|  |  |  |
| --- | --- | --- |
|  |  | Set |
| Creation | Simple Method | thisSet = {"apple", "banana", "cherry", "orange"} |
|  | by USING Constructor | thisSet = Set(("apple", "banana", "cherry", "orange")) |
| Access / Slicing Items |  | You cannot access items in a Set by referring to an index, since Sets are unordered the items has no index.  But you can loop through the Set items using a for loop, |
| Loop Through |  | #To print all value:- |
| for x in thisSet:  print(x) |
| apple banana cherry orange kiwi melon mango |
| Length |  | len(abc) |
| Check if Item Exists |  | print("banana" in thisSet) |
| Changing the item |  | Once a Set is created, you cannot change its items, but you can add new items. |
| Adding an Item | To add an item at the Last index | thisSet.add("orange") |
|  |  | thisSet.update(["Litchies", "Kaju", "Angoor"]) |
|  | To add an item at the specified index | Since tuple is unidexed hence element can not be added at specific index |
| Remove | To removes the specified item: | thisSet.remove("banana") |
|  |  | thisSet.discard("banana") |
|  | To removes the Last item: | thisSet.pop() |
| Clear | To empty the Set: | thisSet.clear() |
| Delete | To delete the Data Completely | del thisSet |
| Copy |  | x = fruits.**copy()** print(x) |
| Join |  | Set Example abc = {"a", "b" , "c"} xyz = {1, 2, 3} |
|  | by + operator | Set can not be joined |
|  | Method 01 | for x in abc:  xyz.**add**(x)  print(xyz) |
| Number of elements in dataSet |  | ***---*** |
| Index of the first element with the specified value |  | ***---*** |
| Reverses the order |  | ***---*** |
| Sorts the Set |  | ***---*** |