In [60]:
```
#AI PROJECT
```

In [61]:
```
#loading datasets
```

In [62]:
```
import pandas as pd
```

In [63]:
```
heart = pd.read_csv(r"C:\Users\zeesh\OneDrive\Desktop\heart.csv")
```

In [64]:
```
heart_data=heart.drop(["target"],axis=1)
```

In [65]:
```
heart_data.head()
```

Out[65]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal
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

In [66]:
```
heart_data.shape
```

Out[66]:
```
(1025, 13)
```

In [67]:
```
heart_data.columns
```

Out[67]:
```
Index(['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalac
h',
       'exang', 'oldpeak', 'slope', 'ca', 'thal'],
      dtype='object')
```

In [68]:
```
heart.isnull().sum()
```

Out[68]:
```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

In [69]:
```
#train test split
```

In [70]:
```
from sklearn.model_selection import train_test_split
```

In [71]:
```
X_train, X_test, Y_train, Y_test = train_test_split(heart_data, heart.targ
```

In [72]:
```
print(X_train)
```

```
     age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak
\
923   54    1   0       140   239    0        1      160      0      1.2
11    43    0   0       132   341    1        0      136      1      3.0
477   57    1   2       128   229    0        0      150      0      0.4
263   55    0   1       132   342    0        1      166      0      1.2
569   42    1   2       120   240    1        1      194      0      0.8
..   ...  ...  ..       ...   ...  ...      ...      ...    ...      ...
299   52    1   1       120   325    0        1      172      0      0.2
233   41    0   1       126   306    0        1      163      0      0.0
772   62    0   0       150   244    0        1      154      1      1.4
595   61    1   0       148   203    0        1      161      0      0.0
820   61    1   0       140   207    0        0      138      1      1.9

     slope  ca  thal
923      2   0     2
11       1   0     3
477      1   1     3
263      2   0     2
569      0   0     3
..     ...  ..   ...
299      2   0     2
233      2   0     2
772      1   0     2
595      2   1     3
820      2   1     3

[820 rows x 13 columns]
```

In [73]:
```
from sklearn import preprocessing
```

In [74]:
```
print(X_train)
```

```
     age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak
\
923   54    1   0       140   239    0        1      160      0      1.2
11    43    0   0       132   341    1        0      136      1      3.0
477   57    1   2       128   229    0        0      150      0      0.4
263   55    0   1       132   342    0        1      166      0      1.2
569   42    1   2       120   240    1        1      194      0      0.8
..   ...  ...  ..       ...   ...  ...      ...      ...    ...      ...
299   52    1   1       120   325    0        1      172      0      0.2
233   41    0   1       126   306    0        1      163      0      0.0
772   62    0   0       150   244    0        1      154      1      1.4
595   61    1   0       148   203    0        1      161      0      0.0
820   61    1   0       140   207    0        0      138      1      1.9

     slope  ca  thal
923      2   0     2
11       1   0     3
477      1   1     3
263      2   0     2
569      0   0     3
..     ...  ..   ...
299      2   0     2
233      2   0     2
772      1   0     2
595      2   1     3
820      2   1     3

[820 rows x 13 columns]
```

In [75]:
```
#import Gaussian Naive Bayes model
```

In [84]:
```
from sklearn.naive_bayes import GaussianNB
model= GaussianNB()
```

In [85]:
```
model.fit(X_train, Y_train)
```

Out[85]:
```
GaussianNB()
```

In [86]:
```
#predict response for test dataset
y_predict = model.predict(X_test)
y_predictS = model.predict([[52,1,0,125,212,0,1,168,0,1,2,2,3]])
print(y_predictS)
```

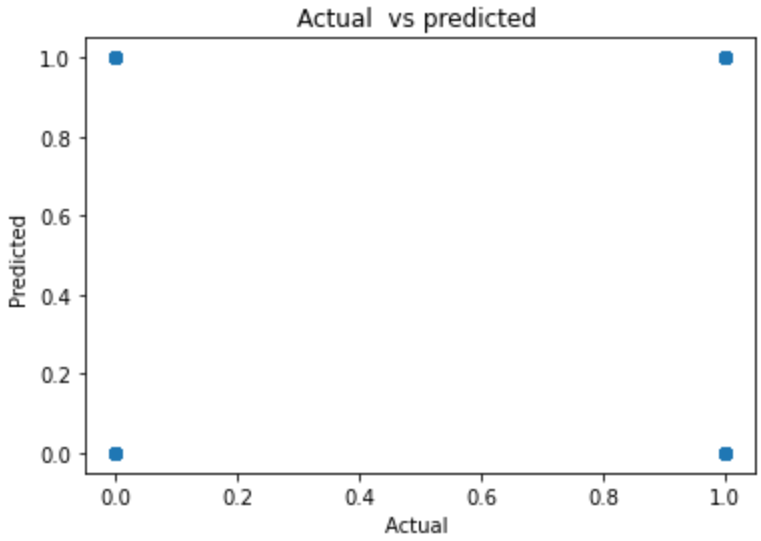
```
[0]
```

In [87]:
```
from sklearn import metrics
print("Accuracy: ", metrics.accuracy_score(Y_test, y_predict))
```

```
Accuracy:  0.824390243902439
```

In [89]:
```
from matplotlib import pyplot as plt
```

In [90]:
```
plt.scatter(Y_test, y_predict)
plt.xlabel("Actual ")
plt.ylabel("Predicted ")
plt.title("Actual  vs predicted ")
plt.show()
```



In [54]:
```
print(heart_data.value_counts())
```

```
age  sex  cp  trestbps  chol  fbs  restecg  thalach  exang  oldpeak  slop
e  ca  thal
38   1    2   138       175   0    1        173      0      0.0      2
4   2       8
61   1    0   148       203   0    1        161      0      0.0      2
1   3       4
                138       166   0    0        125      1      3.6      1
1   2       4
52   1    0   112       230   0    1        160      0      0.0      2
1   2       4
                125       212   0    1        168      0      1.0      2
2   3       4

..
53   0    0   130       264   0    0        143      0      0.4      1
0   2       3
                138       234   0    0        160      0      0.0      2
0   2       3
             2  128       216   0    0        115      0      0.0      2
0   0       3
          1    0   142       226   0    0        111      1      0.0      2
0   3       3
77   1    0   125       304   0    0        162      1      0.0      2
3   2       3
Length: 302, dtype: int64
```

In [ ]: