# Week 9

# Name: Siddharth Dudugu

# Data Cleansing and Transformation

1. **Data Type Correction:**
    - **Objective:** Correct the data types of specific columns.
    - **Code:**

```
df['Date'] = pd.to_datetime(df['Date'], format='%m/%d/%Y')
```

    - **Explanation:** Converts the 'Date' column to the datetime format for consistency and proper handling of date-related operations.
2. **Handling Missing Values:**
    - **Objective:** Ensure that missing values are appropriately addressed.
    - **Code:**

```
df = df.dropna() # Drop rows with missing values
```

    - **Explanation:** Removes rows with missing values, ensuring that the dataset remains complete for subsequent analyses.
3. **Outlier Handling:**
    - **Objective:** Address potential outliers in the 'Sales' column.
    - **Code:**

```
df = df[df['Sales'] <= 100000] # Remove sales data points above 100,000
```

    - **Explanation:** Filters out data points with sales values above 100,000 to mitigate the impact of outliers on summary statistics and models.
4. **Boolean Column Removal:**
    - **Objective:** Remove boolean columns with limited occurrences.
    - **Code:**

```
df = df.drop(['V_DAY', 'EASTER', 'CHRISTMAS'], axis=1)
```

    - **Explanation:** Removes columns 'V_DAY,' 'EASTER,' and 'CHRISTMAS' as they have limited occurrences and may not contribute significantly to the analysis.

5. **Duplicate Entry Removal:**

- **Objective:** Ensure data integrity by handling duplicate entries.
- **Code:**

```
df = df.drop_duplicates(subset=['Product', 'Sales', 'Price Discount (%)', 'Date'])
```

- **Explanation:** Drops duplicate entries based on a subset of columns to prevent distortions in subsequent analyses.

6. **Product Code Transformation:**
   - **Objective:** Convert 'Product' codes to integer format.
   - **Code:**

```
df['Product'] = df['Product'].str.replace('SKU', '').astype(int)
```

   - **Explanation:** Replaces 'SKU' with integer values in the 'Product' column, facilitating numerical operations and modeling.