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

In [1]:

```
import numpy as np
a=np.zeros(2)
b=np.ones(2)
print(np.concatenate((a,b)))
```

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

# 2. Create an array and print the output

In [3]:

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

```
[1 2 3 4]
```

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

In [5]:

```
print(np.empty(5,dtype=np.int8))
```

```
[  32   34 -111 -113 -101]
```

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

In [6]:

```
print(np.arange(2,20,+2))
```

```
[ 2  4  6  8 10 12 14 16 18]
```

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

In [7]:

```
print(np.linspace(0,20,num=3))
```

```
[ 0. 10. 20.]
```

## 6. Access and manipulate elements in the array

In [8]:

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

```
[1 2 3 4 5]
```

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

In [9]:

```
a=np.array([[10,20],[30,40]])
print(np.shape(a))
```

```
(2, 2)
```

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

In [14]:

```
a=np.linspace(0,100,num=11)
b=np.arange(10)
print(a)
print(b)
```

```
[  0.  10.  20.  30.  40.  50.  60.  70.  80.  90. 100.]
[0 1 2 3 4 5 6 7 8 9]
```

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

In [15]:

```
a=np.linspace(0,14,num=3)
print(a)
```

```
[ 0.  7. 14.]
```

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

In [17]:

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

```
[1 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6]
[1 2 3 4 5 6 1 2 3 4 5 6 1 2 3 4 5 6]
```

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

In [19]:

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

```
(2, 4)
8
```

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

In [20]:

```
print(np.size(a))
```

```
8
```

# 13. To find the number of dimensions of the array

In [21]:

```
print(np.ndim(a))
```

```
2
```

# 14. Create an array and reshape into a new array

In [22]:

```
x=np.array([22,11,34,50])
print(x.reshape(2,2))
```

```
[[22 11]
 [34 50]]
```

# 15. Create a null array of size 10

In [23]:

```
print(np.empty(10,dtype=np.int8))
```

```
[ -96   81   98 -109 -101    1    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 [38]:

```
y=np.arange(10,49,+2)
print(y)
```

```
[10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48]
```

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

In [28]:

```
a=np.array([12,34,14,67,24,15,77])
print(a[(a>15)&(a<34)])
```

```
[24]
```

# 18. Use Arithmetic operator and print the output using array

In [29]:

```
print(y[3]+y[7])
```

```
30
```

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

In [30]:

```
print(a[a%3==0])
```

```
[12 24 15]
```

# 20. Difference between python and ipython

IPython is an interactive command-line terminal for Python.

In [ ]: