File Edit View Run Kernel Git Tabs Settings Help

Launcher | notebook.ipynb

Code | git | Python 3 (ipykernel)

[ ]:
```
# Importing Libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
# Library for Missing Value for Dataset
import missingno as msno
```

[ ]:
```
# Load your dataset into a DataFrame
df = pd.read_csv("Salary_Dataset_with_Extra_Features.csv")
```

[ ]:
```
# Print the number of rows and columns
print("Number of rows and columns:", df.shape)
```

```
Number of rows and columns: (22770, 8)
```

[ ]:
```
# Print out the first five rows
df.head()
```

[5]:

	Rating	Company Name	Job Title	Salary	Salaries Reported	Location	Employment Status	Job Roles
0	3.8	Sasken	Android Developer	400000	3	Bangalore	Full Time	Android
1	4.5	Advanced Millennium Technologies	Android Developer	400000	3	Bangalore	Full Time	Android
2	4.0	Unacademy	Android Developer	1000000	3	Bangalore	Full Time	Android
3	3.8	SnapBizz Cloudtech	Android Developer	300000	3	Bangalore	Full Time	Android
4	4.4	Appoids Tech Solutions	Android Developer	600000	3	Bangalore	Full Time	Android

File Edit View Run Kernel Git Tabs Settings Help

Launcher | notebook.ipynb

Code | git | Python 3 (ipykernel)

[ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 22770 entries, 0 to 22769
Data columns (total 8 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   Rating             22770 non-null  float64
 1   Company Name       22770 non-null  object 
 2   Job Title          22770 non-null  object 
 3   Salary             22770 non-null  int64  
 4   Salaries Reported  22770 non-null  int64  
 5   Location           22770 non-null  object 
 6   Employment Status  22770 non-null  object 
 7   Job Roles          22770 non-null  object 
dtypes: float64(1), int64(2), object(5)
memory usage: 1.4+ MB
```

[ ]:
```
# Finding Exact Count of Missing Values in Each Column
df.isna().sum()
```

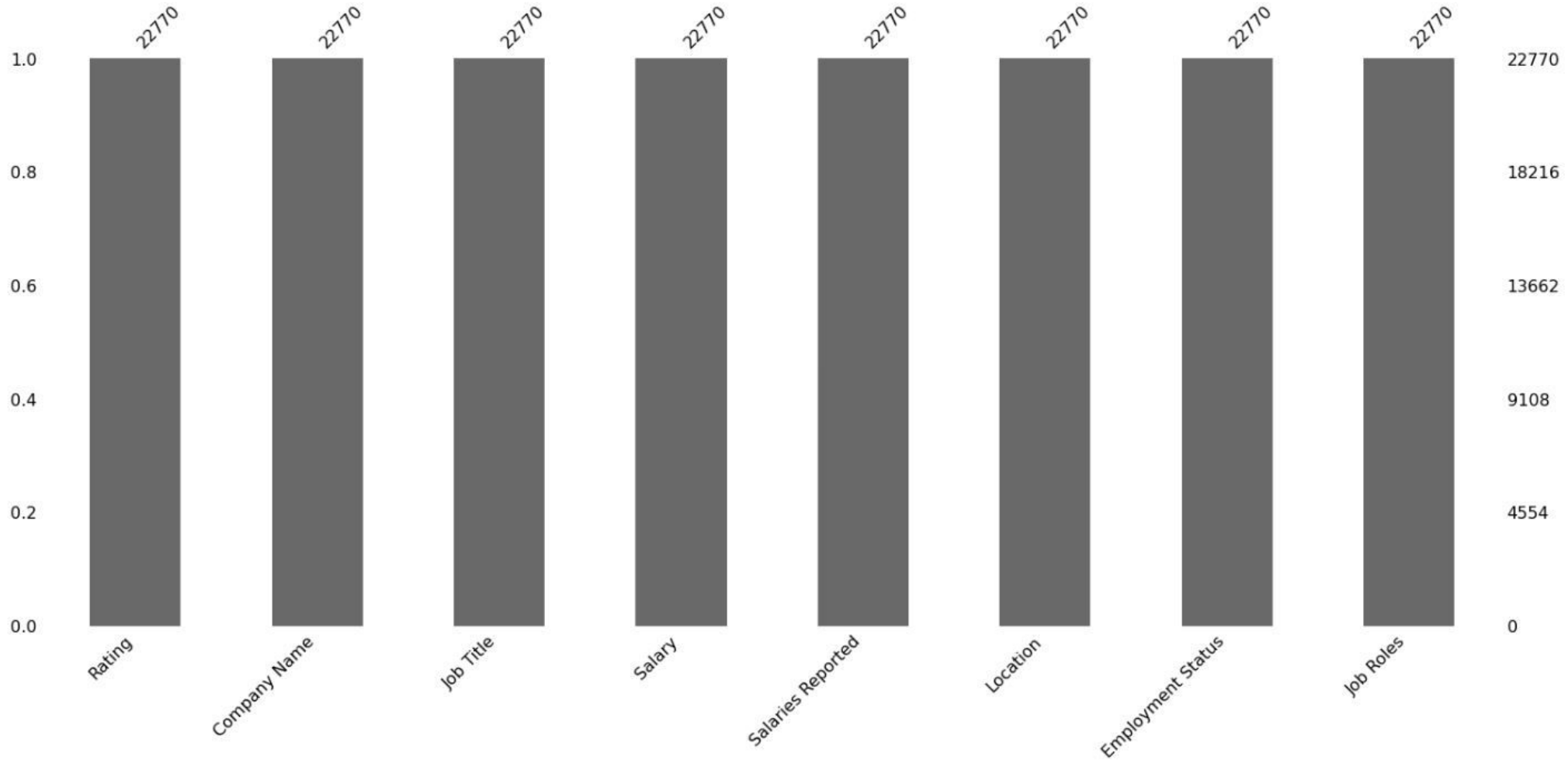
[7]:
```
Rating              0
Company Name        0
Job Title           0
Salary              0
Salaries Reported   0
Location            0
Employment Status   0
Job Roles           0
dtype: int64
```

Edit View Run Kernel Git Tabs Settings Help

Launcher | notebook.ipynb

Code

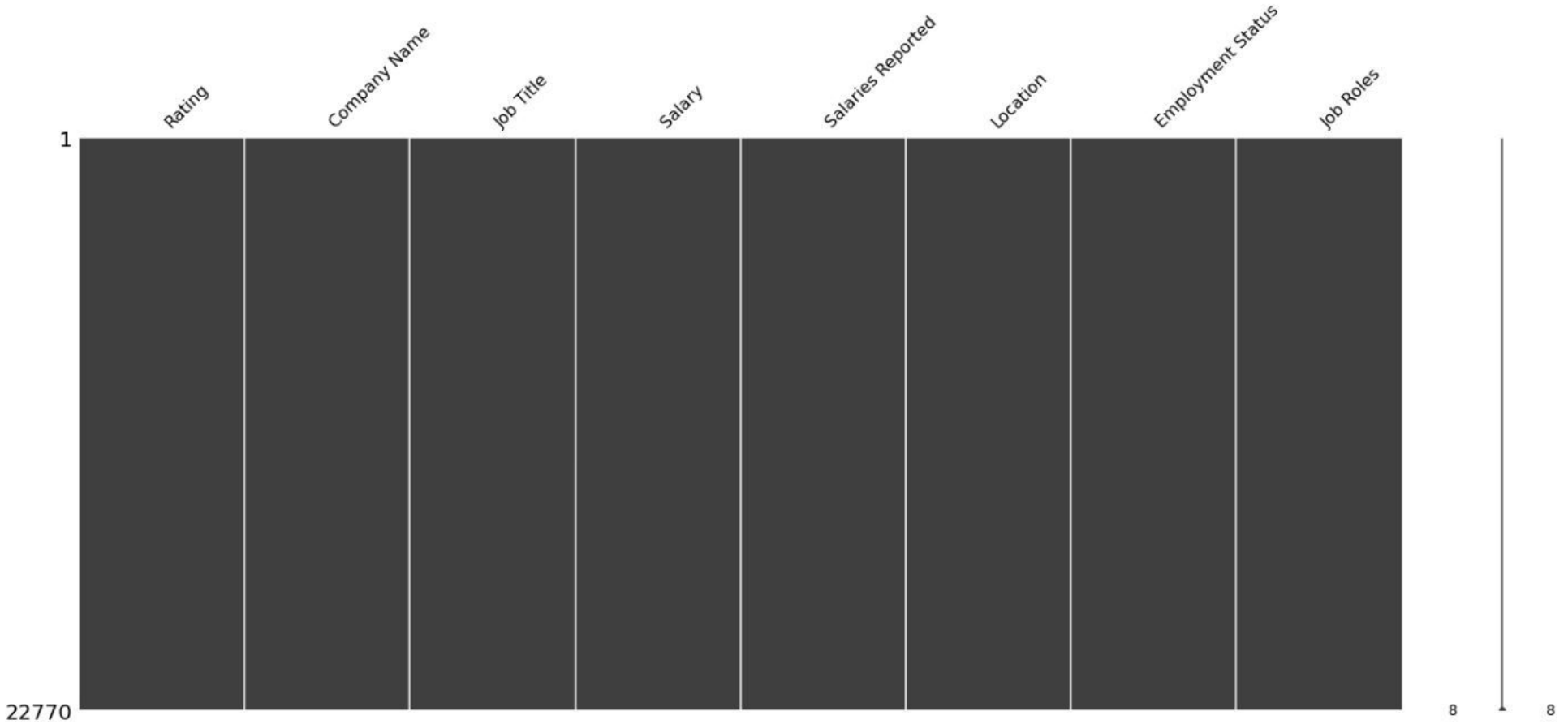
[ ]:
```
# Creating Bar Chart for Missing Values
msno.bar(df)
plt.show()
```



uncher | notebook.ipynb

Code

[ ]:
```
# Creating Bar Chart for Missing Values
msno.matrix(df)
plt.show()
```



File Edit View Run Kernel Git Tabs Settings Help

Launcher | notebook.ipynb

Code | git | Python 3 (ipykernel)

[ ]:
```
# Descriptive statistics for each numeric column , No Missing values
df.describe()
```

[17]:

	Rating	Salary	Salaries Reported
count	22770.000000	2.277000e+04	22770.000000
mean	3.918213	6.953872e+05	1.855775
std	0.519675	8.843990e+05	6.823668
min	1.000000	2.112000e+03	1.000000
25%	3.700000	3.000000e+05	1.000000
50%	3.900000	5.000000e+05	1.000000
75%	4.200000	9.000000e+05	1.000000
max	5.000000	9.000000e+07	361.000000

[ ]:
```
# Print out the unique values of a column:
df['Company Name'].unique()
```

[22]:
```
array(['Sasken', 'Advanced Millennium Technologies', 'Unacademy', ...,
       'Unicon Systems', 'Expert Solutions', 'Nextgen Innovation Labs'],
      dtype=object)
```

File Edit View Run Kernel Git Tabs Settings Help

Launcher | notebook.ipynb

Code | git | Python 3 (ipykernel)

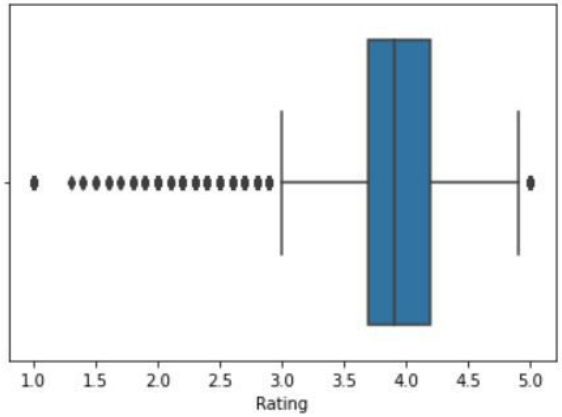
[ ]:
```
# Print out the unique values of a column:
df['Company Name'].unique()
```

[22]:
```
array(['Sasken', 'Advanced Millennium Technologies', 'Unacademy', ...,
       'Unicon Systems', 'Expert Solutions', 'Nextgen Innovation Labs'],
      dtype=object)
```

[ ]:
```
#  Print out the number of rows for each unique value:
df['Company Name'].value_counts(dropna= True)
```

[25]:
```
Tata Consultancy Services         271
Amazon                            184
Infosys                           169
Accenture                         150
Cognizant Technology Solutions    144
                                 ...
Arizona Tile                        1
Highbrow Diligence Services         1
Praveen kumar R                     1
Banao                               1
Emvento Technologies                1
Name: Company Name, Length: 11261, dtype: int64
```

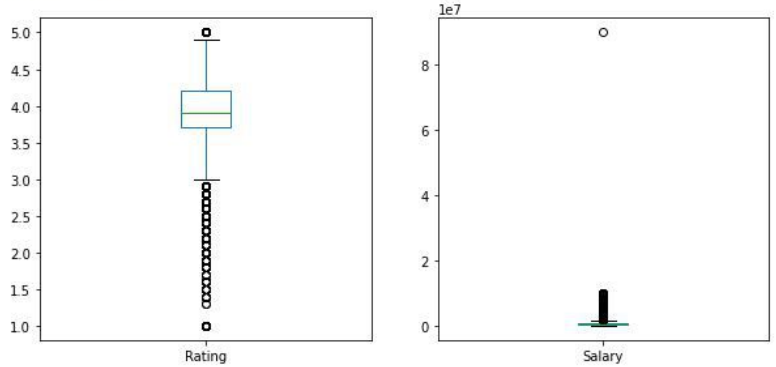
[ ]:
```
# Detecting Outlier in Column Rating through seaborn
sns.boxplot(df['Rating'])
plt.show()
```

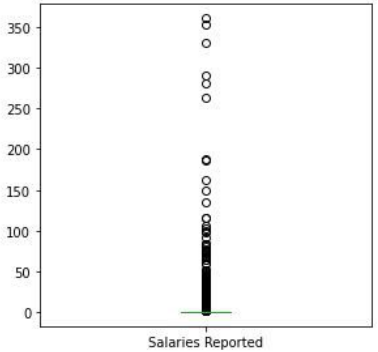


Rating

[ ]:
```
# Detecting Outlier in Columns
df.plot(kind="box", subplots=True, layout=(2,2), figsize=(10,10))
```

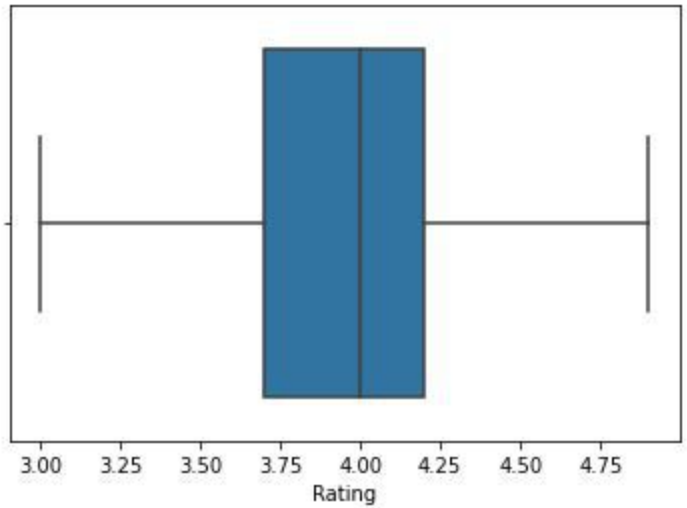
[76]:
```
Rating                 AxesSubplot(0.125,0.536818;0.352273x0.343182)
Salary               AxesSubplot(0.547727,0.536818;0.352273x0.343182)
Salaries Reported          AxesSubplot(0.125,0.125;0.352273x0.343182)
dtype: object
```





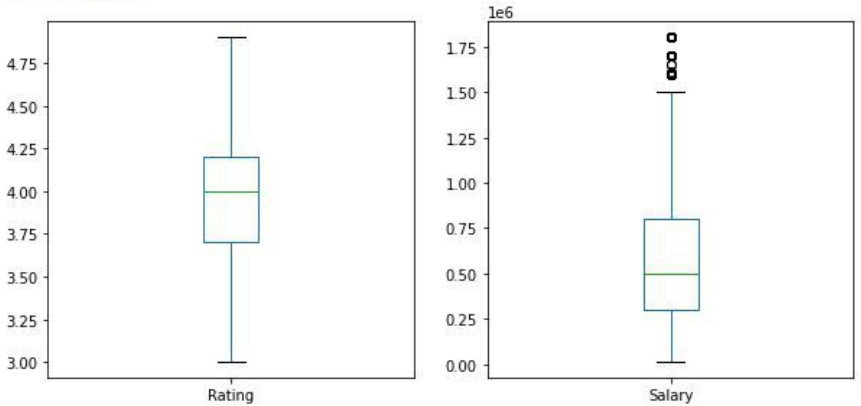
[ ]:
```
# Removing Oulier in a Datset
df = df[~((df < (Q1 - 1.5 * IQR)) |(df > (Q3 + 1.5 * IQR))).any(axis=1)]
```

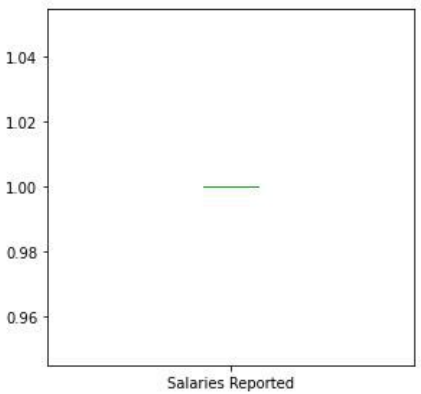
[ ]:
```
sns.boxplot(df['Rating'])
plt.show()
```



[ ]:
```
# Checking if Outliers are Removed
df.plot(kind="box", subplots=True, layout=(2,2), figsize=(10,10))
```

[84]:
```
Rating                AxesSubplot(0.125,0.536818;0.352273x0.343182)
Salary              AxesSubplot(0.547727,0.536818;0.352273x0.343182)
Salaries Reported         AxesSubplot(0.125,0.125;0.352273x0.343182)
dtype: object
```





[ ]:
```
# Checking Statitics of Each Column
df.describe()
```

86]:

	Rating	Salary	Salaries Reported
count	15863.000000	1.586300e+04	15863.0
mean	3.941055	5.944658e+05	1.0
std	0.395131	3.988337e+05	0.0
min	3.000000	1.200000e+04	1.0
25%	3.700000	3.000000e+05	1.0
50%	4.000000	5.000000e+05	1.0
75%	4.200000	8.000000e+05	1.0
max	4.900000	1.800000e+06	1.0

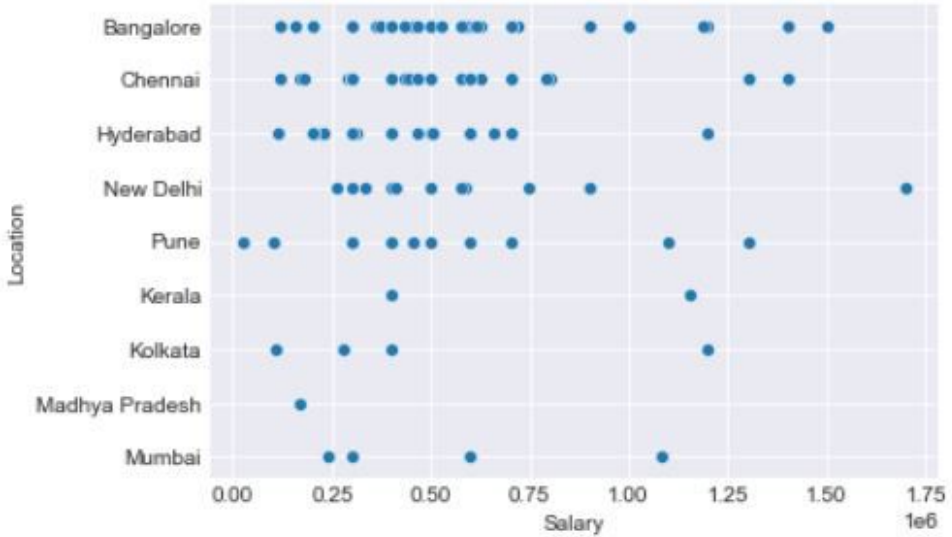
[ ]:
```
# creating Different Comapnies Name dataframe from Company Column
df_TataConsultancyServices = df.loc[df['Company Name'] == 'Tata Consultancy Services']
df_Amazon = df.loc[df['Company Name'] == 'Amazon']
df_Infosys = df.loc[df['Company Name'] == 'Infosys']
df_Accenture = df.loc[df['Company Name'] == 'Accenture']
```

Edit View Run Kernel Git Tabs Settings Help

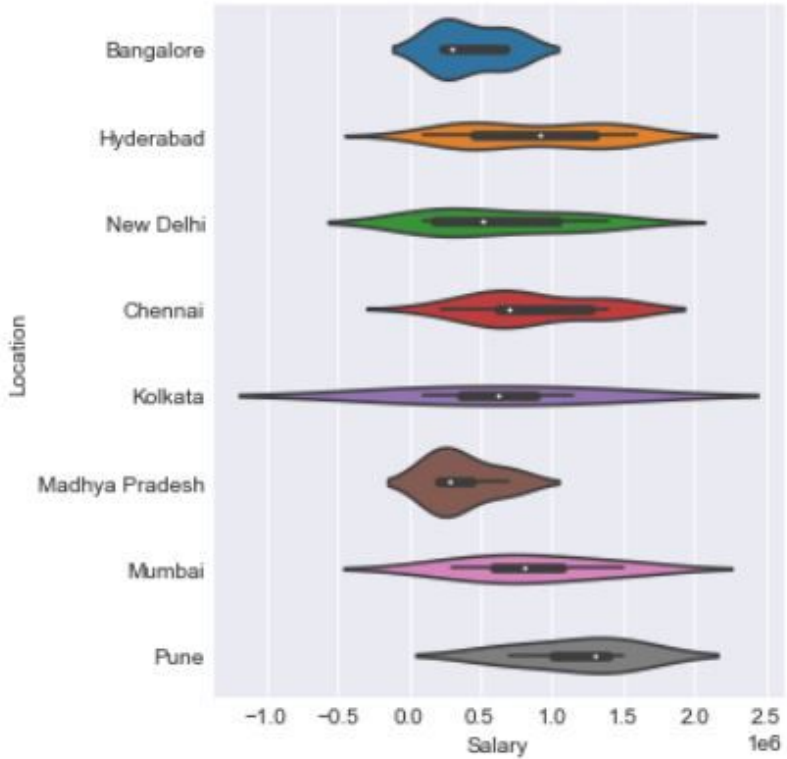
auncher × notebook.ipynb ×

+ ✂ ⧉ 📋 ▶ ■ ↻ ▶▶ Code ⌄ 🕐 git

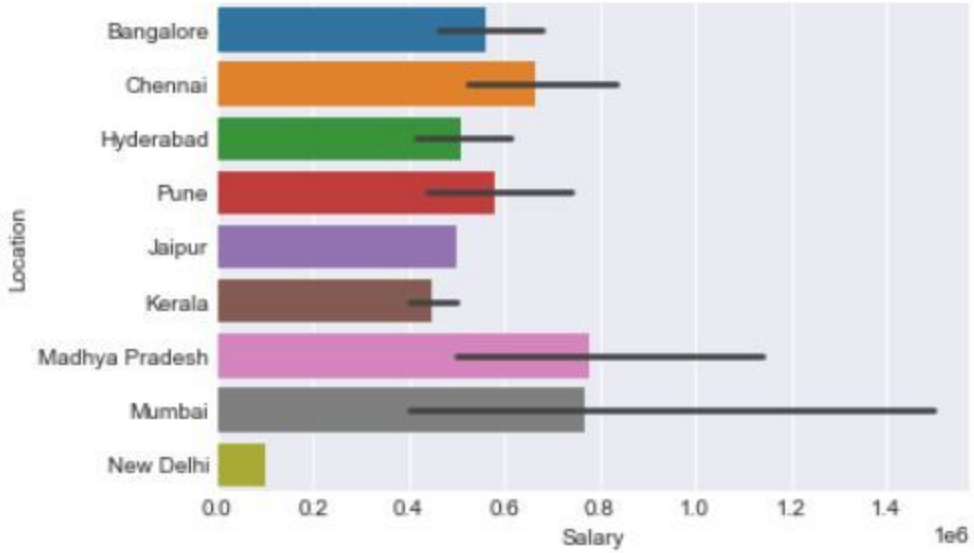
[ ]:
```
# Creating Hypothesis 1  salary within the same company fluctuate between locations
sns.scatterplot(x ='Salary', y ='Location', data=df_TataConsultancyServices)
plt.show()
```



[ ]:
```
# Creating Hypothesis 2  Salary within the same company fluctuate between locations
sns.catplot(x='Salary', y='Location', data=df_Amazon, kind='violin')
plt.show()
```



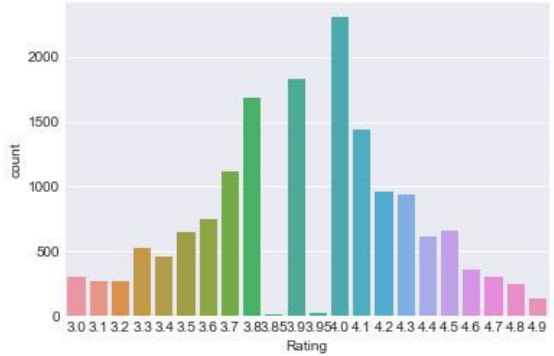
[ ]:
```
# Creating Hypothesis 3  Salary within the same company fluctuate between locations
sns.barplot(x='Salary', y='Location', data=df_Infosys)
plt.show()
```



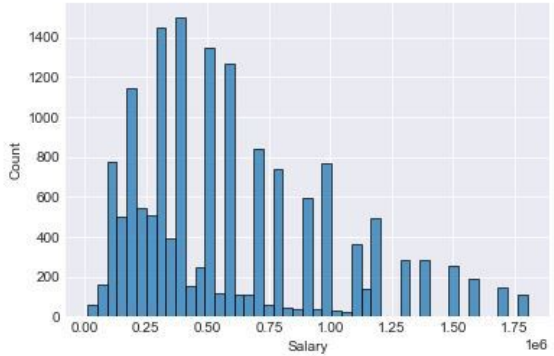
[ ]:
```
# Creating Hypothesis 4  Salary within the same company fluctuate between locations
sns.barplot(x="Salary", y="Location", data=df_Accenture)
plt.show()
```

[ ]:
```
# Creating Univariate ANalysis For 5 Variables
```

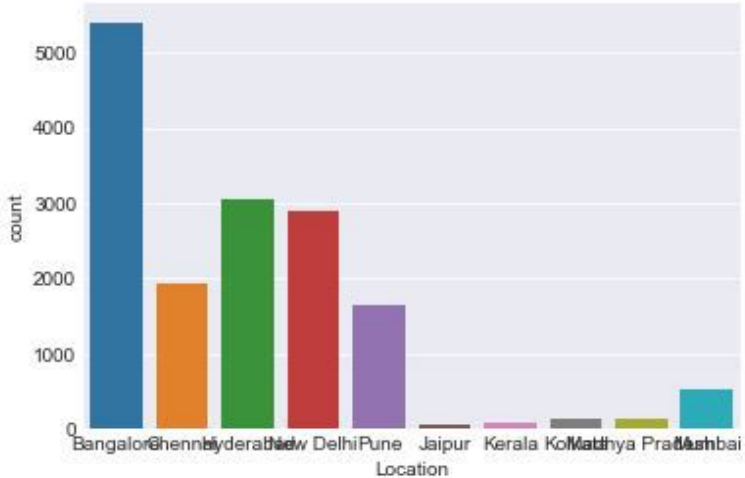
[ ]:
```
# Creating Univariate 1
sns.countplot(df.Rating)
plt.show()
```



[ ]:
```
# Creating Univariate 2
sns.histplot(df.Salary)
plt.show()
```



[ ]:
```
# Creating Univariate 3
sns.countplot(df.Location)
plt.show()
```



[ ]: