import numpy as np

b=[1,2,3]

a=np.array(b)

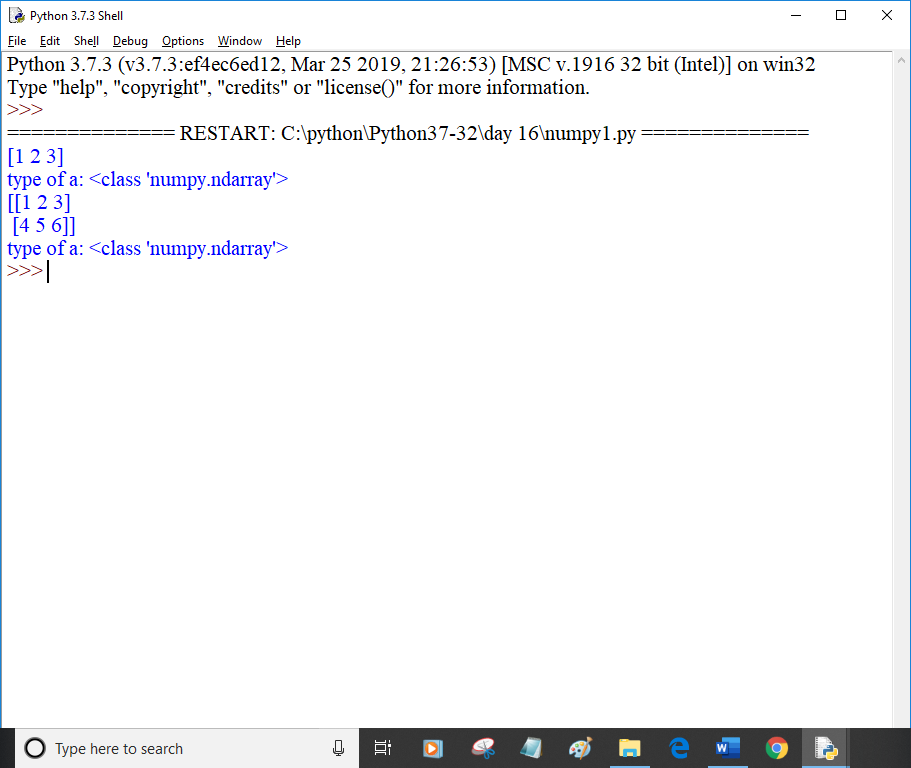
print(a)

print("type of a:",type(a))

a=np.array([(1,2,3),(4,5,6)],)

print(a)

print("type of a:",type(a))



import numpy as np

import sys

S=list(range(1000))

print("Memory occupied by list:",sys.getsizeof(999)\*len(S))

D=np.arange(1000)

print("Memory occupied by Numpy Array:",D.itemsize \* D.size)



import numpy as np

import time

import sys

SIZE = 10000000

L1 = list(range(SIZE))

L2 = list(range(SIZE))

start = time.time()

result = [(x+y) for x,y in zip(L1,L2)]

print("Time taken by list operation:",(time.time()-start)\*100000)

A1 = np.arange(SIZE)

A2 = np.arange(SIZE)

start = time.time()

result = A1+A2

print("Time taken by numpy array operation:",(time.time()-start)\*100000)



import numpy as np

a= np.array([1,2,3])

print(a.ndim)

a = np.array([(1,2,3),(2,3,4)])

print(a.ndim)

print(a.itemsize)

print(a.dtype)

a = np.array([1,2,3])

print(a.size)

a = np.array([1,2,3,4,5,6,7])

print(a.size)

a = np.array([1,2,3])

print(a.shape)

a = np.array([(1,2,3),(2,3,4)])

print(a.shape)



import numpy as np

a = np.array([(1,2,3,4),(3,4,5,6)])

print(a)

a = a.reshape(4,2)

print(a)

#slicing:Extracting particular set of elements from the array

a = np.array([(1,2,3,4),(13,14,15,16)])

print(a[0,2])

print(a[1,2])

print(a[0:,3])

