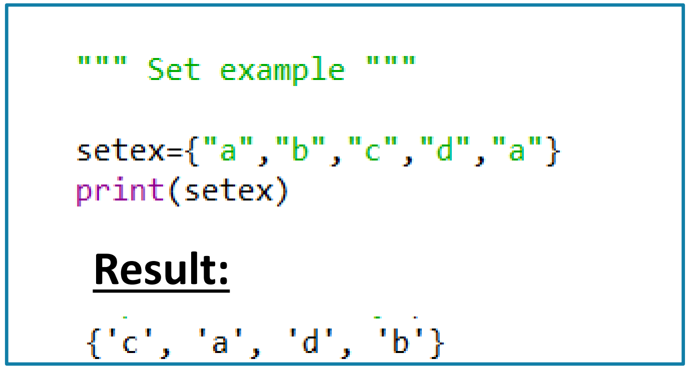
Sets, Tuples & Dictionary

Sets

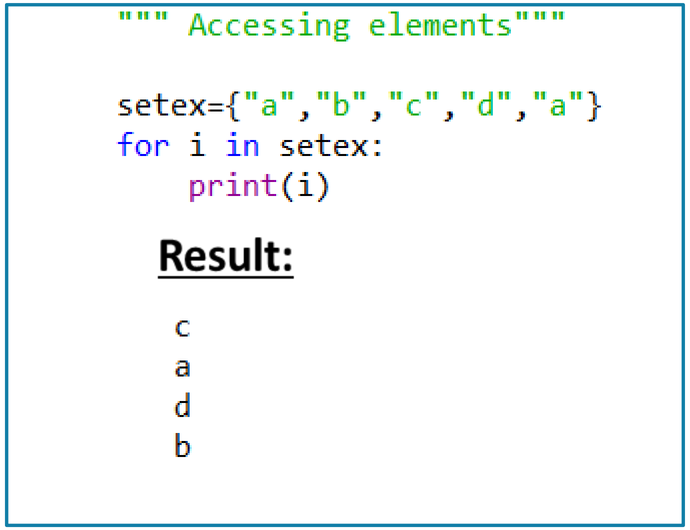
- Immutable
- No Duplication
- Unordered



```python
""" Set example """

setex={"a","b","c","d","a"}
print(setex)
```

Result:

```
{'c', 'a', 'd', 'b'}
```

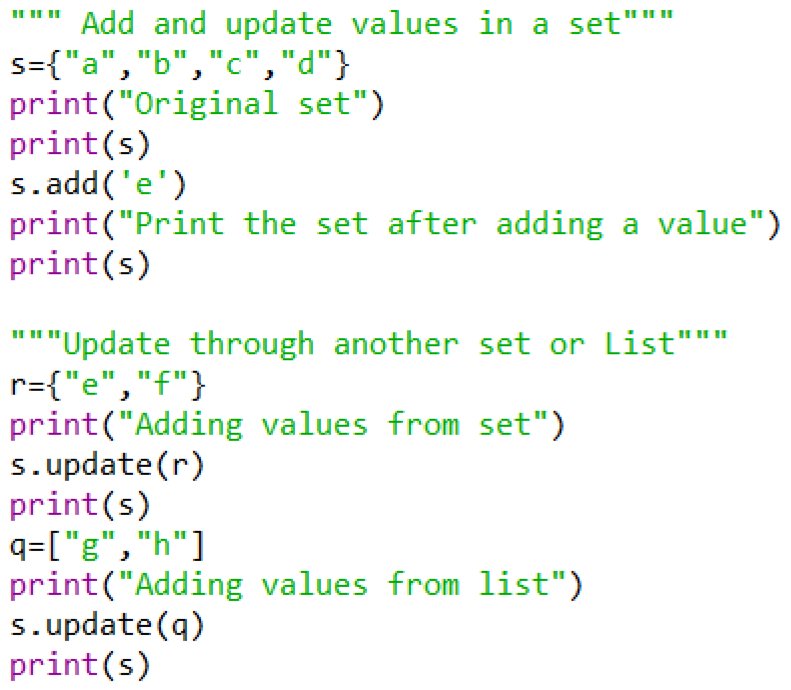


```python
""" Accessing elements"""

setex={"a","b","c","d","a"}
for i in setex:
    print(i)
```

Result:

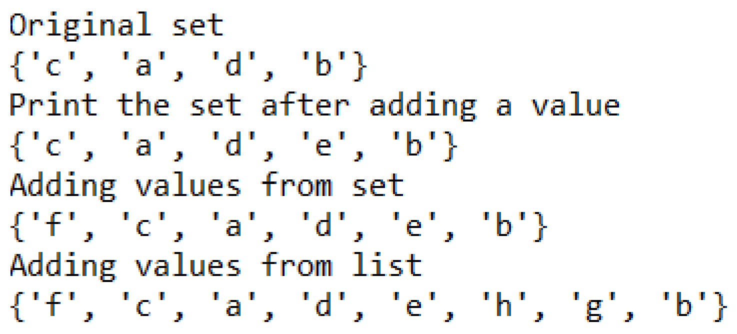
```
c
a
d
b
```

Sets- Add and Update



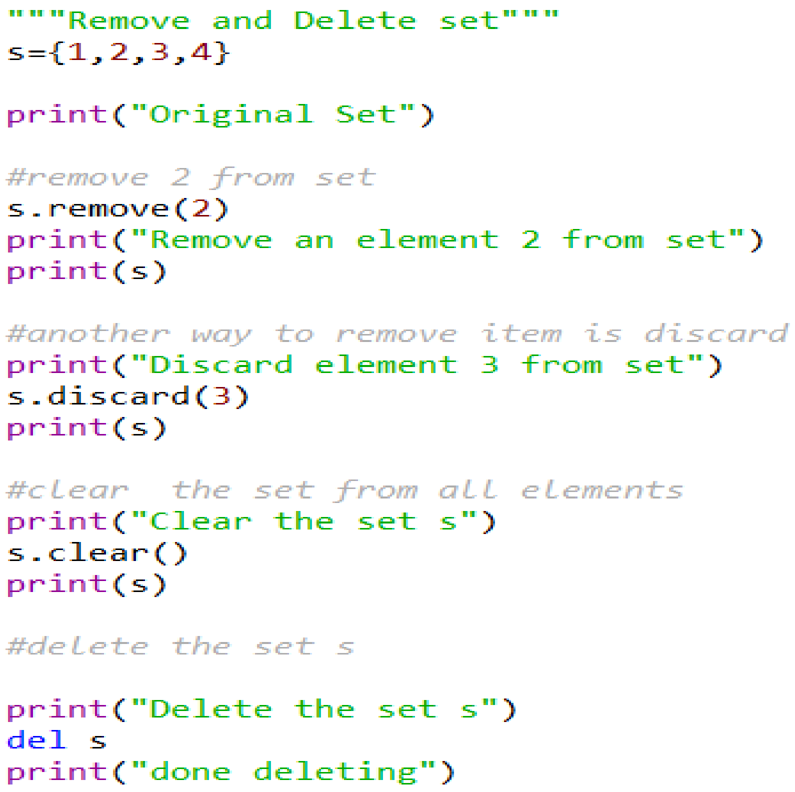
```
""" Add and update values in a set"""
s={"a","b","c","d"}
print("Original set")
print(s)
s.add('e')
print("Print the set after adding a value")
print(s)

"""Update through another set or List"""
r={"e","f"}
print("Adding values from set")
s.update(r)
print(s)
q=["g","h"]
print("Adding values from list")
s.update(q)
print(s)
```

Result:



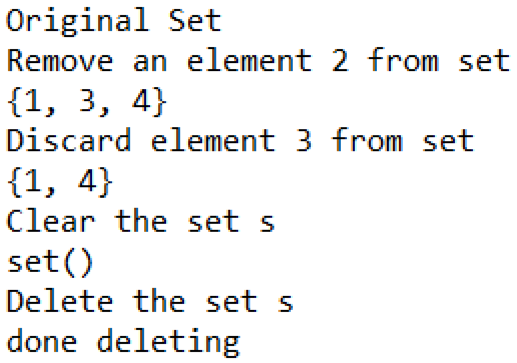
```
Original set
{'c', 'a', 'd', 'b'}
Print the set after adding a value
{'c', 'a', 'd', 'e', 'b'}
Adding values from set
{'f', 'c', 'a', 'd', 'e', 'b'}
Adding values from list
{'f', 'c', 'a', 'd', 'e', 'h', 'g', 'b'}
```

Sets - Remove



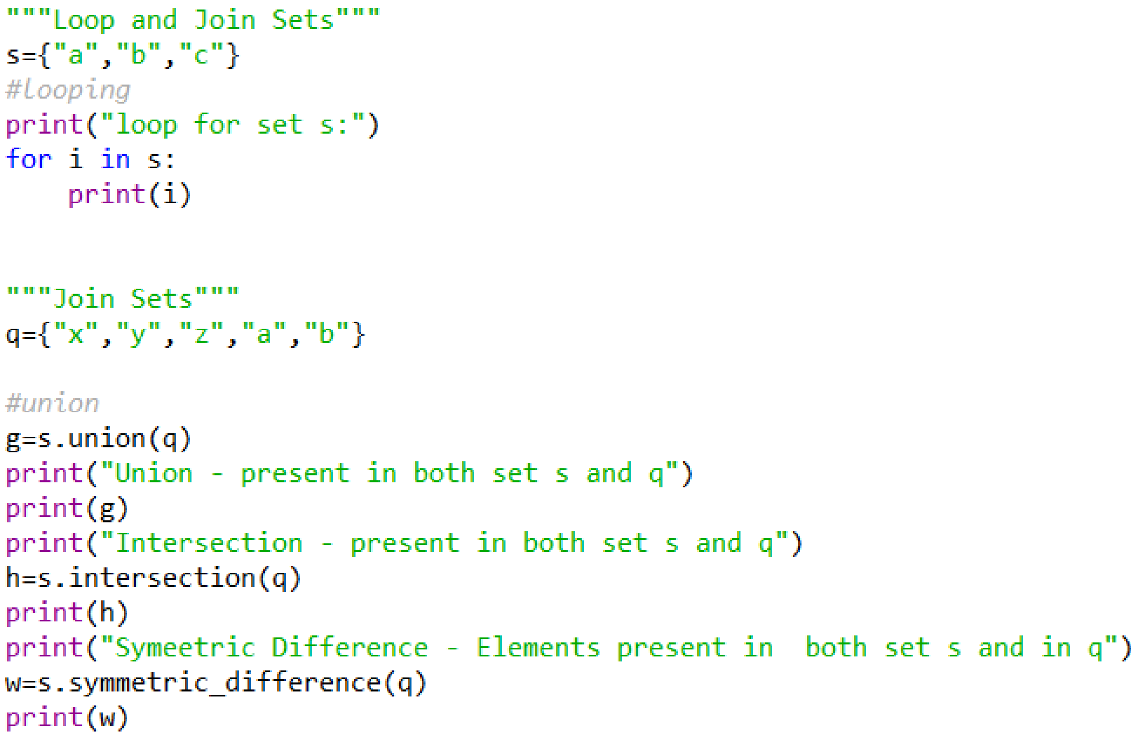
```python
"""Remove and Delete set"""
s={1,2,3,4}

print("Original Set")

#remove 2 from set
s.remove(2)
print("Remove an element 2 from set")
print(s)

#another way to remove item is discard
print("Discard element 3 from set")
s.discard(3)
print(s)

#clear  the set from all elements
print("Clear the set s")
s.clear()
print(s)

#delete the set s

print("Delete the set s")
del s
print("done deleting")
```

Result:



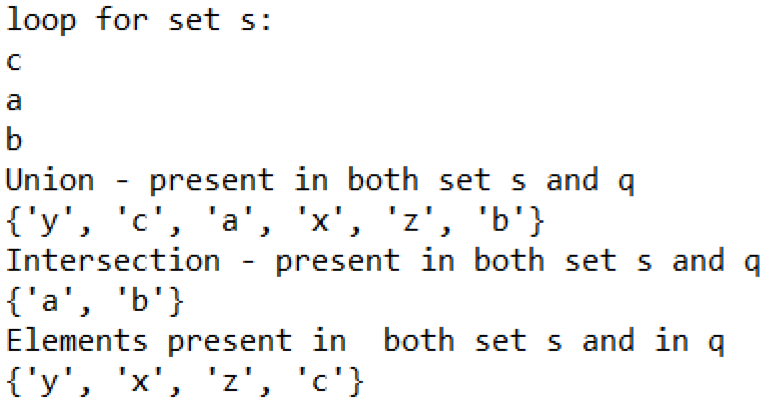
```
Original Set
Remove an element 2 from set
{1, 3, 4}
Discard element 3 from set
{1, 4}
Clear the set s
set()
Delete the set s
done deleting
```

Sets - Loop & Join



```
"""Loop and Join Sets"""
s={"a","b","c"}
#looping
print("loop for set s:")
for i in s:
    print(i)


"""Join Sets"""
q={"x","y","z","a","b"}

#union
g=s.union(q)
print("Union - present in both set s and q")
print(g)
print("Intersection - present in both set s and q")
h=s.intersection(q)
print(h)
print("Symeetric Difference - Elements present in  both set s and in q")
w=s.symmetric_difference(q)
print(w)
```

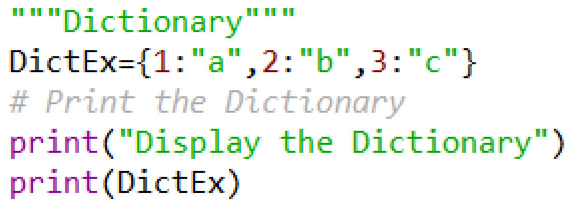
Result:



```
loop for set s:
c
a
b
Union - present in both set s and q
{'y', 'c', 'a', 'x', 'z', 'b'}
Intersection - present in both set s and q
{'a', 'b'}
Elements present in  both set s and in q
{'y', 'x', 'z', 'c'}
```

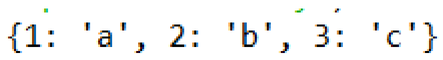
Dictionaries

- Key/value pair
- Ordered
- Changeable
- No Duplication



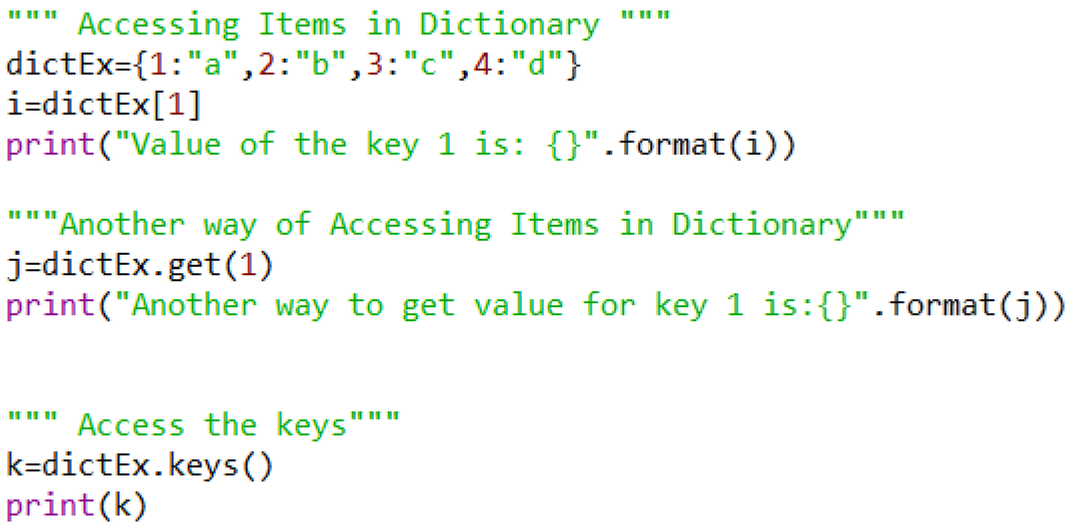
```python
"""Dictionary"""
DictEx={1:"a",2:"b",3:"c"}
# Print the Dictionary
print("Display the Dictionary")
print(DictEx)
```

Result:



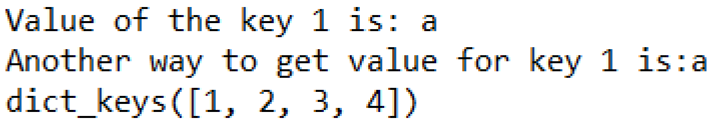
```
{1: 'a', 2: 'b', 3: 'c'}
```

Dictionary – Accessing Items



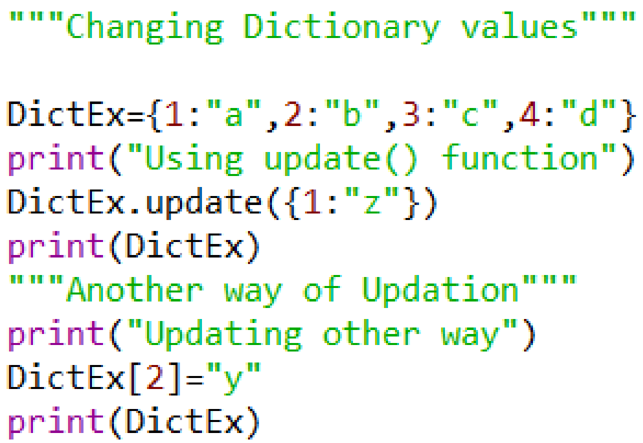
```
""" Accessing Items in Dictionary """
dictEx={1:"a",2:"b",3:"c",4:"d"}
i=dictEx[1]
print("Value of the key 1 is: {}".format(i))

"""Another way of Accessing Items in Dictionary"""
j=dictEx.get(1)
print("Another way to get value for key 1 is:{}".format(j))


""" Access the keys"""
k=dictEx.keys()
print(k)
```

Result:



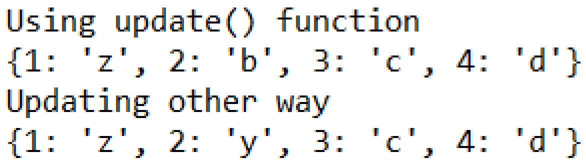
```
Value of the key 1 is: a
Another way to get value for key 1 is:a
dict_keys([1, 2, 3, 4])
```

Dictionary – Updating Values



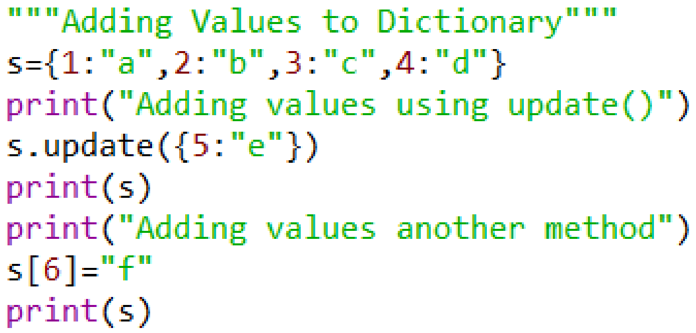
```
"""Changing Dictionary values"""

DictEx={1:"a",2:"b",3:"c",4:"d"}
print("Using update() function")
DictEx.update({1:"z"})
print(DictEx)
"""Another way of Updation"""
print("Updating other way")
DictEx[2]="y"
print(DictEx)
```

Result:



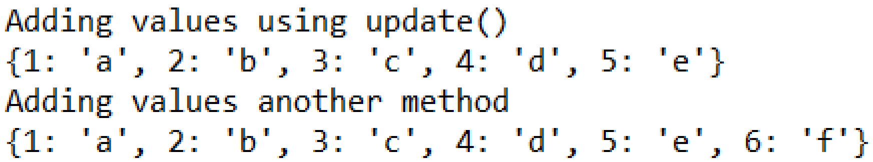
```
Using update() function
{1: 'z', 2: 'b', 3: 'c', 4: 'd'}
Updating other way
{1: 'z', 2: 'y', 3: 'c', 4: 'd'}
```

Dictionary – Adding Values



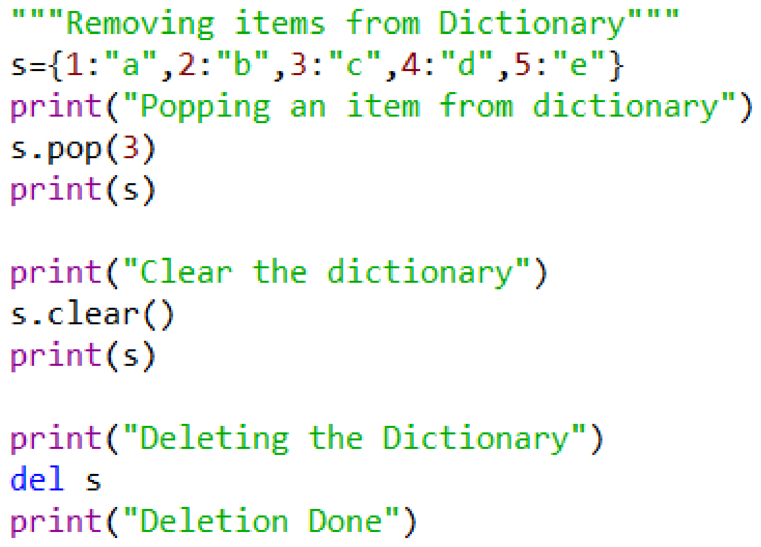
```
"""Adding Values to Dictionary"""
s={1:"a",2:"b",3:"c",4:"d"}
print("Adding values using update()")
s.update({5:"e"})
print(s)
print("Adding values another method")
s[6]="f"
print(s)
```

Result:



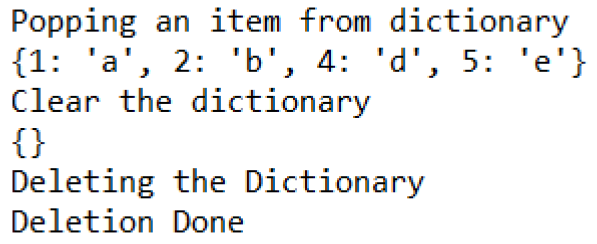
```
Adding values using update()
{1: 'a', 2: 'b', 3: 'c', 4: 'd', 5: 'e'}
Adding values another method
{1: 'a', 2: 'b', 3: 'c', 4: 'd', 5: 'e', 6: 'f'}
```

Dictionary – Removing item



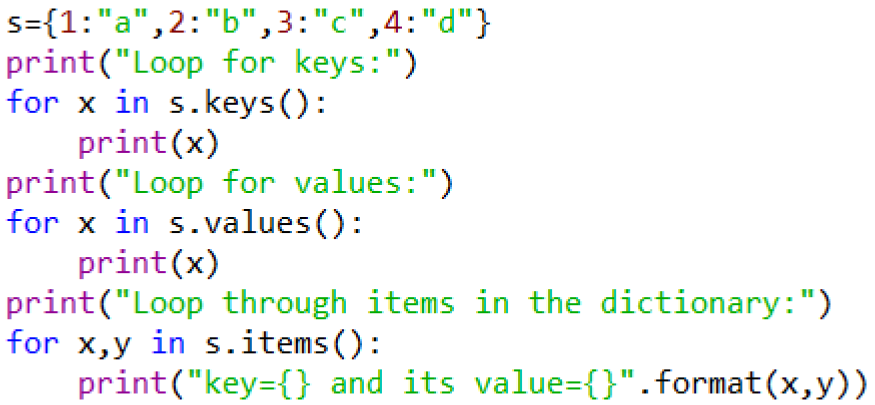
```python
"""Removing items from Dictionary"""
s={1:"a",2:"b",3:"c",4:"d",5:"e"}
print("Popping an item from dictionary")
s.pop(3)
print(s)

print("Clear the dictionary")
s.clear()
print(s)

print("Deleting the Dictionary")
del s
print("Deletion Done")
```

Result:



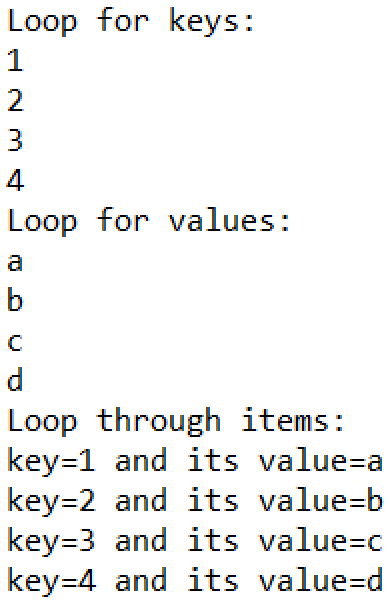
```
Popping an item from dictionary
{1: 'a', 2: 'b', 4: 'd', 5: 'e'}
Clear the dictionary
{}
Deleting the Dictionary
Deletion Done
```

Dictionary Loops



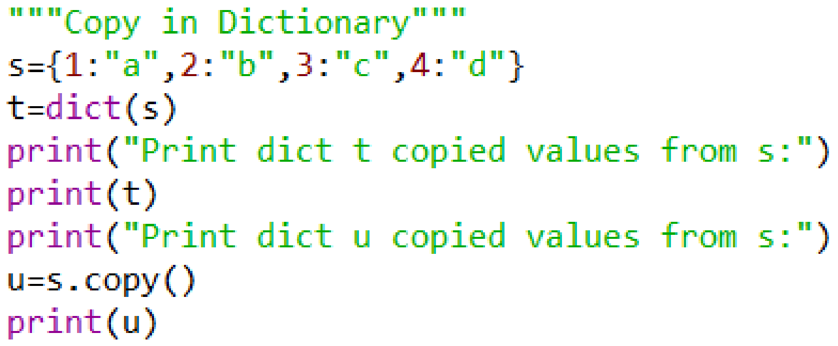
```python
s={1:"a",2:"b",3:"c",4:"d"}
print("Loop for keys:")
for x in s.keys():
    print(x)
print("Loop for values:")
for x in s.values():
    print(x)
print("Loop through items in the dictionary:")
for x,y in s.items():
    print("key={} and its value={}".format(x,y))
```

Result:



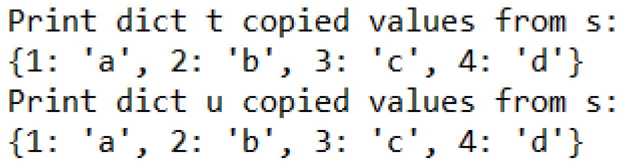
```
Loop for keys:
1
2
3
4
Loop for values:
a
b
c
d
Loop through items:
key=1 and its value=a
key=2 and its value=b
key=3 and its value=c
key=4 and its value=d
```

Dictionary - Copy



```
"""Copy in Dictionary"""
s={1:"a",2:"b",3:"c",4:"d"}
t=dict(s)
print("Print dict t copied values from s:")
print(t)
print("Print dict u copied values from s:")
u=s.copy()
print(u)
```

Result:



```
Print dict t copied values from s:
{1: 'a', 2: 'b', 3: 'c', 4: 'd'}
Print dict u copied values from s:
{1: 'a', 2: 'b', 3: 'c', 4: 'd'}
```

Sample Programs - Dictionary

1. Write a Python script to sort (ascending and descending) a dictionary by value.

2. Write a Python script to add a key to a dictionary.

Sample Dictionary : {0: 10, 1: 20}
Expected Result : {0: 10, 1: 20, 2: 30}

3. Write a Python script to concatenate following dictionaries to create a new one.

Sample Dictionary :
dic1={1:10, 2:20}
dic2={3:30, 4:40}
dic3={5:50,6:60}
Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

4. Write a Python script to check whether a given key already exists in a dictionary.

5. Write a Python program to iterate over dictionaries using for loops.

6. Write a Python script to generate and print a dictionary that contains a number (between 1 and n) in the form (x, x*x).
Sample Dictionary (n = 5) :
Expected Output : {1: 1, 2: 4, 3: 9, 4: 16, 5: 25}

7. Write a Python script to print a dictionary where the keys are numbers between 1 and 15 (both included) and the values are square of keys.
Sample Dictionary
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100, 11: 121, 12: 144, 13: 169, 14: 196, 15: 225}

8. Write a Python script to merge two Python dictionaries.

9. Write a Python program to iterate over dictionaries using for loops.

10. Write a Python program to sum all the items in a dictionary.

11. Write a Python program to multiply all the items in a dictionary.

12. Write a Python program to remove a key from a dictionary.

13. Write a Python program to map two lists into a dictionary.

14. Write a Python program to sort a dictionary by key.

15. Write a Python program to get the maximum and minimum value in a dictionary.

16. Write a Python program to get a dictionary from an object's fields.

17. Write a Python program to remove duplicates from Dictionary.

18. Write a Python program to check a dictionary is empty or not.

19. Write a Python program to combine two dictionary adding values for common keys.
d1 = {'a': 100, 'b': 200, 'c':300}
d2 = {'a': 300, 'b': 200, 'd':400}
Sample output: Counter({'a': 400, 'b': 400, 'd': 400, 'c': 300})

20. Write a Python program to print all unique values in a dictionary.
Sample Data : [{"V":"S001"}, {"V": "S002"}, {"VI": "S001"}, {"VI": "S005"}, {"VII":"S005"}, {"V":"S009"},{"VIII":"S007"}]
Expected Output : Unique Values: {'S005', 'S002', 'S007', 'S001', 'S009'}

21. Write a Python program to create and display all combinations of letters, selecting each letter from a different key in a dictionary.
Sample data : {'1':['a','b'], '2':['c','d']}
Expected Output:

```
ac
ad
bc
bd
```

22. Write a Python program to find the highest 3 values of corresponding keys in a dictionary.

23. Write a Python program to combine values in python list of dictionaries.
Sample data: [{'item': 'item1', 'amount': 400}, {'item': 'item2', 'amount': 300}, {'item': 'item1', 'amount': 750}]
Expected Output: Counter({'item1': 1150, 'item2': 300})

24. Write a Python program to create a dictionary from a string.
Note: Track the count of the letters from the string.
Sample string : 'w3resource'
Expected output: {'w': 1, '3': 1, 'r': 2, 'e': 2, 's': 1, 'o': 1, 'u': 1, 'c': 1}

25. Write a Python program to print a dictionary in table format

26. Write a Python program to count the values associated with key in a dictionary.
Expected Output:

```
6
2
```

27. Write a Python program to convert a list into a nested dictionary of keys.

28. Write a Python program to sort a list alphabetically in a dictionary.

29. Write a Python program to remove spaces from dictionary keys.

30. Write a Python program to get the top three items in a shop.
Sample data: {'item1': 45.50, 'item2':35, 'item3': 41.30, 'item4':55, 'item5': 24}
Expected Output:

```
item4 55
item1 45.5
item3 41.3
```

31. Write a Python program to get the key, value and item in a dictionary.

32. Write a Python program to print a dictionary line by line.

33. Write a Python program to check multiple keys exists in a dictionary.

34. Write a Python program to count number of items in a dictionary value that is a list.

35. Write a Python program to sort Counter by value.
Sample data : {'Math':81, 'Physics':83, 'Chemistry':87}
Expected data: [('Chemistry', 87), ('Physics', 83), ('Math', 81)]

36. Write a Python program to create a dictionary from two lists without losing duplicate values.

```
Sample lists: ['Class-V', 'Class-VI', 'Class-VII', 'Class-VIII'], [1, 2, 2, 3]
Expected Output: defaultdict(<class 'set'>, {'Class-V': {1}, 'Class-VI': {2}, 'Class-VII':
{2}, 'Class-VIII': {3}})
```

37. Write a Python program to replace dictionary values with their average.

38. Write a Python program to match key values in two dictionaries.

```
Sample dictionary: {'key1': 1, 'key2': 3, 'key3': 2}, {'key1': 1, 'key2': 2}
Expected output: key1: 1 is present in both x and y
```

39. Write a Python program to store a given dictionary in a json file.

```
Original dictionary:
{'students': [{'firstName': 'Nikki', 'lastName': 'Roysden'}, {'firstName': 'Mervin',
'lastName': 'Friedland'}, {'firstName': 'Aron ', 'lastName': 'Wilkins'}], 'teachers':
[{'firstName': 'Amberly', 'lastName': 'Calico'}, {'firstName': 'Regine', 'lastName':
'Agtarap'}]}
<class 'dict'>
Json file to dictionary:
{'students': [{'firstName': 'Nikki', 'lastName': 'Roysden'}, {'firstName': 'Mervin',
'lastName': 'Friedland'}, {'firstName': 'Aron ', 'lastName': 'Wilkins'}], 'teachers':
[{'firstName': 'Amberly', 'lastName': 'Calico'}, {'firstName': 'Regine', 'lastName':
'Agtarap'}]}
```

40. Write a Python program to create a dictionary of keys x, y, and z where each key has as value a list from 11-20, 21-30, and 31-40 respectively. Access the fifth value of each key from the dictionary.

```
{'x': [11, 12, 13, 14, 15, 16, 17, 18, 19],
'y': [21, 22, 23, 24, 25, 26, 27, 28, 29],
'z': [31, 32, 33, 34, 35, 36, 37, 38, 39]}
15
25
35
x has value [11, 12, 13, 14, 15, 16, 17, 18, 19]
y has value [21, 22, 23, 24, 25, 26, 27, 28, 29]
z has value [31, 32, 33, 34, 35, 36, 37, 38, 39]
```

41. Write a Python program to drop empty Items from a given Dictionary.

```
Original Dictionary:
{'c1': 'Red', 'c2': 'Green', 'c3': None}
New Dictionary after dropping empty items:
{'c1': 'Red', 'c2': 'Green'}
```

42. Write a Python program to filter a dictionary based on values.

```
Original Dictionary:
{'Cierra Vega': 175, 'Alden Cantrell': 180, 'Kierra Gentry': 165, 'Pierre Cox': 190}
Marks greater than 170:
{'Cierra Vega': 175, 'Alden Cantrell': 180, 'Pierre Cox': 190}
```

43. Write a Python program to convert more than one list to nested dictionary.

```
Original strings:
['S001', 'S002', 'S003', 'S004']
['Adina Park', 'Leyton Marsh', 'Duncan Boyle', 'Saim Richards']
[85, 98, 89, 92]
Nested dictionary:
[{'S001': {'Adina Park': 85}}, {'S002': {'Leyton Marsh': 98}}, {'S003': {'Duncan Boyle':
89}}, {'S004': {'Saim Richards': 92}}]
```

44. Write a Python program to filter the height and width of students, which are stored in a dictionary.

```
Original Dictionary:
{'Cierra Vega': (6.2, 70), 'Alden Cantrell': (5.9, 65), 'Kierra Gentry': (6.0, 68), 'Pierre
Cox': (5.8, 66)}
Height > 6ft and Weight> 70kg:
{'Cierra Vega': (6.2, 70)}
```

45. Write a Python program to check all values are same in a dictionary.

```
Original Dictionary:
{'Cierra Vega': 12, 'Alden Cantrell': 12, 'Kierra Gentry': 12, 'Pierre Cox': 12}
Check all are 12 in the dictionary.
True
Check all are 10 in the dictionary.
False
```

46. Write a Python program to create a dictionary grouping a sequence of key-value pairs into a dictionary of lists.

```
Original list:
[('yellow', 1), ('blue', 2), ('yellow', 3), ('blue', 4), ('red', 1)]
Grouping a sequence of key-value pairs into a dictionary of lists:
{'yellow': [1, 3], 'blue': [2, 4], 'red': [1]}
```

47. Write a Python program to split a given dictionary of lists into list of dictionaries.

```
Original dictionary of lists:
{'Science': [88, 89, 62, 95], 'Language': [77, 78, 84, 80]}
Split said dictionary of lists into list of dictionaries:
[{'Science': 88, 'Language': 77}, {'Science': 89, 'Language': 78}, {'Science': 62,
'Language': 84}, {'Science': 95, 'Language': 80}]
```

48. Write a Python program to remove a specified dictionary from a given list.

```
Original list of dictionary:
[{'id': '#FF0000', 'color': 'Red'}, {'id': '#800000', 'color': 'Maroon'}, {'id': '#FFFF00', 'color':
'Yellow'}, {'id': '#808000', 'color': 'Olive'}]
Remove id #FF0000 from the said list of dictionary:
[{'id': '#800000', 'color': 'Maroon'}, {'id': '#FFFF00', 'color': 'Yellow'}, {'id': '#808000',
'color': 'Olive’}]
```

49. Write a Python program to convert string values of a given dictionary, into integer/float datatypes.

```
Original list:
[{'x': '10', 'y': '20', 'z': '30'}, {'p': '40', 'q': '50', 'r': '60'}]
String values of a given dictionary, into integer types:
[{'x': 10, 'y': 20, 'z': 30}, {'p': 40, 'q': 50, 'r': 60}]
Original list:
[{'x': '10.12', 'y': '20.23', 'z': '30'}, {'p': '40.00', 'q': '50.19', 'r': '60.99'}]
String values of a given dictionary, into float types:
[{'x': 10.12, 'y': 20.23, 'z': 30.0}, {'p': 40.0, 'q': 50.19, 'r': 60.99}]
```

50. A Python Dictionary contains List as value. Write a Python program to clear the list values in the said dictionary.

```
Original Dictionary:
{'C1': [10, 20, 30], 'C2': [20, 30, 40], 'C3': [12, 34]}
Clear the list values in the said dictionary:
{'C1': [], 'C2': [], 'C3': []}
```

51. A Python Dictionary contains List as value. Write a Python program to update the list values in the said dictionary.

```
Original Dictionary:
{'Math': [88, 89, 90], 'Physics': [92, 94, 89], 'Chemistry': [90, 87, 93]}
Update the list values of the said dictionary:
{'Math': [89, 90, 91], 'Physics': [90, 92, 87], 'Chemistry': [90, 87, 93]}
```

52. Write a Python program to extract a list of values from a given list of dictionaries.

```
Original Dictionary:
[{'Math': 90, 'Science': 92}, {'Math': 89, 'Science': 94}, {'Math': 92, 'Science': 88}]
Extract a list of values from said list of dictionaries where subject = Science
[92, 94, 88]
Original Dictionary:
[{'Math': 90, 'Science': 92}, {'Math': 89, 'Science': 94}, {'Math': 92, 'Science': 88}]
Extract a list of values from said list of dictionaries where subject = Math
[90, 89, 92]
```

53. Write a Python program to find the length of a given dictionary values.

```
Original Dictionary:
{1: 'red', 2: 'green', 3: 'black', 4: 'white', 5: 'black'}
Length of dictionary values:
{'red': 3, 'green': 5, 'black': 5, 'white': 5}
Original Dictionary:
{'1': 'Austin Little', '2': 'Natasha Howard', '3': 'Alfred Mullins', '4': 'Jamie Rowe'}
Length of dictionary values:
{'Austin Little': 13, 'Natasha Howard': 14, 'Alfred Mullins': 14, 'Jamie Rowe': 10}
```

54. Write a Python program to get the depth of a dictionary.

```
Expected Output:
4
```

55. Write a Python program to access dictionary key's element by index.

```
Expected Output:
physics
math
chemistry
```

56. Write a Python program to convert a given dictionary into a list of lists.

```
Original Dictionary:
{1: 'red', 2: 'green', 3: 'black', 4: 'white', 5: 'black'}
Convert the said dictionary into a list of lists:
[[1, 'red'], [2, 'green'], [3, 'black'], [4, 'white'], [5, 'black']]
Original Dictionary:
{'1': 'Austin Little', '2': 'Natasha Howard', '3': 'Alfred Mullins', '4': 'Jamie Rowe'}
Convert the said dictionary into a list of lists:
[['1', 'Austin Little'], ['2', 'Natasha Howard'], ['3', 'Alfred Mullins'], ['4', 'Jamie Rowe’]]
```

57. Write a Python program to filter even numbers from a given dictionary values.

```
Original Dictionary:
{'V': [1, 4, 6, 10], 'VI': [1, 4, 12], 'VII': [1, 3, 8]}
Filter even numbers from said dictionary values:
{'V': [4, 6, 10], 'VI': [4, 12], 'VII': [8]}
Original Dictionary:
{'V': [1, 3, 5], 'VI': [1, 5], 'VII': [2, 7, 9]}
Filter even numbers from said dictionary values:
{'V': [], 'VI': [], 'VII': [2]}
```

58. Write a Python program to get all combinations of key-value pairs in a given dictionary.

```
Original Dictionary:
{'V': [1, 4, 6, 10], 'VI': [1, 4, 12], 'VII': [1, 3, 8]}
Combinations of key-value pairs of the said dictionary:
[{'V': [1, 4, 6, 10], 'VI': [1, 4, 12]}, {'V': [1, 4, 6, 10], 'VII': [1, 3, 8]}, {'VI': [1, 4, 12], 'VII': [1, 3, 8]}]
Original Dictionary:
{'V': [1, 3, 5], 'VI': [1, 5]}
Combinations of key-value pairs of the said dictionary:
[{'V': [1, 3, 5], 'VI': [1, 5]}]
```

59. Write a Python program to find the specified number of maximum values in a given dictionary.

```
Original Dictionary:
{'a': 5, 'b': 14, 'c': 32, 'd': 35, 'e': 24, 'f': 100, 'g': 57, 'h': 8, 'i': 100}
1 maximum value(s) in the said dictionary:
['f']
2 maximum value(s) in the said dictionary:
['f', 'i']
5 maximum value(s) in the said dictionary:
['f', 'i', 'g', 'd', 'c’]
```

60. Write a Python program to find shortest list of values with the keys in a given dictionary
Original Dictionary: {'V': [10, 12], 'VI': [10], 'VII': [10, 20, 30, 40], 'VIII': [20], 'IX': [10, 30, 50, 70], 'X': [80]} Shortest list of values with the keys of the said dictionary: ['VI', 'VIII', 'X’]

61. Write a Python program to count the frequency in a given dictionary.

```
Original Dictionary:
{'V': 10, 'VI': 10, 'VII': 40, 'VIII': 20, 'IX': 70, 'X': 80, 'XI': 40, 'XII': 20}
Count the frequency of the said dictionary:
Counter({10: 2, 40: 2, 20: 2, 70: 1, 80: 1})
```

64. Write a Python program to create a key-value list pairings in a given dictionary.
Original dictionary:
{1: ['Jean Castro'], 2: ['Lula Powell'], 3: ['Brian Howell'], 4: ['Lynne Foster'], 5: ['Zachary Simon']}
A key-value list pairings of the said dictionary:
[{1: 'Jean Castro', 2: 'Lula Powell', 3: 'Brian Howell', 4: 'Lynne Foster', 5: 'Zachary Simon’}]

65. Write a Python program to get the total length of all values of a given dictionary with string values.
Original dictionary:
{'#FF0000': 'Red', '#800000': 'Maroon', '#FFFF00': 'Yellow', '#808000': 'Olive'}
Total length of all values of the said dictionary with string values:
20

66. Write a Python program to check if a specific Key and a value exist in a dictionary.
Original dictionary:
[{'student_id': 1, 'name': 'Jean Castro', 'class': 'V'}, {'student_id': 2, 'name': 'Lula Powell', 'class': 'V'}, {'student_id': 3, 'name': 'Brian Howell', 'class': 'VI'}, {'student_id': 4, 'name': 'Lynne Foster', 'class': 'VI'}, {'student_id': 5, 'name': 'Zachary Simon', 'class': 'VII'}]
Check if a specific Key and a value exist in the said dictionary:
True
True
True
False
False
False

62. Write a Python program to extract values from a given dictionaries and create a list of lists from those values.
Original Dictionary:
[{'student_id': 1, 'name': 'Jean Castro', 'class': 'V'}, {'student_id': 2, 'name': 'Lula Powell', 'class': 'V'}, {'student_id': 3, 'name': 'Brian Howell', 'class': 'VI'}, {'student_id': 4, 'name': 'Lynne Foster', 'class': 'VI'}, {'student_id': 5, 'name': 'Zachary Simon', 'class': 'VII'}]
Extract values from the said dictionarie and create a list of lists using those values:
[[1, 'Jean Castro', 'V'], [2, 'Lula Powell', 'V'], [3, 'Brian Howell', 'VI'], [4, 'Lynne Foster', 'VI'], [5, 'Zachary Simon', 'VII']]
[[1, 'Jean Castro'], [2, 'Lula Powell'], [3, 'Brian Howell'], [4, 'Lynne Foster'], [5, 'Zachary Simon']]
[['Jean Castro', 'V'], ['Lula Powell', 'V'], ['Brian Howell', 'VI'], ['Lynne Foster', 'VI'], ['Zachary Simon', 'VII’]]

63. Write a Python program to convert a given list of lists to a dictionary.
Original list of lists:
[[1, 'Jean Castro', 'V'], [2, 'Lula Powell', 'V'], [3, 'Brian Howell', 'VI'], [4, 'Lynne Foster', 'VI'], [5, 'Zachary Simon', 'VII']]
Convert the said list of lists to a dictionary:
{1: ['Jean Castro', 'V'], 2: ['Lula Powell', 'V'], 3: ['Brian Howell', 'VI'], 4: ['Lynne Foster', 'VI'], 5: ['Zachary Simon', 'VII']}

Tuples

1. Write a Python program to create a tuple.

2. Write a Python program to create a tuple with different data types

3. Write a Python program to create a tuple with numbers and print one item

4. Write a Python program to unpack a tuple in several variables.

5. Write a Python program to add an item in a tuple.

6. Write a Python program to convert a tuple to a string.

7. Write a Python program to get the 4th element and 4th element from last of a tuple.

8. Write a Python program to create the colon of a tuple.

9. Write a Python program to find the repeated items of a tuple.

10. Write a Python program to check whether an element exists within a tuple.

11. Write a Python program to convert a list to a tuple.

12. Write a Python program to remove an item from a tuple.

13. Write a Python program to slice a tuple.

14. Write a Python program to find the index of an item of a tuple.

15. Write a Python program to find the length of a tuple.

16. Write a Python program to convert a tuple to a dictionary.

17. Write a Python program to unzip a list of tuples into individual lists.

18. Write a Python program to reverse a tuple.

19. Write a Python program to convert a list of tuples into a dictionary.

20. Write a Python program to print a tuple with string formatting.
Sample tuple : (100, 200, 300)
Output : This is a tuple (100, 200, 300)

21. Write a Python program to replace last value of tuples in a list.
Sample list: [(10, 20, 40), (40, 50, 60), (70, 80, 90)]
Expected Output: [(10, 20, 100), (40, 50, 100), (70, 80, 100)]

22. Write a Python program to remove an empty tuple(s) from a list of tuples.
Sample data: [(), (), ('',), ('a', 'b'), ('a', 'b', 'c'), ('d')]
Expected output: [('',), ('a', 'b'), ('a', 'b', 'c'), 'd']

23. Write a Python program to sort a tuple by its float element.
Sample data: [('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5')]
Expected Output: [('item3', '24.5'), ('item2', '15.10'), ('item1', '12.20’)]

24. Write a Python program to count the elements in a list until an element is a tuple.

25. Write a Python program convert a given string list to a tuple.
Original string: python 3.0
<class 'str'>
Convert the said string to a tuple:
('p', 'y', 't', 'h', 'o', 'n', '3', '.', '0')
<class 'tuple’>

26. Write a Python program calculate the product, multiplying all the numbers of a given tuple.
Original Tuple:
(4, 3, 2, 2, -1, 18)
Product - multiplying all the numbers of the said tuple: -864
Original Tuple:
(2, 4, 8, 8, 3, 2, 9)
Product - multiplying all the numbers of the said tuple: 27648

27. Write a Python program to calculate the average value of the numbers in a given tuple of tuples.
Original Tuple:
((10, 10, 10, 12), (30, 45, 56, 45), (81, 80, 39, 32), (1, 2, 3, 4))
Average value of the numbers of the said tuple of tuples:
[30.5, 34.25, 27.0, 23.25]
Original Tuple:
((1, 1, -5), (30, -15, 56), (81, -60, -39), (-10, 2, 3))
Average value of the numbers of the said tuple of tuples:
[25.5, -18.0, 3.75]

28. Write a Python program to convert a tuple of string values to a tuple of integer values.
Original tuple values:
(('333', '33'), ('1416', '55'))
New tuple values:
((333, 33), (1416, 55))

29. Write a Python program to convert a given tuple of positive integers into an integer.
Original tuple:
(1, 2, 3)
Convert the said tuple of positive integers into an integer:
123
Original tuple:
(10, 20, 40, 5, 70)
Convert the said tuple of positive integers into an integer:
102040570

30. Write a Python program to check if a specified element presents in a tuple of tuples.

```
Original list:
(('Red', 'White', 'Blue'), ('Green', 'Pink', 'Purple'), ('Orange', 'Yellow', 'Lime'))
Check if White presenet in said tuple of tuples!
True
Check if White presenet in said tuple of tuples!
True
Check if Olive presenet in said tuple of tuples!
False
```

31. Write a Python program to compute element-wise sum of given tuples.

```
Original lists:
(1, 2, 3, 4)
(3, 5, 2, 1)
(2, 2, 3, 1)
Element-wise sum of the said tuples:
(6, 9, 8, 6)
```

32. Write a Python program to compute the sum of all the elements of each tuple stored inside a list of tuples.

```
Original list of tuples:
[(1, 2), (2, 3), (3, 4)]
Sum of all the elements of each tuple stored inside the said list of tuples:
[3, 5, 7]
Original list of tuples:
[(1, 2, 6), (2, 3, -6), (3, 4), (2, 2, 2, 2)]
Sum of all the elements of each tuple stored inside the said list of tuples:
[9, -1, 7, 8]
```

33. Write a Python program to convert a given list of tuples to a list of lists.

```
Original list of tuples: [(1, 2), (2, 3), (3, 4)]
Convert the said list of tuples to a list of lists: [[1, 2], [2, 3], [3, 4]]
Original list of tuples: [(1, 2), (2, 3, 5), (3, 4), (2, 3, 4, 2)]
Convert the said list of tuples to a list of lists: [[1, 2], [2, 3, 5], [3, 4], [2, 3, 4, 2]]
```

Sets

1. Write a Python program to create a set.

2. Write a Python program to iterate over sets.

3. Write a Python program to add member(s) in a set.

4. Write a Python program to remove item(s) from set

5. Write a Python program to remove an item from a set if it is present in the set.

6. Write a Python program to create an intersection of sets.

7. Write a Python program to create a union of sets.

8. Write a Python program to create set difference.

9. Write a Python program to create a symmetric difference.

10. Write a Python program to check if a set is a subset of another set.

11. Write a Python program to create a shallow copy of sets.

Note : Shallow copy is a bit-wise copy of an object. A new object is created that has an exact copy of the values in the original object.

12. Write a Python program to clear a set

13. Write a Python program to use of frozensets.
Note: Frozensets behave just like sets except they are immutable.

14. Write a Python program to find maximum and the minimum value in a set

15. Write a Python program to find the length of a set.

16. Write a Python program to check if a given value is present in a set or

17. Write a Python program to check if two given sets have no elements in common.

18. Write a Python program to check if a given set is superset of itself and superset of another given set.

19. Write a Python program to find the elements in a given set that are not in another set.

20. Write a Python program to check a given set has no elements in common with other given set

21. Write a Python program to remove the intersection of a 2nd set from the 1st set.