20-07-2023

In [1]:
```
import numpy as np
```

1.Create an array with zeros and ones and print the output

In [2]:
```
a = np.zeros(5,dtype=np.int64)
b = np.ones(5,dtype=np.int64)
print(a)
print(b)
```

```
[0 0 0 0 0]
[1 1 1 1 1]
```

2.Create an array and print the output

In [3]:
```
arr = np.array([1,2,3,6,8,3,7,10,34,65])
print(arr)
```

```
[ 1  2  3  6  8  3  7 10 34 65]
```

3.Create an array whose initial content is random and print the output

In [4]:
```
arr1 = np.arange(21)
print(arr1)
```

```
[ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20]
```

4.Create an array with the range of values with even intervals

In [5]:
```
ab = np.arange(1,20,4)
print(ab)
```

```
[ 1  5  9 13 17]
```

5.Create an array with values that are spaced linearly in a specified interval

In [8]:
```
abc = np.linspace(1,50,num=10,dtype=np.int64)
print(abc)
```

```
[ 1  6 11 17 22 28 33 39 44 50]
```

6.Access and manipulate elements in the array

In [10]:
```
arr1 = np.array([22,3,4,9,10,24,6])
print(arr[1:5])
```

```
[2 3 6 8]
```

In [11]:
```
arr1[1]
```

Out[11]: 3

In [12]:
```
arr1[-1]
```

Out[12]: 6

In [13]:
```
arr1[-2]
```

Out[13]: 24

7. Create a 2-dimensional array and check the shape of the array

In [16]:
```
arr2=np.array([[1,2,3,4],[9,8,7,6]])
print(arr2)
```

```
[[1 2 3 4]
 [9 8 7 6]]
```

8. Using the arange() and linspace() function to evenly space values in a specified interval

In [17]:
```
xyz = np.arange(25)
abcd = np.linspace(5,95,num=9,dtype=np.int64)
print(xyz)
print(abcd)
```

```
[ 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]
[ 5 16 27 38 50 61 72 83 95]
```

9. Create an array of random values between 0 and 1 in a given shape

In [18]:
```
aj = np.array([])
print(np.shape(aj))
```

```
(0,)
```

In [19]:
```
bj = np.array([1])
print(np.shape(bj))
```

```
(1,)
```

10. Repeat each element of an array by a specified number of times using repeat() and tile() functions

In [21]:
```
m=np.array([1,2,3])
print(np.repeat(m,2))
print(np.tile(m,3))
```

```
[1 1 2 2 3 3]
[1 2 3 1 2 3 1 2 3]
```

11.How do you know the shape and size of an array?

In [23]:
```
# shape = To identify the rows and columns
# size = To identify the size of the elements
```

12. Create an array that indicates the total number of elements in an array

In [24]:
```
print(np.size(m))
```

```
3
```

13.To find the number of dimensions of the array

In [25]:
```
print(np.ndim(m))
```

```
1
```

14. Create an array and reshape into a new array

In [27]:
```
cbi = np.array([1,3,5,7,9,11])
print(cbi.reshape(3,2))
```

```
[[ 1  3]
 [ 5  7]
 [ 9 11]]
```

15. Create a null array of size 10

In [31]:
```
null_array = np.zeros(10)
print(null_array)
```

```
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
```

16.Create any array with values ranging from 10 to 49 and print the numbers whose remainders are zero when divided by 7

In [32]:
```
arr = np.arange(10, 50)
result = arr[arr % 7 == 0]
print(result)
```

```
[14 21 28 35 42 49]
```

17. Create an array and check any two conditions and print the output

In [38]:
```
ghj =  np.arange(10, 50)
ac = ghj[(ghj > 20) & (ghj < 30)]
print(ac)
```

```
[21 22 23 24 25 26 27 28 29]
```

18. Use Arithmetic operator and print the output using array

In [40]:
```
bvc = np.arange(1,11)
cxz = np.arange(11,21)
print(bvc)
print(cxz)
print(bvc+cxz)
```

```
[ 1  2  3  4  5  6  7  8  9 10]
[11 12 13 14 15 16 17 18 19 20]
[12 14 16 18 20 22 24 26 28 30]
```

In [41]:
```
print(bvc-cxz)
print(bvc*cxz)
print(bvc/cxz)
```

```
[-10 -10 -10 -10 -10 -10 -10 -10 -10 -10]
[ 11  24  39  56  75  96 119 144 171 200]
[0.09090909 0.16666667 0.23076923 0.28571429 0.33333333 0.375
 0.41176471 0.44444444 0.47368421 0.5       ]
```

19.Use Relational operators and print the results using array ¶

In [42]:
```
hg = np.array([10,20,30,40,50])
print(hg[hg<30])
print(hg[hg>30])
```

```
[10 20]
[40 50]
```

20. Difference between python and ipython"

In [44]:
```
# Python
# Python is a general-purpose programming language. It was created in the late 1980s by Guido van Rossum. It is now one of the most popular languages in the world. It is rou

# ipyhton
# IPython is an interactive command-line terminal for Python. It was created by Fernando Perez in 2001. IPython offers an enhanced read-eval-print loop (REPL) environment pa
```