Finding Union

```
arr1 = np.array([1,2,3,4])
arr2 = np.array([3,4,5,6])
newarr = np.union1d(arr1,arr2,
                    assume_unique=True)
print(newarr)
```

O/P

[1 2 3 4 5 6]

Finding Difference

To find only the values in the first set that is NOT present in second set, use the setdiff1d() method.

```
set1 = np.array([1,2,3,4])
set2 = np.array([3,4,5,6])
newarr2 = np.setdiff1d(set1, set2, assume_unique=
                       True)
print(newarr)
```

Finding Symmetric Difference

To find only the values that are NOT present in BOTH sets, use the setxor1d() method.

```
set1 = np.array([1,2,3,4])
set2 = np.array([3,4,5,6])
newarr2 = np.setxor1d(set1, set2, assume_unique=
                      True)
print(newarr)
```

O/P

[1 2 5 6]

Finding Intersections

```
arr1 = np.array([1,2,3,4])
arr2 = np.array([3,4,5,6])
newarr = np.intersect1d(arr1, arr2, assume_unique=
                        True)
print(newarr)
```

O/P

[3 4]