﻿"""

Created on Wed Jan 30 13:30:54 2019

@author: roryeggleston

"""

import numpy as Numpy

﻿a = Numpy.array([0, 1, 2, 3])

a

Out[2]: array([0, 1, 2, 3])

﻿a = Numpy.array([0, 1, 2, 3])

﻿a.ndim

Out[4]: 1

﻿a.shape

Out[5]: (4,)

﻿len(a)

Out[6]: 4

﻿b = Numpy.array([[0, 1, 2], [3, 4, 5]])

b

Out[7]:

array([[0, 1, 2],

[3, 4, 5]])

﻿b.ndim

Out[8]: 2

﻿b.shape

Out[9]: (2, 3)

﻿len(b)

Out[10]: 2

﻿c = Numpy.array([[[1], [2]], [[3],[4]]])

c

Out[11]:

array([[[1],

[2]],

[[3],

[4]]])

﻿c.shape

Out[12]: (2, 2, 1)

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

d

Out[13]:

array([[7, 5, 3, 1],

[8, 6, 4, 2]])

﻿d.shape

Out[14]: (2, 4)

﻿len(d)

Out[15]: 2

﻿n [16]: d.ndim

Out[16]: 2

﻿a = Numpy.arange(10)

a

Out[17]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

﻿b = Numpy.arange(1, 9, 2)

b

Out[18]: array([1, 3, 5, 7])

﻿c = Numpy.linspace(0,1,6)

#(start, stop, number of points)

c

Out[19]: array([0. , 0.2, 0.4, 0.6, 0.8, 1. ])

﻿d = Numpy.arange(2, 32, 2)

d

Out[20]: array([ 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30])

﻿e = Numpy.arange(-2,-32, -2)

#(start, stop, +stride)

e

Out[21]:

array([ -2, -4, -6, -8, -10, -12, -14, -16, -18, -20, -22, -24, -26,

-28, -30])

﻿d = Numpy.linspace(0,1,5,endpoint=False)

d

Out[22]: array([0. , 0.2, 0.4, 0.6, 0.8])

﻿a = Numpy.zeros([3,5])

a

Out[23]:

array([[0., 0., 0., 0., 0.],

[0., 0., 0., 0., 0.],

[0., 0., 0., 0., 0.]])

﻿a[2,3] = 3

a

Out[24]:

array([[0., 0., 0., 0., 0.],

[0., 0., 0., 0., 0.],

[0., 0., 0., 3., 0.]])

﻿a = Numpy.random.rand(4)

a

Out[25]: array([0.15909185, 0.35581894, 0.28641742, 0.38229414])

﻿b = Numpy.random.randn(4)

b

b \* 20

Out[26]: array([ -3.17617165, -25.70636017, -13.05316325, 36.49989974])

﻿Numpy.random.seed(1234)

﻿a = Numpy.arange(1,45,3)

a

Out[28]: array([ 1, 4, 7, 10, 13, 16, 19, 22, 25, 28, 31, 34, 37, 40, 43])

﻿b = Numpy.linspace(0, 2, 7)

b

Out[29]:

array([0. , 0.33333333, 0.66666667, 1. , 1.33333333,

1.66666667, 2. ])

﻿c = Numpy.ones([3,6])

c

Out[30]:

array([[1., 1., 1., 1., 1., 1.],

[1., 1., 1., 1., 1., 1.],

[1., 1., 1., 1., 1., 1.]])

﻿d = Numpy.zeros([45, 23])

d

Out[31]:

array([[0., 0., 0., ..., 0., 0., 0.],

[0., 0., 0., ..., 0., 0., 0.],

[0., 0., 0., ..., 0., 0., 0.],

...,

[0., 0., 0., ..., 0., 0., 0.],

[0., 0., 0., ..., 0., 0., 0.],

[0., 0., 0., ..., 0., 0., 0.]])

﻿e = Numpy.random.rand(15)

e

Out[32]:

array([0.19151945, 0.62210877, 0.43772774, 0.78535858, 0.77997581,

0.27259261, 0.27646426, 0.80187218, 0.95813935, 0.87593263,

0.35781727, 0.50099513, 0.68346294, 0.71270203, 0.37025075])

﻿f = Numpy.random.randn(23)

f

Out[33]:

array([ 0.1458091 , 2.89409095, -0.30426018, 0.861661 , -0.68992667,

0.18749737, 0.60430874, -0.18301422, -1.12650247, 1.65887284,

-0.66044141, 1.04108597, 1.05317489, 0.1942955 , 1.35835198,

-0.28694235, 0.44143825, -0.76987271, -0.97456841, 0.70900741,

-0.10578273, 0.77677573, 0.91395164])

﻿Numpy.random.seed(16)

a = Numpy.random.rand(6)

a

Out[34]:

array([0.22329108, 0.52316334, 0.55070146, 0.04560195, 0.36072884,

0.22308094])

﻿a = Numpy.empty([3,4], dtype = int)

a

Out[36]:

array([[5764607523034234880, 5764607523034234880, 13,

0],

[ 0, 0, 0,

0],

[ 0, 0, 0,

0]])

﻿a = Numpy.array([1,2,3])

a.dtype

Out[37]: dtype('int64')

﻿b = Numpy.array([1.,2.,3.])

b.dtype

Out[38]: dtype('float64')

c = Numpy.array([1,2,3], dtype=float)

c.dtype

Out[39]: dtype('float64')

﻿a = Numpy.ones((3,3))

a.dtype

Out[40]: dtype('float64')

﻿d = Numpy.array([1+2j, 3+4j, 5+6j])

d.dtype

Out[41]: dtype('complex128')

﻿x = Numpy.linspace(0,3,20)

y = Numpy.linspace(0,9,20)

plt.plot(x,y)

plt.plot(x,y, 'o')

Out[45]: [<matplotlib.lines.Line2D at 0x11e3814e0>]



﻿

image = Numpy.random.rand(30,30)

plt.imshow(image,cmap=plt.cm.hot)

plt.colorbar()

Out[47]: <matplotlib.colorbar.Colorbar at 0x11e860358>



﻿y = Numpy.sin(x)

plt.plot(y)

Out[48]: [<matplotlib.lines.Line2D at 0x11e8d3cf8>]



﻿

y = Numpy.cos(x)

plt.plot(x,y)

plt.xlabel('time')

plt.ylabel('cosx')

Out[49]: Text(0, 0.5, 'cosx')



﻿image = Numpy.random.rand(20,20)

plt.imshow(image,cmap=plt.cm.jet)

plt.colorbar()

Out[50]: <matplotlib.colorbar.Colorbar at 0x11eb48278>



﻿a = Numpy.arange(10)

a

Out[51]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

﻿a[0], a[2], a[-1]

Out[52]: (0, 2, 9)

﻿a[::-1]

Out[53]: array([9, 8, 7, 6, 5, 4, 3, 2, 1, 0])

﻿a = Numpy.diag(Numpy.arange(3))

a

Out[54]:

array([[0, 0, 0],

[0, 1, 0],

[0, 0, 2]])

﻿a[1,1]

Out[55]: 1

﻿a[2,1] = 10

a

Out[56]:

array([[ 0, 0, 0],

[ 0, 1, 0],

[ 0, 10, 2]])

﻿a[1]

Out[57]: array([0, 1, 0])

﻿a = Numpy.arange(10)

a

Out[58]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

﻿a[2:9:3]

Out[59]: array([2, 5, 8])

﻿a[:4]

Out[60]: array([0, 1, 2, 3])

﻿a[1:3]

Out[61]: array([1, 2])

﻿a[::2]

Out[62]: array([0, 2, 4, 6, 8])

﻿a[3:]

Out[63]: array([3, 4, 5, 6, 7, 8, 9])

﻿a = Numpy.arange(10)

a[5:] = 10

a

Out[64]: array([ 0, 1, 2, 3, 4, 10, 10, 10, 10, 10])

﻿b = Numpy.arange(5)

a[5:] = b[::b-1]

a

Traceback (most recent call last):

File "<ipython-input-65-09cf1f061e11>", line 2, in <module>

a[5:] = b[::b-1]

TypeError: only integer scalar arrays can be converted to a scalar index

﻿a = Numpy.arange(35)

a

Out[66]:

array([ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,

17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33,

34])

﻿a[1:31:4]

Out[67]: array([ 1, 5, 9, 13, 17, 21, 25, 29])

﻿b = Numpy.linspace(0,120,120,dtype='int')

b

Out[68]:

array([ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12,

13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25,

26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,

39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51,

52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64,

65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77,

78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90,

91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103,

104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116,

117, 118, 120])

﻿b[120:1:-2]

Out[69]:

array([120, 117, 115, 113, 111, 109, 107, 105, 103, 101, 99, 97, 95,

93, 91, 89, 87, 85, 83, 81, 79, 77, 75, 73, 71, 69,

67, 65, 63, 61, 59, 57, 55, 53, 51, 49, 47, 45, 43,

41, 39, 37, 35, 33, 31, 29, 27, 25, 23, 21, 19, 17,

15, 13, 11, 9, 7, 5, 3])

﻿Numpy.arange(6) + Numpy.arange(0, 51, 10)[:, Numpy.newaxis]

Out[70]:

array([[ 0, 1, 2, 3, 4, 5],

[10, 11, 12, 13, 14, 15],

[20, 21, 22, 23, 24, 25],

[30, 31, 32, 33, 34, 35],

[40, 41, 42, 43, 44, 45],

[50, 51, 52, 53, 54, 55]])

a = Numpy.array([[1,1,1,1],[1,1,1,1],[1,1,1,2],[1,6,1,1]])

a

Out[71]:

array([[1, 1, 1, 1],

[1, 1, 1, 1],

[1, 1, 1, 2],

[1, 6, 1, 1]])

﻿b = Numpy.array([[0.,0.,0.,0.,0.],[2.,0.,0.,0.,0.],[0.,3.,0.,0.,0.],[0.,0.,4.,0.,0.],[0.,0.,0.,5.,0.],[0.,0.,0.,0.,6.]])

b

Out[72]:

array([[0., 0., 0., 0., 0.],

[2., 0., 0., 0., 0.],

[0., 3., 0., 0., 0.],

[0., 0., 4., 0., 0.],

[0., 0., 0., 5., 0.],

[0., 0., 0., 0., 6.]])

a = Numpy.arange(7,dtype='float')

a

Out[73]: array([0., 1., 2., 3., 4., 5., 6.])

﻿b = Numpy.diag(a)

b

Out[74]:

array([[0., 0., 0., 0., 0., 0., 0.],

[0., 1., 0., 0., 0., 0., 0.],

[0., 0., 2., 0., 0., 0., 0.],

[0., 0., 0., 3., 0., 0., 0.],

[0., 0., 0., 0., 4., 0., 0.],

[0., 0., 0., 0., 0., 5., 0.],

[0., 0., 0., 0., 0., 0., 6.]])

﻿a= Numpy.array([[4,3], [2,1]])

a

Numpy.tile(a, (2,3))

Out[76]:

array([[4, 3, 4, 3, 4, 3],

[2, 1, 2, 1, 2, 1],

[4, 3, 4, 3, 4, 3],

[2, 1, 2, 1, 2, 1]])