1. Import Libraries

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

2. Import Dataset

```python
In [2]: dataset=pd.read_csv("Titanic-Dataset.csv")
        dataset
```

Out[2]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN
...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN

891 rows × 12 columns

In [3]:
```python
dataset.shape
```

Out[3]:
```
(891, 12)
```

In [4]: `dataset.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

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

Out[5]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

Handling Null Values

In [6]: `dataset.isnull().any()`

Out[6]:
```
PassengerId    False
Survived       False
Pclass         False
Name           False
Sex            False
Age             True
SibSp          False
Parch          False
Ticket         False
Fare           False
Cabin           True
Embarked        True
dtype: bool
```

```
In [7]:  dataset["Age"]=dataset["Age"].fillna(dataset["Age"].mean())
```

```
In [8]:  dataset["Cabin"]=dataset["Cabin"].fillna(method = 'pad')
```

```
In [9]:  dataset["Embarked"]=dataset["Embarked"].fillna(dataset["Embarked"].mode()[0])
```
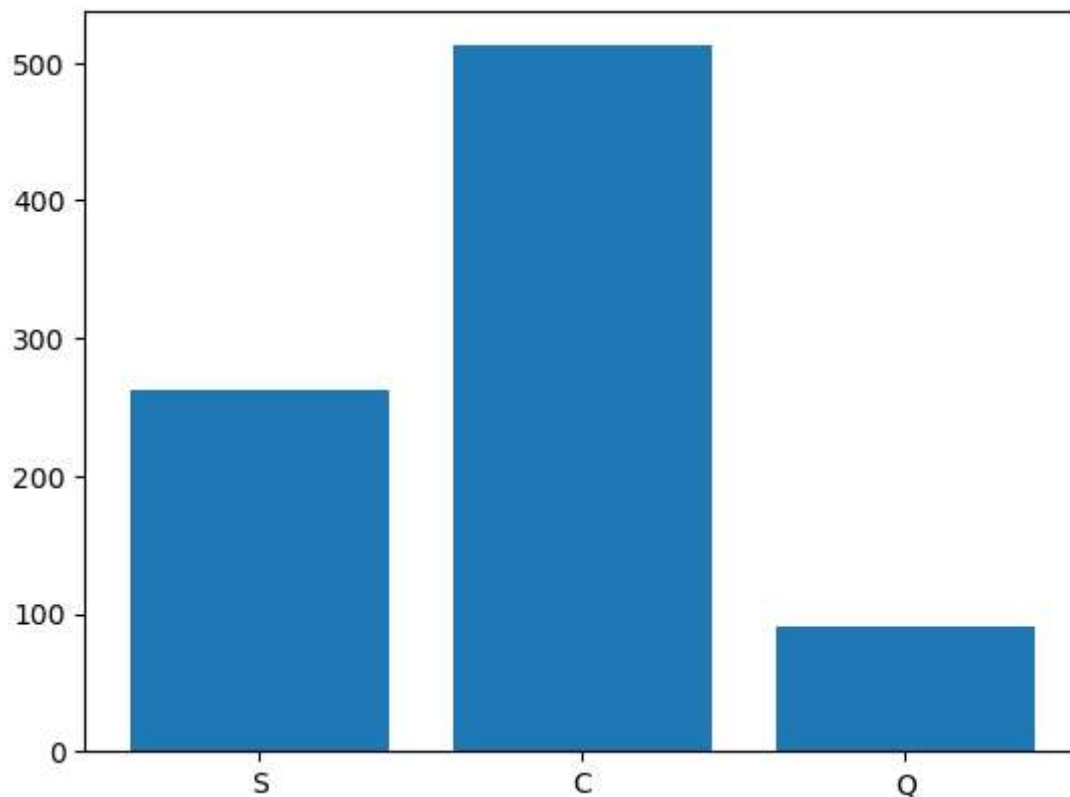
```
In [10]:  dataset.isnull().any()
```

```
Out[10]:  PassengerId     False
          Survived        False
          Pclass          False
          Name            False
          Sex             False
          Age             False
          SibSp           False
          Parch           False
          Ticket          False
          Fare            False
          Cabin            True
          Embarked        False
          dtype: bool
```

Data Visualization

```
In [11]:  plt.bar("Embarked","Fare",data=dataset)
```
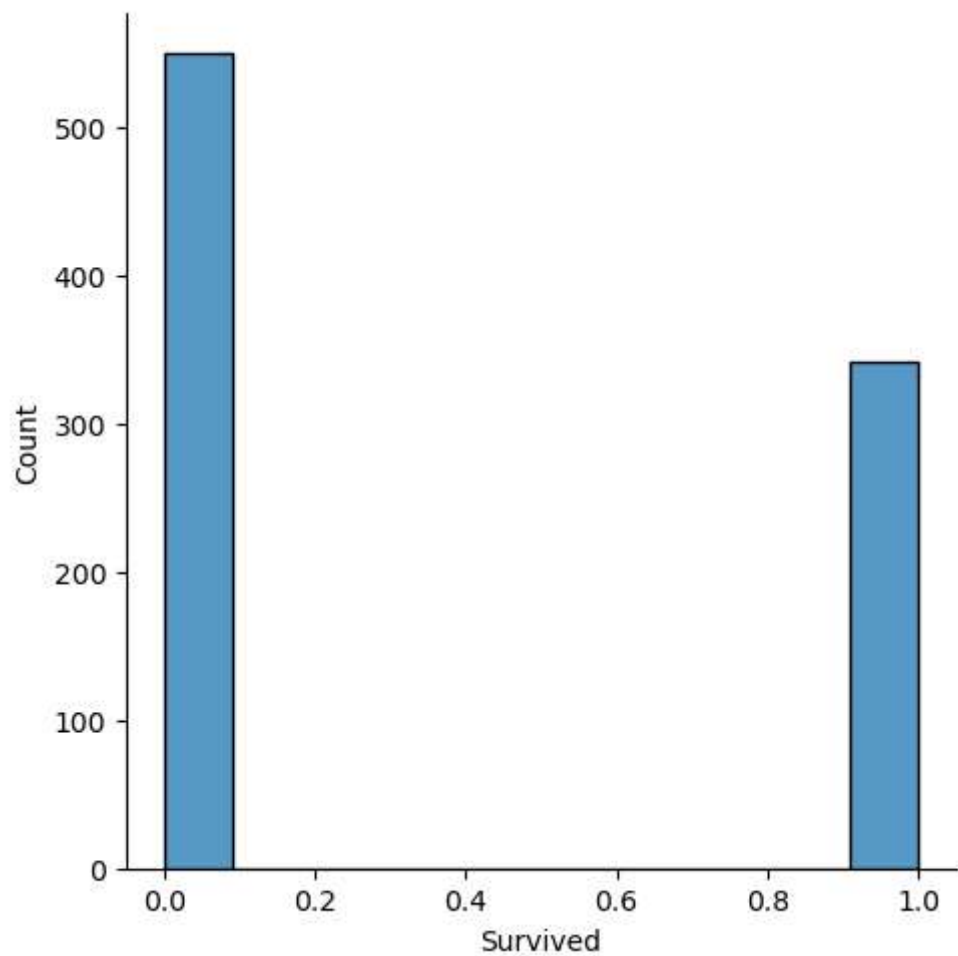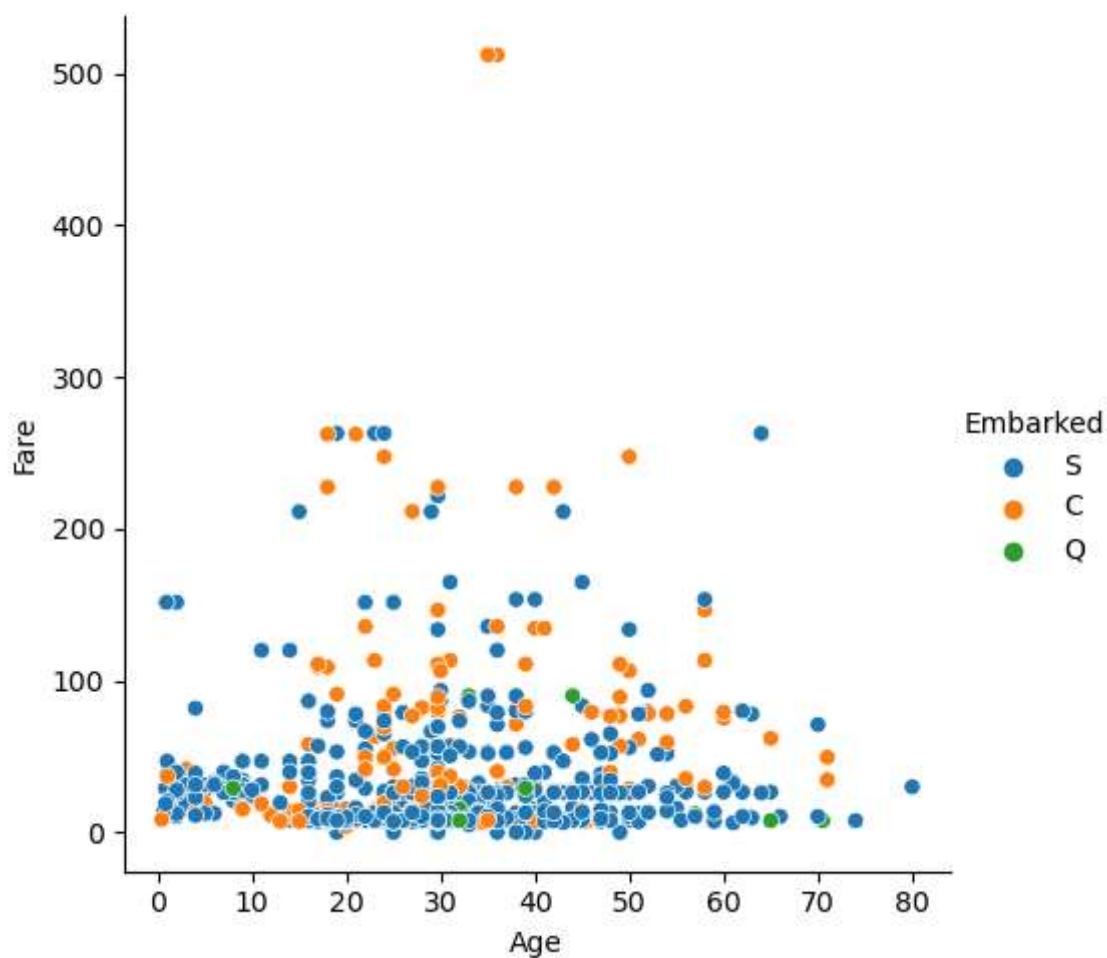
```
