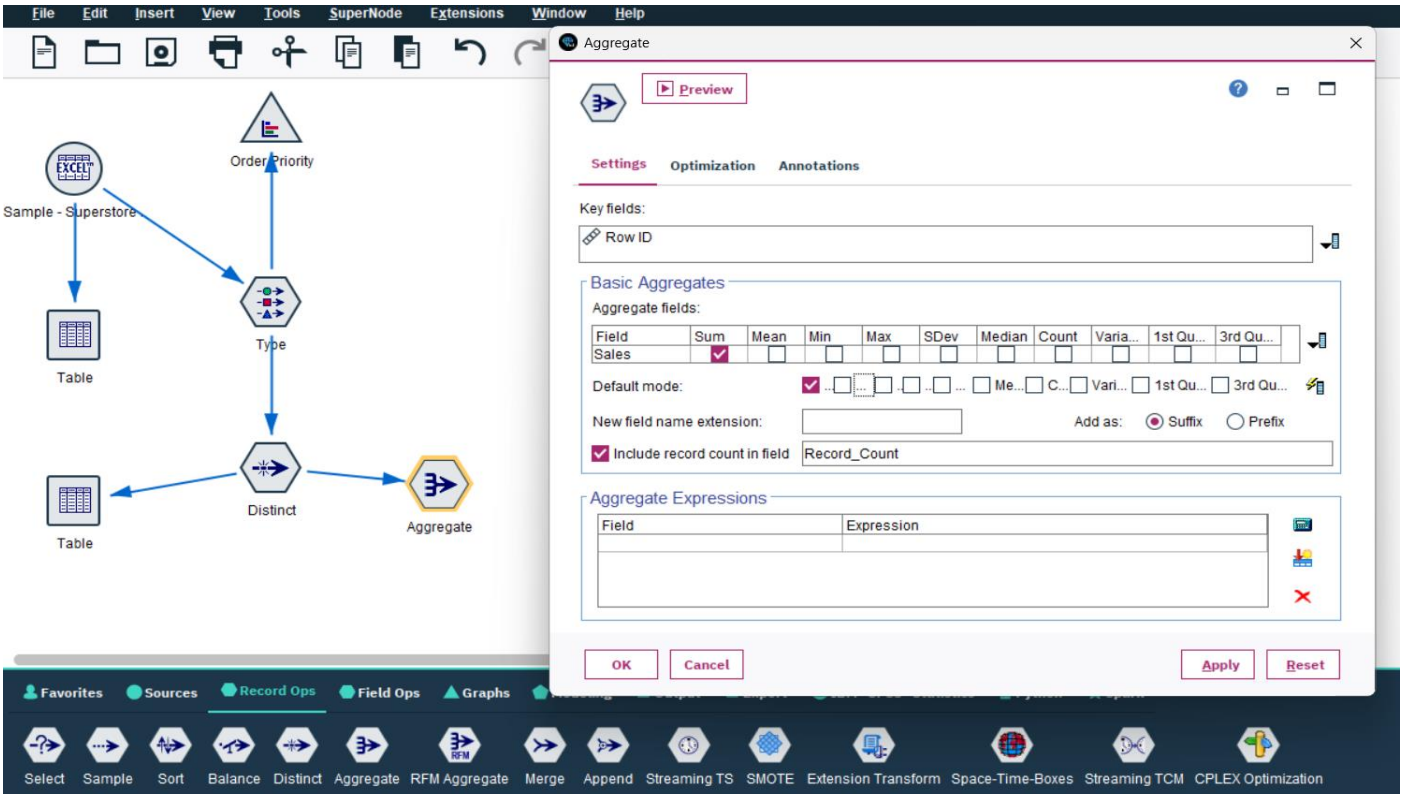


	Row ID	Order ID	Order Date	Order Priority	Order Quantity	Sales	Discount	Ship Mode
1	1.000	3.000	2010-10-13	Low	6.000	261.540	0.040	Regular Air
2	2.000	6.000	2012-02-20	Not Specified	2.000	6.930	0.010	Regular Air
3	3.000	32.000	2011-07-15	High	26.000	2808.080	0.070	Regular Air
4	4.000	32.000	2011-07-15	High	24.000	1761.400	0.090	Delivery Truck
5	5.000	32.000	2011-07-15	High	23.000	160.233	0.040	Regular Air
6	6.000	32.000	2011-07-15	High	15.000	140.560	0.040	Regular Air
7	7.000	35.000	2011-10-22	Not Specified	30.000	288.560	0.030	Regular Air
8	8.000	35.000	2011-10-22	Not Specified	14.000	1892.848	0.010	Regular Air
9	9.000	36.000	2011-11-02	Critical	46.000	2484.745	0.100	Regular Air
10	10.000	65.000	2011-03-17	Critical	32.000	3812.730	0.020	Regular Air

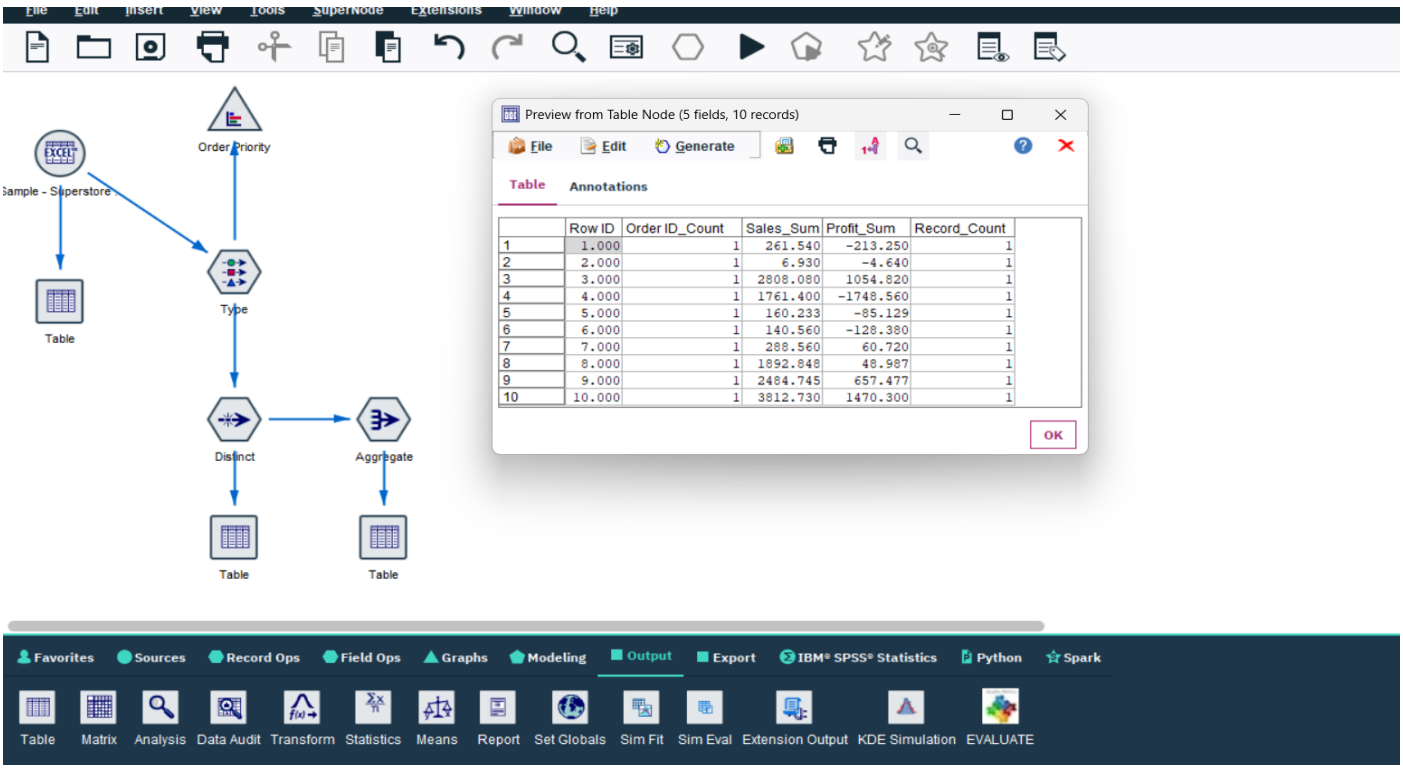
Distinct Node Table – Displays unique records identified using *Row ID*, confirming duplicate rows were successfully removed for cleaner data.

Step 4 : “ Aggregate Data”

Description :
Applied the **Aggregate node** using *Row ID* as the key field to calculate the total sales and record count for each unique record, summarizing data for further analysis.



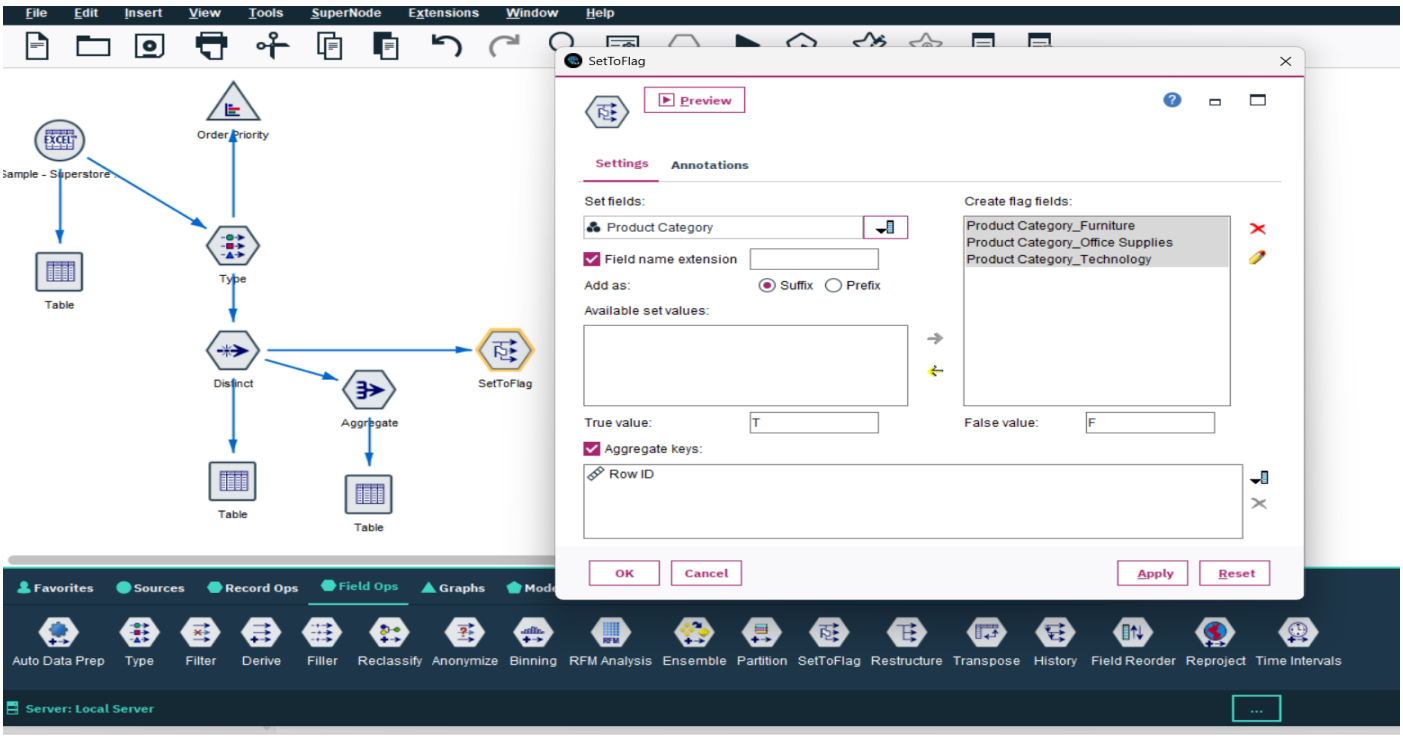
Aggregate Node – Shows the configuration where *Sales* is aggregated using the **Sum** function, and a new field **Record_Count** is added to count the number of records per **Row ID.**



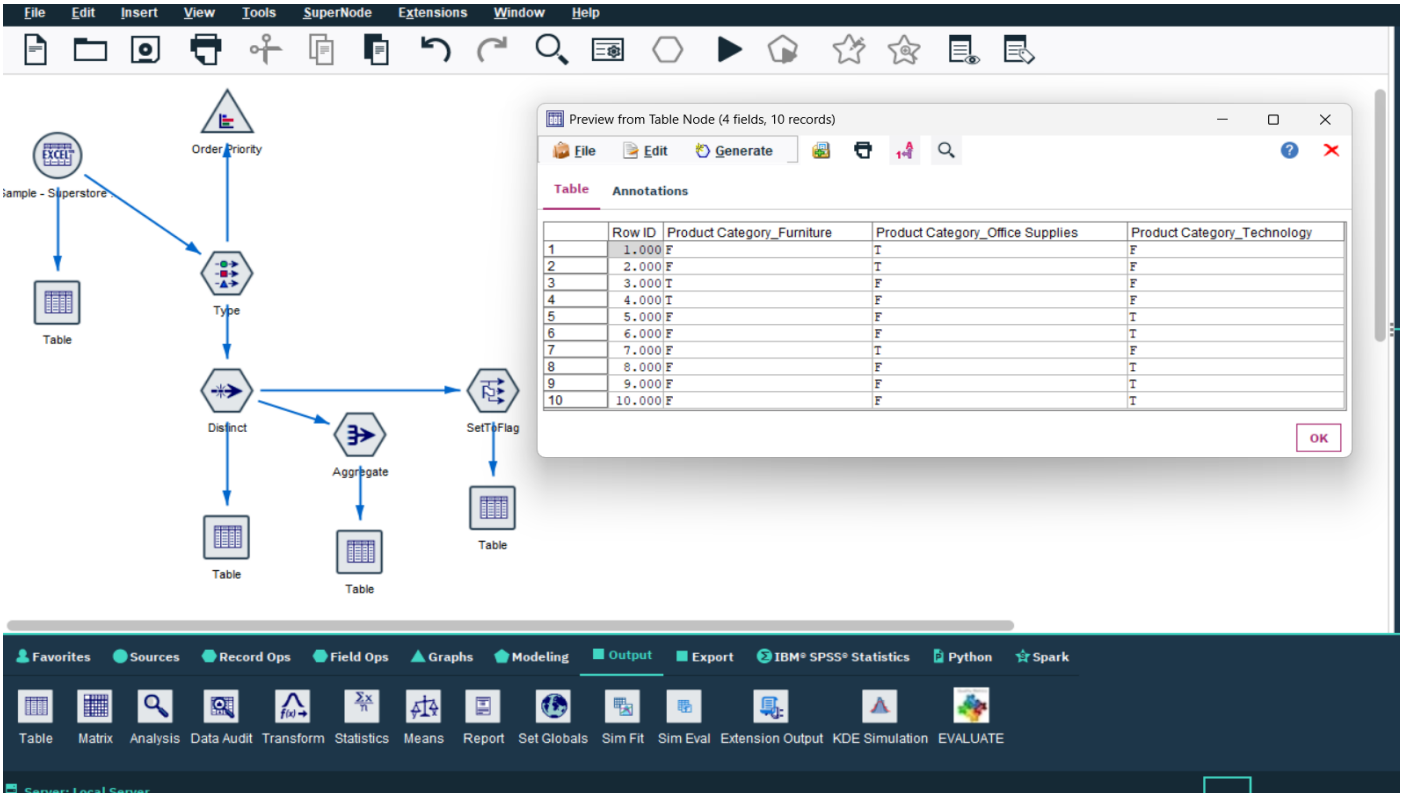
Aggregate Node Output Table – Displays the summarized dataset with total *Sales* and **Record_Count** fields, confirming successful aggregation of records.

Step 5 : “Create Category Flags”

Description :
Used the **SetToFlag** node to convert the “**Product Category**” field into separate binary flag variables for easy analysis and modeling.



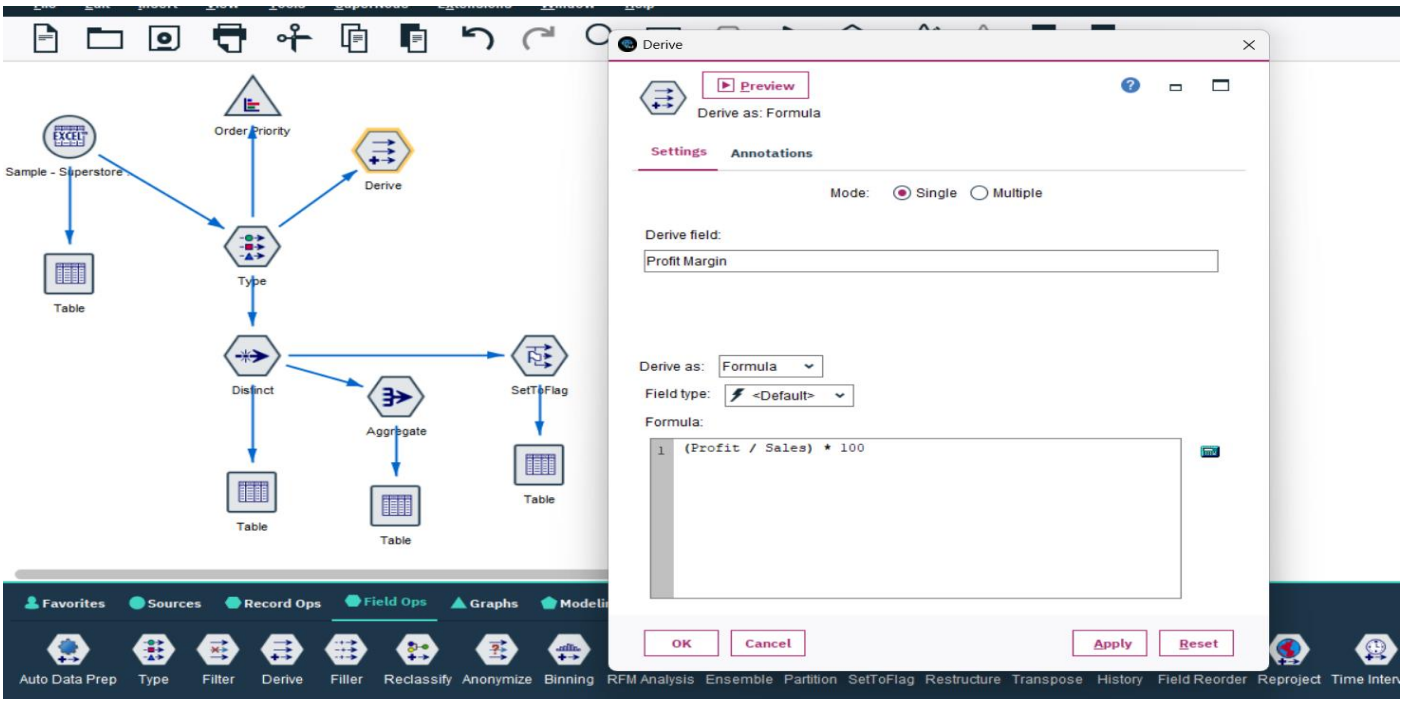
SetToFlag Node – Configured **Product Category** as the set field with flag variables created for **Furniture, Office Supplie***s*, and **Technology**. Each category is assigned **‘T’** for True and **‘F’** for False



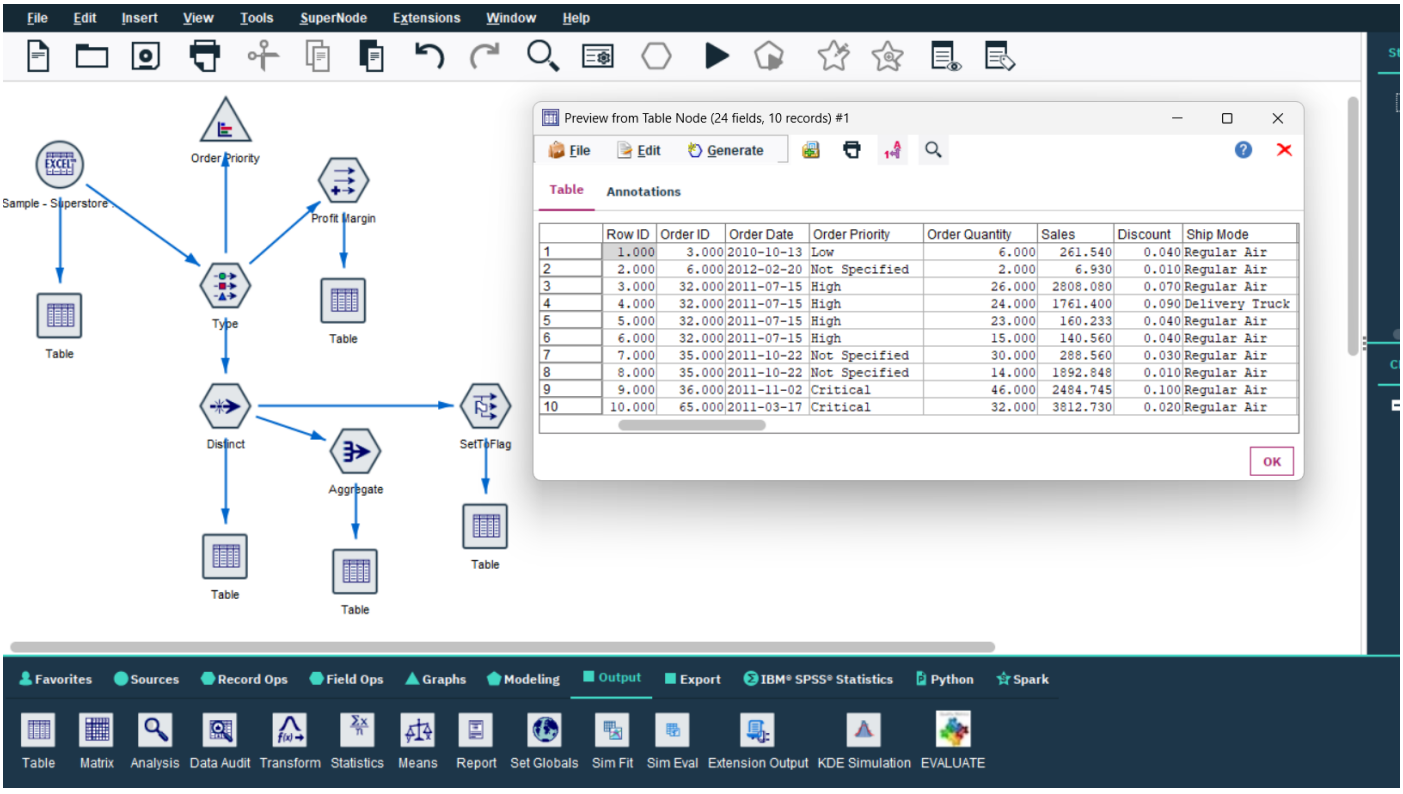
SetToFlag Output Table – Displays the newly created binary columns (**ProductCategory_Furniture, ProductCategory_OfficeSupplies, ProductCategory_Technology),** confirming successful transformation of categorical data into flags.

Step 6 : "Derive New Fields"

Description :
Used the Derive node to create a new calculated field named **Profit Margin** that measures profitability percentage for each record.



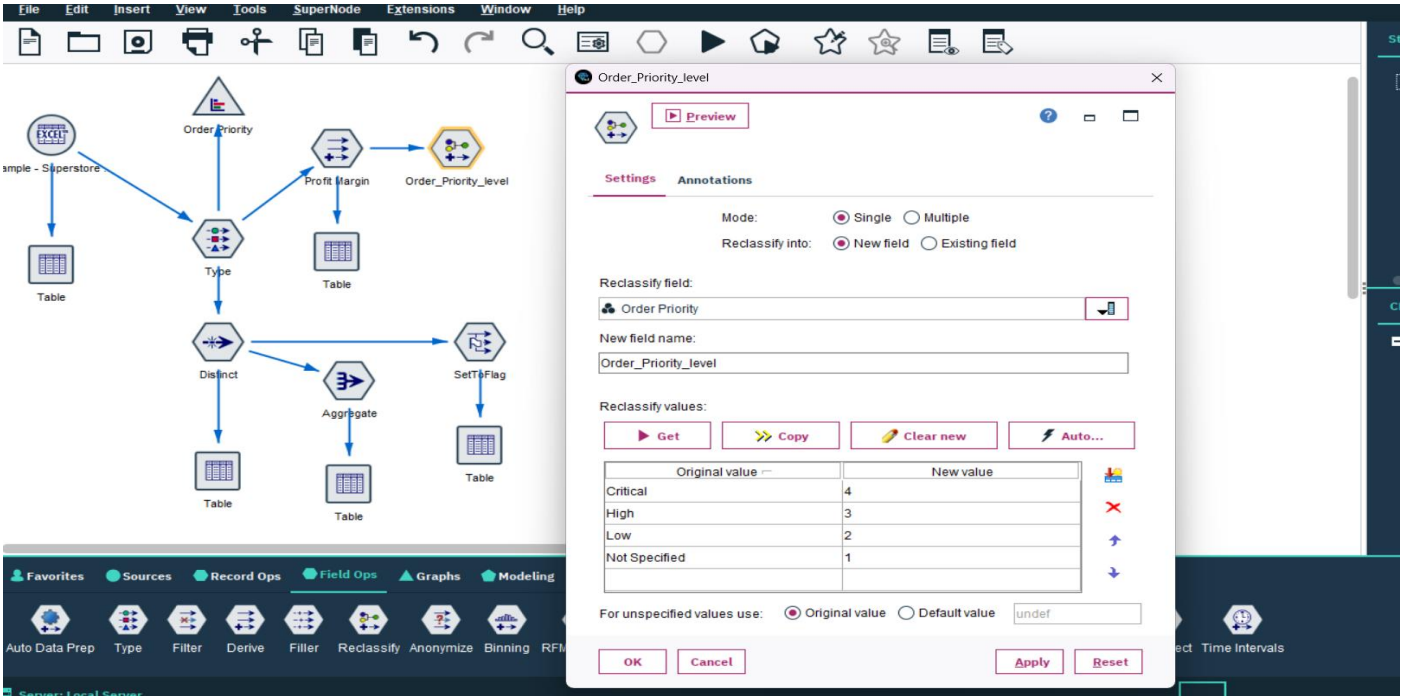
Derive Node – Configured to create the new field **Profit Margin** using the formula "***(Profit / Sales) × 100***", which calculates the profit percentage based on each order's sales value.



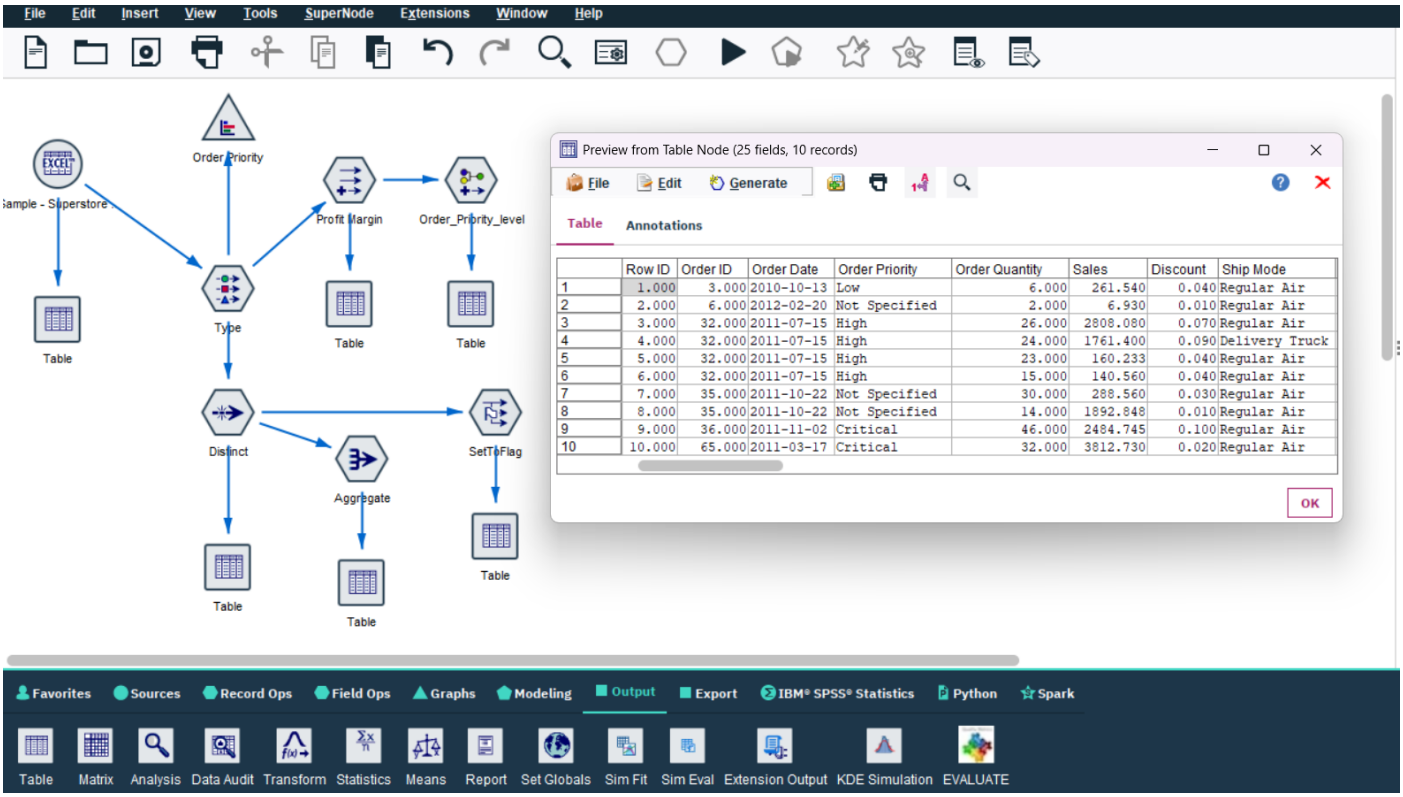
Derived Field Output Table – Displays the dataset including the newly added ***Profit Margin*** column, confirming successful field creation and correct integration with existing data.

Step 7 : "Reclassify Field Values"

Description :
Used the Reclassify node to assign numeric levels to different ***Order Priority*** categories for easier analysis and modeling.



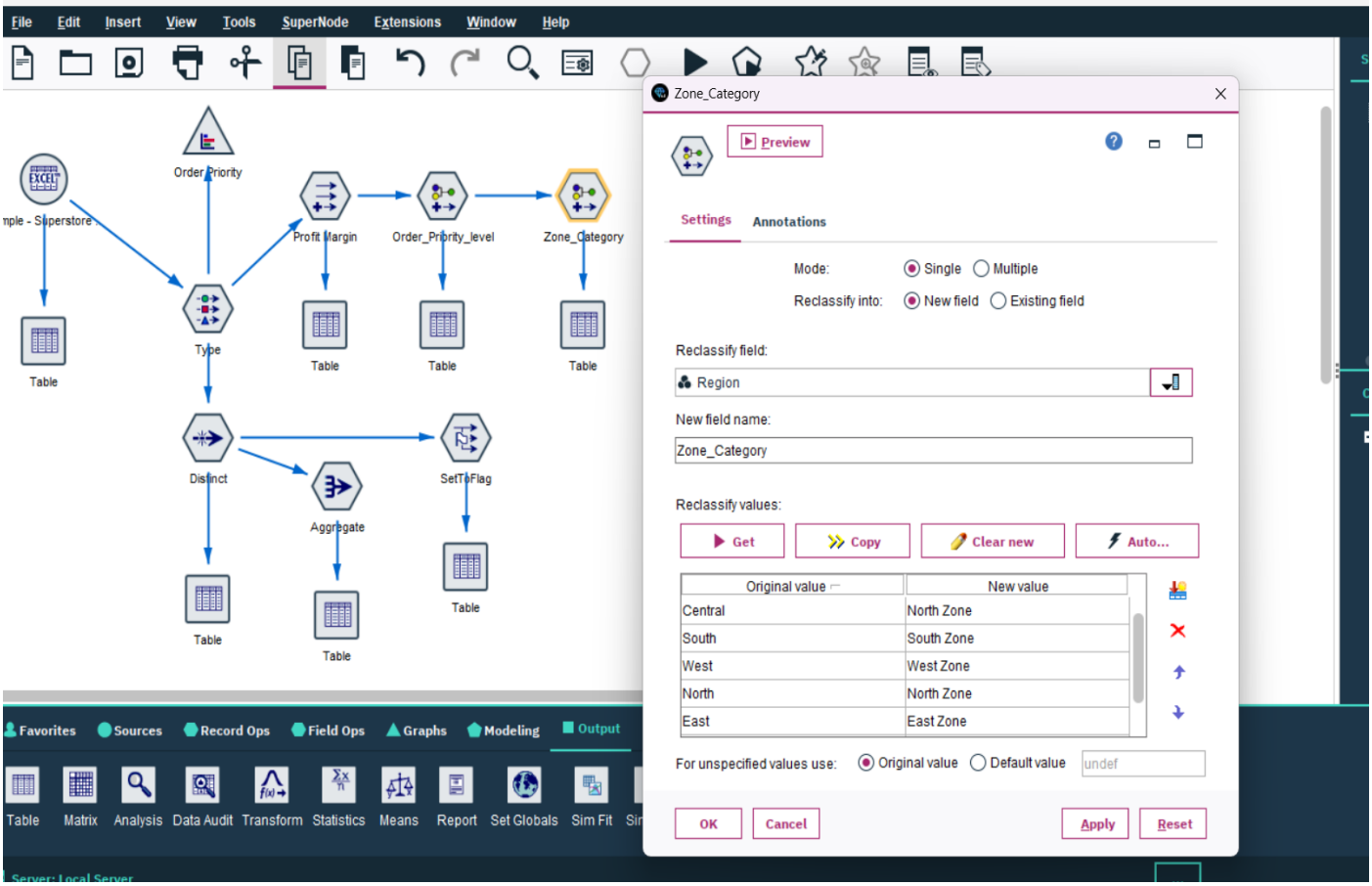
Reclassify Node – Configured the *Order Priority* field to create a new field named **Order_Priority_Level**, assigning numeric values (Critical = 4, High = 3, Low = 2, Not Specified = 1).



Reclassify Output Table – Displays the dataset including the new ***Order_Priority_Level*** column, confirming successful mapping of priority categories into numeric values.

Step 8 – "Zone_Category Node"

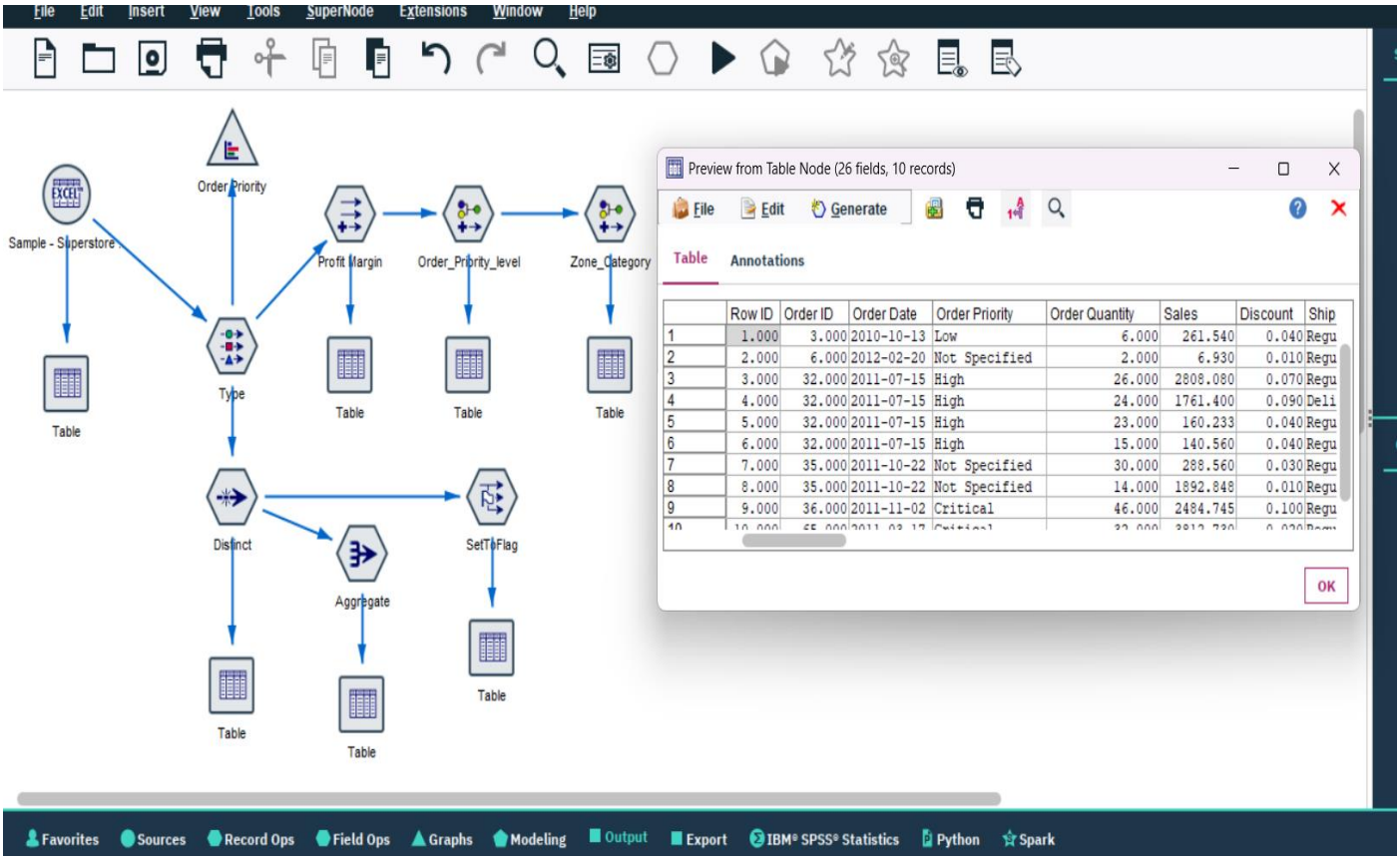
Description: In this step, a Reclassify node is used to create a new field named Zone_Category based on the *Region* field.
The purpose of this node is to convert the existing regional names into zone categories for easier analysis.



The first screenshot shows the **Zone_Category** node configuration window, where the *Region* field is reclassified into new categories. The second screenshot displays the **output preview table** showing the new derived field added to the dataset.

Process:

- The **Region** field is selected as the reclassify field.
- A new field named **Zone_Category** is created.
- Original region values are mapped as follows:
  - *Central* → *North Zone*
  - *South* → *South Zone*
  - *West* → *West Zone*
  - *North* → *North Zone*
  - *East* → *East Zone*
- After applying, the dataset is updated with the new "Zone_Category" column.



Output:

The output table preview shows the new **Zone_Category** field added successfully with the respective zone values according to each record

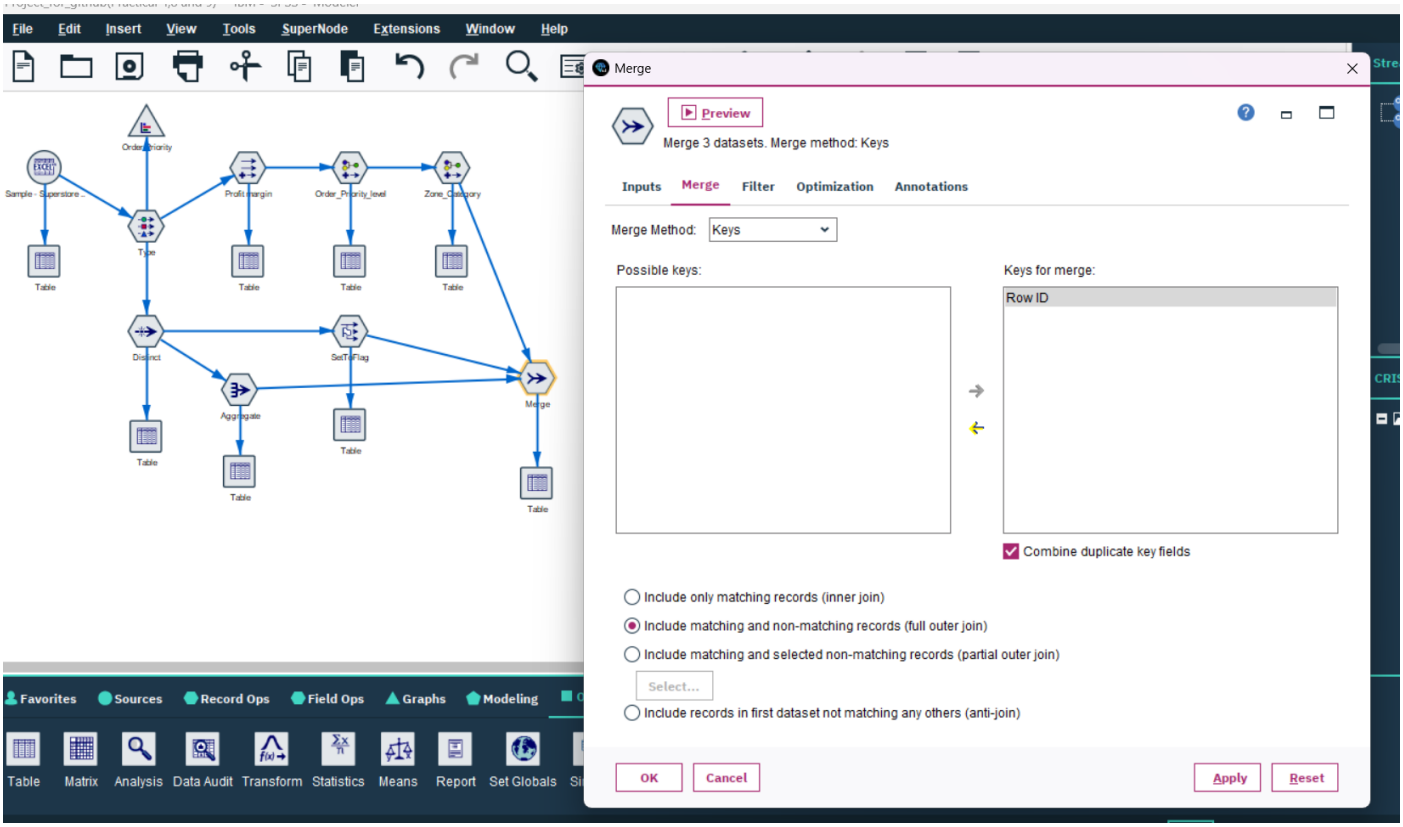
Step 9 : "Final Merge (Aggregate + SetToFlag + Zone_Category)"

Description :

This step combines all processed data—**Aggregate**, **SetToFlag**, and **Zone_Category**—into one complete dataset using the **Merge Node** with Row ID as the key field.

Detailed Process:

1. Connected the outputs from the **Aggregate**, **SetToFlag**, and **Zone_Category** nodes to a single **Merge Node**.
2. Opened the Merge Node and selected **Merge Method = Keys**.
3. Added **Row ID** as the key for merging the records.
4. Selected **Include matching and non-matching records (Full Outer Join)** to ensure all records from all inputs are included.
5. Checked **Combine duplicate key fields** to avoid repeating columns with the same names.
6. Clicked **Apply → OK**, then attached a **Table Node** to view the final integrated dataset.



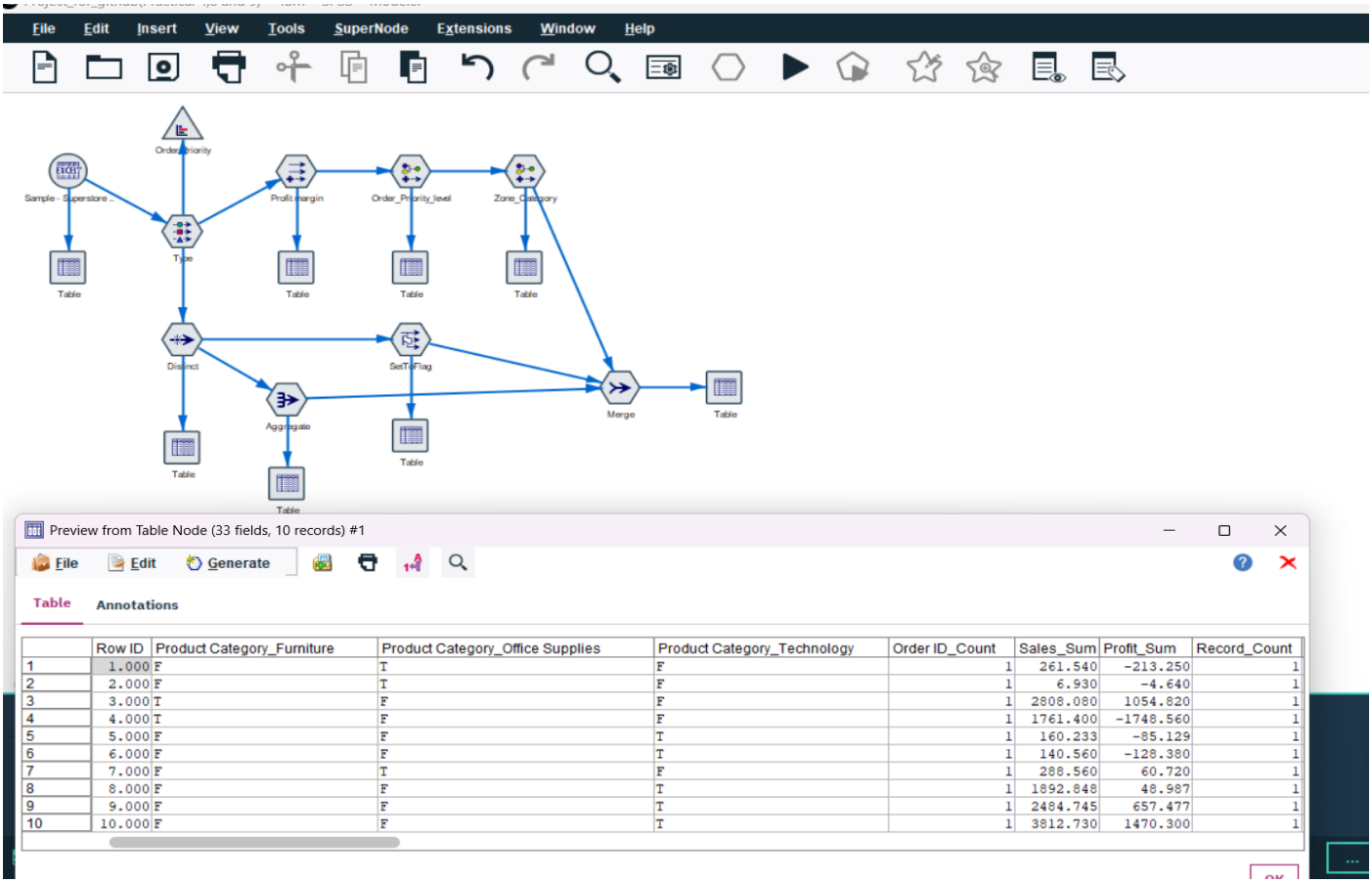
Merge Node Settings – *Using Row ID with Full Outer Join to merge Aggregate, SetToFlag, and Zone_Category outputs.*

Output Table Description:

The final output table shows a **complete view of the Superstore dataset** after all transformations. It includes:

- Product Category Flags (Furniture, Office Supplies, Technology)
- Aggregated Sales and Profit values
- Record Count and Order ID Count
- Derived fields like Profit Margin, Order Priority Level, and Zone Category

This final dataset provides an overall summary of all business aspects — making it ready for reporting, visualization, or further statistical analysis.



	Row ID	Product Category_Furniture	Product Category_Office Supplies	Product Category_Technology	Order ID_Count	Sales_Sum	Profit_Sum	Record_Count
1	1.000	F	T	F	1	261.540	-213.250	1
2	2.000	F	T	F	1	6.930	-4.640	1
3	3.000	T	F	F	1	2808.080	1054.820	1
4	4.000	T	F	F	1	1761.400	-1748.560	1
5	5.000	F	F	T	1	160.233	-85.129	1
6	6.000	F	F	T	1	140.560	-128.380	1
7	7.000	F	T	F	1	288.560	60.720	1
8	8.000	F	F	T	1	1892.848	48.987	1
9	9.000	F	F	T	1	2484.745	657.477	1
10	10.000	F	F	T	1	3812.730	1470.300	1

Final Merged Output Table *– Complete dataset displaying all combined fields including Sales, Profit, Category Flags, and Zone details.*

Conclusion:

1. The project was performed on the **Sample – Superstore Sales** dataset using **IBM SPSS Modeler**.
2. Multiple nodes such as **Type, Distinct, Aggregate, SetToFlag, Derive, and Reclassify** were used for data preparation and transformation.
3. Each node helped in:
    - Cleaning and identifying data types (Type Node)
    - Removing duplicate rows (Distinct Node)
    - Summarizing key metrics like *Sales* and *Profit* (Aggregate Node)
    - Creating category-based flags (SetToFlag Node)
    - Calculating *Profit Margin* using a derived formula (Derive Node)
    - Reclassifying order priority and region into numeric and zone categories (Reclassify Node)
4. Finally, all processed datasets were merged using the **Merge Node** based on the common key field Row ID.
5. The final merged output displays:
    - Total Sales, Profit, and Record Count

- Product Category Flags
- Profit Margin and Order Priority Level
- Zone Category (North, South, East, West)

6. The final dataset gives a **complete business overview**, helping to analyze product performance, sales trends, and regional distribution.
7. This project demonstrates **data cleaning, transformation, integration, and summarization** techniques in SPSS Modeler.
8. The final output can now be used for **visualization in Power BI or Excel**, and to support **business insights and decision-making**.

Thank You

==========X==========