Data collection and Data Visualization

Data collection

Data collection is the process of gathering and measuring information on targeted variables in an established system, which then enables one to answer relevant questions and evaluate outcomes. Data collection is a research component in all study fields, including physical and social sciences, humanities, and business. While methods vary by discipline, the emphasis on ensuring accurate and honest collection remains the same

Data collection from Datasets from csv files and Excel files

Data Visualization

Data visualization is an interdisciplinary field that deals with the graphic representation of data. It is a particularly efficient way of communicating when the data is numerous as for example a time series.

Creating a dataframe

In [2]:
```
my_dict = { 'name' : ["a","b","c","d","e","f","g"],'age' :[20,27,35,55,18,21,35],'designation': ["VP","CEO","CFO","VP","VP","CEO","MD"]}
import pandas as pd
import numpy as np
df=pd.DataFrame(my_dict)
df
```

Out[2]:

	name	age	designation
0	a	20	VP
1	b	27	CEO
2	c	35	CFO
3	d	55	VP
4	e	18	VP
5	f	21	CEO
6	g	35	MD

Saving Dataframe to CSV file

In [4]:
```
df.to_csv('csv_fds')
df
```

Out[4]:

	name	age	designation
0	a	20	VP
1	b	27	CEO
2	c	35	CFO
3	d	55	VP
4	e	18	VP
5	f	21	CEO
6	g	35	MD

In [5]:
```
df.to_csv('csv_fds',index=False)
df_csv=pd.read_csv('csv_fds')
df_csv
```

Out[5]:

	name	age	designation
0	a	20	VP
1	b	27	CEO
2	c	35	CFO
3	d	55	VP
4	e	18	VP
5	f	21	CEO
6	g	35	MD

In [9]:
```
import pandas as pd
Location = "F:/MSC 1/FDS/notes/student-mat.csv"
df = pd.read_csv(Location, header=None)
df.head()
```

Out[9]:

	0	1	2	3	4	5	6	7	8	9	...	23	24	25	26	27	28	29	30	31	32
0	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	famrel	freetime	goout	Dalc	Walc	health	absences	G1	G2	G3
1	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	4	3	4	1	1	3	6	5	6	6
2	GP	F	17	U	GT3	T	1	1	at_home	other	...	5	3	3	1	1	3	4	5	5	6
3	GP	F	15	U	LE3	T	1	1	at_home	other	...	4	3	2	2	3	3	10	7	8	10
4	GP	F	15	U	GT3	T	4	2	health	services	...	3	2	2	1	1	5	2	15	14	15

5 rows × 33 columns

Creating multiple lists using dataframe

In [11]:
```
import pandas as pd
names = ['Anjali','Seema','Ganesh','Govind','Samay']
grades = [74,84,75,88,90]
bsdegrees = [1,0,1,1,0]
msdegrees = [2,1,2,1,1]
phddegrees = [0,1,0,1,0]
Degrees = zip(names,grades,bsdegrees,msdegrees,phddegrees)
columns = ['Names','Grades','BS','MS','PhD']
df = pd.DataFrame(data = Degrees, columns=columns)
df
```

Out[11]:

	Names	Grades	BS	MS	PhD
0	Anjali	74	1	2	0
1	Seema	84	0	1	1
2	Ganesh	75	1	2	0
3	Govind	88	1	1	1
4	Samay	90	0	1	0

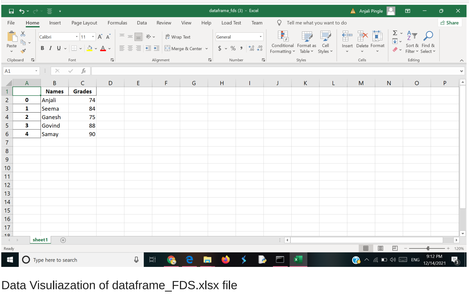
Loading data from Excel files

In [13]:
```
import pandas as pd
Location = 'F:/MSC 1/FDS/notes/share_fds.xlsx'
df = pd.read_excel(Location)
df.columns = ['Roll no','firstname','lastname','gender','age','exer','hrs','grd','addr']
df.head()
```

Out[13]:

	Roll no	firstname	lastname	gender	age	exer	hrs	grd	addr
0	1	Anjali	Kadam	F	22	3	10	75	Anjurphata
1	2	Seema	Kadam	F	21	2	5	80	Thane
2	3	Ganesh	Kadam	M	20	1	8	81	Bhiwandi
3	4	Govind	Kadam	M	23	2	9	85	Kalyan
4	5	Samay	Pingle	M	25	3	5	90	Ghatkopar

In [46]:
```
import pandas as pd
names = ['Anjali','Seema','Ganesh','Govind','Samay']
grades = [74,84,75,88,90]
Gradelist = zip(names,grades)
df = pd.DataFrame(data = Gradelist,columns=['Names','Grades'])
writer = pd.ExcelWriter('dataframe_fds.xlsx', engine='xlsxwriter')
df.to_excel(writer, sheet_name='sheet1')
writer.save()
```



Data Visuliazation of dataframe_FDS.xlsx file

Showing the Result of 'dataframe_FDS.xlsx' file in Bar Chart,Line Chart,Pie Chart Format.

Scatter plot

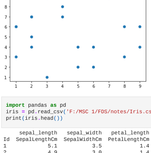
Scatter plots are used to visualize the relationship between two (or sometimes three) variables in a data set.

In [39]:
```
import matplotlib.pyplot as plt
# create a figure and axis
fig, ax = plt.subplots()


x =[2, 4, 6, 6, 9, 2, 7, 2, 6, 1, 8, 4, 5, 9, 1, 2, 3, 7, 5, 8, 1, 3]
y =[7, 8, 2, 4, 6, 4, 9, 5, 6, 3, 6, 7, 2, 4, 6, 7, 1, 9, 4, 3, 6, 9]

ax.scatter(x, y)
```

Out[39]:
```
<matplotlib.collections.PathCollection at 0x1a75d67a030>
```



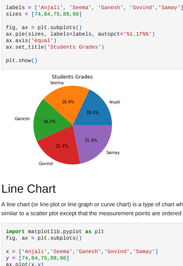
In [40]:
```
import pandas as pd
iris = pd.read_csv('F:/MSC 1/FDS/notes/Iris.csv', names=['sepal_length', 'sepal_width', 'petal_length', 'petal_width', 'class'])
print(iris.head())
```

```
   sepal_length   sepal_width   petal_length   petal_width         class
Id  SepalLengthCm  SepalWidthCm  PetalLengthCm  PetalWidthCm      Species
1            5.1           3.5            1.4           0.2  Iris-setosa
2            4.9           3.0            1.4           0.2  Iris-setosa
3            4.7           3.2            1.3           0.2  Iris-setosa
4            4.6           3.1            1.5           0.2  Iris-setosa
```

In [45]:
```
import matplotlib.pyplot as plt
# create a figure and axis
fig, ax = plt.subplots()

# scatter the sepal_length against the sepal_width
ax.scatter(iris['sepal_length'], iris['sepal_width']) # set a title and labels
ax.set_title('Iris Dataset')
ax.set_xlabel('sepal_length')
ax.set_ylabel('sepal_width')
```

Out[45]:
```
Text(0, 0.5, 'sepal_width')
```



Pie Chart

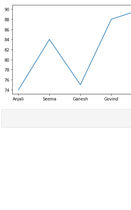
A pie chart (or a circle chart) is a circular statistical graphic, which is divided into slices to illustrate numerical proportion. In a pie chart, the arc length of each slice (and consequently its central angle and area), is proportional to the quantity it represents Pie Chart with labels

In [35]:
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline

labels = ['Anjali', 'Seema', 'Ganesh', 'Govind','Samay']
sizes = [74,84,75,88,90]

fig, ax = plt.subplots()
ax.pie(sizes, labels=labels, autopct='%1.1f%%')
ax.axis('equal')
ax.set_title('Students Grades')

plt.show()
```



Line Chart

A line chart (or line plot or line graph or curve chart) is a type of chart which displays information as a series of data points called 'markers' connected by straight line segments. It is similar to a scatter plot except that the measurement points are ordered (typically with their x-axis value) and joined with straight line segments.

In [43]:
```
import matplotlib.pyplot as plt
fig, ax = plt.subplots()

x = ['Anjali','Seema','Ganesh','Govind','Samay']
y = [74,84,75,88,90]
ax.plot(x,y)
```

Out[43]:
```
[<matplotlib.lines.Line2D at 0x1a75de89fd0>]
```


In [ ]: