

## Experiment No - 03

Aim : Write a program to implement intersection of two list

Theory

Set - set can perform unique value and ordered cannot maintain and repetition is removed from the sets.

- ~~Intersection of set~~
- The set contain only items that exist in both sets, or in all sets if the comparsion is done with more than two sets.
- The intersection () method returns a set that contains the similarity between two or more sets.

The intersection () method returns a set that contains the similarity between two or more sets.

Syntax :  
`set.intersection(set1, set2, etc)`

- AND (&) operator
- we can also get intersections using '&' operator.

e.g

`set 1 = { 2, 4, 5, 6 }`

`set 2 = { 4, 6, 7, 8 }`

`set 3 = { 1, 0, 12 }`

`print ( set 1 & set 2 )`

`print ( set 1 & set 3 )`

Output :

`{ 4, 6 }`

`set c )`

## Programs:-

```

1] list 1 = [ "ncer", 1, 4, 2]
   list 2 = [ "abc", 1, 5, 6, "ncer"]
   new_list = []
   for element in list 1:
       if element in list 2:
           new_list.append(element)
   print(new_list)
  
```

Output : [1, 'ncer']

### Explanation :-

In this above python program for intersection of two list we use for loop and append() function. Here, we created a new list to print the intersection of two sets using append() that is new list append the elements which are common in list 1 & list 2.

```

2] list 1 = [ 1, 2, 3, 4, 5, 6, 1, 2]
   list 2 = [ " abc", 1, 2, 3, 4]
   s = set(list 1)
   s1 = set(list 2)
   print(s)
   print(s1)
   new_set = s & s1
   print(new_set)
  
```

Output :  $\{1, 2, 3, 4, 5, 6\}$

$\{1, 2, 3, 4, 5, 6\}$  is the output of the set.

$\{1, 2, 3, 4, \text{abc}\}$  is the output of the set.

$\{1, 2, 3, 4\}$  is the output of the set.

So, we can see that the output of the set is unique.

**Explanation:** Now about the difference.

In the above example, we have created a list of six elements using square brackets enclosing the elements.

Moreover, we can also observe that one element is repeated in the list, which implies that list elements are ordered, changeable, and allows duplicate value.

In the output all this duplicate values are removed because of typecasting of list to set.

~~3] def intersection (list1, list2):~~

~~batsortemp = set (list2)~~

~~list3 = [value for value in list1 if value~~

~~in temp]~~

~~if list3 == list1:~~

~~return list3~~

~~else:~~

~~return list3~~

**Output:**  $[9, 9, 11]$

$[9, 9, 11]$  is the output of the set.

**Explanation :** The intersection of two lists means we need to get all the familiar elements to both of the initial list. Python is known for its excellent built-in data structure. Python list is one of the famous and valuable built-in data types of python. It can store the various data-types value in sorted order.

4] **Intersection of 2 list using hybrid approach.**

```
set1 = {"apple", "banana", "cherry"}  
set2 = {"google", "microsoft", "apple"}
```

```
new_set = set1.intersection(set2)
```

```
print(new_set)
```

**Output :**

```
{apple}
```

**Explanation :** In above program simply we perform the operation of intersection of two set i.e. set1 & set2 here we created new variable new\_set which perform the operation by using function set1.intersection(set2) and assign the set of common elements of set1 & set2.

**Conclusion :-**

In this experiment we performed the operation intersection of two list & two sets by using different methods, functions like append(), set intersection set() etc. & conversion list to set.