

```
In [1]: import sys

In [2]: p,q,r=[1,"yogita",12],[2.4,"hi",4*7],('a','b','c')

In [3]: print(p)
print(type(p))

In [3]: 11
'yogita', 12]
<class 'list'>

In [4]: print(q)
print(type(q))

In [4]: (2.4, 'hi', (4+7))
<class 'tuple'>

In [5]: print(r)
print(type(r))

In [5]: ('b', 'a', 'c')
<class 'set'>

In [6]:

In [6]: Bin(25)

Out[6]: '0b1001'

In [7]: y=oct(2)

In [7]: 2

Out[7]: '0o2'

In [8]: x=0o56
print(x)
print(type(x))

In [8]: 46
<class 'int'>

ARITHMETIC OPERATIONS

In [9]: a=15
b=10

In [10]: a+b

Out[10]: 25

In [11]: print(a-b)
print(a/b)

In [11]: 5
1.5

In [12]: #find square root

In [13]: import math          #importing package

In [14]: math.sqrt(81)        #sqrt is function which return square root of passing parameter

Out[14]: 9.0

In [15]: math

In [15]: <module 'math' (built-in)>

Out[15]:

In [16]: a=round(math.sqrt(64))    #round function return an integer value
a

Out[16]: 8

In [17]: s="HI"
t=" GOOD MORNING"

In [17]: s+t

In [18]: s+p+p+p              #it will display concatenation of two strings
s+p

Out[18]: 'HI GOOD MORNING'

In [19]: 4<5    #check that 4 is less than 5 & display output as boolean value(True or False)

Out[19]: True

In [20]: 4==5    #it checks 4 is not equal to 5 then result is True

Out[20]: True

In [21]: 5>=5    #it having two conditions(greater than or equal to),if one is True then result is True

Out[21]: True

In [22]: True or True    #or operation required atleast is True

Out[22]: True

In [23]: True or False

Out[23]: True

In [24]: True and True    #and operation required both are True

Out[24]: True

In [25]: True and False

Out[25]: False

In [26]: True or False and True

Out[26]: True

In [27]: True and False or True

Out[27]: True

While loop

In [36]: num=0
while num<5:
    print(num)
    num=num+1
    print(num)
    '''here we have pass the value of num is 10
    first the condition is check that the number if less than 15
    after print that number
    then add 1 in that number and then check condition and print that next number
    loop will execute till condition become false'''

10
11
12
13
14
15

In [29]: range(5)

Out[29]: range(0, 5)

In [35]: range(10,20,5)

Out[35]: range(10, 20, 5)

for loop

In [38]: for i in range(5):
print("Welcome")    #it will print the message 5 times

Welcome
Welcome
Welcome
Welcome
Welcome

In [40]: for j in range(5):
print("Welcome",j)    #it will print the message 5 times with range
print("Good Morning")

Welcome 0
"Good Morning"
Welcome 1
"Good Morning"
Welcome 2
"Good Morning"
Welcome 3
"Good Morning"
Welcome 4
"Good Morning"

check version of package

In [41]: import numpy as np    #check version of numpy package
np.__version__

Out[41]: '1.23.5'

In [42]: import pandas as pd    #check version of pandas package
pd.__version__

Out[42]: '1.5.3'

In [43]: import numpy as np
from numpy.random import randn    #import random module from numpy package

In [44]: import random as r

In [45]: #simple OTP generator

In [48]: def otpgen():
otp=""
for i in range(4):
    otp+=str(random.randint(1,6))
    print("your one time password is")
    print(otp)
    otpgen()

