Skills
Network

Exploratory Data Analysis Lab

Estimated time needed: **30** minutes

In this module you get to work with the cleaned dataset from the previous module.

In this assignment you will perform the task of exploratory data analysis. You will find out the distribution of data, presence of outliers and also determine the correlation between different columns in the dataset.

Objectives

In this lab you will perform the following:

- Identify the distribution of data in the dataset.
- Identify outliers in the dataset.
- Remove outliers from the dataset.
- Identify correlation between features in the dataset.

Hands on Lab

Import the pandas module.

```
In [1]: import pandas as pd
```

```
In [2]: #importing libraries for plotting and arrays
        import numpy as np
        from matplotlib import pyplot as plt
        import seaborn as sns
```

Load the dataset into a dataframe.

```
In [3]: df = pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DA0321EN-SkillsNetwork/LargeData/m2_
```

```
In [4]: df.head()
```

Out[4]:

	Respondent	MainBranch	Hobbyist	OpenSourcer	OpenSource	Employment	Country	Student	EdLevel	UndergradMajor	...	Wel
0	4	I am a developer by profession	No	Never	The quality of OSS and closed source software ...	Employed full-time	United States	No	Bachelor's degree (BA, BS, B.Eng., etc.)	Computer science, computer engineering, or sof...	...	Jus no
1	9	I am a developer by profession	Yes	Once a month or more often	The quality of OSS and closed source software ...	Employed full-time	New Zealand	No	Some college/university study without earning ...	Computer science, computer engineering, or sof...	...	Jus no
2	13	I am a developer by profession	Yes	Less than once a month but more than once per ...	OSS is, on average, of HIGHER quality than pro...	Employed full-time	United States	No	Master's degree (MA, MS, M.Eng., MBA, etc.)	Computer science, computer engineering, or sof...	...	So
3	16	I am a developer by profession	Yes	Never	The quality of OSS and closed source software ...	Employed full-time	United Kingdom	No	Master's degree (MA, MS, M.Eng., MBA, etc.)	NaN	...	Jus no
4	17	I am a developer by profession	Yes	Less than once a month but more than once per ...	The quality of OSS and closed source software ...	Employed full-time	Australia	No	Bachelor's degree (BA, BS, B.Eng., etc.)	Computer science, computer engineering, or sof...	...	Jus no

5 rows × 85 columns

Distribution

The column `ConvertedComp` contains Salary converted to annual USD salaries using the exchange rate on 2019-02-01.

This assumes 12 working months and 50 working weeks.

Determine how the data is distributed

Plot the distribution curve for the column `ConvertedComp`.

In [5]:

```
df['ConvertedComp'].head()
```

Out[5]:

```
0     61000.0
1     95179.0
2     90000.0
3    455352.0
4     65277.0
Name: ConvertedComp, dtype: float64
```

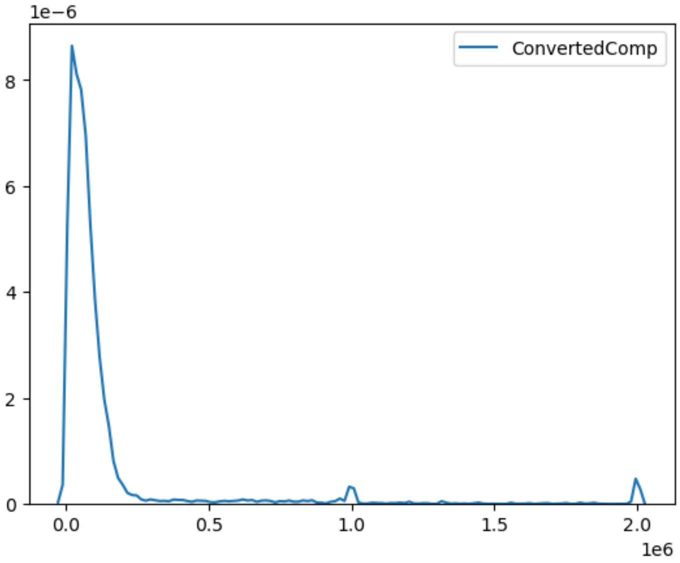
In [6]:

```
X=df['ConvertedComp']
sns.kdeplot(X)

##can also use
#sns.distplot(X,hist=False)
```

Out[6]:

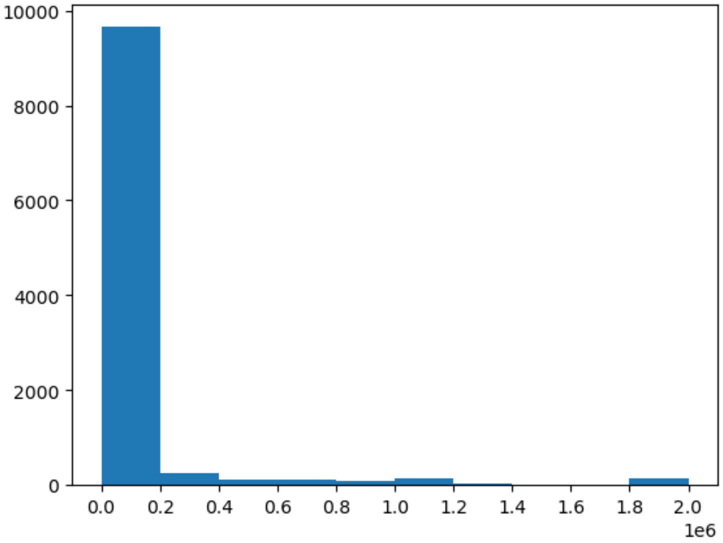
```
<AxesSubplot:>
```



Plot the histogram for the column `ConvertedComp`.

```
In [7]: bins=10
        plt.hist(X,bins)

        # set the ticks to be at the edges of the bins.
        plt.xticks(ticks=np.arange(min(X),max(X)+1,(max(X)-min(X))/bins))
        plt.show()
```



What is the median of the column `ConvertedComp`?

```
In [8]: df['ConvertedComp'].median()
```

```
Out[8]: 57745.0
```

How many responders identified themselves only as a **Man**?

```
In [9]: df['Gender'].value_counts()
```

```
Out[9]: Man                                                            10480
        Woman                                                            731
        Non-binary, genderqueer, or gender non-conforming                 63
        Man;Non-binary, genderqueer, or gender non-conforming             26
        Woman;Non-binary, genderqueer, or gender non-conforming           14
        Woman;Man                                                          9
        Woman;Man;Non-binary, genderqueer, or gender non-conforming        2
        Name: Gender, dtype: int64
```

Find out the median ConvertedComp of responders identified themselves only as a **Woman**?

In [10]:

```
tempdf=df.loc[(df['Gender'] == 'Woman')]
tempdf['ConvertedComp'].median()
```

Out[10]:

```
57708.0
```

Give the five number summary for the column `Age`?

Double click here for hint.

In [11]:

```
df['Age'].describe()
```

Out[11]:

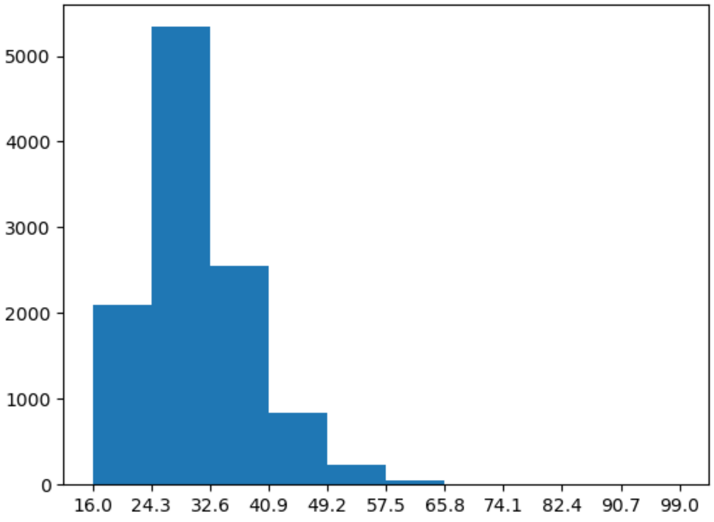
```
count    11111.000000
mean        30.778895
std          7.393686
min         16.000000
25%         25.000000
50%         29.000000
75%         35.000000
max         99.000000
Name: Age, dtype: float64
```

Plot a histogram of the column `Age`.

In [12]:

```
X=df['Age']
bins=10
plt.hist(X,bins)

# set the ticks to be at the edges of the bins.
plt.xticks(ticks=np.arange(min(X),max(X)+1,(max(X)-min(X))/bins))
plt.show()
```

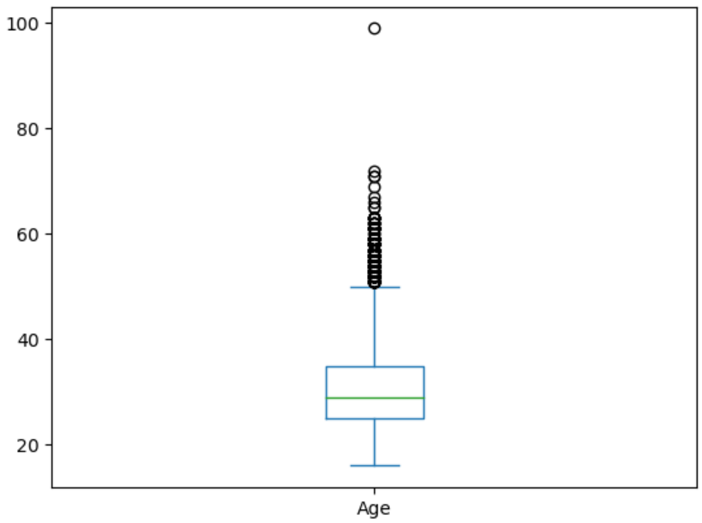


In [16]:

```
#Based on the boxplot of ‘Age’ how many outliers do you see below Q1?
X=df['Age']
X.plot(kind='box')
#Answer: 0
```

Out[16]:

```
<AxesSubplot:>
```



Outliers

Finding outliers

Find out if outliers exist in the column `ConvertedComp` using a box plot?

