```
import array
# Creating an array
print(array.array('i', [1, 2, 3, 4, 5]))
```

```
    array('i', [1, 2, 3, 4, 5])
```

```
import numpy as np
# Creating a list named "a"
a = [1, 2, 3, 4, 5]
print(type(a))
print(np.array(a))
np.array([range(i, i + 4) for i in a])
```

```
    <class 'list'>
    [1 2 3 4 5]
    array([[1, 2, 3, 4],
           [2, 3, 4, 5],
           [3, 4, 5, 6],
           [4, 5, 6, 7],
           [5, 6, 7, 8]])
```

```
a = np.matrix('1 2; 3 4')
a
```

```
matrix([[1, 2],
        [3, 4]])
```

```
np.matrix([[1, 2], [3, 4]])
```

```
    matrix([[1, 2],
            [3, 4]])
```

```
#Create a Numpy Array containing numbers from 5 to 30 but at equal interval of 2
import numpy as np
# Start = 5, Stop = 30, Step Size = 2
arr = np.arange(5, 30, 2)
print(arr)
```

```
    [ 5  7  9 11 13 15 17 19 21 23 25 27 29]
```

```
#Create a Numpy Array containing elements from 1 to 10 with default interval i.e. 1
import numpy as np
# Start = 1, Stop = 10. As Step Size is not provided, so default value be 1
arr = np.arange(1, 10)
print(arr)
```

```
    [1 2 3 4 5 6 7 8 9]
```

```
#Create a Numpy Array containing elements up to 20 with default start and step size
import numpy as np
def main():
```

```
    print('*** Create numpy array using numpy.arange() ***')
    print('Create a Numpy Array containing elements from 5 to 30 but at equal interv
    # Start = 5, Stop = 30, Step Size = 2
    arr = np.arange(5, 30, 2)
    print('Contents of the Array : ', arr)
    print('Create a Numpy Array containing elements from 1 to 10 with default interv
    # Start = 1, Stop = 10. As Step Size is not provided, so default value be 1
    arr = np.arange(1, 10)
    print('Contents of the Array : ', arr)
    print('Create a Numpy Array containing elements up to 10 with default start and
    # Stop = 20. As Start &amp;amp; Step Size is not provided, so default value be 0 &ar
    arr = np.arange(20)
    print('Contents of the Array : ', arr)
if __name__ == '__main__':
    main()
```

```
*** Create numpy array using numpy.arange() ***
Create a Numpy Array containing elements from 5 to 30 but at equal interval o
Contents of the Array :  [ 5  7  9 11 13 15 17 19 21 23 25 27 29]
Create a Numpy Array containing elements from 1 to 10 with default interval i
Contents of the Array :  [1 2 3 4 5 6 7 8 9]
Create a Numpy Array containing elements up to 10 with default start and defa
Contents of the Array :  [ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
```

```
import numpy as np

# Creating array object
arr = np.array( [[ 1, 2, 3],
                 [ 4, 2, 5]] )

# Printing type of arr object
print("Array is of type: ", type(arr))

# Printing array dimensions (axes)
print("No. of dimensions: ", arr.ndim)

# Printing shape of array
print("Shape of array: ", arr.shape)

# Printing size (total number of elements) of array
print("Size of array: ", arr.size)

# Printing type of elements in array
print("Array stores elements of type: ", arr.dtype)
```

```
Array is of type:  <class 'numpy.ndarray'>
No. of dimensions:  2
Shape of array:  (2, 3)
Size of array:  6
Array stores elements of type:  int64
```

```
#Reshape From 1-D to 2-D
import numpy as np
```

```
arr = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12])

newarr = arr.reshape(4, 3)

print(newarr)
```

```
[[ 1  2  3]
 [ 4  5  6]
 [ 7  8  9]
 [10 11 12]]
```

```
#Reshape From 1-D to 3-D
import numpy as np

arr = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12])

newarr = arr.reshape(2, 3, 2)

print(newarr)
```

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

 [[ 7  8]
  [ 9 10]
  [11 12]]]
```

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

```
array([1, 2, 3, 4])
```

```
x = np.arange(4).reshape((2,2))
np.transpose(x)
```

```
array([[0, 2],
       [1, 3]])
```

```
x = np.ones((1, 2, 3))
np.transpose(x, (1, 0, 2)).shape
```

```
(2, 1, 3)
```

```
x = np.ones((2, 3, 4, 5))
np.transpose(x).shape
```

```
(5, 4, 3, 2)
```