In [1]:

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```python
data=pd.read_csv("Titanic-Dataset.csv")
```

In [3]:

```python
data
```

Out[3]:

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

891 rows × 12 columns

In [4]:

```python
data.head()
```

Out[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

In [5]:

```python
data.tail()
```

Out[5]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	NaN	Q

In [6]:

```python
data.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 [7]:

```
data.describe()
```

Out[7]:

	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 [8]:

```
data.isnull().any()
```

Out[8]:

```
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 [9]:

```
data.isnull().sum()
```

Out[9]:

```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

Filling null values in Age column with Mean

In [11]:

```
mean=data["Age"].mean()
mean
```

Out[11]:

```
29.69911764705882
```

In [18]:

```
data["Age"]=data["Age"].fillna(mean)
data["Age"].head()
```

```
Out[18]: 0    22.0
         1    38.0
         2    26.0
         3    35.0
         4    35.0
         Name: Age, dtype: float64
```

```
In [17]: data["Age"].tail()
```

```
Out[17]: 886    27.000000
         887    19.000000
         888    29.699118
         889    26.000000
         890    32.000000
         Name: Age, dtype: float64
```

```
In [19]: data["Age"].isnull().sum()
```

```
Out[19]: 0
```

Filling null values in Cabin column with Mode

```
In [21]: data["Cabin"]
```

```
Out[21]: 0       NaN
         1       C85
         2       NaN
         3      C123
         4       NaN
                ... 
         886     NaN
         887     B42
         888     NaN
         889    C148
         890     NaN
         Name: Cabin, Length: 891, dtype: object
```

```
In [20]: mode=data["Cabin"].mode()
         mode
```

```
Out[20]: 0        B96 B98
         1    C23 C25 C27
         2             G6
         Name: Cabin, dtype: object
```

```
In [26]: data["Cabin"]=data["Cabin"].fillna(mode[1])
         data["Cabin"].head()
```

```
Out[26]: 0    B96 B98
         1        C85
         2         G6
         3       C123
         4         G6
         Name: Cabin, dtype: object
```

```
In [27]: data["Cabin"].tail()
```

```
Out[27]: 886      G6
         887     B42
         888      G6
         889    C148
         890      G6
         Name: Cabin, dtype: object
```

```
In [28]: data["Cabin"].isnull().sum()
```

```
Out[28]: 0
```

Filling null values in Embarked with Mode

```
In [29]: data["Embarked"]
```

```
Out[29]: 0      S
         1      C
         2      S
         3      S
         4      S
               ..
         886    S
         887    S
         888    S
         889    C
         890    Q
         Name: Embarked, Length: 891, dtype: object
```

```
In [31]: mode=data["Embarked"].mode()
```

```
mode
```

Out[31]:

```
0    S
Name: Embarked, dtype: object
```

In [32]:

```
data["Embarked"]=data["Embarked"].fillna(mode[0])
data["Embarked"].head()
```

Out[32]:

```
0    S
1    C
2    S
3    S
4    S
Name: Embarked, dtype: object
```

In [33]:

```
data["Embarked"].tail()
```

Out[33]:

```
886    S
887    S
888    S
889    C
890    Q
Name: Embarked, dtype: object
```

In [34]:

```
data["Embarked"].isnull().sum()
```

Out[34]:

```
0
```

In [36]:

```
data.isnull().sum()
```

Out[36]:

```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Cabin          0
Embarked       0
dtype: int64
```

Data Visualisation

In [37]:

```
cor=data.corr()
```

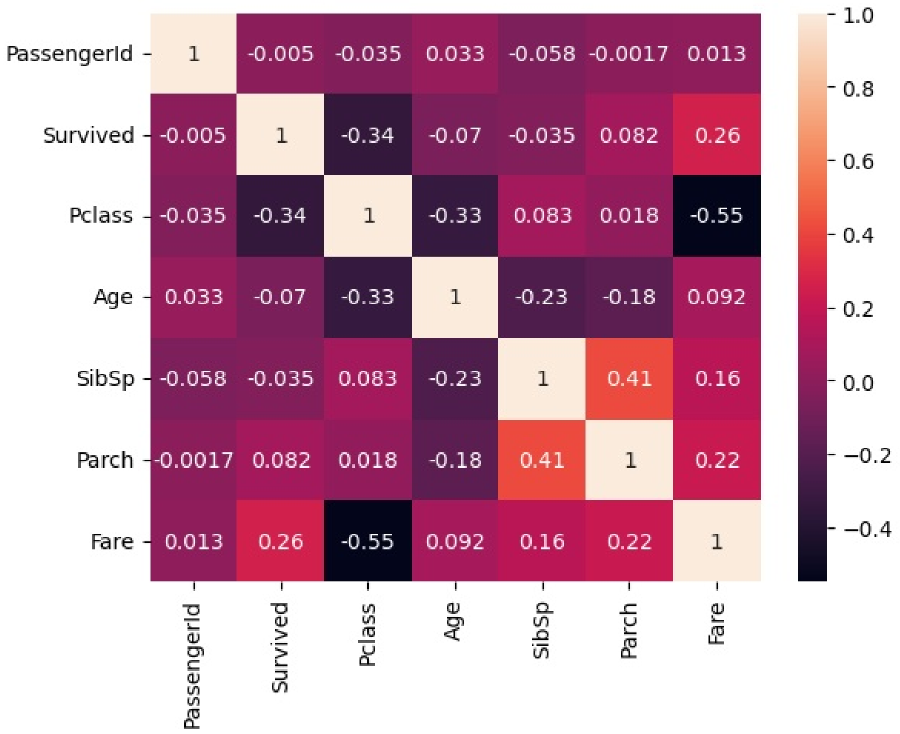
```
C:\Users\karth\AppData\Local\Temp\ipykernel_1392\1426905697.py:1: FutureWarning: The default value of numeric_o
nly in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns o
r specify the value of numeric_only to silence this warning.
  cor=data.corr()
```

In [38]:

```
sns.heatmap(cor,annot=True)
```

Out[38]:

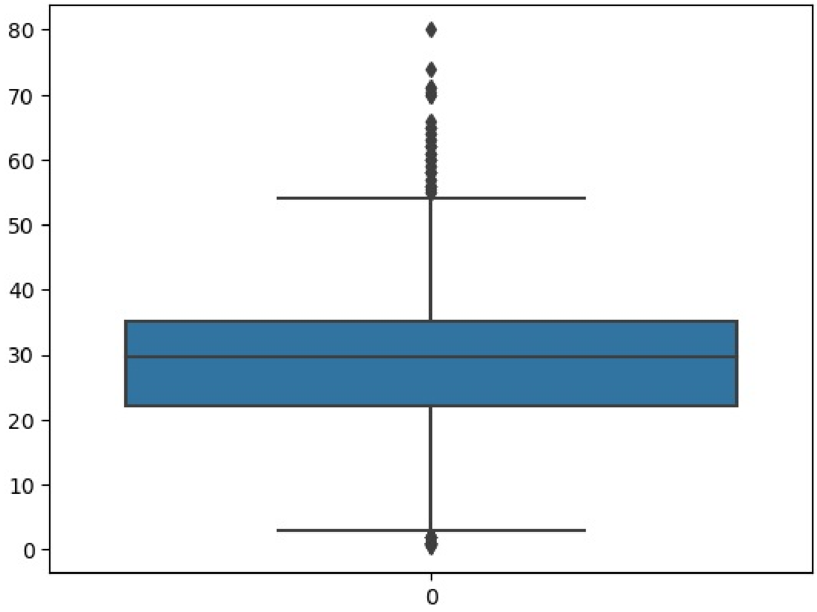
```
<Axes: >
```



Outlier Handling

In [39]:
```
sns.boxplot(data["Age"])
```

Out[39]:
```
<Axes: >
```



Outliers

In [41]:
```
Age_q1=data.Age.quantile(0.25)
Age_q3=data.Age.quantile(0.75)
print(Age_q1)
print(Age_q3)
```

```
22.0
35.0
```

In [42]:
```
IQR_Age=Age_q3-Age_q1
IQR_Age
```

Out[42]:
```
13.0
```

In [43]:
```
upperlimit_Age=Age_q3+1.5*IQR_Age
upperlimit_Age
```

Out[43]:
```
54.5
```

In [44]:
```
lowerlimit_Age=Age_q1-1.5*IQR_Age
lowerlimit_Age
```

Out[44]:
```
2.5
```

In [45]:
```
median_Age=data["Age"].median()
median_Age
```

Out[45]:
```
29.69911764705882
```

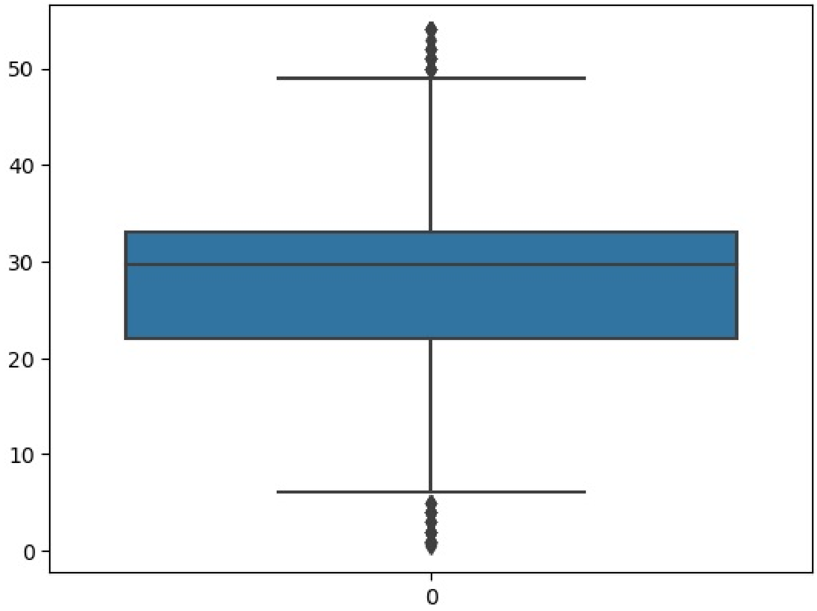
In [46]:
```
data["Age"]=np.where(data["Age"]>upperlimit_Age,median_Age,data["Age"])
```

In [47]:
```
(data["Age"]>54.5).sum()
```

Out[47]:
```
0
```

In [48]:
```
sns.boxplot(data["Age"])
```

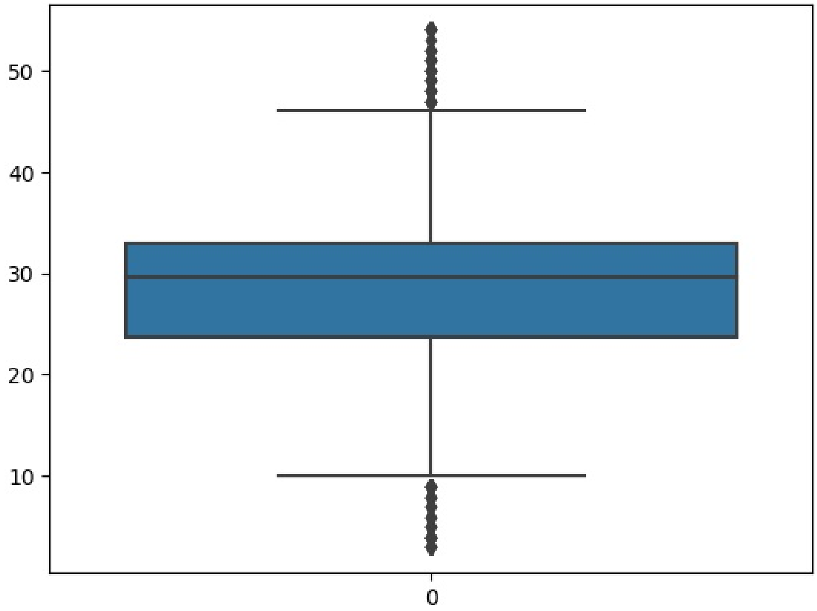
Out[48]:
```
<Axes: >
```



In [49]:
```
data["Age"]=np.where(data["Age"]<lowerlimit_Age,median_Age,data["Age"])
```

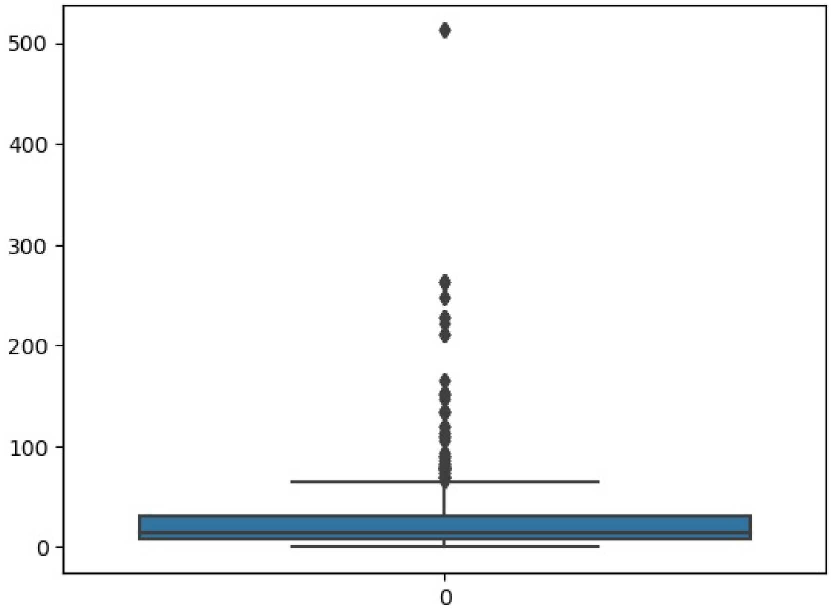
In [50]:
```
sns.boxplot(data["Age"])
```

Out[50]:
```
<Axes: >
```



In [51]:
```
sns.boxplot(data["Fare"])
```

Out[51]:
```
<Axes: >
```



In [52]:
```
Fare_q1=data.Fare.quantile(0.25)
Fare_q3=data.Fare.quantile(0.75)
print(Fare_q1)
print(Fare_q3)
```

```
7.9104
31.0
```

In [53]:
```
IQR_Fare=Fare_q3-Fare_q1
IQR_Fare
```

Out[53]:
```
23.0896
```

In [54]:
```
upperlimit_Fare=Fare_q3+1.5*IQR_Fare
upperlimit_Fare
```

Out[54]:
```
65.6344
```

In [55]:
```
lowerlimit_Fare=Fare_q1-1.5*IQR_Fare
lowerlimit_Fare
```

Out[55]:
```
-26.724
```

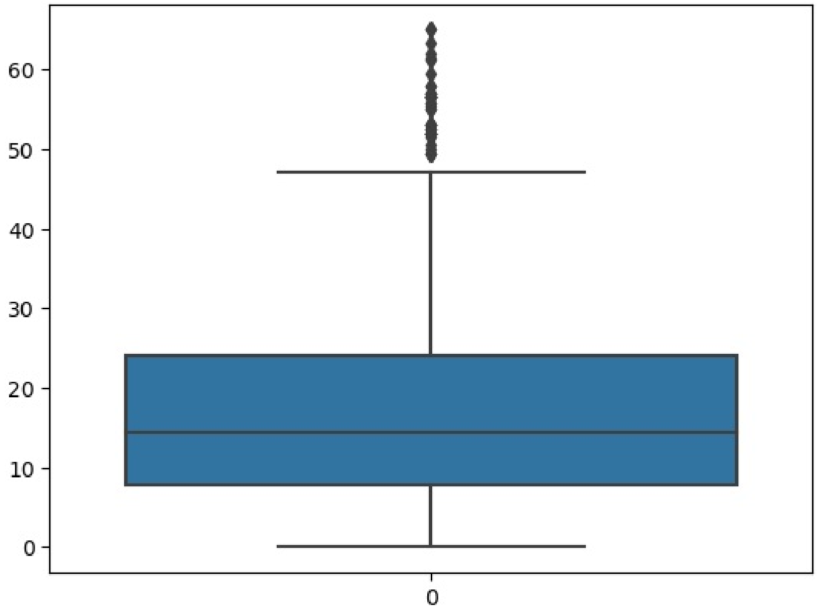
In [56]:
```
median_Fare=data["Fare"].median()
median_Fare
```

Out[56]:
```
14.4542
```

In [57]:
```
data["Fare"]=np.where(data["Fare"]>upperlimit_Fare,median_Fare,data["Fare"])
```

In [58]:
```
sns.boxplot(data["Fare"])
```

Out[58]:
```
<Axes: >
```



```
In [59]: (data["Fare"]>65).sum()
```

```
Out[59]: 0
```

Dropping the variables

```
In [87]: data.drop(["Name"],axis=1,inplace=True)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[87], line 1
----> 1 data.drop(["Name"],axis=1,inplace=True)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\frame.py:5399, in DataFrame.drop(self, labels, axis, index, columns, level, inplace, errors)
   5251 @deprecate_nonkeyword_arguments(version=None, allowed_args=["self", "labels"])
   5252 def drop(  # type: ignore[override]
   5253     self,
   (...)
   5260     errors: IgnoreRaise = "raise",
   5261 ) -> DataFrame | None:
   5262     """
   5263     Drop specified labels from rows or columns.
   5264
   (...)
   5397             weight  1.0     0.8
   5398     """
-> 5399     return super().drop(
   5400         labels=labels,
   5401         axis=axis,
   5402         index=index,
   5403         columns=columns,
   5404         level=level,
   5405         inplace=inplace,
   5406         errors=errors,
   5407     )

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4505, in NDFrame.drop(self, labels, axis, index, columns, level, inplace, errors)
   4503 for axis, labels in axes.items():
   4504     if labels is not None:
-> 4505         obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   4507 if inplace:
   4508     self._update_inplace(obj)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4546, in NDFrame._drop_axis(self, labels, axis, level, errors, only_slice)
   4544         new_axis = axis.drop(labels, level=level, errors=errors)
   4545     else:
-> 4546         new_axis = axis.drop(labels, errors=errors)
   4547     indexer = axis.get_indexer(new_axis)
   4549 # Case for non-unique axis
   4550 else:

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:6934, in Index.drop(self, labels, errors)
   6932 if mask.any():
   6933     if errors != "ignore":
-> 6934         raise KeyError(f"{list(labels[mask])} not found in axis")
   6935     indexer = indexer[~mask]
   6936 return self.delete(indexer)

KeyError: "['Name'] not found in axis"
```

```
In [88]: data
```

Out[88]:

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	3	1	22.000000	1	0	7.2500	S
1	1	0	38.000000	1	0	14.4542	C
2	3	0	26.000000	0	0	7.9250	S
3	1	0	35.000000	1	0	53.1000	S
4	3	1	35.000000	0	0	8.0500	S
...	...	...	...	...	...	...	...
886	2	1	27.000000	0	0	13.0000	S
887	1	0	19.000000	0	0	30.0000	S
888	3	0	29.699118	1	2	23.4500	S
889	1	1	26.000000	0	0	30.0000	C
890	3	1	32.000000	0	0	7.7500	Q

891 rows × 7 columns

In [91]:

```
data.drop(["Ticket"],axis=1,inplace=True)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[91], line 1
----> 1 data.drop(['Ticket'],axis=1,inplace=True)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_argumen
ts.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\frame.py:5399, in DataFrame.drop(self, labels, axis
, index, columns, level, inplace, errors)
   5251 @deprecate_nonkeyword_arguments(version=None, allowed_args=["self", "labels"])
   5252 def drop(  # type: ignore[override]
   5253     self,
   (...)
   5260     errors: IgnoreRaise = "raise",
   5261 ) -> DataFrame | None:
   5262     """
   5263     Drop specified labels from rows or columns.
   5264 
   (...)
   5397             weight  1.0     0.8
   5398     """
-> 5399     return super().drop(
   5400         labels=labels,
   5401         axis=axis,
   5402         index=index,
   5403         columns=columns,
   5404         level=level,
   5405         inplace=inplace,
   5406         errors=errors,
   5407     )

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_argumen
ts.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4505, in NDFrame.drop(self, labels, axis
, index, columns, level, inplace, errors)
   4503 for axis, labels in axes.items():
   4504     if labels is not None:
-> 4505         obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   4507 if inplace:
   4508     self._update_inplace(obj)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4546, in NDFrame._drop_axis(self, labels
, axis, level, errors, only_slice)
   4544         new_axis = axis.drop(labels, level=level, errors=errors)
   4545     else:
-> 4546         new_axis = axis.drop(labels, errors=errors)
   4547     indexer = axis.get_indexer(new_axis)
   4549 # Case for non-unique axis
   4550 else:

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:6934, in Index.drop(self, labels, e
rrors)
   6932 if mask.any():
   6933     if errors != "ignore":
-> 6934         raise KeyError(f"{list(labels[mask])} not found in axis")
   6935     indexer = indexer[~mask]
   6936 return self.delete(indexer)

KeyError: "['Ticket'] not found in axis"
```

```
In [92]: data
```

Out[92]:

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	3	1	22.000000	1	0	7.2500	S
1	1	0	38.000000	1	0	14.4542	C
2	3	0	26.000000	0	0	7.9250	S
3	1	0	35.000000	1	0	53.1000	S
4	3	1	35.000000	0	0	8.0500	S
...	...	...	...	...	...	...	...
886	2	1	27.000000	0	0	13.0000	S
887	1	0	19.000000	0	0	30.0000	S
888	3	0	29.699118	1	2	23.4500	S
889	1	1	26.000000	0	0	30.0000	C
890	3	1	32.000000	0	0	7.7500	Q

891 rows × 7 columns

In [93]:

```
data.drop(["PassengerId"],axis=1,inplace=True)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[93], line 1
----> 1 data.drop(["PassengerId"],axis=1,inplace=True)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\frame.py:5399, in DataFrame.drop(self, labels, axis, index, columns, level, inplace, errors)
   5251 @deprecate_nonkeyword_arguments(version=None, allowed_args=["self", "labels"])
   5252 def drop(  # type: ignore[override]
   5253     self,
   (...)
   5260     errors: IgnoreRaise = "raise",
   5261 ) -> DataFrame | None:
   5262     """
   5263     Drop specified labels from rows or columns.
   5264
   (...)
   5397             weight  1.0     0.8
   5398     """
-> 5399     return super().drop(
   5400         labels=labels,
   5401         axis=axis,
   5402         index=index,
   5403         columns=columns,
   5404         level=level,
   5405         inplace=inplace,
   5406         errors=errors,
   5407     )

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4505, in NDFrame.drop(self, labels, axis, index, columns, level, inplace, errors)
   4503 for axis, labels in axes.items():
   4504     if labels is not None:
-> 4505         obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   4507 if inplace:
   4508     self._update_inplace(obj)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4546, in NDFrame._drop_axis(self, labels, axis, level, errors, only_slice)
   4544         new_axis = axis.drop(labels, level=level, errors=errors)
   4545     else:
-> 4546         new_axis = axis.drop(labels, errors=errors)
   4547     indexer = axis.get_indexer(new_axis)
   4549 # Case for non-unique axis
   4550 else:

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:6934, in Index.drop(self, labels, errors)
   6932 if mask.any():
   6933     if errors != "ignore":
-> 6934         raise KeyError(f"{list(labels[mask])} not found in axis")
   6935     indexer = indexer[~mask]
   6936 return self.delete(indexer)

KeyError: "['PassengerId'] not found in axis"
```

In [94]:

```
data
```

Out[94]:

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	3	1	22.000000	1	0	7.2500	S
1	1	0	38.000000	1	0	14.4542	C
2	3	0	26.000000	0	0	7.9250	S
3	1	0	35.000000	1	0	53.1000	S
4	3	1	35.000000	0	0	8.0500	S
...	...	...	...	...	...	...	...
886	2	1	27.000000	0	0	13.0000	S
887	1	0	19.000000	0	0	30.0000	S
888	3	0	29.699118	1	2	23.4500	S
889	1	1	26.000000	0	0	30.0000	C
890	3	1	32.000000	0	0	7.7500	Q

891 rows × 7 columns

In [95]:

```
data.drop(["Cabin"],axis=1,inplace=True)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[95], line 1
----> 1 data.drop(["Cabin"],axis=1,inplace=True)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\frame.py:5399, in DataFrame.drop(self, labels, axis, index, columns, level, inplace, errors)
   5251 @deprecate_nonkeyword_arguments(version=None, allowed_args=["self", "labels"])
   5252 def drop(  # type: ignore[override]
   5253     self,
   (...)
   5260     errors: IgnoreRaise = "raise",
   5261 ) -> DataFrame | None:
   5262     """
   5263     Drop specified labels from rows or columns.
   5264 
   (...)
   5397             weight  1.0     0.8
   5398     """
-> 5399     return super().drop(
   5400         labels=labels,
   5401         axis=axis,
   5402         index=index,
   5403         columns=columns,
   5404         level=level,
   5405         inplace=inplace,
   5406         errors=errors,
   5407     )

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\util\_decorators.py:331, in deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*args, **kwargs)
    325 if len(args) > num_allow_args:
    326     warnings.warn(
    327         msg.format(arguments=_format_argument_list(allow_args)),
    328         FutureWarning,
    329         stacklevel=find_stack_level(),
    330     )
--> 331 return func(*args, **kwargs)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4505, in NDFrame.drop(self, labels, axis, index, columns, level, inplace, errors)
   4503 for axis, labels in axes.items():
   4504     if labels is not None:
-> 4505         obj = obj._drop_axis(labels, axis, level=level, errors=errors)
   4507 if inplace:
   4508     self._update_inplace(obj)

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\generic.py:4546, in NDFrame._drop_axis(self, labels, axis, level, errors, only_slice)
   4544         new_axis = axis.drop(labels, level=level, errors=errors)
   4545     else:
-> 4546         new_axis = axis.drop(labels, errors=errors)
   4547     indexer = axis.get_indexer(new_axis)
   4549 # Case for non-unique axis
   4550 else:

File C:\ProgramData\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:6934, in Index.drop(self, labels, errors)
   6932 if mask.any():
   6933     if errors != "ignore":
-> 6934         raise KeyError(f"{list(labels[mask])} not found in axis")
   6935     indexer = indexer[~mask]
   6936 return self.delete(indexer)

KeyError: "['Cabin'] not found in axis"
```

In [96]:

```
data
```

Out[96]:

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	3	1	22.000000	1	0	7.2500	S
1	1	0	38.000000	1	0	14.4542	C
2	3	0	26.000000	0	0	7.9250	S
3	1	0	35.000000	1	0	53.1000	S
4	3	1	35.000000	0	0	8.0500	S
...	...	...	...	...	...	...	...
886	2	1	27.000000	0	0	13.0000	S
887	1	0	19.000000	0	0	30.0000	S
888	3	0	29.699118	1	2	23.4500	S
889	1	1	26.000000	0	0	30.0000	C
890	3	1	32.000000	0	0	7.7500	Q

891 rows × 7 columns

Splitting the Data

In [97]:

```
y=data["Pclass"]
```

In [98]:

```
y.head()
```

Out[98]:

```
0    3
1    1
2    3
3    1
4    3
Name: Pclass, dtype: int64
```

Encoding

In [68]:

```
from sklearn.preprocessing import LabelEncoder
```

In [69]:

```
le=LabelEncoder()
```

In [70]:

```
data["Sex"]=le.fit_transform(data["Sex"])
data["Sex"]
```

Out[70]:

```
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 [99]:

```
data.head()
```

Out[99]:

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	3	1	22.0	1	0	7.2500	S
1	1	0	38.0	1	0	14.4542	C
2	3	0	26.0	0	0	7.9250	S
3	1	0	35.0	1	0	53.1000	S
4	3	1	35.0	0	0	8.0500	S

In [100…

```
data["Embarked"]=le.fit_transform(data["Embarked"])
```

In [101…

```
data.head()
```

Out[101]:

	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	3	1	22.0	1	0	7.2500	2
1	1	0	38.0	1	0	14.4542	0
2	3	0	26.0	0	0	7.9250	2
3	1	0	35.0	1	0	53.1000	2
4	3	1	35.0	0	0	8.0500	2

In [103…

```
data["Pclass"].nunique()
```

Out[103]:

```
3
```

In [104…

```
data["Pclass"].unique()
```

Out[104]:

```
array([3, 1, 2], dtype=int64)
```

In [105…

```
data["Sex"].unique()
```

Out[105]:

```
array([1, 0])
```

In [106…

```
data["Embarked"].unique()
```

Out[106]:

```
array([2, 0, 1])
```

Splitting train and test data

In [107…

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(data,y,test_size=0.3,random_state=0)
```

In [108…

```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

Out[108]:

```
((623, 7), (268, 7), (623,), (268,))
```

Feature Scaling

In [109…

```
from sklearn.preprocessing import StandardScaler
```

In [110…

```
sc=StandardScaler()
```

In [111…

```
x_train=sc.fit_transform(x_train)
x_train
```

Out[111]:

```
array([[-1.5325562 ,  0.72592065,  2.25384386, ..., -0.47299765,
         0.67925137,  0.56710989],
       [-1.5325562 , -1.37756104,  2.04763778, ..., -0.47299765,
        -0.26059483, -2.03075381],
       [ 0.84844757,  0.72592065,  0.05765817, ...,  1.93253327,
         2.26045064,  0.56710989],
       ...,
       [ 0.84844757,  0.72592065,  0.05765817, ..., -0.47299765,
        -0.78281017, -0.73182196],
       [ 0.84844757, -1.37756104,  0.70729828, ..., -0.47299765,
        -0.03170555,  0.56710989],
       [-0.34205431,  0.72592065,  0.05765817, ...,  0.72976781,
         1.64661898,  0.56710989]])
```

In [112…

```
x_test=sc.fit_transform(x_test)
x_test
```

Out[112]:

```
array([[ 0.77963055,  0.76537495,  0.12227065, ..., -0.47809977,
        -0.15813988, -1.76531134],
       [ 0.77963055,  0.76537495,  0.12227065, ..., -0.47809977,
        -0.72165412,  0.63014911],
       [ 0.77963055,  0.76537495, -2.14803266, ...,  0.87064484,
         1.03823178, -0.56758111],
       ...,
       [ 0.77963055,  0.76537495, -0.14768767, ..., -0.47809977,
        -0.15847431, -1.76531134],
       [ 0.77963055, -1.30654916, -1.04784292, ..., -0.47809977,
        -0.72607524,  0.63014911],
       [-1.64991582,  0.76537495,  0.12227065, ..., -0.47809977,
         0.92369033, -1.76531134]])
```

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js