

```
In [1]: # NumPy program to create an element-wise comparison (greater, greater_equal, Less.py
import numpy as np
x = np.array([3,5,1,2,3])
y = np.array([2,5,3,2,1])
print("Array A")
print(x)
print("\nArray B")
print(y)
print("\nA>B")
print(np.greater(x, y))
print("\nA>=B")
print(np.greater_equal(x, y))
print("\nA<B")
print(np.less(x, y))
print("\nA<=B")
print(np.less_equal(x, y))
```

Array A
[3 5 1 2 3]

Array B
[2 5 3 2 1]

A>B
[True False False False True]

A>=B
[True True False True True]

A<B
[False False True False False]

A<=B
[False True True True False]

```
In [2]: # NumPy program to create an array of all the even integers from 30 to 70.py
import numpy as np
x = np.arange(start=30, stop=71, step=2)
print(x)
```

[30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66 68 70]

```
In [3]: # _NumPy program to create a 3x3 identity matrix
import numpy as np
x = np.identity(3)
print(x)
```

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

```
In [4]: # NumPy program to create a vector with values from 0 to 20 and change the sign of.py
import numpy as np
x = np.arange(21)
print("Vectors ")
print(x)
print("\nAfter changing the sign of the numbers in the range from 9 to 15:")
x[(x >= 9) & (x <= 15)] *= -1
print(x)
```

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

After changing the sign of the numbers in the range from 9 to 15:
[0 1 2 3 4 5 6 7 8 -9 -10 -11 -12 -13 -14 -15 16 17
18 19 20]

In [5]: *# NumPy program to create a 5x5 zero matrix with elements on the main diagonal.py*

```
import numpy as n
x = np.diag([1, 2, 3, 4, 5])
print(x)
```

```
[[1 0 0 0 0]
 [0 2 0 0 0]
 [0 0 3 0 0]
 [0 0 0 4 0]
 [0 0 0 0 5]]
```

In [6]: *# NumPy program to compute sum of all elements, sum of each column and sum of.*

```
import numpy as np
x = np.array([[1,0],[0,1]])
print("Array")
print(x)
print("\nSum of all elements")
print(np.sum(x))
print("\nSum of each column")
print(np.sum(x, axis=0))
print("\nSum of each row")
print(np.sum(x, axis=1))
```

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

```
Sum of all elements
2
```

```
Sum of each column
[1 1]
```

```
Sum of each row
[1 1]
```

In [7]: *# NumPy program to save a given array to a text file and load it.py*

```
import numpy as np
import os
x = np.arange(16).reshape(4,4)
print("Array:")
print(x)
header = 'C1 C2 C3 C4'
np.savetxt('7_array.txt', x, fmt="%d", header=header)
print("\nAfter loading, content of the text file:")
print(np.loadtxt('7_array.txt'))
```

```
Array:
[[ 0  1  2  3]
 [ 4  5  6  7]
 [ 8  9 10 11]
 [12 13 14 15]]
```

```
After loading, content of the text file:
[[ 0.  1.  2.  3.]
 [ 4.  5.  6.  7.]
 [ 8.  9. 10. 11.]
 [12. 13. 14. 15.]]
```

```
In [8]: # NumPy program to check whether two arrays are equal (element wise) or not.py
import numpy as np
nums1 = np.array([2,2,3,2,1])
nums2 = np.array([2,3,4,3,1])
print("Original arrays:")
print(nums1)
print(nums2)
print("\nTest said two arrays are equal (element wise) or not:?)")
print(nums1 == nums2)
print(np.equal(nums1, nums2))
```

Original arrays:

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

Test said two arrays are equal (element wise) or not:?

```
[ True False False False  True]
[ True False False False  True]
```

```
In [9]: # NumPy program to create a 4x4 array with random values, now create a new array.py
import numpy as np
nums = np.arange(16, dtype='int').reshape(-1, 4)
print("Original array:")
print(nums)
print("\nNew array after swapping first and last rows of the said array:")
#new_nums = nums[3:3:-1]
nums = nums[[-1,1,2,0]]
print(nums)
"""
num0 = list(nums[0])
num3 = list(nums[3])
nums[0] = num3
nums[3] = num0
print(nums)
"""
```

Original array:

```
[[ 0  1  2  3]
 [ 4  5  6  7]
 [ 8  9 10 11]
 [12 13 14 15]]
```

New array after swapping first and last rows of the said array:

```
[[12 13 14 15]
 [ 4  5  6  7]
 [ 8  9 10 11]
 [ 0  1  2  3]]
```

```
Out[9]: '\nnum0 = list(nums[0])\nnum3 = list(nums[3])\nnums[0] = num3\nnums[3] = num0\nprint(nums)\n'
```

```
In [11]: # NumPy program to multiply two given arrays of same size element-by-element.py /
import numpy as np
nums1 = np.array([[2, 5, 2],[1, 5, 5]])
nums2 = np.array([[5, 3, 4],[3, 2, 5]])
print("Array1:")
print(nums1)
print("Array2:")
print(nums2)
print("\nMultiply said arrays of same size element-by-element:")
print(np.multiply(nums1, nums2))
```

Array1:

```
[[2 5 2]
 [1 5 5]]
```

Array2:

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

Multiply said arrays of same size element-by-element:

```
[[10 15  8]
 [ 3 10 25]]
```