```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset
df = pd.read_csv('heart_disease.csv')

# 1. Data Cleaning
# - Handle missing values (replace with mean/median or drop rows)
df = df.fillna(df.mean())  # Replace missing values with mean

# - Remove outliers (e.g., using IQR)
Q1 = df['age'].quantile(0.25)
Q3 = df['age'].quantile(0.75)
IQR = Q3 - Q1
df = df[~((df['age'] < (Q1 - 1.5 * IQR)) | (df['age'] > (Q3 + 1.5 * IQR)))]

# 2. Exploratory Data Analysis (EDA)
# - Summary statistics
print(df.describe())

# - Data distribution
plt.figure(figsize=(10, 6))
sns.histplot(df['age'], bins=20)
plt.title('Age Distribution')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()

# - Correlation matrix
plt.figure(figsize=(10, 8))
sns.heatmap(df.corr(), annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()

# 3. Question Formulation and Analysis
# - Q1: What is the average age of patients with heart disease?
avg_age_hd = df[df['target'] == 1]['age'].mean()
print(f"Average age of patients with heart disease: {avg_age_hd:.2f}")

# - Q2: Is there a relationship between cholesterol and heart disease?
sns.boxplot(x='target', y='chol', data=df)
plt.title('Cholesterol Levels by Heart Disease')
plt.xlabel('Heart Disease (0: No, 1: Yes)')
plt.ylabel('Cholesterol')
plt.show()

# - Q3: How does smoking affect the risk of heart disease?
sns.countplot(x='target', hue='smoker', data=df)
plt.title('Smoking and Heart Disease')
plt.xlabel('Heart Disease (0: No, 1: Yes)')
plt.ylabel('Count')
plt.show()

# - Q4: ... (Add more questions and analysis here)

# 4. Data Visualization
# - Create more visualizations based on the analysis, such as bar plots, scatter plots, etc.

# Note: This code is a basic example and can be further enhanced with more sophisticated data cleaning, feature engineering, and machine learning model
```

shell

Copy Edit

```
              age        chol  ...  other_numeric_columns
count   290.000000  290.000000  ...
mean     55.634483  246.692345  ...
std       9.623245   51.776462  ...
min      29.000000  126.000000  ...
25%      48.000000  211.000000  ...
50%      56.000000  240.000000  ...
75%      63.000000  274.000000  ...
max      77.000000  564.000000  ...
```