**For the Training Set:**

**1 Analysed the Distribution of Variables:**

◦ Conducted an exploratory data analysis (EDA) to examine the distribution patterns of both numerical and categorical features within the training dataset. This process helps to understand the characteristics and behaviours of each variable, including identifying potential issues like skewness, outliers, or anomalies.

**2 Imputed Missing Values:**

◦ For numerical columns where the data followed a roughly uniform or normal distribution, missing values were imputed using the **mean** of the respective column. This method ensures minimal bias while maintaining consistency in the dataset.

◦ For categorical columns, missing values were handled using two strategies:

**▪ Label Encoding:** Categorical variables were converted into numerical form by assigning integer labels to each unique category.

**▪ KNN Imputer:** For a more sophisticated approach, missing values were imputed by leveraging the K-Nearest Neighbours (KNN) algorithm, which predicts the missing values based on the nearest neighbours' values in the feature space.

**3 Outlier Treatment:**

◦ Extreme values, defined as outliers beyond the 99th percentile of the data distribution, were detected and replaced to ensure the model was not biased by anomalous data points.

◦ The outliers were replaced with the value at the **99th percentile**, ensuring the data remained within a reasonable range while preventing skewed results during model training.

**4 Scaling of Numerical Features:**

◦ Numerical columns were standardised using the **Min-Max Scaler** to ensure that all features were on the same scale. This prevents variables with larger numerical ranges from dominating the training process, allowing for more accurate model learning.

**5 One-Hot Encoding for Categorical Variables:**

◦ Categorical variables were transformed using **One-Hot Encoding**, which creates binary columns for each category within the variable. This ensures that the model can effectively learn from non-ordinal categorical data, preventing the model from interpreting any inherent order in the data.

**6 Model Training and Hyperparameter Tuning:**

◦ Multiple machine learning models were trained with a focus on minimizing **Root Mean Squared Error (RMSE)** as the evaluation metric. This is a robust choice for regression tasks, as it penalizes large errors more significantly.

◦ Hyperparameter optimization was performed using techniques such as **Grid Search** or **Randomized Search**, aiming to find the best combination of parameters for each model to improve its predictive accuracy.

**7 Model Selection and Fitting:**

◦ After evaluating the performance of the models, the best-performing model was selected based on cross-validation results and its ability to minimize RMSE.

◦ The selected model was then fit on the entire training dataset, ensuring that the model is fully trained with all available information before applying it to the test set.

**For the Test Set:**

**1 Imputation of Missing Values in Numerical Columns:**

◦ Missing values in numerical columns in the test set were imputed with the **mean** of the corresponding numerical column from the training set. This ensures consistency, as the model is trained with the training set's characteristics and we aim to apply the same transformation to the test set to avoid data leakage.

**2 Imputation of Missing Values in Categorical Columns:**

◦ The same **KNN Imputer** or **Label Encoder** model used in the training phase was applied to the test set's categorical variables. This ensures that any missing categorical data in the test set is handled identically to how it was managed during model training.

**3 Outlier Replacement in Test Set:**

◦ Outliers in the test set that exceed the 99th percentile observed in the training set were replaced with the **99th percentile value** from the training set. This approach guarantees that the test set adheres to the same data constraints as the training set, preventing the model from making predictions based on unrealistic test data values.

**4 Scaling of Test Set Features:**

◦ The **Min-Max Scaler** fitted on the training set was used to transform the numerical columns in the test set. This ensures that the test set features are scaled consistently with the training data, preserving the integrity of the model's performance.

**5 Sales Prediction Using Trained Model:**

◦ Finally, the trained model was used to predict the target variable (sales values) on the test data. The model applies the learned patterns from the training phase to generate sales predictions for unseen data, providing valuable insights for decision-making.