Data Mining

Project Work

137 | Vishal Baraiya | 23010101014

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

1. Read Online Retail Data Set.

In [2]:
```
dt = pd.read_csv('D:\\VS_CODES\\DataMining\\ProjectDataMining\\Dataset\\Heart.csv',
dt
```

Out[2]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	tha
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2
...	...	...	...	...	...	...	...	...	...	...	...	...	..
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3

1025 rows × 14 columns

2.Read First 10 Data.

In [3]:
```
dt.head(10)
```

Out[3]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	t
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	
5	58	0	0	100	248	0	0	122	0	1.0	1	0	2	
6	58	1	0	114	318	0	2	140	0	4.4	0	3	1	
7	55	1	0	160	289	0	0	145	1	0.8	1	1	3	
8	46	1	0	120	249	0	0	144	0	0.8	2	0	3	
9	54	1	0	122	286	0	0	116	1	3.2	1	2	2	

3.Read Last 10 Data.

In [4]:
```
dt.tail(10)
```

Out[4]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	tha
1015	58	1	0	128	216	0	0	131	1	2.2	1	3	3
1016	65	1	3	138	282	1	0	174	0	1.4	1	1	2
1017	53	1	0	123	282	0	1	95	1	2.0	1	2	3
1018	41	1	0	110	172	0	0	158	0	0.0	2	0	3
1019	47	1	0	112	204	0	1	143	0	0.1	2	0	2
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3

4. Summary of statiscal data

In [5]:
```
dt.describe()
```

Out[5]:

	age	sex	cp	trestbps	chol	fbs	r
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.0
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.5
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.5
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.0
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.0
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.0
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.0
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.0

In [6]:
```
dt.describe(include='all')
```

Out[6]:

	age	sex	cp	trestbps	chol	fbs	r
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.00000	1025.000000	1025.0
mean	54.434146	0.695610	0.942439	131.611707	246.00000	0.149268	0.5
std	9.072290	0.460373	1.029641	17.516718	51.59251	0.356527	0.5
min	29.000000	0.000000	0.000000	94.000000	126.00000	0.000000	0.0
25%	48.000000	0.000000	0.000000	120.000000	211.00000	0.000000	0.0
50%	56.000000	1.000000	1.000000	130.000000	240.00000	0.000000	1.0
75%	61.000000	1.000000	2.000000	140.000000	275.00000	0.000000	1.0
max	77.000000	1.000000	3.000000	200.000000	564.00000	1.000000	2.0

5.Data Types of all the Columns

In [7]:
```
dt.dtypes
```

Out[7]:
```
age           int64
sex           int64
cp            int64
trestbps      int64
chol          int64
fbs           int64
restecg       int64
thalach       int64
exang         int64
oldpeak     float64
slope         int64
ca            int64
thal          int64
target        int64
dtype: object
```

6. Number of Rows

In [8]:
```
dt.shape[0]
```

Out[8]:
```
1025
```

7.Number of Columns

In [9]:
```
dt.shape[1]
```

Out[9]:
```
14
```

8.Sum of Any Column

In [10]:
```
dt['sex'].sum()
```

Out[10]: 713

9.Average Of Any Column

In [11]:
```
dt['sex'].mean()
```

Out[11]: 0.6956097560975609

10. Max in Column

In [12]:
```
dt['age'].max()
```

Out[12]: 77

11.Min in Columns

In [13]:
```
dt['age'].min()
```

Out[13]: 29

12.Standard deviation of column

In [14]:
```
dt['age'].std()
```

Out[14]: 9.072290233244281

13.location of column using iloc

In [15]:
```
dt.iloc[3]
```

Out[15]:
```
age          61.0
sex           1.0
cp            0.0
trestbps    148.0
chol        203.0
fbs           0.0
restecg       1.0
thalach     161.0
exang         0.0
oldpeak       0.0
slope         2.0
ca            1.0
thal          3.0
target        0.0
Name: 3, dtype: float64
```

14.copy

In [16]:
```
x = dt.copy()
x
```

Out[16]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	tha
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2
...	...	...	...	...	...	...	...	...	...	...	...	...	..
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3

1025 rows × 14 columns

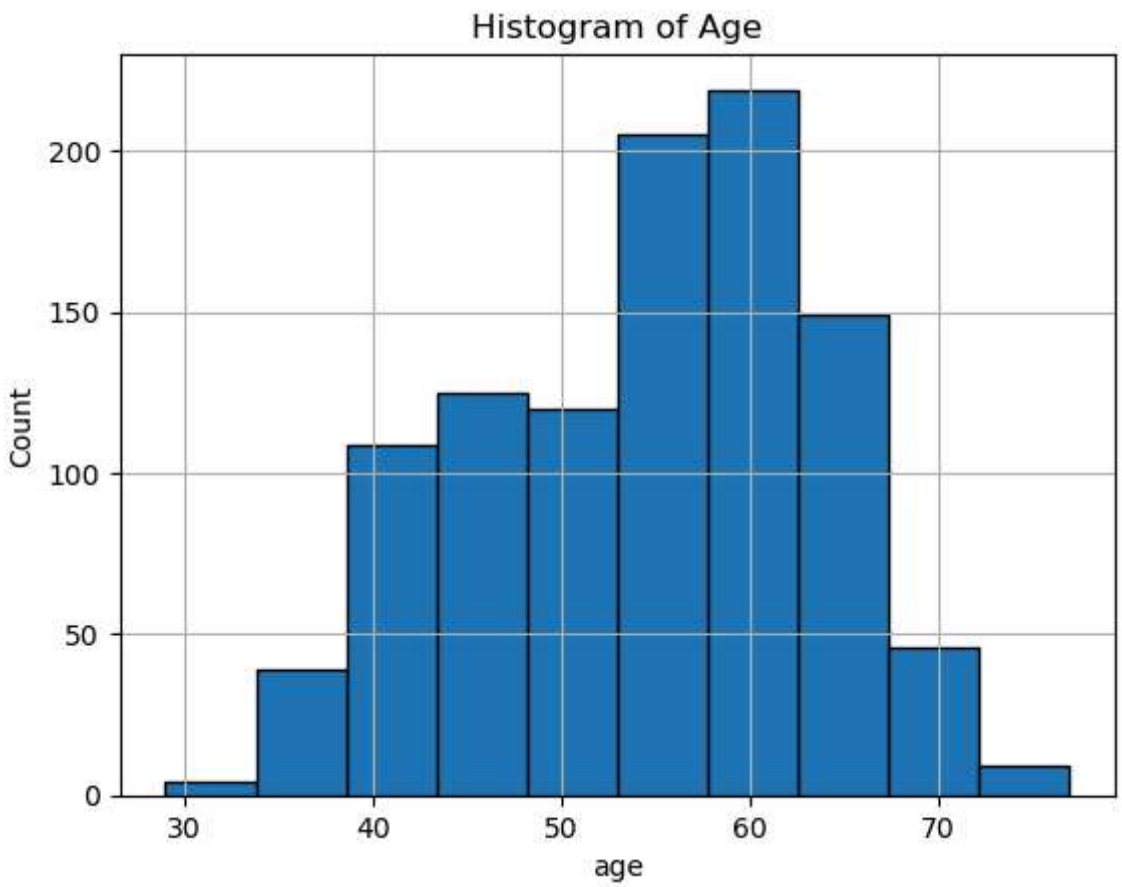
15.Unique data

In [17]:
```
dt['fbs'].nunique()
```

Out[17]: 2

16.Histogram of Age

In [18]:
```
dt['age'].hist(edgecolor = 'black')
plt.title("Histogram of Age")
plt.xlabel('age')
plt.ylabel('Count')
plt.show()
```



17.return value having 6 quantity

In [19]:
```
q_counts = dt['age'].value_counts()
q_counts[q_counts == 6]
```

Out[19]:
```
age
37    6
34    6
Name: count, dtype: int64
```

19.drop

In [20]:
```
dt.drop("cp", axis=1, inplace=True)
```

In [21]:
```
dt
```

Out[21]:

	age	sex	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	ta
0	52	1	125	212	0	1	168	0	1.0	2	2	3	
1	53	1	140	203	1	0	155	1	3.1	0	0	3	
2	70	1	145	174	0	1	125	1	2.6	0	0	3	
3	61	1	148	203	0	1	161	0	0.0	2	1	3	
4	62	0	138	294	1	1	106	0	1.9	1	3	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	
1020	59	1	140	221	0	1	164	1	0.0	2	0	2	
1021	60	1	125	258	0	0	141	1	2.8	1	1	3	
1022	47	1	110	275	0	0	118	1	1.0	1	1	2	
1023	50	0	110	254	0	0	159	0	0.0	2	0	2	
1024	54	1	120	188	0	1	113	0	1.4	1	1	3	

1025 rows × 13 columns

18.Condition

In [22]:
```
dt[dt["sex"] == 1]
```

Out[22]:

	age	sex	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	ta
0	52	1	125	212	0	1	168	0	1.0	2	2	3	
1	53	1	140	203	1	0	155	1	3.1	0	0	3	
2	70	1	145	174	0	1	125	1	2.6	0	0	3	
3	61	1	148	203	0	1	161	0	0.0	2	1	3	
6	58	1	114	318	0	2	140	0	4.4	0	3	1	
...	...	...	...	...	...	...	...	...	...	...	...	...	
1019	47	1	112	204	0	1	143	0	0.1	2	0	2	
1020	59	1	140	221	0	1	164	1	0.0	2	0	2	
1021	60	1	125	258	0	0	141	1	2.8	1	1	3	
1022	47	1	110	275	0	0	118	1	1.0	1	1	2	
1024	54	1	120	188	0	1	113	0	1.4	1	1	3	

713 rows × 13 columns

20.length

In [23]:
```
len(dt[dt["age"] == 2])
```

Out[23]:
```
0
```

21.Groupby

In [24]:
```
quant = dt.groupby('sex')['age'].sum()
total = quant[quant > 100]
total
```

Out[24]:
```
sex
0    17425
1    38370
Name: age, dtype: int64
```

22.return Index

In [25]:
```
dt.index
```

Out[25]:
```
RangeIndex(start=0, stop=1025, step=1)
```

23.return highest column value

In [26]:
```
dt['age'].value_counts().head(1)
```

Out[26]:
```
age
58    68
Name: count, dtype: int64
```

24.using loc

In [27]:
```
dt.loc[1,'age']
```

Out[27]:
```
53
```

25.set Index

In [28]:
```
dt.set_index('age')
```

Out[28]:

age	sex	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
52	1	125	212	0	1	168	0	1.0	2	2	3	0
53	1	140	203	1	0	155	1	3.1	0	0	3	0
70	1	145	174	0	1	125	1	2.6	0	0	3	0
61	1	148	203	0	1	161	0	0.0	2	1	3	0
62	0	138	294	1	1	106	0	1.9	1	3	2	0
...	...	...	...	...	...	...	...	...	...	...	...	...
59	1	140	221	0	1	164	1	0.0	2	0	2	1
60	1	125	258	0	0	141	1	2.8	1	1	3	0
47	1	110	275	0	0	118	1	1.0	1	1	2	0
50	0	110	254	0	0	159	0	0.0	2	0	2	1
54	1	120	188	0	1	113	0	1.4	1	1	3	0

1025 rows × 12 columns

In [ ]: