```
# Step 1: Import Libraries import pandas as pd import
numpy as np import seaborn as sns import
matplotlib.pyplot as plt from sklearn.ensemble import
RandomForestRegressor from sklearn.model_selection
import train_test_split from sklearn.preprocessing import
LabelEncoder from sklearn.metrics import
mean_squared_error, r2_score

# Step 2: Load Data df = pd.read_csv("customer_survey_data.csv")  #
Update with your filename

# Step 3: Initial Data Check
print(df.head()) print(df.info())
print(df.describe())

# Step 4: Handle Missing Values df.fillna(df.mean(numeric_only=True),
inplace=True) df.dropna(inplace=True)  # Drop if essential fields still
have NA

# Step 5: Encode Categorical Variables
label_encoders = {} for col in
df.select_dtypes(include='object').columns:
    le = LabelEncoder()     df[col] =
le.fit_transform(df[col])
label_encoders[col] = le

# Step 6: Correlation Matrix plt.figure(figsize=(12,8))
sns.heatmap(df.corr(), annot=True,
cmap="coolwarm") plt.title("Correlation Heatmap")
plt.show()
```

```
# Step 7: Define Features and Target target = 'Customer_Satisfaction'
# Replace with actual column name X = df.drop(columns=[target]) y
= df[target]

# Step 8: Train-Test Split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Step 9: Model Training model =
RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

# Step 10: Predictions and Evaluation y_pred =
model.predict(X_test) print("R² Score:", r2_score(y_test,
y_pred)) print("RMSE:", np.sqrt(mean_squared_error(y_test,
y_pred)))

# Step 11: Feature Importance importances = model.feature_importances_
feature_importance_df = pd.DataFrame({'Feature': X.columns, 'Importance': importances})
feature_importance_df.sort_values(by='Importance', ascending=False, inplace=True)

# Step 12: Plot Feature Importance
plt.figure(figsize=(10,6)) sns.barplot(x='Importance', y='Feature',
data=feature_importance_df) plt.title("Key Drivers of Customer
Satisfaction") plt.show()

# Step 13: Top Insights print("\nTop Drivers of
Customer Satisfaction:")
print(feature_importance_df.head())
```

Expected Outputs:

1. Initial Data Check

python

CopyEdit

```
print(df.head())
print(df.info())
print(df.describe())
```

- **df.head()**: Displays the first 5 rows of the dataset.
- **df.info()**: Summary of the dataset, including column types and non-null counts.
- **df.describe()**: Statistical summary of numerical columns (mean, std, min, 25%, 50%, 75%, max).

2. Correlation Heatmap

- A heatmap showing the correlation between all numeric columns, using color gradients (from blue to red). This visualizes how each feature is related to others, including the target (Customer_Satisfaction).

3. Model Evaluation Metrics

python

CopyEdit

```
print("R² Score:", r2_score(y_test, y_pred))
print("RMSE:",
np.sqrt(mean_squared_error(y_test, y_pred)))
```

- **R^2 Score**: Indicates how well the regression model fits the test data. A value closer to 1 means a better fit.

- **RMSE (Root Mean Squared Error)**: Shows average prediction error in the same units as the target variable. Lower is better.

4. Feature Importance Plot

- A horizontal bar chart that shows the top features driving customer satisfaction, based on how influential they were in the Random Forest model.

5. Top 5 Features

python

CopyEdit

```
print("\nTop Drivers of Customer Satisfaction:")
print(feature_importance_df.head())
```

- Displays a table of the top 5 most important features (columns) in determining customer satisfaction.
