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 [6]: import pandas as pd
        import matplotlib.pyplot as plt
```

Load the dataset into a dataframe.

```
In [7]: df = pd.read_csv("https://cf-courses-data.s3.us.cloud-object-storage.appdomain.clou
```

Distribution

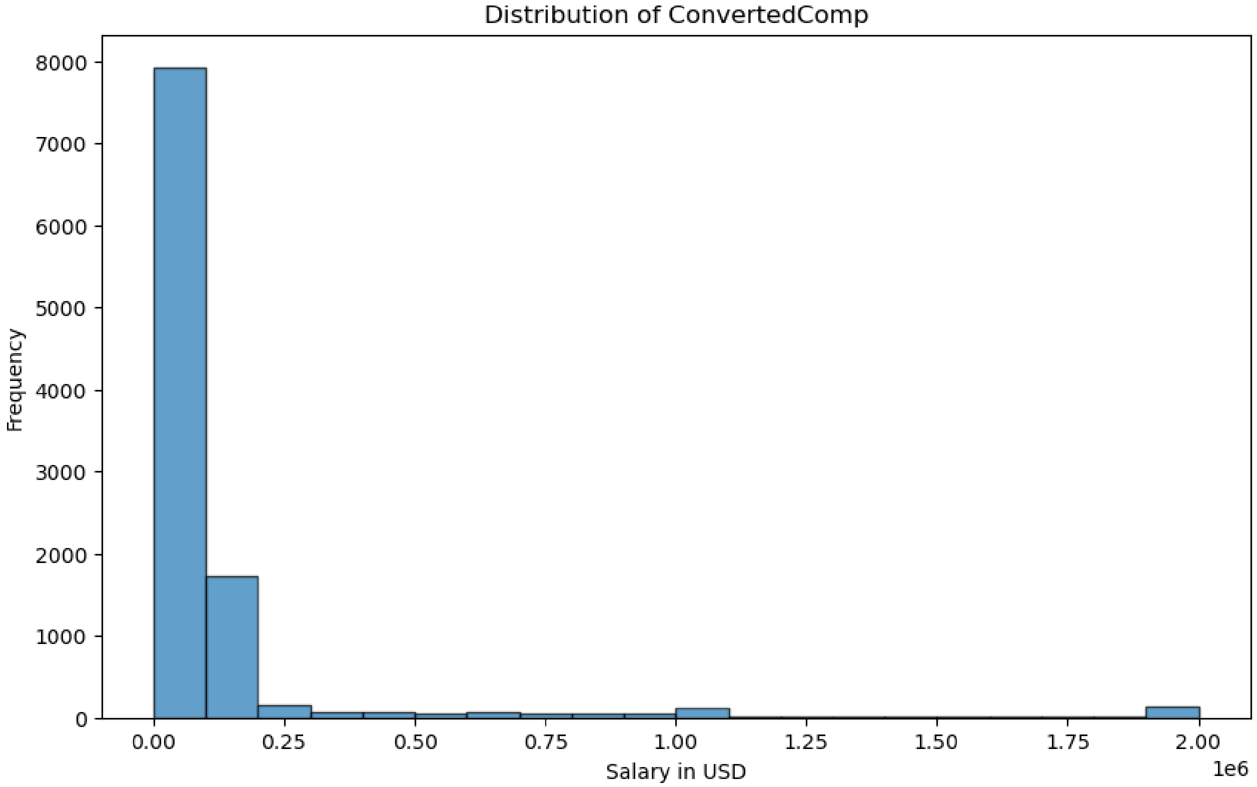
Determine how the data is distributed

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.

Plot the distribution curve for the column `ConvertedComp`.

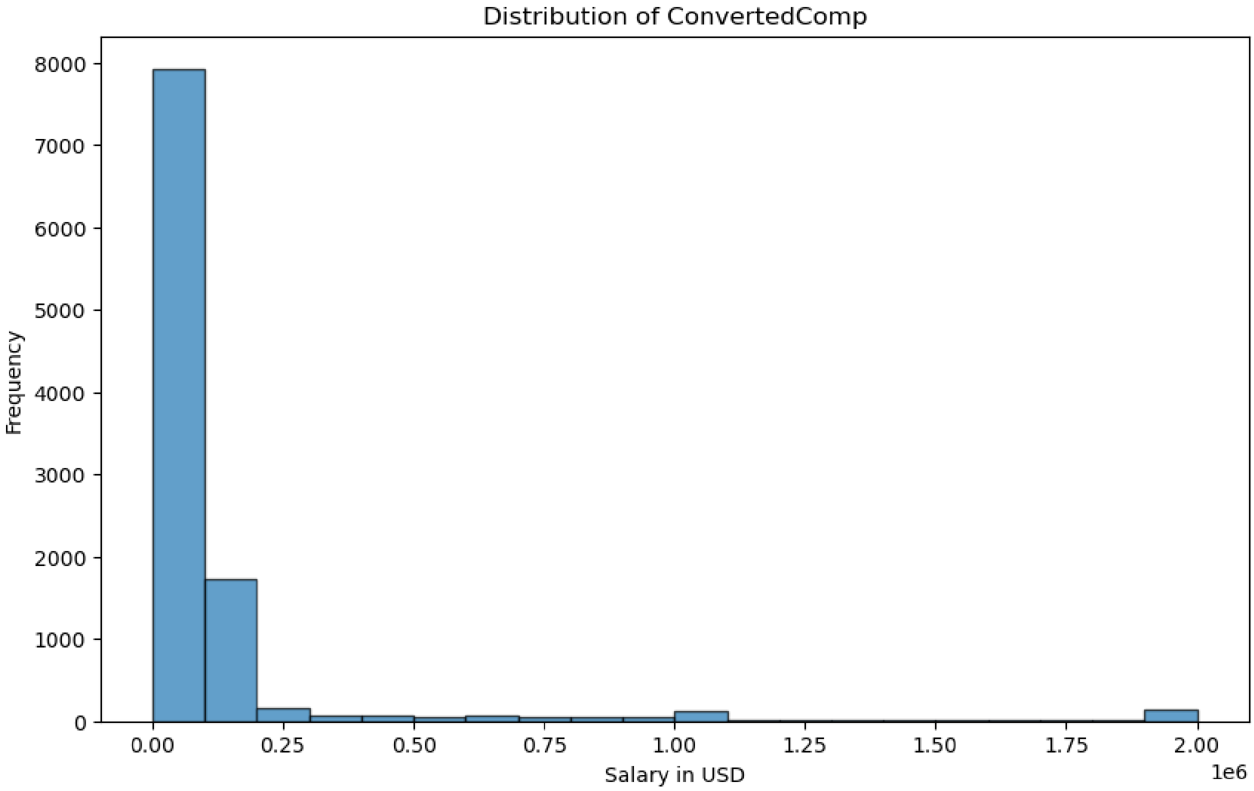
In [12]:
```
# your code goes here
df_con = df['ConvertedComp']

# Create figure and axes
fig, ax = plt.subplots(figsize=(10, 6))

# Plot the histogram
ax.hist(df_con, bins=20, edgecolor='k', alpha=0.7)  # Adjust the number of bins as

# Set labels and title
ax.set_title('Distribution of ConvertedComp')
ax.set_xlabel('Salary in USD')
ax.set_ylabel('Frequency')

# Display the plot
plt.show()
```



Plot the histogram for the column `ConvertedComp`.

In [13]:
```
# your code goes here
df_con = df['ConvertedComp']

# Create figure and axes
```

```
fig, ax = plt.subplots(figsize=(10, 6))

# Plot the histogram
ax.hist(df_con, bins=20, edgecolor='k', alpha=0.7)  # Adjust the number of bins as

# Set labels and title
ax.set_title('Distribution of ConvertedComp')
ax.set_xlabel('Salary in USD')
ax.set_ylabel('Frequency')

# Display the plot
plt.show()
```



What is the median of the column `ConvertedComp` ?

In [15]:
```
# your code goes here
df['ConvertedComp'].median()
```

Out[15]:
```
57745.0
```

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

In [21]:
```
# your code goes here
# Filter the DataFrame to include only rows where 'Gender' is 'Man'

#EASIER CODE USING PANDAS
#man_responses = df[df['Gender'] == 'Man']

# Get the count of responders who identified themselves as 'Man'
#count_man_responses = man_responses.shape[0]
```

```
#print("Number of responders who identified as 'Man':", count_man_responses)


# Initialize a variable to count 'Man' responses
man_count = 0

# Iterate through the 'Gender' column
for gender in df['Gender']:
    if gender == 'Man':
        man_count += 1

print("Number of responders who identified as 'Man':", man_count)
```

```
Number of responders who identified as 'Man': 10480
```

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

In [23]:

```
# your code goes here

woman_count=0

for gen in df['Gender']:
    if gen=='Woman':
        woman_count +=1
print("Number of responders who identified as 'Wman':", woman_count)
```

```
Number of responders who identified as 'Wman': 731
```

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

Double click here for hint.

In [24]:

```
# your code goes here
df_age=df['Age']
df_age.head()
```

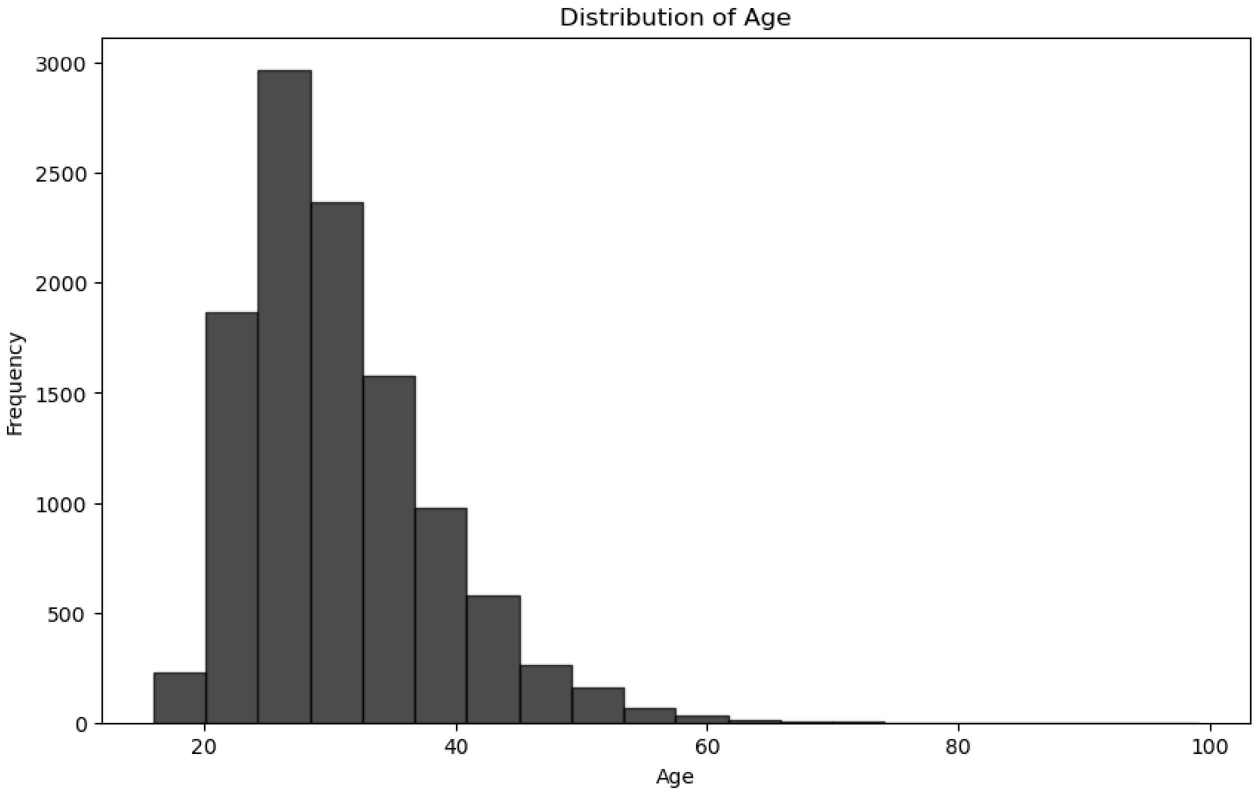
Out[24]:

```
0    22.0
1    23.0
2    28.0
3    26.0
4    29.0
Name: Age, dtype: float64
```

Plot a histogram of the column `Age`.

In [31]:

```
# your code goes here
df_age=df['Age']

# Create figure and axes
fig, ax = plt.subplots(figsize=(10, 6))

# Plot the histogram
ax.hist(df_age, bins=20,color='black', edgecolor='k', alpha=0.7)  # Adjust the num

# Set labels and title
ax.set_title('Distribution of Age')
ax.set_xlabel('Age')
ax.set_ylabel('Frequency')

# Display the plot
plt.show()
```



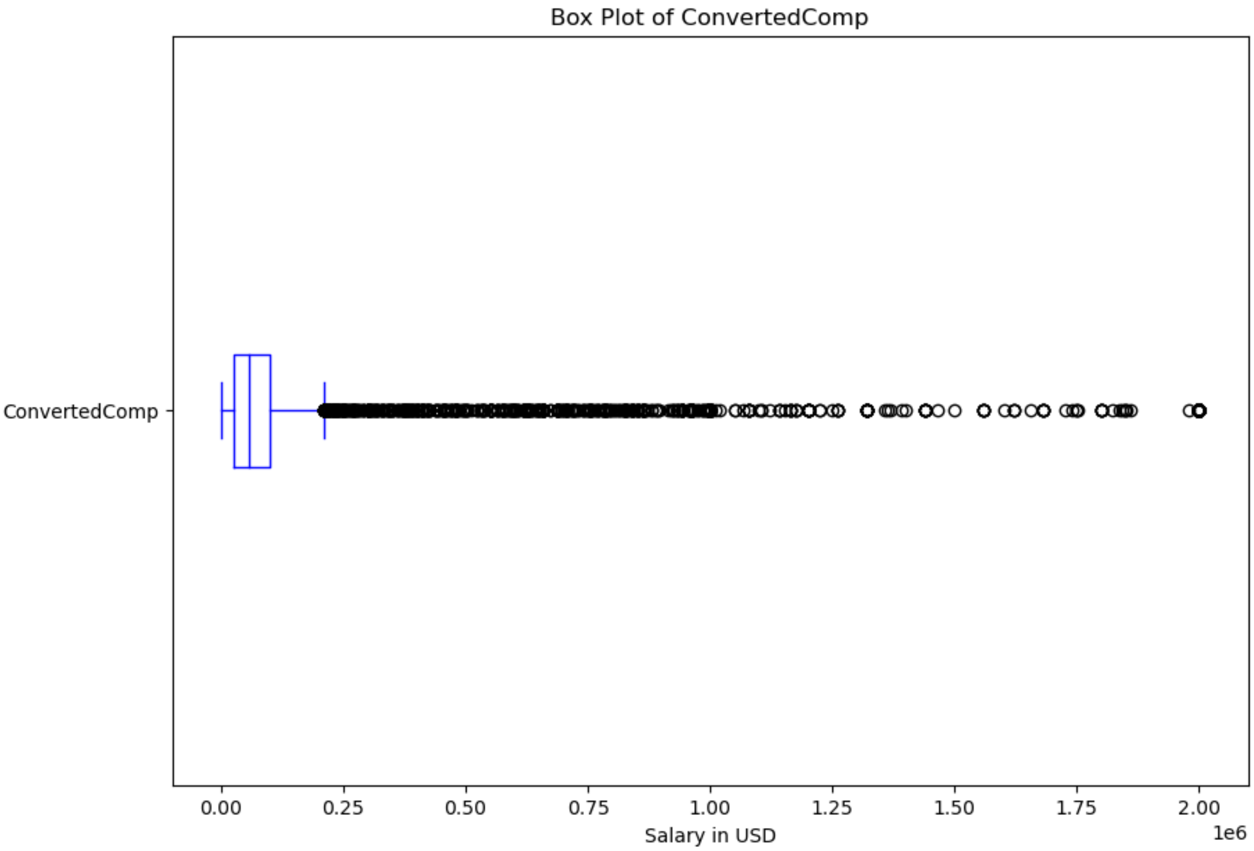
Outliers

Finding outliers

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

In [60]:

```
# your code goes here
import matplotlib.pyplot as plt

plt.figure(figsize=(8, 6))
df_con.plot(kind='box', figsize=(10, 7), color='blue', vert=False)
plt.title('Box Plot of ConvertedComp')
plt.xlabel('Salary in USD')
plt.show()
```



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

```
In [62]: # Calculate the first quartile (Q1)
q1 = df['ConvertedComp'].quantile(0.25)

# Calculate the third quartile (Q3)
q3 = df['ConvertedComp'].quantile(0.75)

# Calculate the Interquartile Range (IQR)
iqr = q3 - q1

print("Interquartile Range (IQR) for ConvertedComp:", iqr)
```

```
Interquartile Range (IQR) for ConvertedComp: 73132.0
```

Find out the upper and lower bounds.

```
In [64]: # your code goes here
# Calculate the lower bound
lower_bound = q1 - 1.5 * iqr

# Calculate the upper bound
upper_bound = q3 + 1.5 * iqr

print("Lower Bound:", lower_bound)
print("Upper Bound:", upper_bound)
```

```
Lower Bound: -82830.0
Upper Bound: 209698.0
```

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

In [67]:

```
# your code goes here
# Identify outliers
outliers = df[(df['ConvertedComp'] < lower_bound) | (df['ConvertedComp'] > upper_b

# Count the number of outliers
num_outliers = outliers.shape[0]

print("Number of outliers in ConvertedComp:", num_outliers)
```

```
Number of outliers in ConvertedComp: 879
```

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

In [69]:

```
# your code goes here
df_remove_outliers = df[(df['ConvertedComp'] >= lower_bound) & (df['ConvertedComp'
df_remove_outliers
```

Out[69]:

	Respondent	MainBranch	Hobbyist	OpenSourcer	OpenSource	Employment	Country	Stu
0	4	I am a developer by profession	No	Never	The quality of OSS and closed source software ...	Employed full-time	United States	
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	
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	
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	
5	19	I am a developer by profession	Yes	Never	The quality of OSS and closed source software ...	Employed full-time	Brazil	
...	...	...	...	...	...	...	...	
11392	25134	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	Ecuador	
11393	25136	I am a developer by profession	Yes	Never	OSS is, on average, of HIGHER quality than pro...	Employed full-time	United States	
11394	25137	I am a developer by profession	Yes	Never	The quality of OSS and closed source software ...	Employed full-time	Poland	
11395	25138	I am a developer by profession	Yes	Less than once per year	The quality of OSS and closed source software ...	Employed full-time	United States	
11396	25141	I am a developer by profession	Yes	Less than once a month but more than once per ...	OSS is, on average, of LOWER quality than prop...	Employed full-time	Switzerland	

9703 rows × 85 columns

Correlation

Finding correlation

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

In [73]:

```
# your code goes here
df.corr()['Age']
```

Out[73]:

```
Respondent       0.004041
CompTotal        0.006970
ConvertedComp    0.105386
WorkWeekHrs      0.036518
CodeRevHrs      -0.020469
Age              1.000000
Name: Age, dtype: float64
```

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