Exploratory Data Analysis

Description of EDA

Exploratory Data Analysis (EDA) is a crucial step in the data analysis process. It involves examining and understanding the structure, patterns, and characteristics of a dataset before applying any formal statistical techniques or modeling. EDA helps us uncover valuable insights, detect outliers, identify data quality issues, and make informed decisions about data preprocessing and modeling approaches.

During EDA, we perform various data manipulaion and visualization techniques to gain a deeper understanding of the dataset. This includes tasks such as data cleaning, handling missing values, removing duplicates, transforming variables, calculating descriptive statistics, visualizing distributions, and exploring relationships between variables.

By conducting EDA, we can:

- Identify data quality issues such as missing values, duplicates or outliers
- Understand the distribution and summary of each variable.
- Explore relationships and correlations between variable.
- Identify patterns, trends, or anomalies in the data.
- Formualate initial hypotheses for further analysis or modeling.

Exploratory Data Analysis (EDA) is an Iterative process that often requires domain knowledge, critical thinking, and creativity. It helps us uncover insights and ask relevant questions that drive subsequent analysis and decision-making.

Questions

> Import the pandas library

This quesiton involves importing the pandas library, a popular Python library used for data manipulation and analysis. It is a fundamental step in working with tabular data and performing various data operations.

Here, we have imported this python library to read the csv files.

In [1]:

```
import pandas as pd
import scipy as stats

data = pd.read_csv('Emp_EDA.csv')
data
```

Out[1]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Se Manager
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000	F
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	F
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	F
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500	
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000	F
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700	
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300	
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000	
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000	
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000	F
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000	F
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900	
16	Shawn	Male	111737	Human Resources	39.0	15	118903.81120	11500	F
17	Gary	Male	109831	Product	39.0	15	116235.24560	11500	F
18	Jeremy	Male	90370	Human Resources	42.0	18	97029.36530	11000	F
19	Kimberly	Female	41426	Finance	44.0	20	44512.23700	11000	
20	Louise	Female	63241	Business Development	45.0	21	72810.62812	10800	
21	Donna	Female	81014	Product	49.0	23	82548.40516	10600	F
22	Ruby	Female	65476	Product	54.0	25	72031.45712	10400	
23	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	F
24	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	

> Remove the irrelevant column 'Senior Management'

Here, the task is to remove the 'Senior Management' column from the dataset as it is deemed irrelevant for the analysis or does not contribute to the research objectives.

In [2]:

```
data.drop(columns = ['Senior Management'], inplace=True)
data
```

Out[2]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900
16	Shawn	Male	111737	Human Resources	39.0	15	118903.81120	11500
17	Gary	Male	109831	Product	39.0	15	116235.24560	11500
18	Jeremy	Male	90370	Human Resources	42.0	18	97029.36530	11000
19	Kimberly	Female	41426	Finance	44.0	20	44512.23700	11000
20	Louise	Female	63241	Business Development	45.0	21	72810.62812	10800
21	Donna	Female	81014	Product	49.0	23	82548.40516	10600
22	Ruby	Female	65476	Product	54.0	25	72031.45712	10400
23	Lillian	Female	59414	Product	55.0	26	60160.23984	10300
24	Lillian	Female	59414	Product	55.0	26	60160.23984	10300

> Remove the duplicate rows and analyse

In this question, we need to identify and remove any duplicate rows from the dataset. Duplicate rows can skew analysis results and leat to incorrect insights. After removing duplicates, further analysis and insights can be derived from the cleaned dataset.

In [3]:

```
data.drop_duplicates(subset = "Gender", keep=False, inplace=True)
data
```

Out[3]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus
7	Frances	NaN	139852	Business Development	34.0	10	150374.4645	15500

> Rename the column 'bonus' to 'Incentive'

This question involves renaming the column 'bonus' to 'Incentive' to align with the terminology or specific requirements of the analysis.

In [4]:

```
data.rename(columns={'Bonus':'Incentive'}, inplace=True)
data
```

Out[4]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Incentive
7	Frances	NaN	139852	Business Development	34.0	10	150374.4645	15500

In [5]:

```
data1 = pd.read_csv('Emp_EDA.csv')
```

> Calculate the central tendency measures for 'Experience'

Here, the task is to calculate the central tendancy measures, such as mean, median, mode, for the 'Experience' Column. These measures provide insights into the typical or central values of the variable.

In [6]:

```
# 4) Calculate the central tendency measures for 'Experience'
print("Mean value for experience column:")
mean = data1[["Experience"]].mean()
print(mean)

print()

print("Median value for the experience column:")
median = data1[["Experience"]].median()
print(median)

print()

print("Mode value for the experience column:")
mode = data1[["Experience"]].mode()
print(mode)
```

```
Mean value for experience column:
Experience    13.68
dtype: float64

Median value for the experience column:
Experience    12.0
dtype: float64

Mode value for the experience column:
   Experience
0           5
1           7
2          10
3          11
4          12
5          13
6          15
7          26
```

> Calculate the variability measures for 'Experience'

This question requires calculating variability measures, such as variance and standard deviation, for the 'Experience' column. These measures quantify the dispersion or spread of values around the central tendency, providing insights into the data's variability.

In [7]:

```
print('Range:',data1['Experience'].max() - data1['Experience'].min())
print('Variance:',data1['Experience'].var())
print('Standard Deviation:',data1['Experience'].std())
```

```
Range: 21
Variance: 43.14333333333334
Standard Deviation: 6.568358496103371
```

> Calculate the IQR using quantile for 'Experience'

The interquartile range (IQR) is calculated using quantiles and provides information about the spread and distribution of data. This question involves calculating the IQR for the 'Experience' Column, which helps identify the range where the middle 50% of this data falls.

In [8]:

```
q1 = data1['Experience'].quantile(0.25)
q3 = data1['Experience'].quantile(0.75)
IQR = q3 - q1
print("IQR Value: ", IQR)
```

```
IQR Value:  9.0
```

> Calculate the z-score for 'Experience'

The z-score measures how many standard deviations a data point is from the mean. This question involves calculating the z-scores for the 'Experience' column, which allows us to assess the relative position of each data point within the distribution.

In [9]:

```
import scipy
from scipy import stats
data1['Experience'].fillna(0, inplace=True)
data1['Experience_zscore']=stats.zscore(data1['Experience'])
data1
```

Out[9]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Se Manager
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000	F
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	F
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	F
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500	
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000	F
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700	
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300	
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000	
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000	
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000	F
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000	F
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900	
16	Shawn	Male	111737	Human Resources	39.0	15	118903.81120	11500	F
17	Gary	Male	109831	Product	39.0	15	116235.24560	11500	F
18	Jeremy	Male	90370	Human Resources	42.0	18	97029.36530	11000	F
19	Kimberly	Female	41426	Finance	44.0	20	44512.23700	11000	
20	Louise	Female	63241	Business Development	45.0	21	72810.62812	10800	
21	Donna	Female	81014	Product	49.0	23	82548.40516	10600	F
22	Ruby	Female	65476	Product	54.0	25	72031.45712	10400	
23	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	F
24	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	

> Add 2 rows at the end of the Dataframe

This question requires adding two rows at the end of the dataframe using the append method. The provided values are used to populate the new rows.

In [10]:

```
# Given values:
# {&#39;First Name&#39;:&#39;Zion&#39;, &#39;Gender&#39;:&#39;Male&#39;, &#39;Team&#39;:&
# &#39;Experience&#39;:90,&#39;New_Salary&#39;:146075.4, &#39;Incentive&#39;:20000
# {&#39;First Name&#39;:&#39;Frances&#39;, &#39;Gender&#39;:&#39;Male&#39;, &#39;Salary&
# Development&#39;, &#39;Age&#39;:34, &#39;Experience&#39;:95, &#39;New_Salary&#39;:1503

data.append({'First Name':'Zion', 'Gender':'Male', 'Salary':'12345', 'Team':'Finance', '
```

Out[10]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Incentive
0	Frances	NaN	139852	Business Development	34.0	10	150374.4645	15500
1	Zion	Male	12345	Finance	37.0	90	146075.4000	20000

In [11]:

```
data.append({'First Name':'Frances', 'Gender':'Male', 'Salary':'13952', 'Team':'Business
```

Out[11]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Incentive
0	Frances	NaN	139852	Business Development	34.0	10	150374.4645	15500
1	Frances	Male	13952	Business Development	39.0	95	150374.5000	15500

> Replace NAN with a given values

Here, the task is to replace any Nan(missing) values in the dataset with a specified value. This ensures consistency and completeness in the dataset.

In [12]:

```python
# Given value (salary=130590)
data2 = pd.read_csv('Emp_EDA.csv')
data2.fillna(130590)
```

Out[12]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Senior Management
0	Maria	Female	130590	Finance	130590.0	5	146075.36220	20000	False
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	True
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	False
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	True
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	True
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	True
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	False
7	Frances	130590	139852	Business	34.0	10	150374.46450	15500	True

> Replace the NaN value in the Salary column with previous value, next value, linear interpolation, and central tendency measures:

This Question involves handling missing values specifically in the Salary Column. Different techniques such as using the previous value, next value, linear interpolation, or central tendency measures(mean, median) can be used to fill in the missing values.

In [13]:

```python
data3 = pd.read_csv('Emp_EDA.csv')
data3['Salary'].fillna(method='pad', inplace=True)
data3
```

Out[13]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Senior Management
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000	False
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	True
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	False
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	True
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	True
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	True
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	False
7	Frances	NaN	139852	Business	34.0	10	150374.46450	15500	True

In [14]:

```
data4 = pd.read_csv('Emp_EDA.csv')
data4['Salary'].fillna(method='bfill', inplace=True)
data4
```

Out[14]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Se Manager
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000	F
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	F
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	F
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500	
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000	F
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700	
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300	
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000	
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000	
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000	F
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000	F
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900	
16	Shawn	Male	111737	Human Resources	39.0	15	118903.81120	11500	F
17	Gary	Male	109831	Product	39.0	15	116235.24560	11500	F
18	Jeremy	Male	90370	Human Resources	42.0	18	97029.36530	11000	F
19	Kimberly	Female	41426	Finance	44.0	20	44512.23700	11000	
20	Louise	Female	63241	Business Development	45.0	21	72810.62812	10800	
21	Donna	Female	81014	Product	49.0	23	82548.40516	10600	F
22	Ruby	Female	65476	Product	54.0	25	72031.45712	10400	
23	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	F
24	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	

In [15]:

```
data5 = pd.read_csv('Emp_EDA.csv')
data5['Salary'].interpolate(method='linear', limit_direction='forward', inplace=True)
data5
```

Out[15]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Se Manager
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000	F
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	F
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	F
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500	
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000	F
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700	
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300	
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000	
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000	
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000	F
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000	F
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900	
16	Shawn	Male	111737	Human Resources	39.0	15	118903.81120	11500	F
17	Gary	Male	109831	Product	39.0	15	116235.24560	11500	F
18	Jeremy	Male	90370	Human Resources	42.0	18	97029.36530	11000	F
19	Kimberly	Female	41426	Finance	44.0	20	44512.23700	11000	
20	Louise	Female	63241	Business Development	45.0	21	72810.62812	10800	
21	Donna	Female	81014	Product	49.0	23	82548.40516	10600	F
22	Ruby	Female	65476	Product	54.0	25	72031.45712	10400	
23	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	F
24	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	

In [16]:

```
data6 = pd.read_csv('Emp_EDA.csv')
data6['Salary'].fillna(data['Salary'].mean(), inplace=True)
data6
```

Out[16]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Se Manage
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000	F
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	F
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	F
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500	
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000	F
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700	
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300	
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000	
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000	
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000	F
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000	F
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900	
16	Shawn	Male	111737	Human Resources	39.0	15	118903.81120	11500	F
17	Gary	Male	109831	Product	39.0	15	116235.24560	11500	F
18	Jeremy	Male	90370	Human Resources	42.0	18	97029.36530	11000	F
19	Kimberly	Female	41426	Finance	44.0	20	44512.23700	11000	
20	Louise	Female	63241	Business Development	45.0	21	72810.62812	10800	
21	Donna	Female	81014	Product	49.0	23	82548.40516	10600	F
22	Ruby	Female	65476	Product	54.0	25	72031.45712	10400	
23	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	F
24	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	

In [17]:

```
data7 = pd.read_csv('Emp_EDA.csv')
data7['Salary'].fillna(data['Salary'].median(), inplace=True)
data7
```

Out[17]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Se Manager
0	Maria	Female	130590	Finance	NaN	5	146075.36220	20000	F
1	Angela	Female	54568	Business Development	27.0	5	64675.63064	19000	
2	Allan	Male	125792	Client Services	28.0	6	132134.43260	18500	F
3	Rohan	Female	45906	Finance	28.0	7	51230.17788	18000	
4	Douglas	Male	97308	Marketing	28.0	7	104066.04060	17000	
5	Brandon	Male	112807	Human Resources	30.0	8	132539.20040	16000	
6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	F
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500	
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000	F
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700	
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300	
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000	
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000	
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000	F
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000	F
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900	
16	Shawn	Male	111737	Human Resources	39.0	15	118903.81120	11500	F
17	Gary	Male	109831	Product	39.0	15	116235.24560	11500	F
18	Jeremy	Male	90370	Human Resources	42.0	18	97029.36530	11000	F
19	Kimberly	Female	41426	Finance	44.0	20	44512.23700	11000	
20	Louise	Female	63241	Business Development	45.0	21	72810.62812	10800	
21	Donna	Female	81014	Product	49.0	23	82548.40516	10600	F
22	Ruby	Female	65476	Product	54.0	25	72031.45712	10400	
23	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	F
24	Lillian	Female	59414	Product	55.0	26	60160.23984	10300	

> Detect outliers in the updated 'Experience' column with boxplot, scatter plot, and histogram

Outliers are extreme values that significantly differ from other data points. This question requires detecting outliers in the updated 'Experience' column using visual techniques such as boxplot, scatter plot, and histogram. These visualizations help identify values that fall outside the expected range.

In [18]:

```
import matplotlib.pyplot as plt
data8 = pd.read_csv('Emp_EDA.csv')
plt.boxplot(data8['Experience'])
plt.show()
```

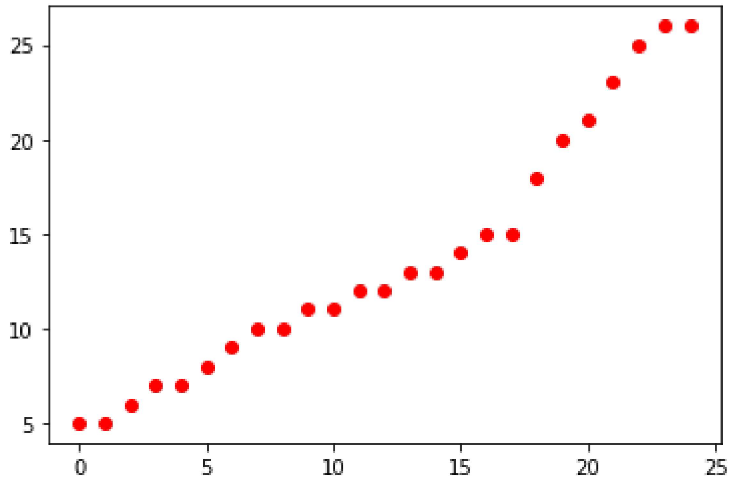
```
<Figure size 640x480 with 1 Axes>
```

In [19]:

```
plt.plot(data8['Experience'], linewidth=0, marker='o', color='red')
```

Out[19]:

```
[<matplotlib.lines.Line2D at 0x7f1e30a9d278>]
```

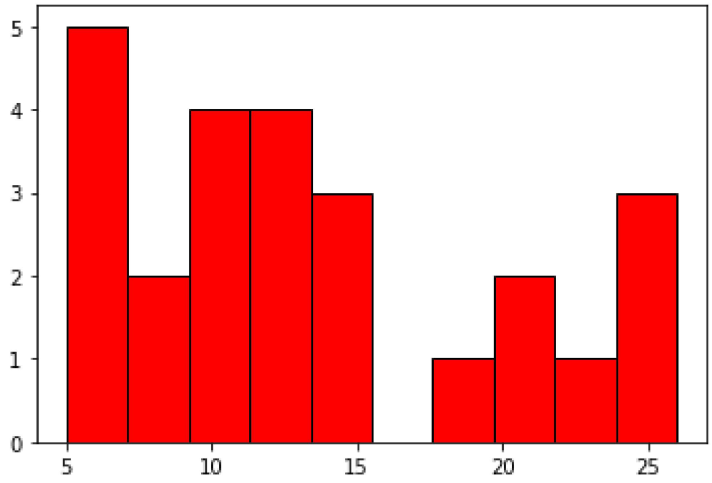


In [20]:

```
plt.hist(data8['Experience'], edgecolor='black', color='red')
```

Out[20]:

```
(array([5., 2., 4., 4., 3., 0., 1., 2., 1., 3.]),
 array([ 5. ,  7.1,  9.2, 11.3, 13.4, 15.5, 17.6, 19.7, 21.8, 23.9, 26.
]),
 <a list of 10 Patch objects>)
```



> Remove the outliers using IQR by recalculating IQR in the updated 'Experience' column and analyze with a box plot.

This question involves removing outliers from the updated 'Experience' column using the interquartile range (IQR) method. By recalculating the IQR and removing values outside a certain range, outliers can be effectively eliminated. The analysis is then visualized using a box plot.

In [21]:

```
Q1c=data7['Experience'].quantile(0.25)
Q3c=data7['Experience'].quantile(0.75)
IQRc = Q3c-Q1c
l=Q1c-1.5*IQRc
h=Q3c+1.5*IQRc
data7['Experience']=data7[(data7['Experience']>l) | (data7['Experience']< h)]
data7
```

Out[21]:

	First Name	Gender	Salary	Team	Age	Experience	New_Salary	Bonus	Se Manager
0	Maria	Female	130590	Finance	NaN	Maria	146075.36220	20000	F
1	Angela	Female	54568	Business Development	27.0	Angela	64675.63064	19000	
2	Allan	Male	125792	Client Services	28.0	Allan	132134.43260	18500	F
3	Rohan	Female	45906	Finance	28.0	Rohan	51230.17788	18000	
4	Douglas	Male	97308	Marketing	28.0	Douglas	104066.04060	17000	
5	Brandon	Male	112807	Human Resources	30.0	Brandon	132539.20040	16000	
6	Diana	Female	132940	Client Services	31.0	Diana	158307.61080	15800	F
7	Frances	NaN	139852	Business Development	34.0	Frances	150374.46450	15500	
8	Matthew	Male	100612	Marketing	34.0	Matthew	114340.50740	15000	F
9	Larry	Male	101004	Client Services	35.0	Larry	102406.94560	14700	
10	Joshua	Male	90816	Client Services	35.0	Joshua	107903.93860	14300	
11	Jerry	Male	72000	Finance	35.0	Jerry	78724.80000	14000	
12	Lois	Female	64714	Legal	35.0	Lois	67906.98876	14000	
13	Dennis	Male	115163	Legal	36.0	Dennis	126823.25380	13000	F
14	John	Male	97950	Client Services	37.0	John	111538.60350	12000	F
15	Thomas	Male	61933	Marketing	38.0	Thomas	68711.56685	11900	
16	Shawn	Male	111737	Human Resources	39.0	Shawn	118903.81120	11500	F
17	Gary	Male	109831	Product	39.0	Gary	116235.24560	11500	F
18	Jeremy	Male	90370	Human Resources	42.0	Jeremy	97029.36530	11000	F
19	Kimberly	Female	41426	Finance	44.0	Kimberly	44512.23700	11000	
20	Louise	Female	63241	Business Development	45.0	Louise	72810.62812	10800	
21	Donna	Female	81014	Product	49.0	Donna	82548.40516	10600	F
22	Ruby	Female	65476	Product	54.0	Ruby	72031.45712	10400	
23	Lillian	Female	59414	Product	55.0	Lillian	60160.23984	10300	F
24	Lillian	Female	59414	Product	55.0	Lillian	60160.23984	10300	

> Remove the outliers using z-score by recalculating z-score in the updated 'Experience' column and analyze it with a box plot.

This question requirres removing outliers from the updated 'Experience' column using the z-score method. By recalculating the z-scores and removing values exceed a certain threshold, outliers can be identified and removed. The analysis is visualized using a box plot.

In [22]:

```
df_zscore = (data7['Age'] - data7['Age'].mean())/data7['Age'].std()
print(df_zscore)
```

```
0           NaN
1     -1.297767
2     -1.180234
3     -1.180234
4     -1.180234
5     -0.945166
6     -0.827633
7     -0.475032
8     -0.475032
9     -0.357498
10    -0.357498
11    -0.357498
12    -0.357498
13    -0.239965
14    -0.122431
15    -0.004897
16     0.112636
17     0.112636
18     0.465237
```

> Plot the Heatmap using correlation

This question involves plotting a heatmap to visualize the correlation between different variables in the dataset. Heatmaps provide an effective way to identify and analyze relationships and dependencies between variables.

In [23]:

```
corr = data7.corr()
corr.style.background_gradient(cmap='coolwarm')
```

Out[23]:

	Salary	Age	New_Salary	Bonus	Senior Management
Salary	1	-0.407259	0.98788	0.368744	-0.49921
Age	-0.407259	1	-0.453648	-0.88841	-0.0972426
New_Salary	0.98788	-0.453648	1	0.403568	-0.474023
Bonus	0.368744	-0.88841	0.403568	1	0.0840114
Senior Management	-0.49921	-0.0972426	-0.474023	0.0840114	1

> Drop the last rows added in the dataframe

Finally, this question requires dropping the last two rows that were previously added to the dataframe. This ensures that the dataframe is reverted to its original state or structure.

In [24]:

```
data2.drop([23, 24], inplace=True)
data2
```

6	Diana	Female	132940	Client Services	31.0	9	158307.61080	15800	False
7	Frances	NaN	139852	Business Development	34.0	10	150374.46450	15500	True
8	Matthew	Male	100612	Marketing	34.0	10	114340.50740	15000	False
9	Larry	Male	101004	Client Services	35.0	11	102406.94560	14700	True
10	Joshua	Male	90816	Client Services	35.0	11	107903.93860	14300	True
11	Jerry	Male	72000	Finance	35.0	12	78724.80000	14000	True
12	Lois	Female	64714	Legal	35.0	12	67906.98876	14000	True
13	Dennis	Male	115163	Legal	36.0	13	126823.25380	13000	False
14	John	Male	97950	Client Services	37.0	13	111538.60350	12000	False
15	Thomas	Male	61933	Marketing	38.0	14	68711.56685	11900	True
16	Shawn	Male	111737	Human	39.0	15	118903.81120	11500	False

In Conclusion Exploratory Data Analysis (EDA) Plays a crucial role in the data analysis process by providing valuable insights, identifying patterns, and uncovering relationships within the dataset. Through various techniques such as data cleaning, visualization, and statistical calculations EDA helps us understand the characteristics and structure of the data, detect outliers or missing values and formulate hypotheses for further analysis.

By conducting EDA, we can make informed decisions about data preprocessing, variable selection, and modeling approaches. It allows us to gain a deeper understanding of the data, identify potential data quality issues, and derive meaningful insights. EDA Serves as a foundation for more advanced analysis techniques, such as predictive modeling, hypothesis testing, and machine learning.

Contact Information:

For any inquiries or further discussions related to this exploratory data analysis notebook, please feel free to reach out to me. I welcome the opportunity to connect and engage in data-related conversations.

Email: info@rubangino.in (https://info@rubangino.in/)

LinkedIn: https://www.linkedin.com/in/ruban-gino-singh/ (https://www.linkedin.com/in/ruban-gino-singh/)

Twitter: https://twitter.com/Rubangino (https://twitter.com/Rubangino)

Github: https://github.com/Ruban2205 (https://github.com/Ruban2205)

Let's connect and Explore the fascinating world of data analysis together!