1.Ans

In [2]:
```
import numpy as np
```

In [2]:
```
# Creating a numpy_array with given range and stepsize of 3

arr=np.array(range(2,51,3))
```

In [3]:
```
arr
```

Out[3]:
```
array([ 2,  5,  8, 11, 14, 17, 20, 23, 26, 29, 32, 35, 38, 41, 44, 47, 50])
```

2.Ans

In [4]:
```
# Creating list of 5 elements

list1=[]
for i in range(0,5):
    ele=int(input())

    list1.append(ele)

print(list1)
```

```
1
9
4
12
32
[1, 9, 4, 12, 32]
```

In [5]:
```
# Creating another list of 5 elements

list2=[]
for i in range(0,5):
    ele=int(input())

    list2.append(ele)

print(list2)
```

```
21
15
8
56
2
[21, 15, 8, 56, 2]
```

In [20]:

```
# Converting list into numpy_array

arr1=np.array(list1)
arr2=np.array(list2)
type(arr1)
```

Out[20]:

```
numpy.ndarray
```

In [32]:

```
# Concatinating two numpy_array's

arr3=np.concatenate((arr1,arr2))
arr3
```

Out[32]:

```
array([ 1,  9,  4, 12, 32, 21, 15,  8, 56,  2])
```

In [34]:

```
# Sorting all the numpy_arrays

print(np.sort(arr1))
print(np.sort(arr2))
print(np.sort(arr3))
```

```
[ 1  4  9 12 32]
[ 2  8 15 21 56]
[ 1  2  4  8  9 12 15 21 32 56]
```

3.Ans

In [3]:

```
# Creating numpy array

arr = np.array( [[ 1, 2, 3],
                 [ 4, 2, 5],
                 [ 3, 6, 7]] )

# Printing array dimensions (axes)

print("No. of dimensions: ", arr.ndim)

# Printing size (total number of elements) of array

print("Size of array: ", arr.size)
```

```
No. of dimensions:  2
Size of array:  9
```

4.Ans

In [4]:

```
# Converting 1D array to 2D array
```

In [13]:

```
# Creating 1D array

arr_1d=np.arange(12)
arr_1d
```

Out[13]:

```
array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11])
```

In [10]:

```
# To convert 1D array to 2D array we use reshape(no_of_rows,no_of_columns)

arr_2d=arr.reshape(4,3)
arr_2d
```

Out[10]:

```
array([[ 0,  1,  2],
       [ 3,  4,  5],
       [ 6,  7,  8],
       [ 9, 10, 11]])
```

In [40]:

```
arr_2d=arr.reshape(2, 6)
arr_2d
```

Out[40]:

```
array([[ 0,  1,  2,  3,  4,  5],
       [ 6,  7,  8,  9, 10, 11]])
```

In [42]:

```
arr_2d=arr.reshape(3, 4)
arr_2d
```

Out[42]:

```
array([[ 0,  1,  2,  3],
       [ 4,  5,  6,  7],
       [ 8,  9, 10, 11]])
```

5.Ans

In [21]:

```
# Creating two numpy Square arrays

m1 = np.array([[1, 2],
               [3, 4]])

m2 = np.array([[5, 6],
               [7, 8]])

# vertical stacking

print("Vertical stacking:\n", np.vstack((m1, m2)))

# horizontal stacking

print("\nHorizontal stacking:\n", np.hstack((m1, m2)))
```

```
Vertical stacking:
 [[1 2]
 [3 4]
 [5 6]
 [7 8]]

Horizontal stacking:
 [[1 2 5 6]
 [3 4 7 8]]
```

6.Ans

In [37]:

```
# Creating lists

list =[1, 2, 1, 1, 3, 4, 3, 3, 5]
list1 = [10, 20, 10, 30, 40, 40]

# Creating Function to get unique values from list.

def unique(x):
  # intilize a null list

    unique_list = []
    # traverse for all elements

    for i in x:
        # check if exists in unique_list or not

        if i not in unique_list:
            unique_list.append(i)
    # print list

    print("UNIQUE ELEMENTS = ", unique_list)

    # Finding sum of unique elements
    sum = 0
    for i in unique_list:
        sum=sum+i
    # printing sum

    print("SUM = ",sum)
```

In [38]:

```
unique(list)
```

```
UNIQUE ELEMENTS =  [1, 2, 3, 4, 5]
SUM =  15
```

In [39]:

```
unique(list1)
```

```
UNIQUE ELEMENTS =  [10, 20, 30, 40]
SUM =  100
```