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


         train_data = pd.read_csv("C:\\firstdata\\tested.csv")
         train_data.head(10)
```

Out[1]:

|	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	0	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	0	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	0	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	1	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S
5	897	0	3	Svensson, Mr. Johan Cervin	male	14.0	0	0	7538	9.2250	NaN	S
6	898	1	3	Connolly, Miss. Kate	female	30.0	0	0	330972	7.6292	NaN	Q
7	899	0	2	Caldwell, Mr. Albert Francis	male	26.0	1	1	248738	29.0000	NaN	S
8	900	1	3	Abrahim, Mrs. Joseph (Sophie Halaut Easu)	female	18.0	0	0	2657	7.2292	NaN	C
9	901	0	3	Davies, Mr. John Samuel	male	21.0	2	0	A/4 48871	24.1500	NaN	S

```
In [2]:  print(len(train_data.index))
```

418

analaysing the data
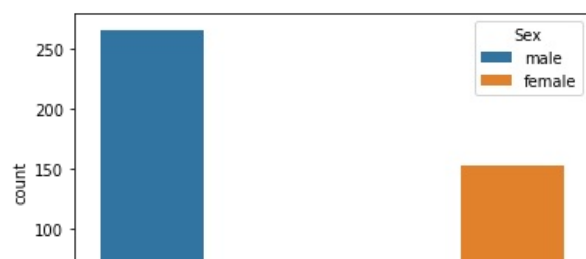
```
In [3]:  sns.countplot(x="Survived",data=train_data)
```

Out[3]: <AxesSubplot:xlabel='Survived', ylabel='count'>
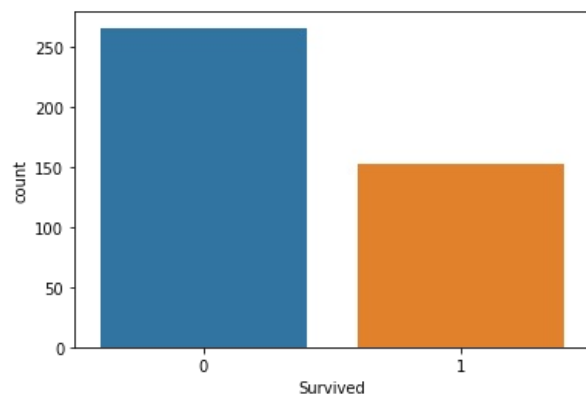

```
In [4]:  sns.countplot(x="Survived",hue="Sex",data=train_data)
```
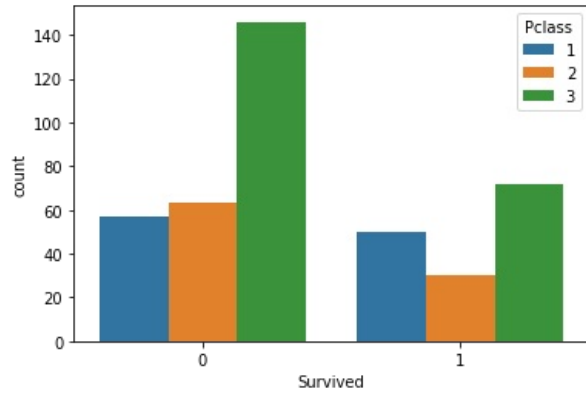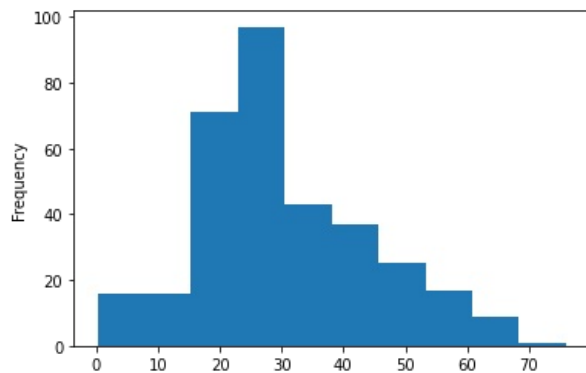
Out[4]: <AxesSubplot:xlabel='Survived', ylabel='count'>

In [5]:
```python
sns.countplot(x="Survived",hue="Pclass",data=train_data)
```

Out[5]: `<AxesSubplot:xlabel='Survived', ylabel='count'>`

In [6]:
```python
train_data["Age"].plot.hist()
```
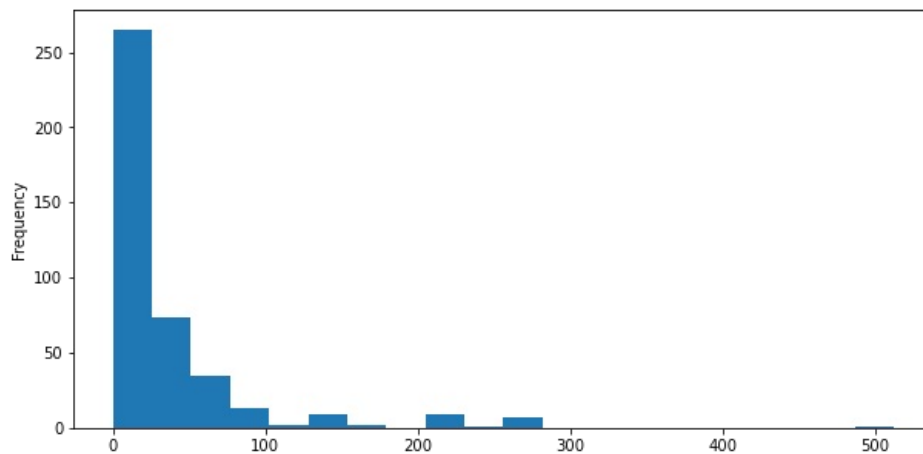
Out[6]: `<AxesSubplot:ylabel='Frequency'>`

In [7]:
```python
train_data["Fare"].plot.hist(bins=20,figsize=(10,5))
```

Out[7]: `<AxesSubplot:ylabel='Frequency'>`


In [8]:
```python
train_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 418 entries, 0 to 417
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  418 non-null    int64
 1   Survived     418 non-null    int64
 2   Pclass       418 non-null    int64
 3   Name         418 non-null    object
 4   Sex          418 non-null    object
 5   Age          332 non-null    float64
 6   SibSp        418 non-null    int64
 7   Parch        418 non-null    int64
 8   Ticket       418 non-null    object
 9   Fare         417 non-null    float64
 10  Cabin        91 non-null     object
 11  Embarked     418 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 39.3+ KB
```

In [9]:
```python
sns.countplot(x="SibSp",data=train_data)
```

Out[9]: <AxesSubplot:xlabel='SibSp', ylabel='count'>
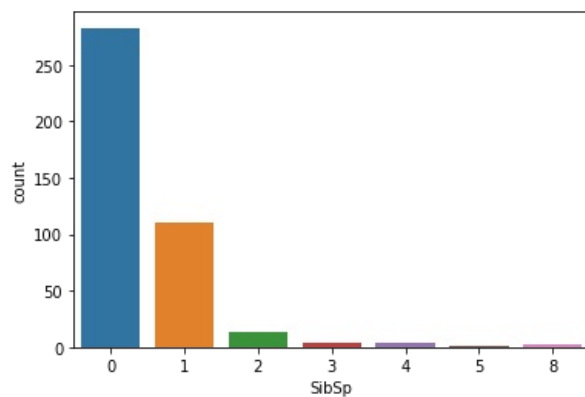
wrangling the data

In [10]:
```python
train_data.isnull().sum()
```

Out[10]:
```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age             86
SibSp            0
Parch            0
Ticket           0
Fare             1
Cabin          327
Embarked         0
dtype: int64
```

In [11]:
```python
sns.heatmap (train_data.isnull(),cmap='viridis')
```

Out[11]: <AxesSubplot:>

```
sns.boxplot(x="Pclass",y="Age",data=train_data)
```

`<AxesSubplot:xlabel='Pclass', ylabel='Age'>`
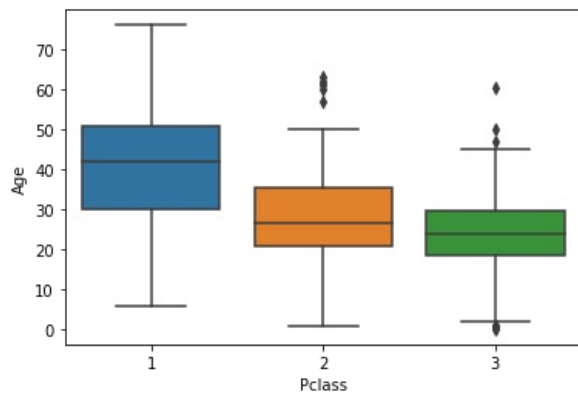
```
train_data.dropna(inplace=True)
```

```
sns.heatmap (train_data.isnull(),cmap='viridis')
```

`<AxesSubplot:>`

```
sns.heatmap (train_data.isnull(),cbar='viridis')
```

`<AxesSubplot:>`

PassengerId Survived Pclass Name Sex Age SibSp Parch Ticket Fare Cabin Embarked

In [16]:
```python
train_data.isnull().sum()
```
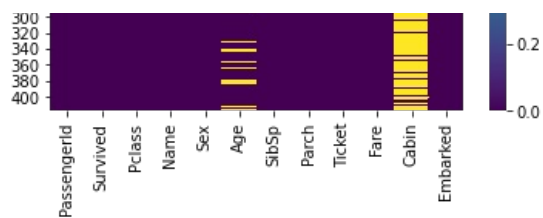
Out[16]:
```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Cabin          0
Embarked       0
dtype: int64
```
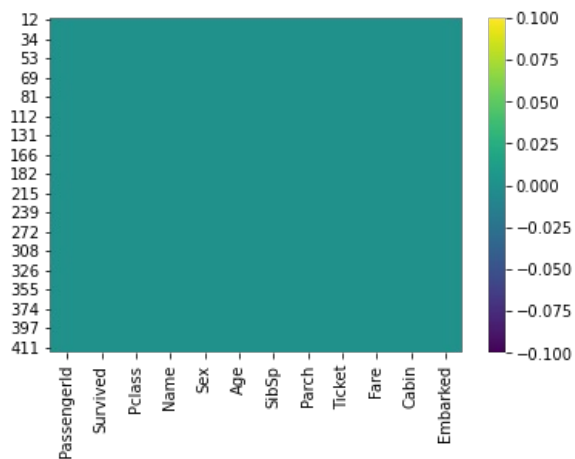
In [17]:
```python
train_data.head(10)
```

Out[17]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
12	904	1	1	Snyder, Mrs. John Pillsbury (Nelle Stevenson)	female	23.0	1	0	21228	82.2667	B45	S
14	906	1	1	Chaffee, Mrs. Herbert Fuller (Carrie Constance...	female	47.0	1	0	W.E.P. 5734	61.1750	E31	S
24	916	1	1	Ryerson, Mrs. Arthur Larned (Emily Maria Borie)	female	48.0	1	3	PC 17608	262.3750	B57 B59 B63 B66	C
26	918	1	1	Ostby, Miss. Helene Ragnhild	female	22.0	0	1	113509	61.9792	B36	C
28	920	0	1	Brady, Mr. John Bertram	male	41.0	0	0	113054	30.5000	A21	S
34	926	0	1	Mock, Mr. Philipp Edmund	male	30.0	1	0	13236	57.7500	C78	C
44	936	1	1	Kimball, Mrs. Edwin Nelson Jr (Gertrude Parsons)	female	45.0	1	0	11753	52.5542	D19	S
46	938	0	1	Chevre, Mr. Paul Romaine	male	45.0	0	0	PC 17594	29.7000	A9	C
48	940	1	1	Bucknell, Mrs. William Robert (Emma Eliza Ward)	female	60.0	0	0	11813	76.2917	D15	C
50	942	0	1	Smith, Mr. Lucien Philip	male	24.0	1	0	13695	60.0000	C31	S

In [18]:
```python
sex=pd.get_dummies(train_data['Sex'],drop_first=True)
sex.head()
```

Out[18]:

	male
12	0
14	0
24	0
26	0
28	1

In [19]:
```python
embark=pd.get_dummies(train_data['Embarked'],drop_first=True)
embark.head()
```

Out[19]:

	Q	S
12	0	1
14	0	1
24	0	0
26	0	0
28	0	1

```
In [20]:    Pclass=pd.get_dummies(train_data['Pclass'],drop_first=True)
            Pclass.head()
```

Out[20]:

	2	3
12	0	0
14	0	0
24	0	0
26	0	0
28	0	0

```
In [21]:    train_data=pd.concat([train_data,Pclass,sex,embark],axis=True)
```

```
In [22]:    train_data.head(5)
```

Out[22]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	2	3	male	Q	S
12	904	1	1	Snyder, Mrs. John Pillsbury (Nelle Stevenson)	female	23.0	1	0	21228	82.2667	B45	S	0	0	0	0	1
14	906	1	1	Chaffee, Mrs. Herbert Fuller (Carrie Constance...	female	47.0	1	0	W.E.P. 5734	61.1750	E31	S	0	0	0	0	1
24	916	1	1	Ryerson, Mrs. Arthur Larned (Emily Maria Borie)	female	48.0	1	3	PC 17608	262.3750	B57 B59 B63 B66	C	0	0	0	0	0
26	918	1	1	Ostby, Miss. Helene Ragnhild	female	22.0	0	1	113509	61.9792	B36	C	0	0	0	0	0
28	920	0	1	Brady, Mr. John Bertram	male	41.0	0	0	113054	30.5000	A21	S	0	0	1	0	1

```
In [23]:    df=train_data.drop(['Sex','Ticket','Name','Embarked','PassengerId','Pclass','Cabin'],axis=1)
            df.head()
```

Out[23]:

	Survived	Age	SibSp	Parch	Fare	2	3	male	Q	S
12	1	23.0	1	0	82.2667	0	0	0	0	1
14	1	47.0	1	0	61.1750	0	0	0	0	1
24	1	48.0	1	3	262.3750	0	0	0	0	0
26	1	22.0	0	1	61.9792	0	0	0	0	0
28	0	41.0	0	0	30.5000	0	0	1	0	1

trainig the data

```
In [24]:    X=df.drop(['Survived'],axis=1)
            y=df['Survived']
            X.head()
            y.head()
```

Out[24]: 12   1
          14   1
          24   1
          26   1
          28   0
          Name: Survived, dtype: int64

```
In [25]:    from sklearn.model_selection import train_test_split
```

```
In [26]:    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=1)
```

```python
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
X_train=sc.fit_transform(X_train)
X_test=sc.transform(X_test)
print(X_train)
```

```
[[ 1.21484321 -0.96673649  0.63529261  4.52744452 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [ 0.14744416 -0.96673649 -0.54944226 -0.67593413 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [ 1.01470588  2.3685044  -0.54944226 -0.86199486 -0.3086067  -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [-0.65310512 -0.96673649 -0.54944226  0.67928235 -0.3086067  -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [ 0.14744416 -0.96673649 -0.54944226 -0.85258107 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [-1.1200922  -0.96673649  0.63529261 -0.22564171 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [ 1.41498053  0.70088395  3.00476234  1.75919097 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [ 0.48100636  0.70088395 -0.54944226  1.37322572 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [ 1.14813076  0.70088395 -0.54944226  0.47610071 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [ 1.01470588  0.70088395  0.63529261 -0.11110694 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [ 1.61511785  0.70088395 -0.54944226 -0.31322647 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [-0.71981756 -0.96673649 -0.54944226  1.30959074 -0.3086067  -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [-0.18611804  0.70088395  0.63529261 -0.22564171 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [ 0.08073172  0.70088395 -0.54944226 -0.57224404 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [ 0.2141566   0.70088395 -0.54944226 -0.53260425 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [-0.58639268 -0.96673649 -0.54944226  0.34296983 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [-1.85392905  2.3685044   0.63529261 -0.71469679  3.24037035 -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [-0.65310512  0.70088395  1.82002747  0.53179932 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [ 1.34826809  0.70088395  4.1894972   1.76611288 -0.3086067  -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [ 0.68114368 -0.96673649 -0.54944226 -0.85867235 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [ 0.74785612 -0.96673649  0.63529261 -0.71026677 -0.3086067  -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [ 1.54840541  0.70088395 -0.54944226  1.30959074 -0.3086067  -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [-0.18611804  0.70088395 -0.54944226 -0.14986959 -0.3086067  -0.12126781
  -0.98561076  8.24621125 -0.95742711]
 [ 0.41429392 -0.96673649 -0.54944226 -0.3133184  -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [ 0.08073172 -0.96673649 -0.54944226 -0.80883465 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [ 0.68114368  0.70088395  0.63529261  1.19574815 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [-0.85324244  0.70088395  1.82002747 -0.57072123 -0.3086067  -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [ 0.61443124 -0.96673649 -0.54944226 -1.14662347 -0.3086067  -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [-1.18680464 -0.96673649  0.63529261 -0.46020115 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [ 1.01470588 -0.96673649 -0.54944226  0.35552118 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [ 0.881281   -0.96673649 -0.54944226 -0.83098474 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [-0.91995488  0.70088395 -0.54944226  0.36821207 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [-2.25420369 -0.96673649  1.82002747  0.34296983 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [-1.18680464 -0.96673649 -0.54944226 -1.03033552  3.24037035 -0.12126781
  -0.98561076 -0.12126781  1.04446594]
 [ 0.34758148  0.70088395  0.63529261  0.34296983 -0.3086067  -0.12126781
   1.01459931 -0.12126781 -0.95742711]
 [ 0.5477188   0.70088395  0.63529261 -0.26948005 -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
 [-1.42029818 -0.96673649 -0.54944226 -1.00264791  3.24037035 -0.12126781
   1.01459931 -0.12126781  1.04446594]
 [ 1.34826809 -0.96673649 -0.54944226 -0.3016896  -0.3086067  -0.12126781
  -0.98561076 -0.12126781 -0.95742711]
```

```
[-0.31954292 -0.96673649 -0.54944226  1.19574815 -0.3086067  -0.12126781
 -0.98561076 -0.12126781 -0.95742711]
[-1.1200922   0.70088395 -0.54944226 -0.23551622 -0.3086067  -0.12126781
 -0.98561076 -0.12126781  1.04446594]
[-1.1200922  -0.96673649 -0.54944226 -0.11110694 -0.3086067  -0.12126781
  1.01459931 -0.12126781  1.04446594]
[ 0.881281    0.70088395  0.63529261 -0.24003926 -0.3086067  -0.12126781
  1.01459931 -0.12126781  1.04446594]
[ 2.41566713  0.70088395 -0.54944226 -0.27335632 -0.3086067  -0.12126781
 -0.98561076 -0.12126781  1.04446594]
[ 0.34758148 -0.96673649 -0.54944226 -0.81769469 -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[-1.32022952 -0.96673649 -0.54944226 -0.99309568  3.24037035 -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[-0.85324244  0.70088395  0.63529261  1.59468007 -0.3086067  -0.12126781
 -0.98561076 -0.12126781 -0.95742711]
[-1.78721661  2.3685044   1.82002747  1.75919097 -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[ 0.68114368  0.70088395  0.63529261  1.19574815 -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[-0.58639268 -0.96673649 -0.54944226 -0.83056942 -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[-0.31954292  0.70088395 -0.54944226 -0.50703973 -0.3086067  -0.12126781
 -0.98561076 -0.12126781 -0.95742711]
[-0.25283048 -0.96673649 -0.54944226 -0.3133184  -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[-0.98666732 -0.96673649 -0.54944226 -1.06189939 -0.3086067   8.24621125
  1.01459931 -0.12126781  1.04446594]
[-0.48632402 -0.96673649 -0.54944226  1.19574815 -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[ 1.81525517  0.70088395 -0.54944226  1.30959074 -0.3086067  -0.12126781
  1.01459931 -0.12126781  1.04446594]
[ 0.5477188   0.70088395 -0.54944226  0.032038   -0.3086067  -0.12126781
 -0.98561076 -0.12126781 -0.95742711]
[-0.25283048 -0.96673649 -0.54944226 -0.79577496 -0.3086067  -0.12126781
 -0.98561076 -0.12126781 -0.95742711]
[ 0.34758148  0.70088395 -0.54944226 -0.56458344 -0.3086067  -0.12126781
 -0.98561076 -0.12126781  1.04446594]
[ 1.28155565  2.3685044  -0.54944226 -0.57648911 -0.3086067  -0.12126781
 -0.98561076 -0.12126781  1.04446594]
[ 1.01470588 -0.96673649 -0.54944226 -0.59287131 -0.3086067  -0.12126781
  1.01459931 -0.12126781  1.04446594]
[ 0.48100636  0.70088395 -0.54944226 -0.4691077  -0.3086067  -0.12126781
 -0.98561076 -0.12126781  1.04446594]
[-0.75317378 -0.96673649 -0.54944226 -0.83961441 -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[-0.91995488 -0.96673649 -0.54944226 -1.00264791  3.24037035 -0.12126781
  1.01459931 -0.12126781  1.04446594]
[-1.4536544   0.70088395 -0.54944226 -0.48212088 -0.3086067  -0.12126781
 -0.98561076 -0.12126781  1.04446594]
[-0.65310512  0.70088395 -0.54944226 -0.50703973 -0.3086067  -0.12126781
  1.01459931 -0.12126781 -0.95742711]
[ 0.94799344  0.70088395  0.63529261 -0.24003926 -0.3086067  -0.12126781
 -0.98561076 -0.12126781  1.04446594]
[-1.05337976  0.70088395 -0.54944226 -0.48212088 -0.3086067  -0.12126781
  1.01459931 -0.12126781  1.04446594]
[-0.98666732  0.70088395 -0.54944226 -0.53260425 -0.3086067  -0.12126781
 -0.98561076 -0.12126781 -0.95742711]
[-0.25283048 -0.96673649 -0.54944226  1.75919097 -0.3086067  -0.12126781
 -0.98561076 -0.12126781 -0.95742711]
[-0.25283048 -0.96673649  3.00476234 -0.71469679  3.24037035 -0.12126781
 -0.98561076 -0.12126781  1.04446594]]
```

In [48]:
```python
from sklearn.linear_model import LogisticRegression
logmodel=LogisticRegression()
```

In [49]:
```python
logmodel.fit(X_train,y_train)
```

Out[49]: LogisticRegression()

In [50]:
```python
predictions=logmodel.predict(X_test)
```

In [51]:
```python
from sklearn.metrics import classification_report
```

In [52]:
```
classification_report(y_test,predictions)
```

```
classification_report(y_test,predictions)
```

Out[52]: '   precision   recall f1-score   support\n\n   0   1.00   1.00   1.00   9\n   1   1.00   1.00   1.00   9\n\n   accuracy   1.00   18\n   macro avg   1.00   1.00   1.00   18\nweighted avg   1.00   1.00   1.00   18\n'

In [53]:
```python
from sklearn.metrics import confusion_matrix
```

In [54]:
```python
confusion_matrix(y_test,predictions)
```

Out[54]: array([[9, 0],
       [0, 9]], dtype=int64)

In [ ]:

In [ ]: