1. Import iris.csv file and plot the species "Iris-versicolor" and "Iris-virginica" 2. Create two 4x4 matrices and add them. 3. Write programs to demonstrate the dot and cross products. 4. Create a 3x2 matrix and print the sum of its elements using for loops. 5. Create a 2x3 matrix and fill it with random numbers. 6. Write a python program to multiply two matrices. 7. Using Matrix inversion, solve the following system of equation

4x1 □ 2x2 + x3 = 11 □2x1 + 4x2 □ 2x3 = □16 x1 □ 2x2 + 4x3 = 17 .

In [2]:

```
import random
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
df=pd.read_csv("iris.csv",sep=",")
type(df)
```

Out[2]:

```
pandas.core.frame.DataFrame
```

In [2]:

```
sepleniris=df[df.Species=="Iris-versicolor"]["SepalLength"]
sepwidiris=df[df.Species=="Iris-versicolor"]["SepalWidth"]
petleniris=df[df.Species=="Iris-versicolor"]["PetalLength"]
petwidiris=df[df.Species=="Iris-versicolor"]["PetalWidth"]
```

In [3]:

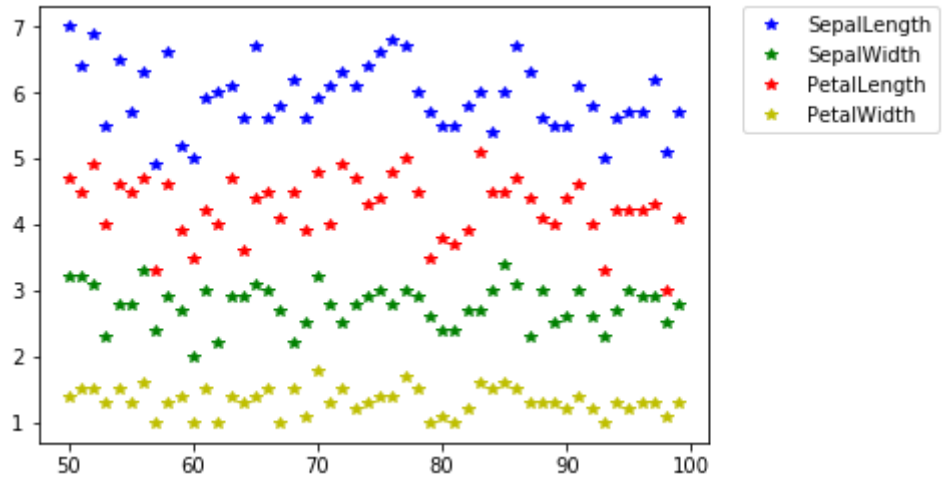
```
plt.plot(sepleniris,"b*")
plt.legend(["sepleniris"])
plt.plot(sepwidiris,"g*")
plt.legend(["sepwidiris"])
plt.plot(petleniris,"r*")
plt.legend(["petleniris"])
plt.plot(petwidiris,"y*")
plt.legend(["petwidiris"])
plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0)
```

Out[3]:

```
<matplotlib.legend.Legend at 0x2110f45f2b0>
```

In [4]:

```
plt.show()
```



In [5]:

```
sepleniris=df[df.Species=="Iris-virginica"]["SepalLength"]
sepwidiris=df[df.Species=="Iris-virginica"]["SepalWidth"]
petleniris=df[df.Species=="Iris-virginica"]["PetalLength"]
petwidiris=df[df.Species=="Iris-virginica"]["PetalWidth"]
```

In [6]:

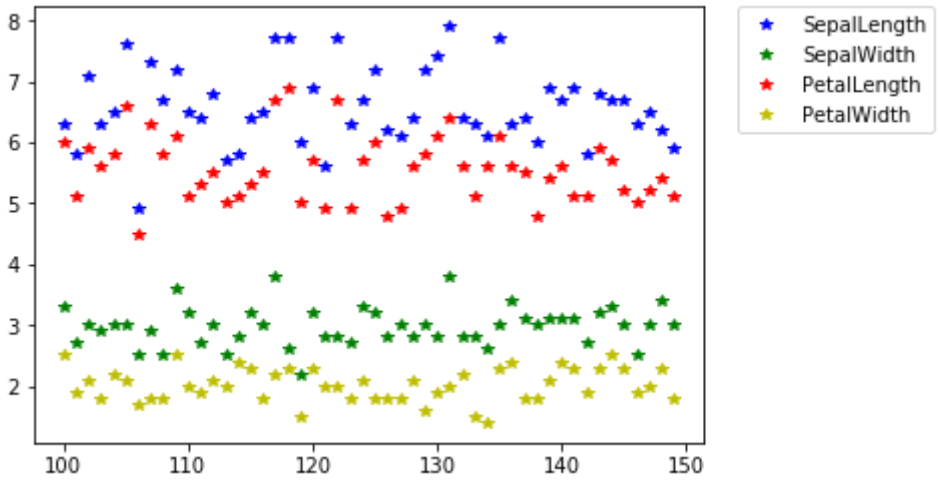
```
plt.plot(sepleniris,"b*")
plt.legend(["sepleniris"])
plt.plot(sepwidiris,"g*")
plt.legend(["sepwidiris"])
plt.plot(petleniris,"r*")
plt.legend(["petleniris"])
plt.plot(petwidiris,"y*")
plt.legend(["petwidiris"])
plt.legend(bbox_to_anchor=(1.05,1),loc=2,borderaxespad=0)
```

Out[6]:

```
<matplotlib.legend.Legend at 0x2110f56ac50>
```

In [7]:

```
plt.show()
```



In [9]:

```
x=np.array([[1,2],[4,5]])
y=np.array([[1,3],[5,6]])
x
```

Out[9]:

```
array([[1, 2],
       [4, 5]])
```

In [6]:

```
A=np.array([[1,5,6,7],[8,9,1,0],[2,3,4,5],[4,5,2,3]])
B=np.array([[1,2,3,4],[5,6,7,8],[9,10,11,12],[13,14,15,16]])
A
```

Out[6]:

```
array([[1, 5, 6, 7],
       [8, 9, 1, 0],
       [2, 3, 4, 5],
       [4, 5, 2, 3]])
```

In [7]:

```
B
```

Out[7]:

```
array([[ 1,  2,  3,  4],
       [ 5,  6,  7,  8],
       [ 9, 10, 11, 12],
       [13, 14, 15, 16]])
```

In [8]:

```
print(np.add(A,B))
```

```
[[ 2  7  9 11]
 [13 15  8  8]
 [11 13 15 17]
 [17 19 17 19]]
```

In [9]:

```
D=np.array([[2,4],[6,8],[10,12]])
D
```

Out[9]:

```
array([[ 2,  4],
       [ 6,  8],
       [10, 12]])
```

In [17]:

```
E=np.random.rand(2,3)
```

In [18]:

```
E
```

Out[18]:

```
array([[ 0.58317298,  0.33877062,  0.41111488],
       [ 0.34777596,  0.07183252,  0.89929276]])
```

In [19]:

```
A=([[1,3],[4,9]])
B=([[4,5],[6,8]])
A_B=np.dot(A,B)
print(A_B)
```

```
[[22 29]
 [70 92]]
```

In [3]:

```
x=np.array([[1,2],[3,4],[5,6]])
print("The 1st matrix is: ")
print(x)
print("\n")
y=np.array([[9,8],[7,6],[5,4]])
print("The 2nd matrix is: ")
print(y)
print("\n")
result=np.array([[0,0],[0,0],[0,0],[0,0]])
for i in range(len(x)):
    for j in range(len(x[0])):
        result[i][j]=x[i][j]+y[i][j]
print("The sum of the elements using for loop is: ")
for r in result:
    print(r)
```

```
The 1st matrix is:
[[1 2]
 [3 4]
 [5 6]]


The 2nd matrix is:
[[9 8]
 [7 6]
 [5 4]]


The sum of the elements using for loop is:
[10 10]
[10 10]
[10 10]
[0 0]
```

In [23]:

```
F=np.array([[4,-2,1],[-2,4,-2],[1,-2,4]])
F
```

Out[23]:

```
array([[ 4, -2,  1],
       [-2,  4, -2],
       [ 1, -2,  4]])
```

In [25]:

```
G=np.linalg.inv(F)
G
```

Out[25]:

```
array([[ 0.33333333,  0.16666667,  0.        ],
       [ 0.16666667,  0.41666667,  0.16666667],
       [ 0.        ,  0.16666667,  0.33333333]])
```