**Guiding Questions.**

1. Importing the Libraries

2.Top 5 Rows of The Dataset

3.Shape of the Dataset (Number of Rows And Number of Columns)

4. Get Information About Our Dataset Like Total Number Rows, Total Number of Columns, Datatypes of Each Column And Memory Requirement

Check Null Values In The Dataset bold text

6. Check For Duplicate Data and Drop Them

7. Get Overall Statistics About The Dataset

8. Draw Correlation Matrix

How Many People Have Heart Disease, And How Many Don't Have Heart Disease In This Dataset? bold text

10. Find the gender that had The Most Heart Disease [ Male or Female ]?

11. Find Gender Distribution According to The Target Variable.

12. Check Age Distribution In The Dataset

14. Check Chest Pain Type

14. Show The Chest Pain Distribution As Per Target Variable

15. Show Fasting Blood Sugar Distribution According To Target Variable

16 Compare Resting Blood Pressure As Per Sex Column

17. Show Distribution of Serum cholesterol

18.Plot the continious variable

## **The steps below provide a structured approach to understand and analyze the heart disease dataset, enabling one to draw meaningful insights and make informed recommendations.**

1. Importing Libraries:

- Imported necessary Python libraries including `pandas`, `seaborn`, `matplotlib.pyplot`, and `numpy`.

2. Loading the Dataset:

- Loaded the heart disease dataset from a CSV file using `pandas`.

3. Displaying Top Rows:

- Displayed the top 5 rows of the dataset to get an initial look at the data using `data.head()`.

4. Checking Dataset Shape:

- Checked the number of rows and columns in the dataset using `data.shape`.

5. Dataset Information:

- Retrieved information about the dataset including the number of rows, columns, data types of each column, and memory usage using `data.info()`.

6. Checking for Null Values:

- Checked for any missing values in the dataset using `data.isnull().sum()`.

7. Checking for Duplicate Records:

- Identified and removed duplicate records in the dataset using `data.duplicated().any()` and `data.drop\_duplicates()`.

8. Statistical Summary:

- Generated overall statistics for the dataset including mean, standard deviation, minimum, and maximum values for each column using `data.describe()`.

9. Correlation Matrix:

- Created a correlation matrix to understand the relationships between different variables using `sns.heatmap(data.corr(), annot=True)`.

10. Heart Disease Distribution:

- Analyzed the distribution of the target variable (presence of heart disease) using `data["target"].value\_counts()`.

- Visualized this distribution with a bar chart using `data['target'].value\_counts().plot(kind='bar')`.

11. Gender Distribution:

- Examined the distribution of gender in the dataset using `data['sex'].value\_counts()`.

- Visualized gender distribution with a bar chart using `data['sex'].value\_counts().plot(kind='bar')`.

12. Heart Disease by Gender:

- Analyzed the prevalence of heart disease by gender using `sns.countplot(x='sex', hue='target', data=data)`.

- Customized the plot with appropriate labels for better interpretation.