Attrition Assignment - Hitik Panchal

In [5]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

Reading the Data

In [4]:

```
gen_data=pd.read_csv('general_data.csv')
gen_data.head()
```

Out[4]:

	Age	Attrition	BusinessTravel	Department	DistanceFromHome	Education	EducationFie
0	51	No	Travel_Rarely	Sales	6	2	Life Scienc
1	31	Yes	Travel_Frequently	Research & Development	10	1	Life Scienc
2	32	No	Travel_Frequently	Research & Development	17	4	Oth
3	38	No	Non-Travel	Research & Development	2	5	Life Scienc
4	32	No	Travel_Rarely	Research & Development	10	1	Medi

5 rows × 24 columns

Features of the Data

In [6]:

```
gen_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4410 entries, 0 to 4409
Data columns (total 24 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Age                      4410 non-null   int64  
 1   Attrition                4410 non-null   object 
 2   BusinessTravel           4410 non-null   object 
 3   Department               4410 non-null   object 
 4   DistanceFromHome         4410 non-null   int64  
 5   Education                4410 non-null   int64  
 6   EducationField           4410 non-null   object 
 7   EmployeeCount            4410 non-null   int64  
 8   EmployeeID               4410 non-null   int64  
 9   Gender                   4410 non-null   object 
 10  JobLevel                 4410 non-null   int64  
 11  JobRole                  4410 non-null   object 
 12  MaritalStatus            4410 non-null   object 
 13  MonthlyIncome            4410 non-null   int64  
 14  NumCompaniesWorked       4391 non-null   float64
 15  Over18                   4410 non-null   object 
 16  PercentSalaryHike        4410 non-null   int64  
 17  StandardHours            4410 non-null   int64  
 18  StockOptionLevel         4410 non-null   int64  
 19  TotalWorkingYears        4401 non-null   float64
 20  TrainingTimesLastYear    4410 non-null   int64  
 21  YearsAtCompany           4410 non-null   int64  
 22  YearsSinceLastPromotion  4410 non-null   int64  
 23  YearsWithCurrManager     4410 non-null   int64  
dtypes: float64(2), int64(14), object(8)
memory usage: 827.0+ KB
```

In [8]:

```
gen_data.shape
```

Out[8]:

```
(4410, 24)
```

In [9]:

```
gen_data.describe()
```

Out[9]:

	Age	DistanceFromHome	Education	EmployeeCount	EmployeeID	JobLe
count	4410.000000	4410.000000	4410.000000	4410.0	4410.000000	4410.000
mean	36.923810	9.192517	2.912925	1.0	2205.500000	2.063
std	9.133301	8.105026	1.023933	0.0	1273.201673	1.106
min	18.000000	1.000000	1.000000	1.0	1.000000	1.000
25%	30.000000	2.000000	2.000000	1.0	1103.250000	1.000
50%	36.000000	7.000000	3.000000	1.0	2205.500000	2.000
75%	43.000000	14.000000	4.000000	1.0	3307.750000	3.000
max	60.000000	29.000000	5.000000	1.0	4410.000000	5.000

In [10]:

```
print(gen_data.columns)
```

```
Index(['Age', 'Attrition', 'BusinessTravel', 'Department', 'DistanceFromHo
me',
       'Education', 'EducationField', 'EmployeeCount', 'EmployeeID', 'Gend
er',
       'JobLevel', 'JobRole', 'MaritalStatus', 'MonthlyIncome',
       'NumCompaniesWorked', 'Over18', 'PercentSalaryHike', 'StandardHour
s',
       'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear',
       'YearsAtCompany', 'YearsSinceLastPromotion', 'YearsWithCurrManage
r'],
      dtype='object')
```

Cleaning the Data

In [12]:

```
gen_data.isnull().any()
```

Out[12]:

```
Age                       False
Attrition                 False
BusinessTravel            False
Department                False
DistanceFromHome          False
Education                 False
EducationField            False
EmployeeCount             False
EmployeeID                False
Gender                    False
JobLevel                  False
JobRole                   False
MaritalStatus             False
MonthlyIncome             False
NumCompaniesWorked         True
Over18                    False
PercentSalaryHike         False
StandardHours             False
StockOptionLevel          False
TotalWorkingYears          True
TrainingTimesLastYear     False
YearsAtCompany            False
YearsSinceLastPromotion   False
YearsWithCurrManager      False
dtype: bool
```

In [13]:

```
gen_data.fillna(0 , inplace=True)
```

In [14]:

```
gen_data.isnull().any()
```

Out[14]:

```
Age                       False
Attrition                 False
BusinessTravel            False
Department                False
DistanceFromHome          False
Education                 False
EducationField            False
EmployeeCount             False
EmployeeID                False
Gender                    False
JobLevel                  False
JobRole                   False
MaritalStatus             False
MonthlyIncome             False
NumCompaniesWorked        False
Over18                    False
PercentSalaryHike         False
StandardHours             False
StockOptionLevel          False
TotalWorkingYears         False
TrainingTimesLastYear     False
YearsAtCompany            False
YearsSinceLastPromotion   False
YearsWithCurrManager      False
dtype: bool
```

In [15]:

```
gen_data.duplicated()
```

Out[15]:

```
0       False
1       False
2       False
3       False
4       False
        ...
4405    False
4406    False
4407    False
4408    False
4409    False
Length: 4410, dtype: bool
```

In [17]:

```
gen_data.drop_duplicates()
```

Out[17]:

	Age	Attrition	BusinessTravel	Department	DistanceFromHome	Education	Educatio
0	51	No	Travel_Rarely	Sales	6	2	Life Sci
1	31	Yes	Travel_Frequently	Research & Development	10	1	Life Sci
2	32	No	Travel_Frequently	Research & Development	17	4	
3	38	No	Non-Travel	Research & Development	2	5	Life Sci
4	32	No	Travel_Rarely	Research & Development	10	1	M
...	...	...	...	...	...	...	
4405	42	No	Travel_Rarely	Research & Development	5	4	M
4406	29	No	Travel_Rarely	Research & Development	2	4	M
4407	25	No	Travel_Rarely	Research & Development	25	2	Life Sci
4408	42	No	Travel_Rarely	Sales	18	2	M
4409	40	No	Travel_Rarely	Research & Development	28	3	M

4410 rows × 24 columns

Working with the Data

In [20]:

```
gen_data1=gen_data[['Age','DistanceFromHome','Education','MonthlyIncome', 'NumCompanies
Worked', 'PercentSalaryHike','TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtComp
any','YearsSinceLastPromotion', 'YearsWithCurrManager']].describe()
gen_data1
```

Out[20]:

	Age	DistanceFromHome	Education	MonthlyIncome	NumCompaniesWorked
count	4410.000000	4410.000000	4410.000000	4410.000000	4410.000000
mean	36.923810	9.192517	2.912925	65029.312925	2.683220
std	9.133301	8.105026	1.023933	47068.888559	2.499737
min	18.000000	1.000000	1.000000	10090.000000	0.000000
25%	30.000000	2.000000	2.000000	29110.000000	1.000000
50%	36.000000	7.000000	3.000000	49190.000000	2.000000
75%	43.000000	14.000000	4.000000	83800.000000	4.000000
max	60.000000	29.000000	5.000000	199990.000000	9.000000

In [21]:

```
gen_data1=gen_data[['Age','DistanceFromHome','Education','MonthlyIncome', 'NumCompanies
Worked', 'PercentSalaryHike','TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtComp
any','YearsSinceLastPromotion', 'YearsWithCurrManager']].median()
gen_data1
```

Out[21]:

```
Age                         36.0
DistanceFromHome             7.0
Education                    3.0
MonthlyIncome            49190.0
NumCompaniesWorked           2.0
PercentSalaryHike           14.0
TotalWorkingYears           10.0
TrainingTimesLastYear        3.0
YearsAtCompany               5.0
YearsSinceLastPromotion      1.0
YearsWithCurrManager         3.0
dtype: float64
```

In [22]:

```
gen_data1=gen_data[['Age','DistanceFromHome','Education','MonthlyIncome', 'NumCompanies
Worked', 'PercentSalaryHike','TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtComp
any','YearsSinceLastPromotion', 'YearsWithCurrManager']].mode()
gen_data1
```

Out[22]:

	Age	DistanceFromHome	Education	MonthlyIncome	NumCompaniesWorked	PercentSalar
0	35	2	3	23420	1.0	

In [23]:

```
gen_data1=gen_data[['Age','DistanceFromHome','Education','MonthlyIncome', 'NumCompanies
Worked', 'PercentSalaryHike','TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtComp
any','YearsSinceLastPromotion', 'YearsWithCurrManager']].var()
gen_data1
```

Out[23]:

```
Age                         8.341719e+01
DistanceFromHome            6.569144e+01
Education                   1.048438e+00
MonthlyIncome               2.215480e+09
NumCompaniesWorked          6.248686e+00
PercentSalaryHike           1.338907e+01
TotalWorkingYears           6.069855e+01
TrainingTimesLastYear       1.661465e+00
YearsAtCompany              3.751728e+01
YearsSinceLastPromotion     1.037935e+01
YearsWithCurrManager        1.272582e+01
dtype: float64
```

In [24]:

```
gen_data1=gen_data[['Age','DistanceFromHome','Education','MonthlyIncome', 'NumCompanies
Worked', 'PercentSalaryHike','TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtComp
any','YearsSinceLastPromotion', 'YearsWithCurrManager']].skew()
gen_data1
```

Out[24]:

```
Age                         0.413005
DistanceFromHome            0.957466
Education                  -0.289484
MonthlyIncome               1.368884
NumCompaniesWorked          1.029836
PercentSalaryHike           0.820569
TotalWorkingYears           1.113489
TrainingTimesLastYear       0.552748
YearsAtCompany              1.763328
YearsSinceLastPromotion     1.982939
YearsWithCurrManager        0.832884
dtype: float64
```

In [25]:

```
gen_data1=gen_data[['Age','DistanceFromHome','Education','MonthlyIncome', 'NumCompanies
Worked', 'PercentSalaryHike','TotalWorkingYears', 'TrainingTimesLastYear', 'YearsAtComp
any','YearsSinceLastPromotion', 'YearsWithCurrManager']].kurt()
gen_data1
```

Out[25]:

```
Age                        -0.405951
DistanceFromHome           -0.227045
Education                  -0.560569
MonthlyIncome               1.000232
NumCompaniesWorked          0.015084
PercentSalaryHike          -0.302638
TotalWorkingYears           0.909606
TrainingTimesLastYear       0.491149
YearsAtCompany              3.923864
YearsSinceLastPromotion     3.601761
YearsWithCurrManager        0.167949
dtype: float64
```

Inference

• All the above variables show positive skewness; while Age & Mean_distance_from_home are leptokurtic and all other variables are platykurtic.

• The Mean_Monthly_Income's IQR is at 54K suggesting company wide attrition across all income bands.

• Mean age forms a near normal distribution with 13 years of IQR

Outliers

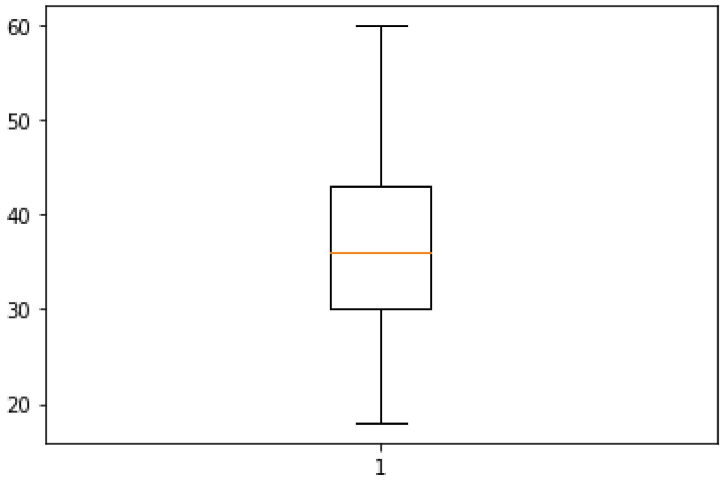
There's no regression found while plotting Age, MonthlyIncome, TotalWorkingYears, YearsAtCompany, etc., on a scatter plot

In [26]:

```
box_plot=gen_data.Age
plt.boxplot(box_plot)
```

Out[26]:

```
{'whiskers': [<matplotlib.lines.Line2D at 0x230ee0a01c8>,
  <matplotlib.lines.Line2D at 0x230ef16b3c8>],
 'caps': [<matplotlib.lines.Line2D at 0x230ef164048>,
  <matplotlib.lines.Line2D at 0x230ef143108>],
 'boxes': [<matplotlib.lines.Line2D at 0x230ee0c7f88>],
 'medians': [<matplotlib.lines.Line2D at 0x230ee00c208>],
 'fliers': [<matplotlib.lines.Line2D at 0x230ee00c0c8>],
 'means': []}
```



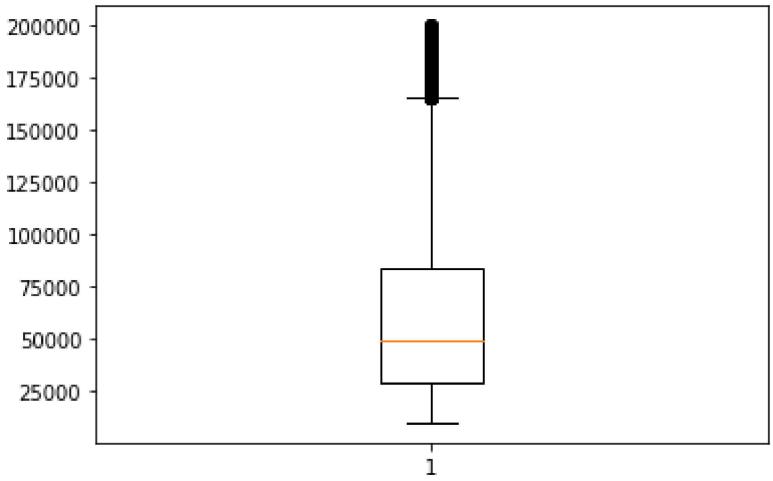
Age is normally distributed without any outliers

In [27]:

```
box_plot=gen_data.MonthlyIncome
plt.boxplot(box_plot)
```

Out[27]:

```
{'whiskers': [<matplotlib.lines.Line2D at 0x230ee765ec8>,
  <matplotlib.lines.Line2D at 0x230ee7b6648>],
 'caps': [<matplotlib.lines.Line2D at 0x230ebd15748>,
  <matplotlib.lines.Line2D at 0x230eefac688>],
 'boxes': [<matplotlib.lines.Line2D at 0x230ece8de88>],
 'medians': [<matplotlib.lines.Line2D at 0x230ef1e3088>],
 'fliers': [<matplotlib.lines.Line2D at 0x230ef18d4c8>],
 'means': []}
```



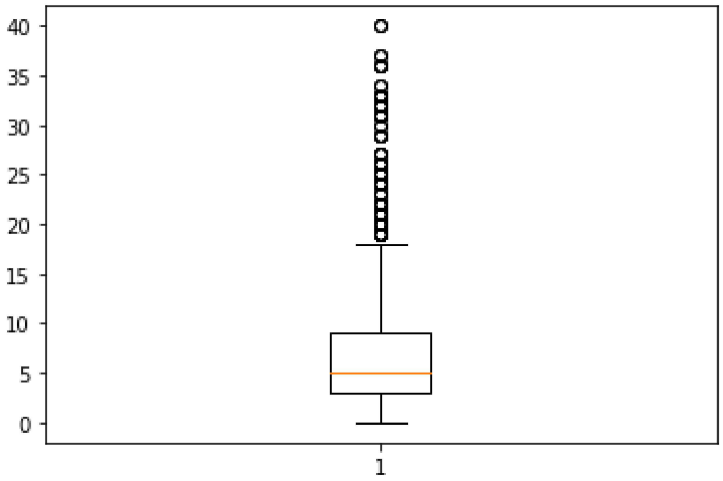
Monthly Income is Right skewed with several outliers

In [28]:

```
box_plot=gen_data.YearsAtCompany
plt.boxplot(box_plot)
```

Out[28]:

```
{'whiskers': [<matplotlib.lines.Line2D at 0x230ee80eb48>,
  <matplotlib.lines.Line2D at 0x230ee82c448>],
 'caps': [<matplotlib.lines.Line2D at 0x230ee848888>,
  <matplotlib.lines.Line2D at 0x230ee857948>],
 'boxes': [<matplotlib.lines.Line2D at 0x230ee808d08>],
 'medians': [<matplotlib.lines.Line2D at 0x230ee87d5c8>],
 'fliers': [<matplotlib.lines.Line2D at 0x230ee8826c8>],
 'means': []}
```



Years at company is also Right Skewed with several outliers observed.

In [ ]: