1. a= [10,12,19,17,-13,18,27,30,-12,-27]

Convert the above list into a numPy array and filter out the numbers with absolute value(modulus value) less than 20.

In [69]:
```
import numpy as np
a = [10,12,19,17,-13,18,27,30,-12,-27]
b = abs(np.array(a))
filter_arr = b>20
print(filter_arr)
newarr = b[filter_arr]
print(newarr)
```

```
[False False False False False False  True  True False  True]
[27 30 27]
```

2. Create a NumPy array with the dimensions 10,2,5 using the arrange function

In [23]:
```
a = np.arange(100).reshape(10,2,5)
a
```

Out[23]:
```
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]]])
```

3. Write a NumPy program to create a vector with values from 0 to 20 and change the sign of the numbers in the range from 9 to 15.

In [29]:
```
x = np.arange(21)
print("Original vector: ")
print(x)
x[(x>=9)&(x<=15)] *= -1
print("Final vector:")
print(x)
```

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

4. Write a NumPy program to create a 3x4 matrix filled with values from 10 to 21

In [32]:
```
x = np.arange(10,22).reshape((3,4))
print(x)
```

```
[[10 11 12 13]
 [14 15 16 17]
 [18 19 20 21]]
```

5. Write a NumPy program to create a 5x5 zero matrix with elements on the main diagonal equal to 1, 2, 3, 4 (Hint: Google how to change individual values in np array)

In [71]:
```
arr = np.diag([1,2,3,4,5])
arr
```

Out[71]:
```
array([[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]])
```

6. Write a NumPy program to multiply two given arrays of the same size element-by-element

In [55]:
```
x = np.arange(5)
print(x)
y = np.arange(5,10)
print(y)
x*y
```

```
[0 1 2 3 4]
[5 6 7 8 9]
```

Out[55]:
```
array([ 0,  6, 14, 24, 36])
```

7. Write a NumPy program to create an array of equal shapes and data types of a given array

In [64]:

```
x= np.array([[11,12,13,14],
             [15,16,17,18],
             [19,20,21,22]])
print(x)
print(x.shape)
print(x.dtype)
y = np.full_like(x,12)
print(y)
print(y.dtype)
```

```
[[11 12 13 14]
 [15 16 17 18]
 [19 20 21 22]]
(3, 4)
int32
[[12 12 12 12]
 [12 12 12 12]
 [12 12 12 12]]
int32
```

In [ ]:

```
1
```

In [ ]:

```
1
```