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

```
def sort_dictionary_by_value(dictionary, ascending=True):
    sorted_dict = dict(sorted(dictionary.items(), key=lambda item: item[1], reverse=not
ascending))
    return sorted_dict

input_dict = {}
num_items = int(input("Enter the number of items in the dictionary: "))
for i in range(num_items):
    key = input("Enter key: ")
    value = input("Enter value: ")
    input_dict[key] = value

ascending_sorted_dict = sort_dictionary_by_value(input_dict)
print("Ascending order:", ascending_sorted_dict)

descending_sorted_dict = sort_dictionary_by_value(input_dict, ascending=False)
print("Descending order:", descending_sorted_dict)
```

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}

```
d = {0: 10, 1: 20}
d[2] = 30
print(d)
```

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}

```
dic1 = {1: 10, 2: 20}
dic2 = {3: 30, 4: 40}
dic3 = {5: 50, 6: 60}
result = {}
result.update(dic1)
result.update(dic2)
result.update(dic3)
print(result)
```

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

```
sample_dict = {0: 10, 1: 20}
```

```
key_to_check = 1

if key_to_check in sample_dict:
    print("Key exists")
else:
    print("Key does not exist")
```

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

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

print("Keys:")
for key in my_dict:
    print(key)

print("\nValues:")
for value in my_dict.values():
    print(value)

print("\nKey-Value Pairs:")
for key, value in my_dict.items():
    print("Key: {}, Value: {}".format(key, value))
```

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}

```
n = 5
square_dict = {x: x*x for x in range(1, n+1)}
print(square_dict)
```

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

```
d1 = {'a': 1, 'b': 2}
d2 = {'c': 3, 'd': 4}
d1.update(d2)
print(d1)
```

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

```
sample_dict = {0: 10, 1: 20}
total = sum(sample_dict.values())
print(total)
```

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

```
sample_dict = {1: 2, 2: 3, 3: 4}
```

```
result = 1
for value in sample_dict.values():
    result *= value
print(result)
```

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

```
d = {'a': 1, 'b': 2, 'c': 3}
key = 'b'
if key in d:
    del d[key]
print(d)
```

OR

```
sample_dict = {0: 10, 1: 20, 2: 30}
key_to_remove = 1
sample_dict.pop(key_to_remove, None)
print(sample_dict)
```

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

```
d = {'b': 1, 'a': 2, 'c': 3}
print(dict(sorted(d.items())))
```

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

```
d = {'a': 1, 'b': 2, 'c': 3}
print(max(d.values()), min(d.values()))
```

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

```
d = {'a': 1, 'b': 1, 'c': 2}
print(dict(d))
```

OR

```
sample_dict = {'a': 10, 'b': 20, 'c': 10, 'd': 30}
unique_dict = {key: value for key, value in sample_dict.items()}
print(unique_dict)
```

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

```
empty_dict = {}
if not bool(empty_dict):
    print("Dictionary is empty")
else:
    print("Dictionary is not empty")
```

15. 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})

```
from collections import Counter
d1 = {'a': 100, 'b': 200, 'c': 300}
d2 = {'a': 300, 'b': 200, 'd': 400}
combined_dict = Counter(d1) + Counter(d2)
print(combined_dict)
```

16. Write a Python program to find the highest 3 values in a dictionary.

```
sample_dict = {'a': 10, 'b': 50, 'c': 20, 'd': 40, 'e': 30}
highest_values = sorted(sample_dict.values(), reverse=True)[:3]
print(highest_values)
```

17. 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

```
dict_x = {'key1': 1, 'key2': 3, 'key3': 2}
dict_y = {'key1': 1, 'key2': 2}
for key in dict_x.keys():
    if key in dict_y and dict_x[key] == dict_y[key]:
        print(f"{key}: {dict_x[key]} is present in both x and y")
```

18. Write a Python program to check if all dictionaries in a list are empty or not.
Sample list : [{},{},{}]
Return value : True
Sample list : [{1,2},{},{}]
Return value : False

```
dicts1 = [{}, {}, {}]
dicts2 = [{1, 2}, {}, {}]
all_empty1 = all(not bool(d) for d in dicts1)
all_empty2 = all(not bool(d) for d in dicts2)
print(all_empty1)  # True
print(all_empty2)  # False
```

19. Write a Python program to remove duplicates from a list of lists.
Sample list : [[10, 20], [40], [30, 56, 25], [10, 20], [33], [40]]
New List : [[10, 20], [30, 56, 25], [33], [40]]

```
sample_list = [[10, 20], [40], [30, 56, 25], [10, 20], [33], [40]]
new_list = [list(item) for item in set(tuple(sublist) for sublist in sample_list)]
print(new_list)
```

20. Write a Python program to extend a list without append.
Sample data: [10, 20, 30]
[40, 50, 60]
Expected output : [40, 50, 60, 10, 20, 30]

```
sample_data = [10, 20, 30]
extension_data = [40, 50, 60]
extended_list = extension_data + sample_data
print(extended_list)
```

OR

```
list1 = [10, 20, 30]
list2 = [40, 50, 60]
list1[:0] = list2
print(list1)
```