1.Import the necessary Libraries

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

2.Import the dastaset

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

```
dataset
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	7
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	5

```
dataset.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	F
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2
1	2	1	1	Cumings, Mrs. John Bradley (Florence	female	38.0	1	0	PC 17599	71.2

```
dataset.head(3)
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp

```
dataset.tail()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibS
				Montvila,			

```
dataset.shape
```

```
(891, 12)
```

```
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
```

```
dataset.describe()
```

	PassengerId	Survived	Pclass	Age	SibS
count	891.000000	891.000000	891.000000	714.000000	891.0000(
mean	446.000000	0.383838	2.308642	29.699118	0.5230(

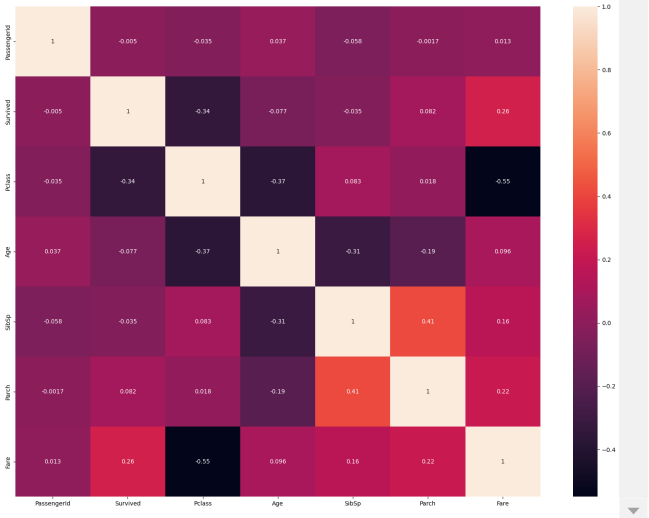
```
corr=dataset.corr()
corr
```

```
<ipython-input-11-f22ca9e9dc13>:1: FutureWarning: The default v
  corr=dataset.corr()
```

	PassengerId	Survived	Pclass	Age	SibS
PassengerId	1.000000	-0.005007	-0.035144	0.036847	-0.0575

```
plt.subplots(figsize=(20,15))
sns.heatmap(corr,annot=True)
```

```
<Axes: >
```




```
dataset.Embarked.value_counts()
```

```
S    644
C    168
Q     77
Name: Embarked, dtype: int64
```

```
dataset.Sex.value_counts()
```

```
male      577
female    314
Name: Sex, dtype: int64
```

```
dataset.head()
```

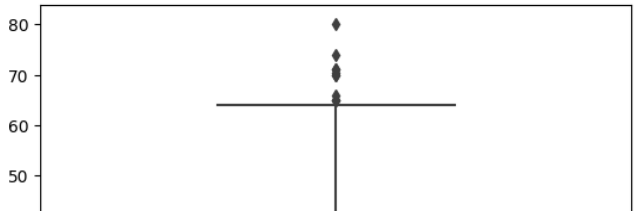
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
				Braund,			

```
dataset.Name.value_counts()
```

```
Braund, Mr. Owen Harris                     1
Boulos, Mr. Hanna                           1
Frolicher-Stehli, Mr. Maxmillian            1
Gilinski, Mr. Eliezer                       1
Murdlin, Mr. Joseph                         1
                                           ..
Kelly, Miss. Anna Katherine "Annie Kate"    1
McCoy, Mr. Bernard                          1
Johnson, Mr. William Cahoone Jr             1
Keane, Miss. Nora A                         1
Dooley, Mr. Patrick                         1
Name: Name, Length: 891, dtype: int64
```

```
sns.boxplot(dataset.Age)
```

```
<Axes: >
```



3.Handling Null Values

```
dataset.isnull().any()
```

```
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
```

```
dataset.isnull().sum()
```

```
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
```

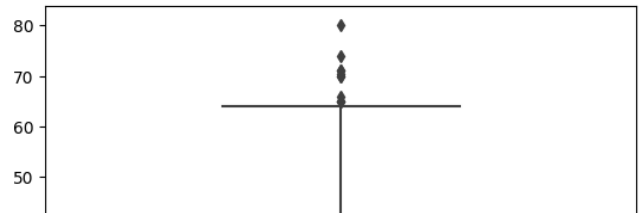
```
dataset.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp
				Braund,			

4.Outliers

```
sns.boxplot(dataset.Age)
```

```
<Axes: >
```



5.Seperate dependent and independent variables

```
x=dataset.iloc[:,3:13]
y=dataset.iloc[:,13:14]
```

```
x.head()
```

	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabi
	Braund,							

```
y.head()
```

0
1

```
dataset.shape
```

```
(891, 12)
```

```
x.shape
```

```
(891, 9)
```

```
y.shape
```

```
(891, 0)
```

6.Encoding

Label encoding on Sex column

```
from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

x["Sex"]=le.fit_transform(x["Sex"])

x["Sex"]
```

```
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: int64
```

```
x["Sex"].value_counts()
```

```
1    577
0    314
Name: Sex, dtype: int64
```

```
x["Sex"].nunique()
```

```
2
```

```
x.head()
```

	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
	Braund,							

```
x.Name.value_counts()
```

```
Braund, Mr. Owen Harris                     1
Boulos, Mr. Hanna                           1
Frolicher-Stehli, Mr. Maxmillian            1
Gilinski, Mr. Eliezer                       1
Murdlin, Mr. Joseph                         1
                                           ..
Kelly, Miss. Anna Katherine "Annie Kate"    1
McCoy, Mr. Bernard                          1
Johnson, Mr. William Cahoone Jr             1
Keane, Miss. Nora A                         1
Dooley, Mr. Patrick                         1
Name: Name, Length: 891, dtype: int64
```

One hot encoding on name column

```
x.shape
```

```
(891, 9)
```

```
Name=pd.get_dummies(x["Name"],drop_first=True)
```

```
Name
```

	Abbott, Mr. Rossmore Edward	Abbott, Mrs. Stanton (Rosa Hunt)	Abelson, Mr. Samuel	Abelson, Mrs. Samuel (Hannah Wizosky)	Adahl, Mr. Mauritz Nils Martin	Adams, Mr. John	Mr
0	0	0	0	0	0	0	

```
x=pd.concat([x,Name],axis=1)
```

```
x.head()
```

	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin

```
x.drop(["Name"],axis=1,inplace=True)
```

```
x.head(10)
```

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	22.0	1	0	A/5 21171	7.2500	NaN	S

```
x.shape
```

```
(891, 898)
```

7.Splitting into trainig and testing set

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

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

```
((623, 898), (268, 898), (623, 0), (268, 0))
```

```
a=[1,2,3,4,5,6]
b=[1,0,1,5,6,3]

for i in range(5):
    a_train,a_test,b_train,b_test=train_test_split(a,b,test_size=0.3,random_state=100)
    print("with random state",a_train)
```

```
with random state [5, 4, 6, 1]
with random state [5, 4, 6, 1]
with random state [5, 4, 6, 1]
with random state [5, 4, 6, 1]
with random state [5, 4, 6, 1]
```

```
a=[1,2,3,4,5,6]    # 4 values for training and 2 for testing
b=[1,0,1,5,6,3]

for i in range(5):
    a_train,a_test,b_train,b_test=train_test_split(a,b,test_size=0.3)
    print("without random state",a_train)
```

```
without random state [3, 2, 4, 6]
without random state [4, 1, 6, 3]
without random state [4, 6, 3, 5]
without random state [4, 1, 3, 6]
without random state [6, 2, 4, 5]
```

8.Feature Scaling

```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()

x_train
```

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
857	1	51.0	0	0	113055	26.5500	E17	S

```
x_test
```

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	Abbott, Mr. Rossmore Edward	Abbott, Mrs. Stanton (Rosa Hunt)	.
495	1	NaN	0	0	2627	14.4583	NaN	C	0	0	
648	1	NaN	0	0	S.O./P.P. 751	7.5500	NaN	S	0	0	
278	1	7.0	4	1	382652	29.1250	NaN	Q	0	0	
31	0	NaN	1	0	PC 17569	146.5208	B78	C	0	0	
255	0	29.0	0	2	2650	15.2458	NaN	C	0	0	
...	...	...	...	...	...	...	...	...	...	...	...
263	1	40.0	0	0	112059	0.0000	B94	S	0	0	
718	1	NaN	0	0	36568	15.5000	NaN	Q	0	0	

✓ 0s completed at 8:08 PM