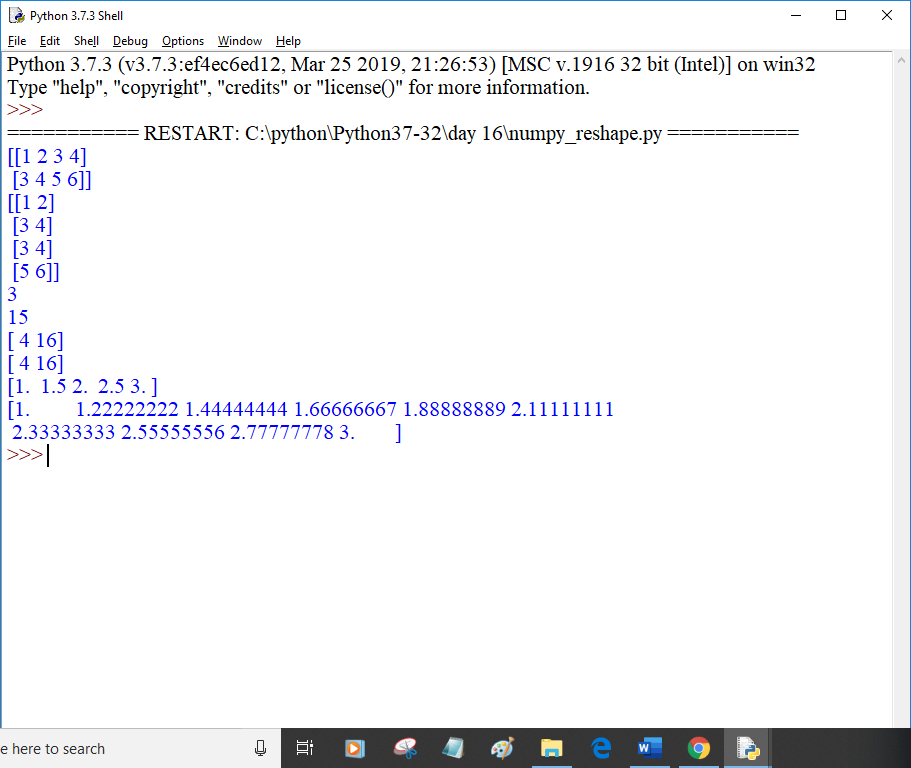
print(a[0:2,3])

a = np.linspace(1,3,5)

print(a)

a = np.linspace(1,3,10)

print(a)



import numpy as np

a = np.array([(1,2,3,4),(3,4,5,6)])

print(a)

a = a.reshape(4,2)

print(a)

#slicing:Extracting particular set of elements from the array

a = np.array([(1,2,3,4),(13,14,15,16)])

print(a[0,2])

print(a[1,2])

print(a[0:,3])

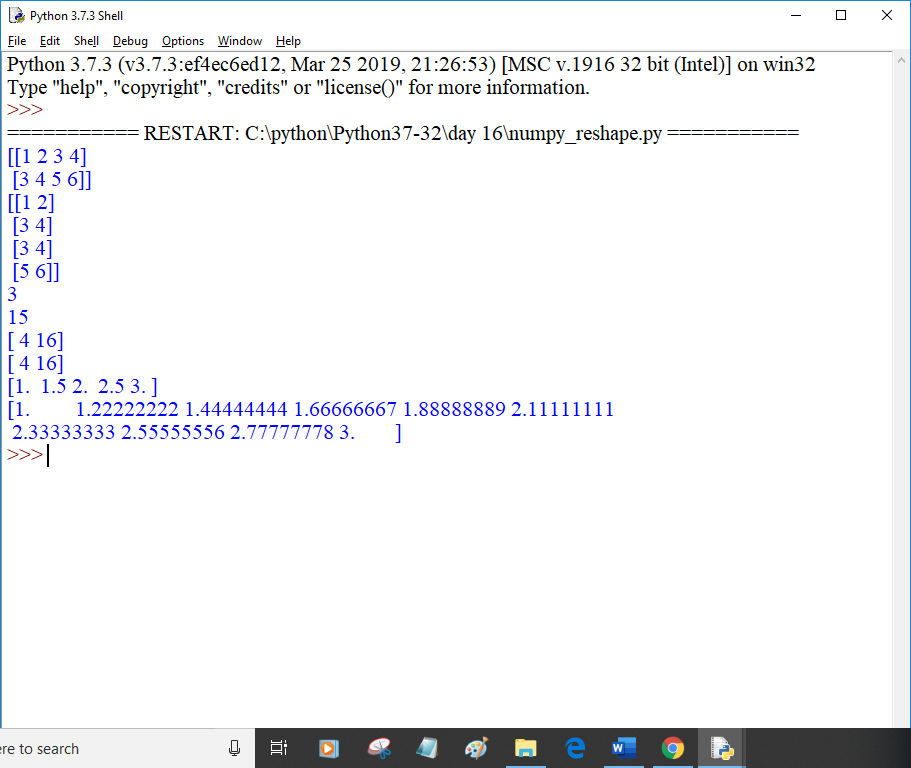
print(a[0:2,3])

a = np.linspace(1,3,5)

print(a)

a = np.linspace(1,3,10)

print(a)



import numpy as np

a = np.array([1,2,3,4])

print("Min:",a.min())

print("Max:",a.max())

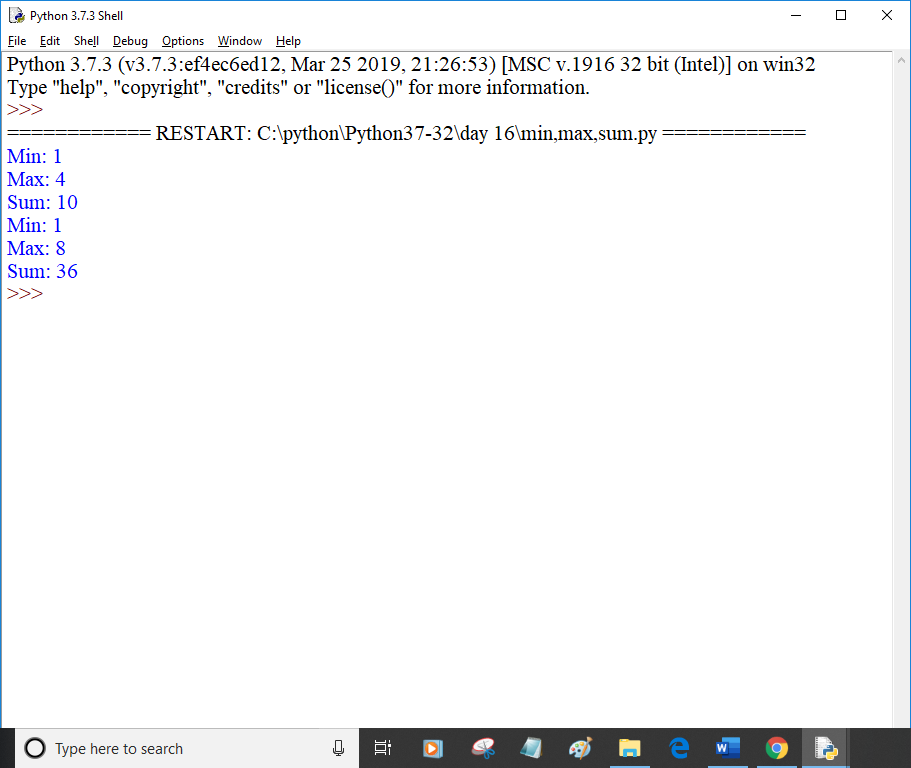
print("Sum:",a.sum())

a = np.array([(1,2,3,4),(5,6,7,8)])

print("Min:",a.min())

print("Max:",a.max())

print("Sum:",a.sum())



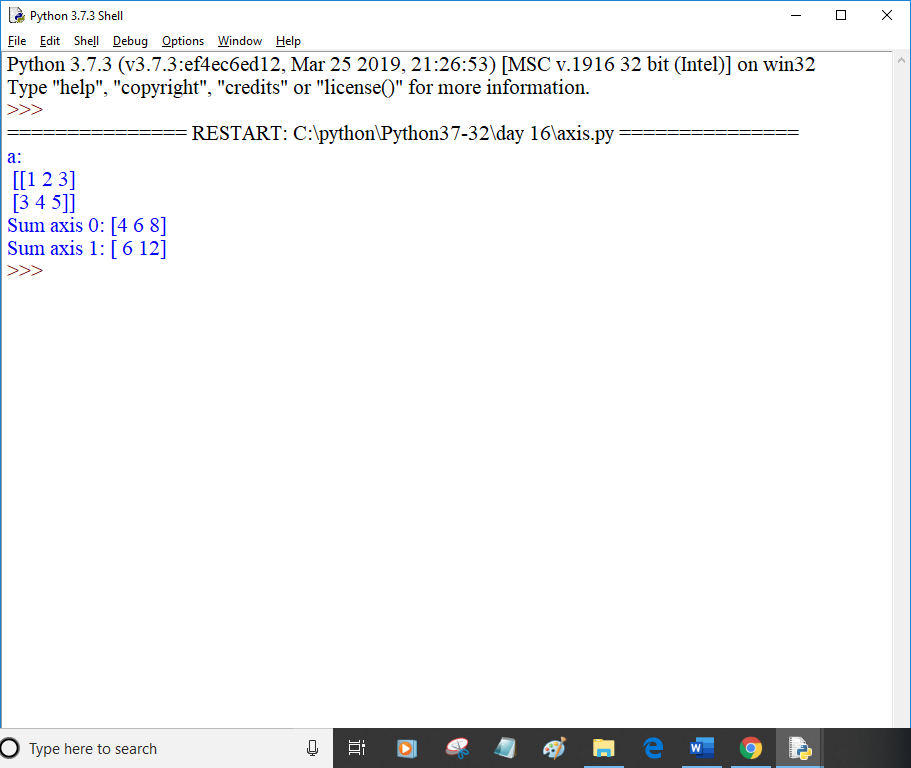
import numpy as np

a = np.array([(1,2,3),(3,4,5)])

print("a:\n",a)

print("Sum axis 0:",a.sum(axis=0))

print("Sum axis 1:",a.sum(axis=1))



import numpy as np

a = np.array([(1,2,3),(3,4,5)])

print("sqrt:",np.sqrt(a))

print("mean:",np.mean(a))

print("std:",np.std(a))



import numpy as np

a = np.array([(1,2,3),(3,4,5)])

b = np.array([(1,2,3),(3,4,5)])

print(a+b)

print(a-b)

print(a\*b)

print(a/b)

#vertical stacking

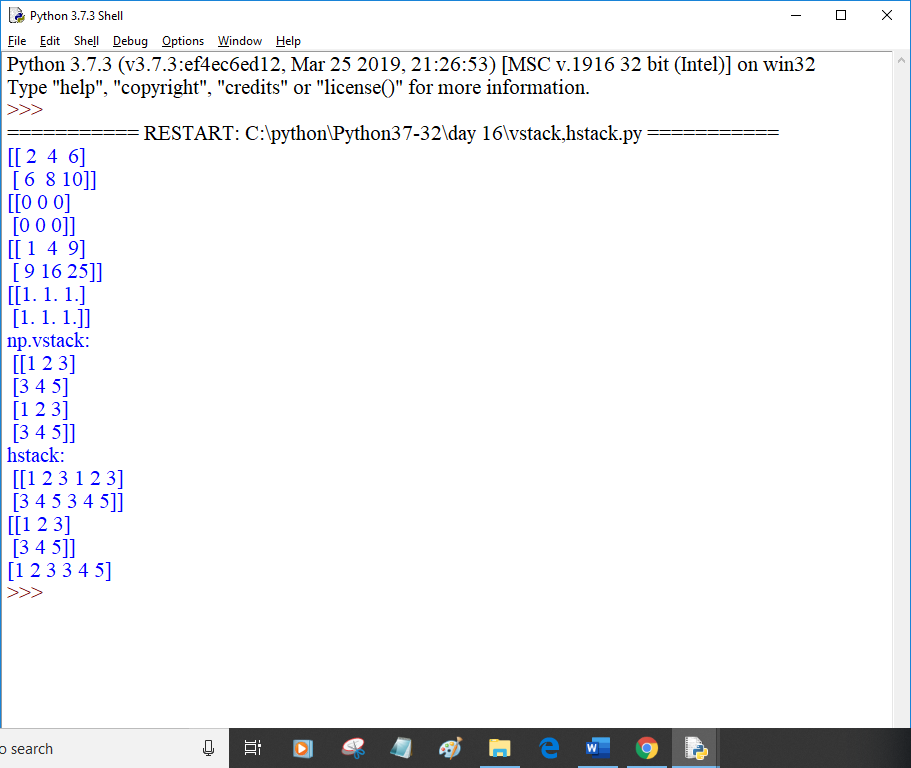
print("np.vstack:\n",np.vstack((a,b)))

#horizontal stacking

print("hstack:\n",np.hstack((a,b)))

print(a)

print(a.ravel())



import numpy as np

import matplotlib.pyplot as plt

x = np.arange(-1,3\*np.pi,0.1)

print("X:\n",x)

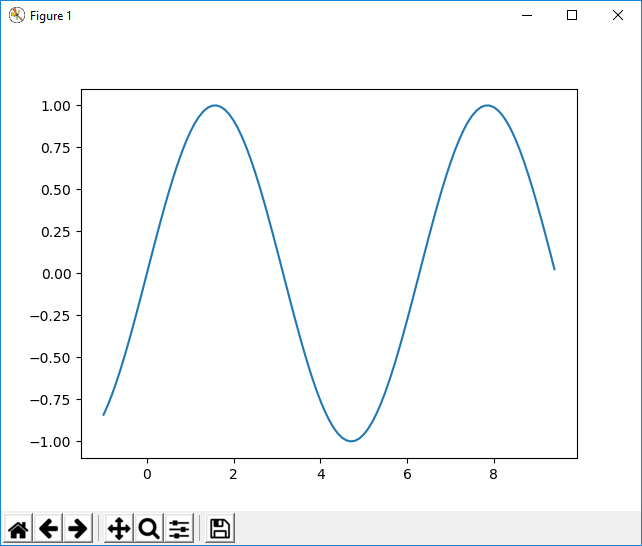
y = np.sin(x)

print("Y:\n",y)

plt.plot(x,y)

plt.show()





import numpy as np

import matplotlib.pyplot as plt

x = np.array([1,2,3,4,5,6,7,8,9,10])

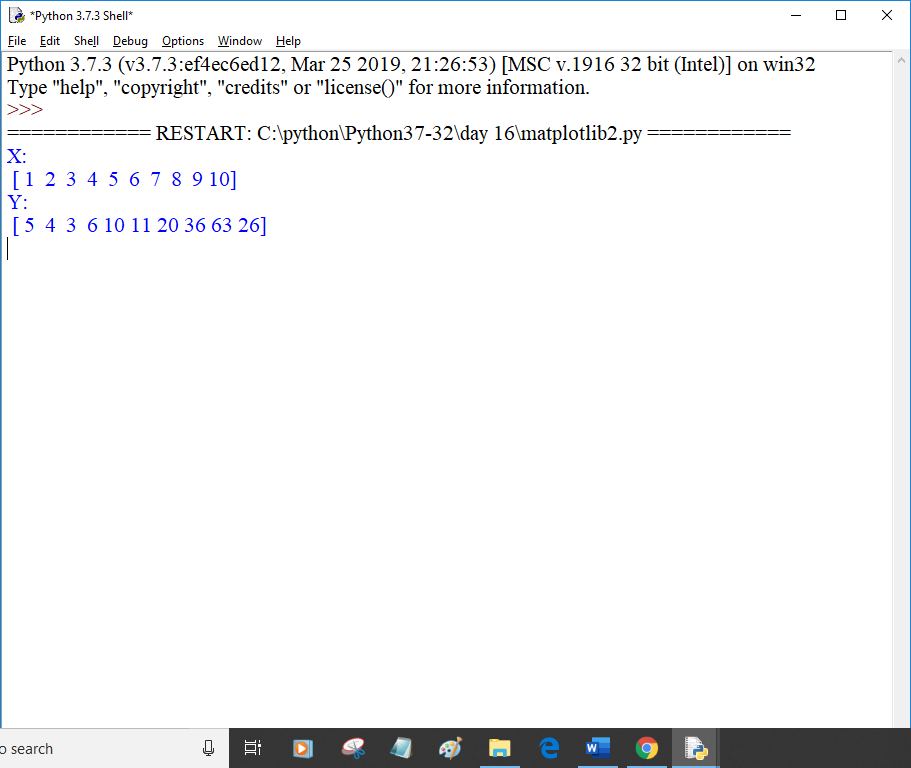
print("X:\n",x)

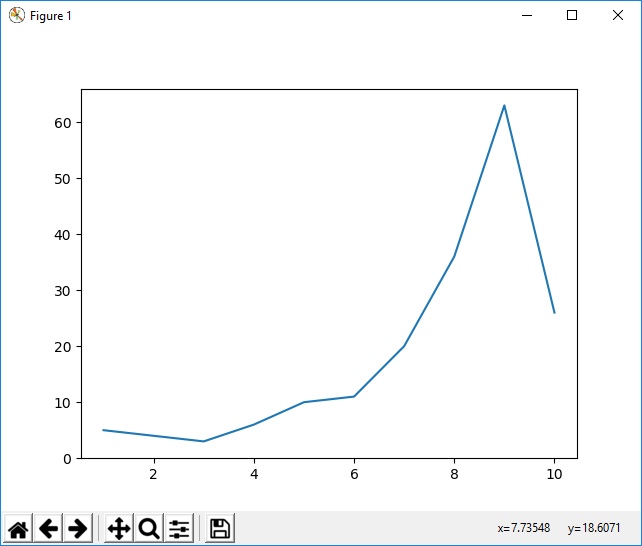
y = np.array([5,4,3,6,10,11,20,36,63,26])

print("Y:\n",y)

plt.plot(x,y)

plt.show()





import matplotlib.pyplot as plt

import numpy as np

objects = ('Python','C++','Java','Perl','Scala','Lisp')

x\_pos = np.arange(len(objects))

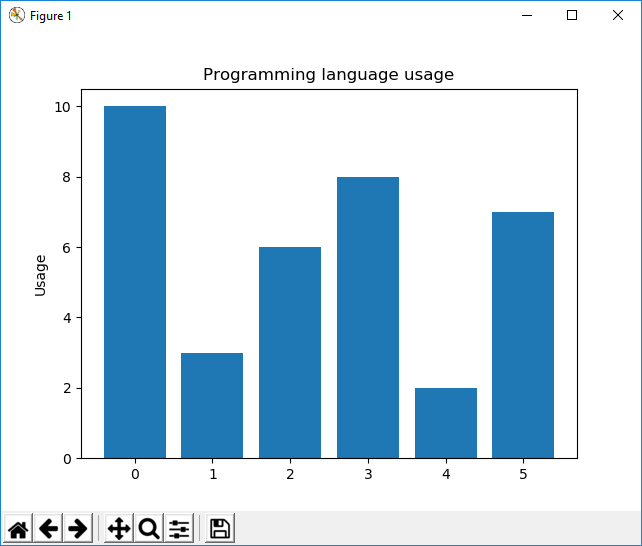
performance = [10,3,6,8,2,7]

plt.bar(x\_pos,performance,align = 'center')

plt.ylabel('Usage')

plt.title('Programming language usage')

plt.show()



import matplotlib.pyplot as plt

import numpy as np

objects = ('Python','C++','Java','Perl','Scala','Lisp')

y\_pos = np.arange(len(objects))

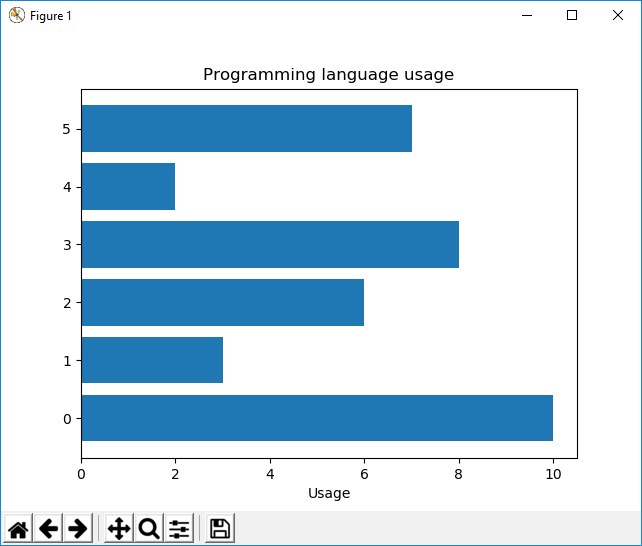
performance = [10,3,6,8,2,7]

plt.barh(y\_pos,performance,align = 'center')

plt.xlabel('Usage')

plt.title('Programming language usage')

plt.show()



import matplotlib.pyplot as plt

labels = ['Python','C++','Ruby','Java']

sizes = [215,130,245,210]

colors = ['gold','yellowgreen','lightcoral','lightskyblue']

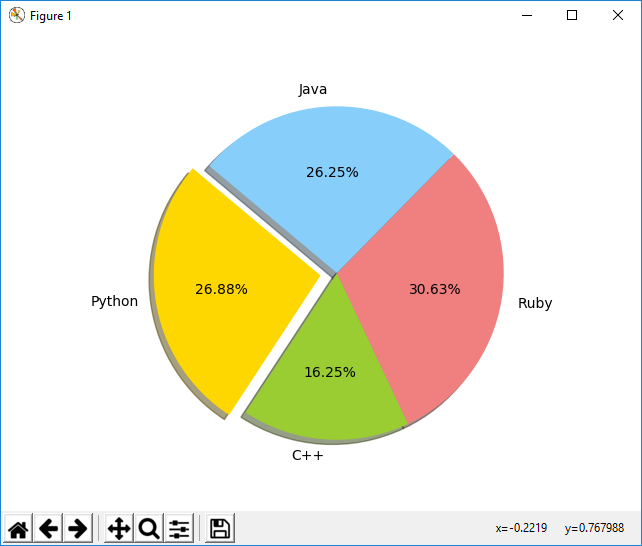
explode = (0.1,0,0,0)

plt.pie(sizes, explode=explode, labels=labels, colors=colors,

autopct='%1.2f%%', shadow = True, startangle = 140)

plt.axis('equal')

plt.show()



import numpy as np

import matplotlib.pyplot as plt

N=500

x = np.random.rand(N)

y = np.random.rand(N)

colors = (0,0,0)

area = np.pi\*3

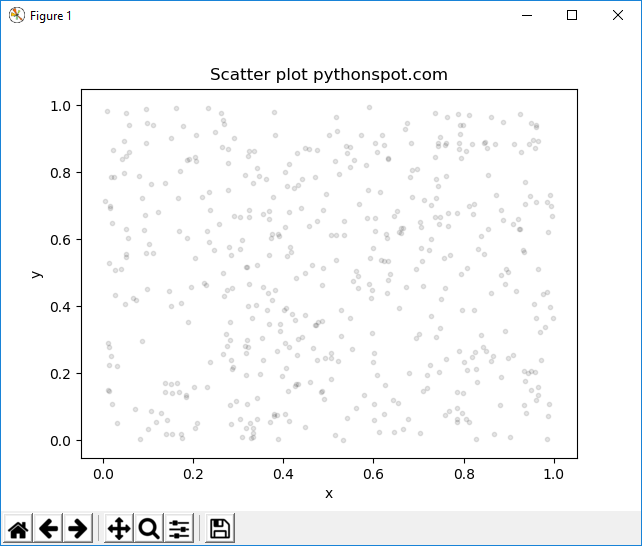
plt.scatter(x,y, s=area,c=colors, alpha=0.1)

plt.title('Scatter plot pythonspot.com')

plt.xlabel('x')

plt.ylabel('y')

plt.show()



import matplotlib.pyplot as plt

import numpy as np

y = [2,4,6,8,10,12,14,16,18,19]

y2 = [10,11,12,13,14,15,16,17,18,19]

x = np.arange(10)

fig = plt.figure()

ax = plt.subplot(111)

ax.plot(x,y, label='Sy=numbers')

ax.plot(x,y2, label='Sy2=other numbers')

plt.title('Legend outside')

chartBox = ax.get\_position()

ax.set\_position([chartBox.x0,chartBox.y0,chartBox.width\*0.6,chartBox.height])

ax.legend(loc='upper center',bbox\_to\_anchor=(1.45,0.8), shadow =True, ncol=1)

plt.show()