Out[11]:  <BarContainer object of 891 artists>
```



```
In [12]:  sns.displot(dataset["Survived"])
```
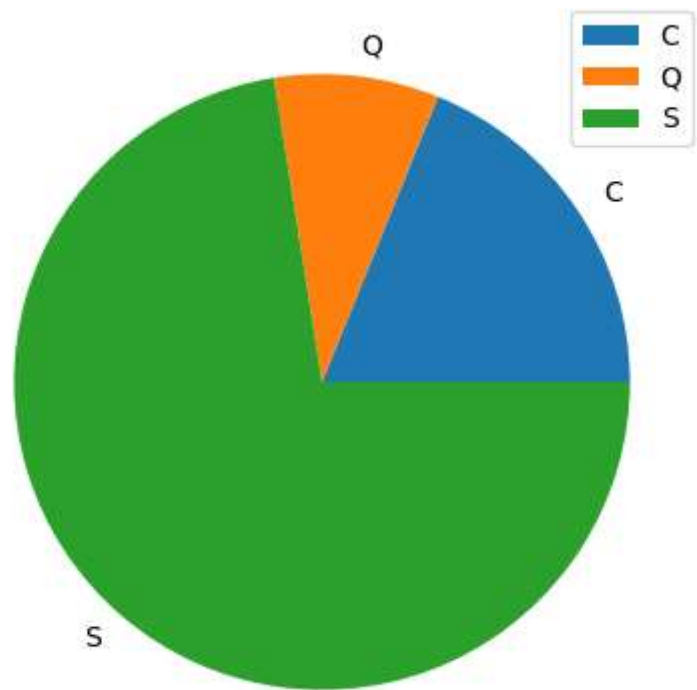
Out[12]:   `<seaborn.axisgrid.FacetGrid at 0x287bb3bd350>`

In [13]:
```python
sns.relplot(x="Age",y="Fare",data=dataset,hue="Embarked")
```

Out[13]:   `<seaborn.axisgrid.FacetGrid at 0x287bc1bff50>`

```
In [14]:  fig=plt.figure()
          axes1=fig.add_axes([0.1,0.1,0.8,0.8])    #[left,bottom,width,height]
          x=dataset.groupby(["Embarked"])["Embarked"].count()
          axes1.pie(x.values,labels=["C","Q","S"],data=x)
          axes1.legend()
```

Out[14]: <matplotlib.legend.Legend at 0x1dfd4b09dd0>

In [15]:
```python
corr=dataset.corr()
corr
```

Out[15]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1.000000	-0.005007	-0.035144	0.033207	-0.057527	-0.001652	0.012658
Survived	-0.005007	1.000000	-0.338481	-0.069809	-0.035322	0.081629	0.257307
Pclass	-0.035144	-0.338481	1.000000	-0.331339	0.083081	0.018443	-0.549500
Age	0.033207	-0.069809	-0.331339	1.000000	-0.232625	-0.179191	0.091566
SibSp	-0.057527	-0.035322	0.083081	-0.232625	1.000000	0.414838	0.159651
Parch	-0.001652	0.081629	0.018443	-0.179191	0.414838	1.000000	0.216225
Fare	0.012658	0.257307	-0.549500	0.091566	0.159651	0.216225	1.000000

In [16]:
```python
sns.heatmap(corr,annot=True)
```
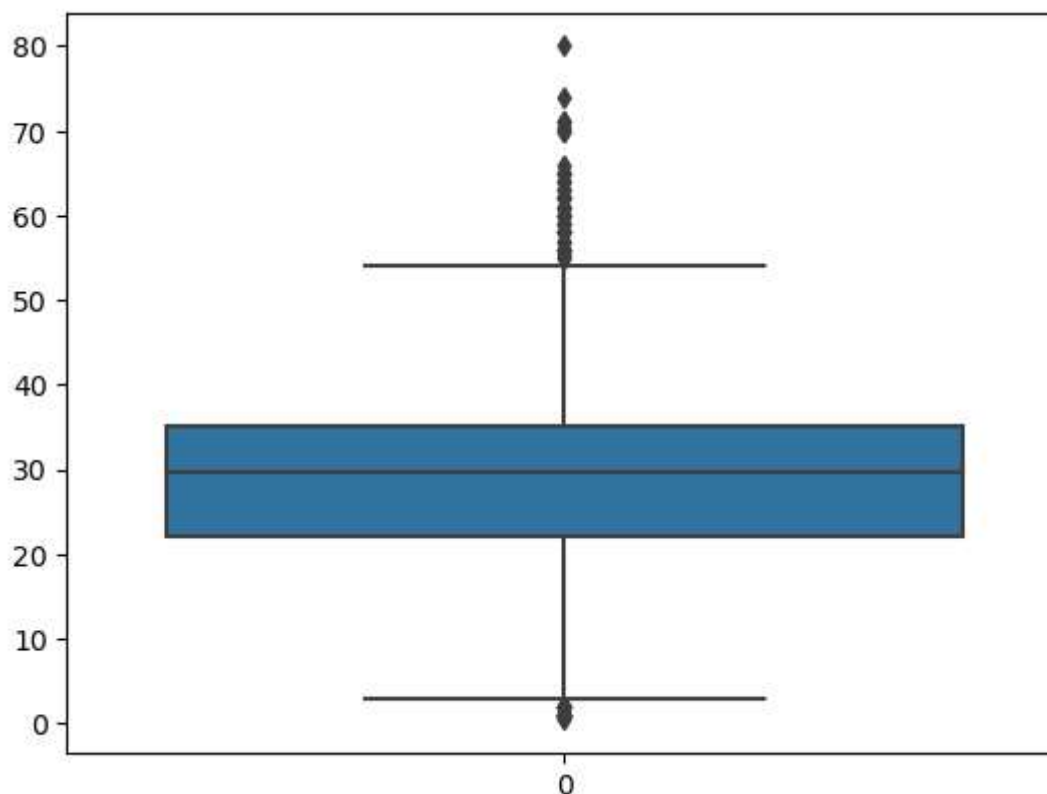
Out[16]:
```
<Axes: >
```
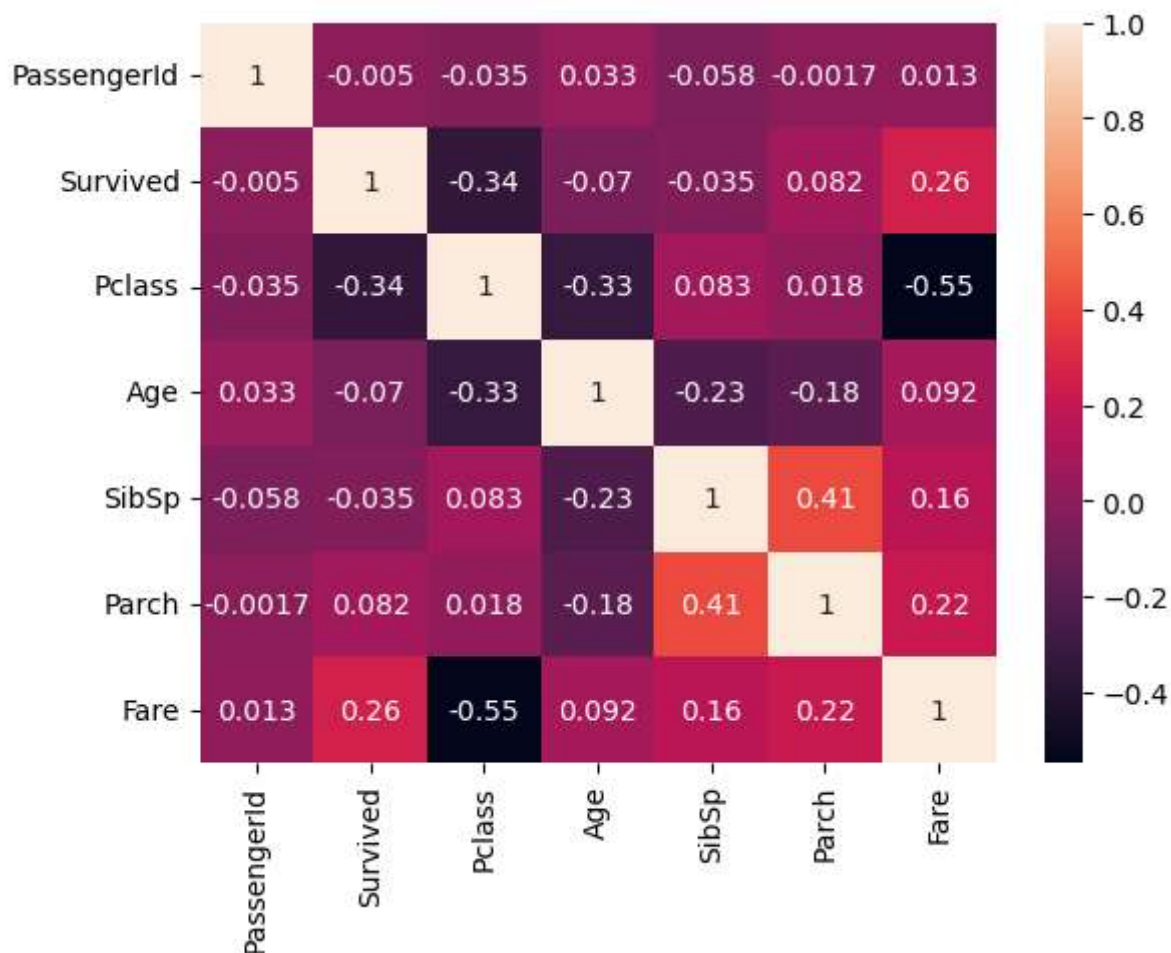
Outlier Detection

In [17]: `sns.boxplot(dataset.Age)`

Out[17]: `<Axes: >`

Splitting Dependent and Independent variables

```
In [18]:  x=dataset.iloc[:,[2,4,5,6,7,10]]
          y=dataset.iloc[:,1:2]
          x
```

Out[18]:

	Pclass	Sex	Age	SibSp	Parch	Cabin
0	3	male	22.000000	1	0	NaN
1	1	female	38.000000	1	0	C85
2	3	female	26.000000	0	0	C85
3	1	female	35.000000	1	0	C123
4	3	male	35.000000	0	0	C123
...	...	...	...	...	...	...
886	2	male	27.000000	0	0	C50
887	1	female	19.000000	0	0	B42
888	3	female	29.699118	1	2	B42
889	1	male	26.000000	0	0	C148
890	3	male	32.000000	0	0	C148

891 rows × 6 columns

Encoding

```
In [19]:   from sklearn.preprocessing import LabelEncoder
           le=LabelEncoder()
```

```
In [21]:   x["Sex"]=le.fit_transform(x["Sex"])
           x["Sex"]
```

```
Out[21]:   0      1
           1      0
           2      0
           3      0
           4      1
                  ..
           886    1
           887    0
           888    0
           889    1
           890    1
           Name: Sex, Length: 891, dtype: int32
```

```
In [33]:   x["Cabin"]=le.fit_transform(x["Cabin"])
           x["Cabin"]
```

```
Out[33]:   0      147
           1       81
           2       81
           3       55
           4       55
                  ...
           886     70
           887     30
           888     30
           889     60
           890     60
           Name: Cabin, Length: 891, dtype: int32
```

Feature Scaling

```
In [34]:   from sklearn.preprocessing import StandardScaler
           sc=StandardScaler()
```

```
In [35]:   x=sc.fit_transform(x)
           x
```

```
Out[35]:    array([[ 0.82737724,  0.73769513, -0.5924806 ,  0.43279337, -0.47367361,
                    1.65629226],
                   [-1.56610693, -1.35557354,  0.63878901,  0.43279337, -0.47367361,
                    0.10733594],
                   [ 0.82737724, -1.35557354, -0.2846632 , -0.4745452 , -0.47367361,
                    0.10733594],
                   ...,
                   [ 0.82737724, -1.35557354,  0.         ,  0.43279337,  2.00893337,
                   -1.08958486],
                   [-1.56610693,  0.73769513, -0.2846632 , -0.4745452 , -0.47367361,
                   -0.3855138 ],
                   [ 0.82737724,  0.73769513,  0.17706291, -0.4745452 , -0.47367361,
                   -0.3855138 ]])
```

Splitting Training and Testing Set

```
In [36]:   from sklearn.model_selection import train_test_split
```

```
In [37]:   x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=0)
```

```
In [38]:   x_train
```

```
Out[38]:    array([[ 0.82737724, -1.35557354,  0.         , -0.4745452 ,  2.00893337,
                   -0.73754933],
                   [-0.36936484,  0.73769513,  0.10010856, -0.4745452 , -0.47367361,
                   -0.3151067 ],
                   [-0.36936484,  0.73769513,  0.10010856,  0.43279337,  0.76762988,
                   -1.41815136],
                   ...,
                   [ 0.82737724,  0.73769513,  0.         , -0.4745452 , -0.47367361,
                    0.8818141 ],
                   [ 0.82737724, -1.35557354,  0.48488031,  0.43279337, -0.47367361,
                    1.3511948 ],
                   [-0.36936484,  0.73769513,  2.33178473,  0.43279337,  0.76762988,
                    0.76446892]])
```

```
In [39]:   x_test
```

```
Out[39]:    array([[ 0.82737724,  0.73769513,  0.         , -0.4745452 , -0.47367361,
                   -0.29163766],
                   [ 0.82737724,  0.73769513,  0.         , -0.4745452 , -0.47367361,
                   -1.62937267],
                   [ 0.82737724,  0.73769513, -1.74679586,  3.15480905,  0.76762988,
                    0.85834506],
                   ...,
                   [-1.56610693, -1.35557354,  0.10010856,  0.43279337, -0.47367361,
                    0.62365471],
                   [ 0.82737724,  0.73769513, -0.51552625, -0.4745452 , -0.47367361,
                   -1.30080618],
                   [ 0.82737724,  0.73769513, -0.82334365, -0.4745452 , -0.47367361,
                    1.2807877 ]])
```