Impement support vector machine on "Iris" dataset using any two attributes as input and compute the following

input and compute the following

(1) accurecy score (2) confusion matrix

In [51]: import pandas as pd

```
import numpy as np
       from sklearn.preprocessing import LabelEncoder
In [38]: df=pd.read csv('C:/satish (coding)/csv files/iris.data.csv')
In [39]: df.head()
Out[39]:
         5.1 3.5 1.4 0.2 Iris-setosa
       0 4.9 3.0 1.4 0.2 Iris-setosa
       1 4.7 3.2 1.3 0.2 Iris-setosa
       2 4.6 3.1 1.5 0.2 Iris-setosa
       3 5.0 3.6 1.4 0.2 Iris-setosa
       4 5.4 3.9 1.7 0.4 Iris-setosa
In [40]: df['Iris-setosa'].value_counts()
Out[40]: Iris-versicolor
       Iris-virginica
                     50
                     49
       Iris-setosa
       Name: Iris-setosa, dtype: int64
In [41]: lr=LabelEncoder()
       x=df.iloc[:,[0,1]]
       y=df.iloc[:,4]
       y=lr.fit_transform(y)
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,
            In [42]: from sklearn.model selection import train test split
       xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.2)
In [43]: print('length of train data=',len(xtrain))
       print('length of test data=',len(xtest))
       length of train data= 119
       length of test data= 30
In [44]: from sklearn.svm import SVC
In [45]: model=SVC()
       model.fit(xtrain,ytrain)
Out[45]:  vC
       sv¢()
In [46]: pre=model.predict(xtest)
In [47]: import sklearn.metrics as mc
       mc.confusion_matrix(ytest,pre)
Out[47]: array([[10, 0, 0],
            [ 0, 11, 3],
            [ 0, 2, 4]], dtype=int64)
```

In [48]:	mc.accuracy_score(ytest,pre)
Out[48]:	0.8333333333334
In []:	
In []:	