In [49]: #if else statement

In [57]: we=round()
if >5:
    ans="greater than 5"
else
    ans="less than 5"

print(ans)
print(ans)

1.434961427357741
less than 5

In [58]: list(range(15))    #print the list of elements from range 0 to 15

Out[58]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]

In [59]: list(range(10,15))    #print the list of elements from range 10 to 15

Out[59]: [10, 11, 12, 13, 14]

In [60]: list(range(10,20,3))    #print the list of elements from range 10 to 20 with three steps

Out[60]: [10, 13, 16, 19]

In [63]: l1=list(range(10,55,5))    #print the list of elements from range 10 to 55 with 5 steps
l1

Out[63]: [10, 15, 20, 25, 30, 35, 40, 45, 50]

In [64]: len(l1)

Out[64]: 9

In [65]: type(l1)

Out[65]: list

In [66]: min(1,500,254)

Out[66]: 1

In [67]: max(1,500,254)

Out[67]: 500

In [2]: import numpy as np
np.random.randint(10,20,5)
a

Out[2]: array([15, 17, 16, 19, 13])

In [3]: list_=10,1,2,3,4,5
list_

Out[3]: [0, 1, 2, 3, 4, 5]

In [4]: type(list_)

Out[4]: list

In [6]: arr=np.array(list_)    #convert the list into array
arr

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

In [8]: type(arr)

Out[8]: numpy.ndarray

In [9]: np.arange(10)    #it prints the list of numbers from range 0 to 10

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

In [10]: np.arange(5)    #it prints the list of numbers from range 0 to 5

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

In [11]: np.arange(10,20)    #it prints the list of numbers from range 10 to 20

Out[11]: array([10, 11, 12, 13, 14, 15, 16, 17, 18, 19])

In [12]: np.arange(10,20)    #starting range should be less than end

Out[12]: array([], dtype=int32)

In [13]: np.arange(10,10)

Out[13]: array([], dtype=int32)

In [14]: np.arange(-30,20)    #it prints the list of numbers from range -30 to 20

Out[14]: array([-30, -29, -28, -27, -26, -25, -24, -23, -22, -21, -20, -19, -18,
-17, -16, -15, -14, -13, -12, -11, -10, -9, -8, -7, -6, -5,
-4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6, 7, 8,
9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19])

In [15]: np.arange(10,30,5)    #it prints the list of numbers from range 10 to 30 with 5 steps

Out[15]: array([10, 15, 20, 25])

In [16]: np.zeros(4,5)    #print zeros with 4 rows and 5 columns

Out[16]: array([[0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0.]])

In [17]: np.zeros(5,4)    #print zeros with 5 rows and 4 columns

Out[17]: array([[0., 0., 0., 0.],
[0., 0., 0., 0.],
[0., 0., 0., 0.],
[0., 0., 0., 0.],
[0., 0., 0., 0.]])

In [18]: np.ones(2,6)    #print ones with 2 rows and 6 columns

Out[18]: array([[1., 1., 1., 1., 1., 1.],
[1., 1., 1., 1., 1., 1.]])

In [19]: np.ones(6,6)    #print ones with 6 rows and 6 columns

Out[19]: array([[1., 1., 1., 1., 1., 1.],
[1., 1., 1., 1., 1., 1.],
[1., 1., 1., 1., 1., 1.],
[1., 1., 1., 1., 1., 1.],
[1., 1., 1., 1., 1., 1.],
[1., 1., 1., 1., 1., 1.]])

In [20]: from numpy import *

In [21]: arange(3)    #print list of elements from range 0 to 3

Out[21]: array([0, 1, 2])

In [22]: np.arange(3)

Out[22]: array([0, 1, 2])

In [23]: zeros(5)

Out[23]: array([0., 0., 0., 0., 0.])

In [28]: np.random.randint(10,30)    #print random number from 10 to 30

Out[28]: 20

In [31]: np.random.randint(10,20,5)    #print 5 random numbers from 10 to 20

Out[31]: array([10, 18, 16, 16, 16])

In [32]: np.random.randint(1)

Out[32]: 0

In [33]: np.random.randint(1,3,5)    #print 5 random numbers from 1 to 3

Out[33]: array([2, 2, 2, 2, 1])

In [35]: np.random.randint(10,20,(4,5))    #print random numbers from 10 to 20 with 4 rows & 5 columns

Out[35]: array([[19, 13, 13, 12, 13],
[14, 16, 15, 19, 11],
[13, 17, 16, 15, 12],
[17, 18, 18, 13, 10]])

In [36]: np.random.randint(10,20,(4,4))    #print random numbers from 10 to 20 with 4 rows & 4 columns

Out[36]: array([[18, 13, 15, 14],
[14, 14, 17, 16],
[17, 19, 19, 17],
[15, 18, 10, 17]])

In [37]: np.random.randint(10,20,5)    #print 5 random numbers from 10 to 20

Out[37]: array([11, 16, 11, 13, 15])

In [38]: arr

Out[38]: array([0, 1, 2, 3, 4, 5])

In [40]: arr=np.random.randint(0,100,10)    #print 10 random numbers from 0 to 100

Out[40]: array([0, 1, 2, 3, 4, 5])

In [41]: arr

Out[41]: array([0, 1, 2, 3, 4, 5])

In [42]: arr.max()    #maximum number in arr

Out[42]: 5

In [43]: arr.min()    #minimum number in arr

Out[43]: 0

In [44]: arr.mean()    #mean of arr

Out[44]: 2.5

In [46]: from numpy import *
a=array([1,2,3,5,6])
median(a)    #print middle value in a

Out[46]: 3.0

In [47]: arr

Out[47]: array([0, 1, 2, 3, 4, 5])

In [48]: arr.reshape(3,2)    #convert 1D array into 3 rows and 2 columns

Out[48]: array([[0, 1],
[2, 3],
[4, 5]])

In [49]: arr.reshape(2,3)    #convert 1D array into 2 rows and 3 columns

Out[49]: array([[0, 1, 2],
[3, 4, 5]])

In [50]: arr.reshape(1,6)    #convert array into 1 row and 6 columns

Out[50]: array([[0, 1, 2, 3, 4, 5]])

INDEXING

In [52]: mat=np.arange(0,100).reshape(10,10)#print numbers from 0 to 100 with 10 rows 10 columns(matrix)
mat

Out[52]: 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]])

In [53]: row=3
col=4

In [54]: mat[row,col]    #print number from matrix having row index 3 and column index 4

Out[54]: 34

In [58]: mat[:]    #print all elements from index 0 to n-1

Out[58]: 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]])

In [56]: mat[:,col]    #print all elements from index 0 to n-1 having column=4

Out[56]: array([4, 14, 24, 34, 44, 54, 64, 74, 84, 94])

In [57]: mat[row,: ]    #print all elements from index 0 to n-1 having row index=3(we have assign already)

Out[57]: array([30, 31, 32, 33, 34, 35, 36, 37, 38, 39])

In [59]: mat[2:6:2:4]    #print elements with row from index 2 to 6-1 & column index 2 to 4-1(3)

Out[59]: array([22, 23],
[32, 33],
[42, 43],
[52, 53]])

In [60]: mat[1:2:2:4]    #print elements with row from index 1 to 2-1(1) & column index 2 to 4-1(3)

Out[60]: array([[12, 13]])

MASKING

In [61]: mat

Out[61]: 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]])

In [62]: mat<50    #it will print result True for value less than 50

Out[62]: array([[ True,  True,  True,  True,  True,  True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True,  True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True,  True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True,  True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True,  True,  True,  True,  True,  True],
[ True,  True,  True,  True,  True,  True,  True,  True,  True,  True],
[False, False, False, False, False, False, False, False, False, False],
[False, False, False, False, False, False, False, False, False, False],
[False, False, False, False, False, False, False, False, False, False],
[False, False, False, False, False, False, False, False, False, False]])

In [63]: mat[mat<50]    #it will values less than 50

Out[63]: 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])

In [64]: mat[mat>50]    #it will values greater than 50

Out[64]: array([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])

In [ ]:
```