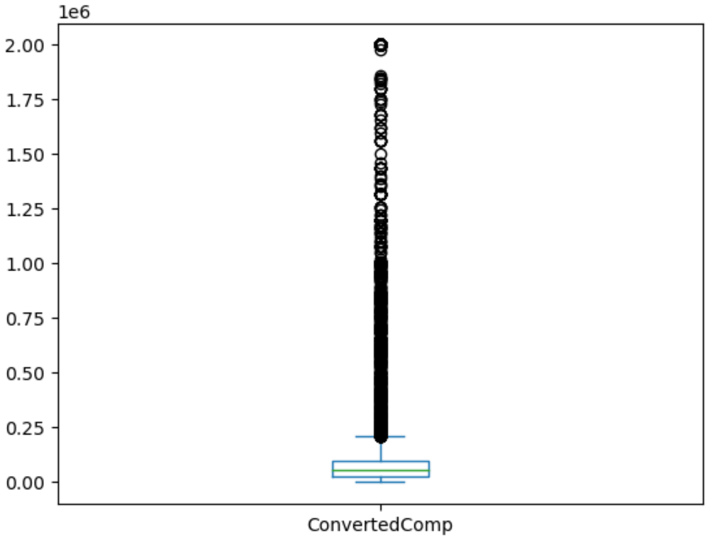
In [14]:

```
X=df['ConvertedComp']

X.plot(kind='box')
```

Out[14]:

```
<AxesSubplot:>
```



Find out the Inter Quartile Range for the column `ConvertedComp`.

In [18]:

```
stats=df['ConvertedComp'].describe() #use for more stats

q1=stats[4] #25%
q2=stats[5] #50% median
q3=stats[6] #75%

#calculate interquartile range
iqr=q3-q1

#print interquartile range
print(iqr)
```

```
73132.0
```

In [19]:

```
#What is the median ConvertedComp before removing outliers?

print(q2)
```

```
57745.0
```

Find out the upper and lower bounds.

In [21]:

```
upper_bound=q3+iqr*1.5
lower_bound=q1-iqr*1.5
print('upper bound:', upper_bound)
print('lower bound:', lower_bound)
```

```
upper bound: 209698.0
lower bound: -82830.0
```

Identify how many outliers are there in the `ConvertedComp` column.

In [23]:

```
def outliers_count(test_column):
    res = 0
    stats=test_column.describe()
    q1=stats[4] #25%
    q2=stats[5] #50% median
    q3=stats[6] #75%
    iqr=q3-q1 #interquartile range
    upper_bound=q3+iqr*1.5
    lower_bound=q1-iqr*1.5

    for i in test_column:
        if (i >upper_bound)or (i<lower_bound):
            res += 1
    print ("The number of outliers: " + str(res))
    return(res)

outliers_count(df['ConvertedComp'])
```

```
The number of outliers: 879
```

Out[23]:

```
879
```

Create a new dataframe by removing the outliers from the `ConvertedComp` column.

In [26]:

```
new_df = df[(df['ConvertedComp'] >= lower_bound) & (df['ConvertedComp'] <= upper_bound)]
```

In [27]:

```
#What is the median ConvertedComp after removing outliers?
new_df['ConvertedComp'].median()
```

Out[27]:

```
52704.0
```

In [28]:

```
#What is the mean ConvertedComp after removing outliers?
new_df['ConvertedComp'].mean()
```

Out[28]:

```
59883.20838915799
```

Correlation

Finding correlation

Find the correlation between `Age` and all other numerical columns.

In [29]:

```
new_df.corr()
```

Out[29]:

	Respondent	CompTotal	ConvertedComp	WorkWeekHrs	CodeRevHrs	Age
Respondent	1.000000	-0.019354	0.010878	-0.016221	0.005293	0.002180
CompTotal	-0.019354	1.000000	-0.063561	0.004910	0.017007	0.006337
ConvertedComp	0.010878	-0.063561	1.000000	0.034351	-0.088934	0.401821
WorkWeekHrs	-0.016221	0.004910	0.034351	1.000000	0.038935	0.032032
CodeRevHrs	0.005293	0.017007	-0.088934	0.038935	1.000000	-0.012878
Age	0.002180	0.006337	0.401821	0.032032	-0.012878	1.000000

Authors

Ramesh Sannareddy

Other Contributors

Rav Ahuja

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2020-10-17	0.1	Ramesh Sannareddy	Created initial version of the lab