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

ANS:

```
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)
```

index1.html index2.html p1.py

C: > Users > shabnam > Documents > Pace Wisdom > p1.py > descending_sorted_dict

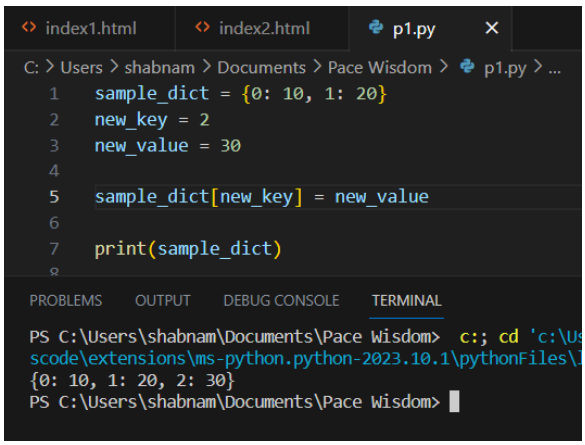
```
def sort_dictionary_by_value(dictionary, ascending=True):
    sorted_dict = dict(sorted(dictionary.items(), key=lambda item: item[1], reverse=not ascending))
    return sorted_dict

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)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Enter key: Fruit
Enter value: Apple
Enter key: Animal
Enter value: Goat
Traceback (most recent call last):
  File "c:\Users\shabnam\Documents\Pace Wisdom\p1.py", line 12, in <module>
    ascending_sorted_dict = sort_dictionary_by_value(dict)
  File "c:\Users\shabnam\Documents\Pace Wisdom\p1.py", line 2, in sort_dictionary_by_value
    sorted_dict = dict(sorted(dictionary.items(), key=lambda item: item[1], reverse=not ascending))
TypeError: 'dict' object is not callable
PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Users\shabnam\Documents\Pace Wisdom'; & 'C:\Users\shabnam\Ap
scode\extensions\ms-python.python-2023.10.1\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '50709'
Enter the number of items in the dictionary: 2
Enter key: Fruit
Enter value: Apple
Enter key: Animal
Enter value: Goat
Ascending order: {'Fruit': 'Apple', 'Animal': 'Goat'}
Descending order: {'Animal': 'Goat', 'Fruit': 'Apple'}
PS C:\Users\shabnam\Documents\Pace Wisdom>
```

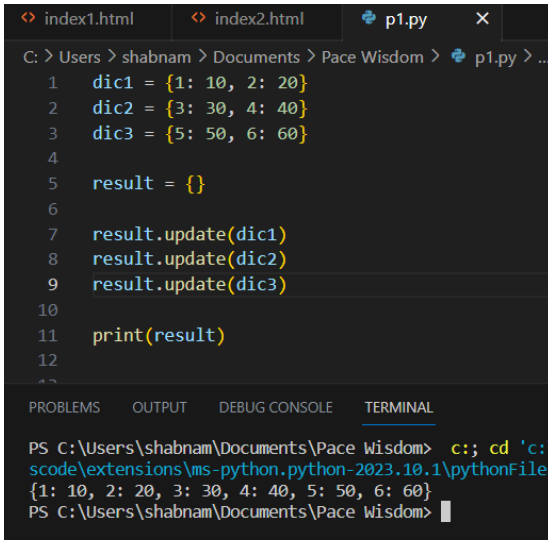
2. Write a Python script to add a key to a dictionary.
Sample Dictionary : {0: 10, 1: 20}
Expected Result : {0: 10, 1: 20, 2: 30}

ANS
sample_dict = {0: 10, 1: 20}
new_key = 2
new_value = 30
sample_dict[new_key] = new_value
print(sample_dict)



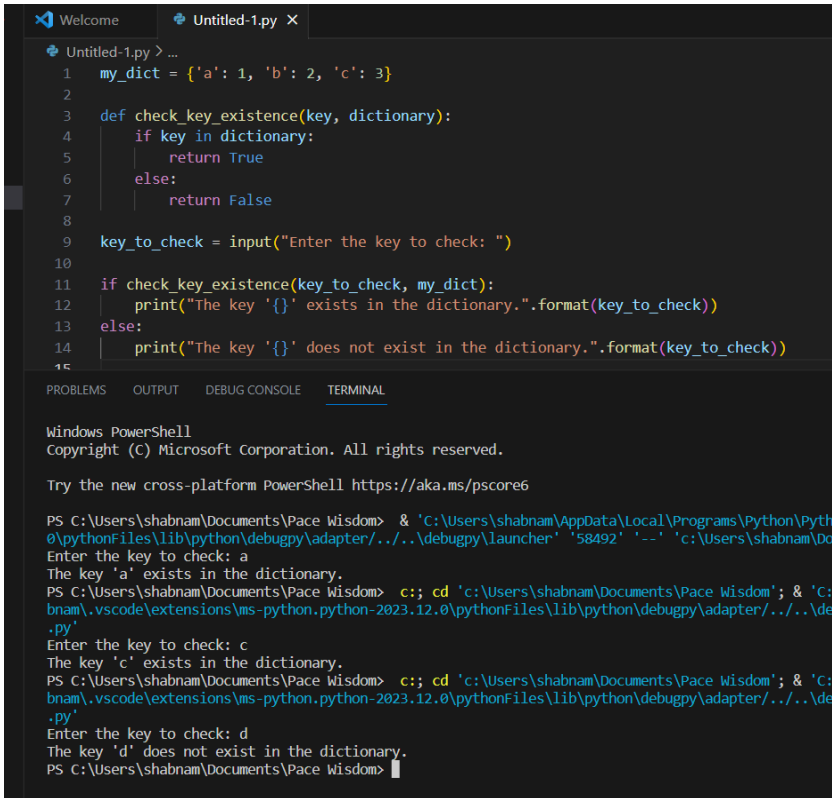
3. Write a Python script to concatenate following dictionaries to create a new one.
Sample Dictionary :
dic1={1:10, 2:20}
dic2={3:30, 4:40}
dic3={5:50,6:60}
Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}

ANS:
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.
ANS:

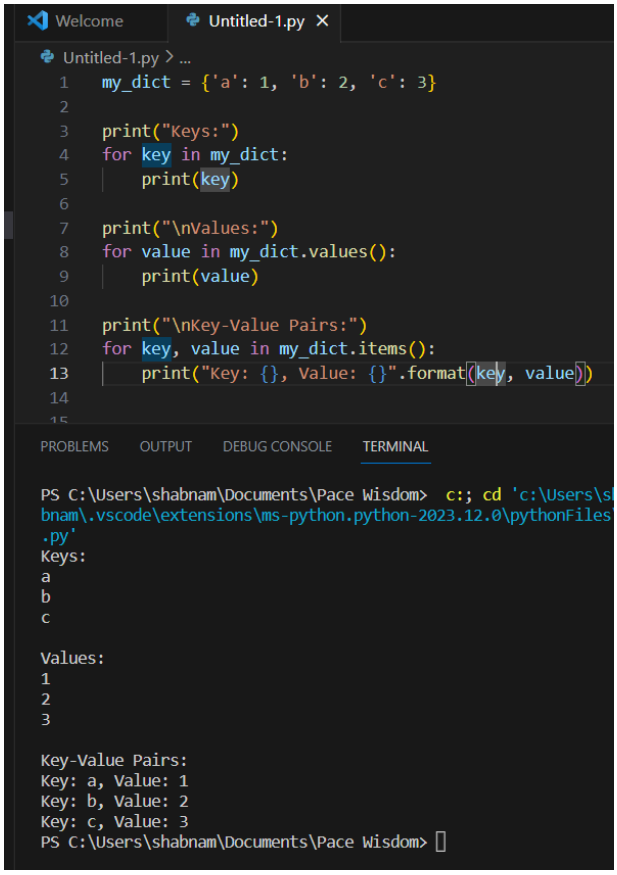
```
my_dict = {'a': 1, 'b': 2, 'c': 3}

def check_key_existence(key, dictionary):
    if key in dictionary:
        return True
    else:
        return False

key_to_check = input("Enter the key to check: ")

if check_key_existence(key_to_check, my_dict):
    print("The key '{}' exists in the dictionary.".format(key_to_check))
else:
    print("The key '{}' does not exist in the dictionary.".format(key_to_check))
```



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

ANS:

```
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))
```



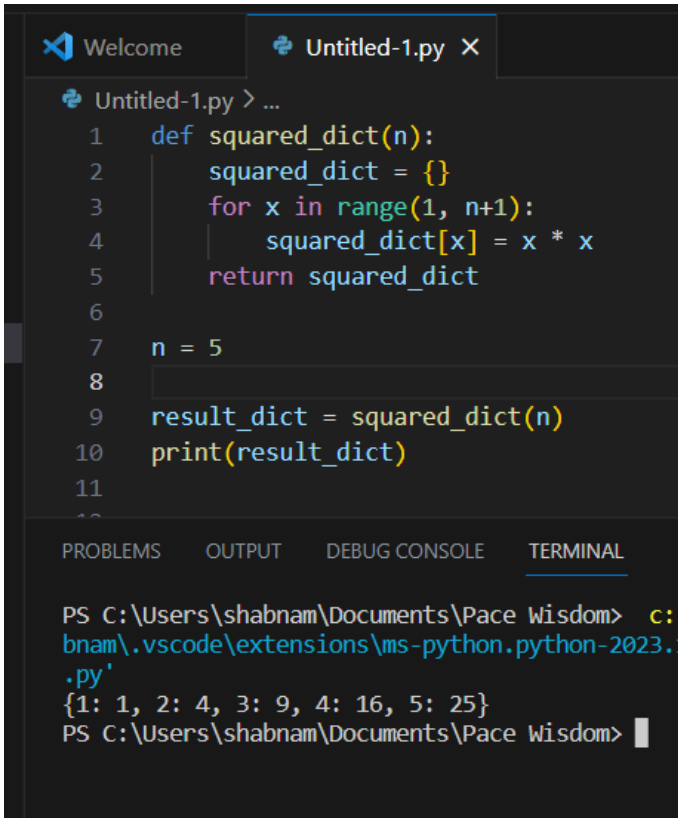
6. Write a Python script to generate and print a dictionary that contains a number (between 1 and n) in the form (x, x*x).

Sample Dictionary (n = 5) :

Expected Output : {1: 1, 2: 4, 3: 9, 4: 16, 5: 25}

ANS:

```
def squared_dict(n):
    squared_dict = {}
    for x in range(1, n+1):
        squared_dict[x] = x * x
    return squared_dict

n = 5

result_dict = squared_dict(n)
print(result_dict)
```



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

ANS:

```
def merge_dicts(dict1, dict2):
    result_dict = dict1.copy()
    result_dict.update(dict2)
    return result_dict

dict1_input = input("Enter the first dictionary in the format {key1: value1, key2: value2, ...}: ")
dict1 = eval(dict1_input)

dict2_input = input("Enter the second dictionary in the format {key1: value1, key2: value2, ...}: ")
dict2 = eval(dict2_input)

merged_dict = merge_dicts(dict1, dict2)

print("Merged Dictionary:")
print(merged_dict)
```

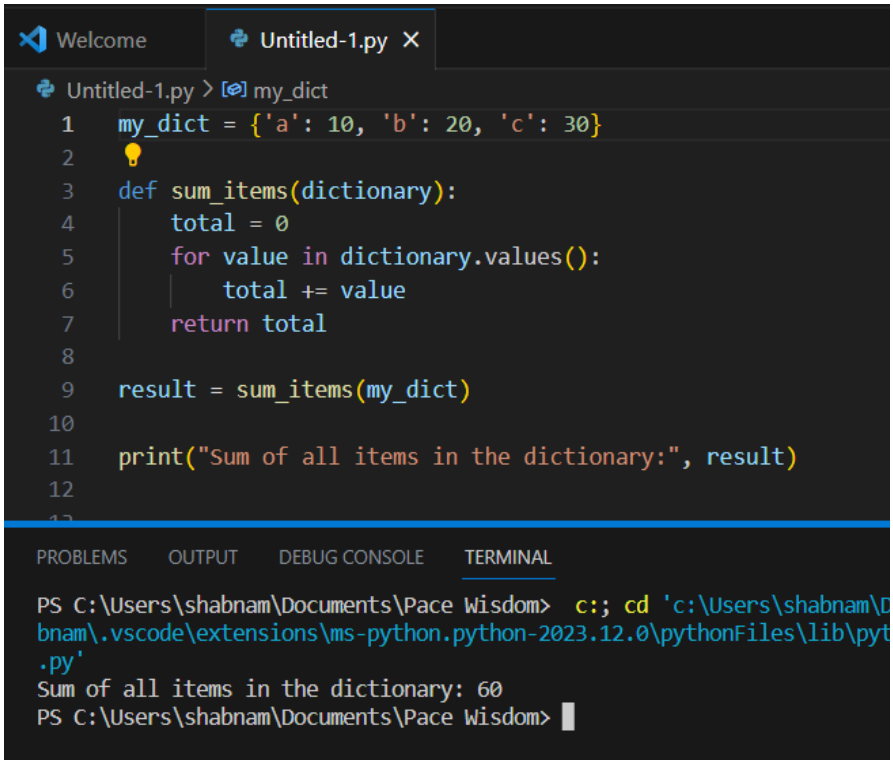
```python
def merge_dicts(dict1, dict2):
    result_dict = dict1.copy()
    result_dict.update(dict2)
    return result_dict

dict1_input = input("Enter the first dictionary in the format {key1: value1, key2: value2, ...}: ")
dict1 = eval(dict1_input)

dict2_input = input("Enter the second dictionary in the format {key1: value1, key2: value2, ...}: ")
dict2 = eval(dict2_input)

merged_dict = merge_dicts(dict1, dict2)

print("Merged Dictionary:")
print(merged_dict)
```

```
PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Users\shabnam\Documents\Pace Wisdom'; & 'C:\Users\shabnam\AppData\Lo
bnam\.vscode\extensions\ms-python.python-2023.12.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '58952' '
.py'
Enter the first dictionary in the format {key1: value1, key2: value2, ...}: {1: 10, 2: 20, 3: 30}
Enter the second dictionary in the format {key1: value1, key2: value2, ...}: {'a': 'apple', 'b': 'banana', 'c': 'cherry'}
Merged Dictionary:
{1: 10, 2: 20, 3: 30, 'a': 'apple', 'b': 'banana', 'c': 'cherry'}
PS C:\Users\shabnam\Documents\Pace Wisdom>
```

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

ANS:

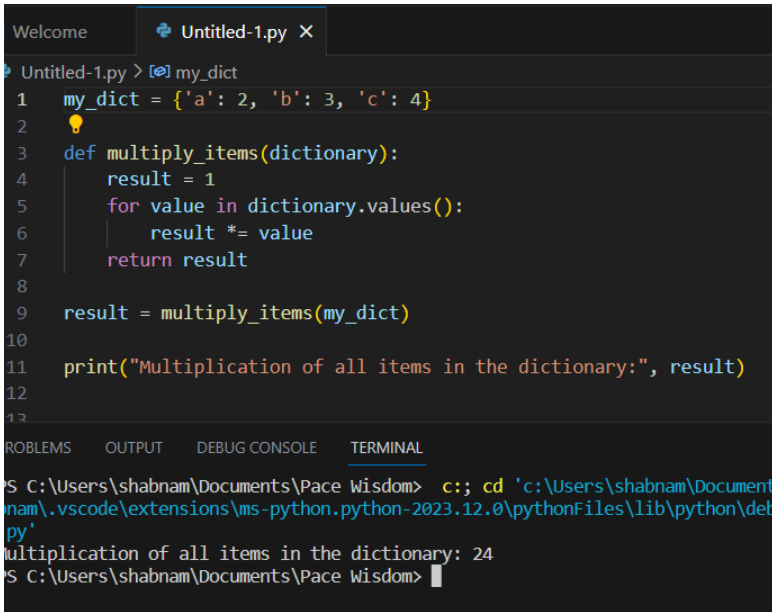
```python
my_dict = {'a': 10, 'b': 20, 'c': 30}

def sum_items(dictionary):
    total = 0
    for value in dictionary.values():
        total += value
    return total

result = sum_items(my_dict)

print("Sum of all items in the dictionary:", result)
```



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

ANS:

```
my_dict = {'a': 2, 'b': 3, 'c': 4}

def multiply_items(dictionary):
    result = 1
    for value in dictionary.values():
        result *= value
    return result

result = multiply_items(my_dict)

print("Multiplication of all items in the dictionary:", result)
```



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

ANS:

```
dict1_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}: ")
dict1 = eval(dict1_input)

key_to_remove = input("Enter the key to remove from the dictionary: ")

if key_to_remove in dict1:
    del dict1[key_to_remove]
    print("Key removed successfully.")
    print("Updated dictionary:", dict1)
else:
   print("The key '{}' does not exist in the dictionary.".format(key_to_remove))
```

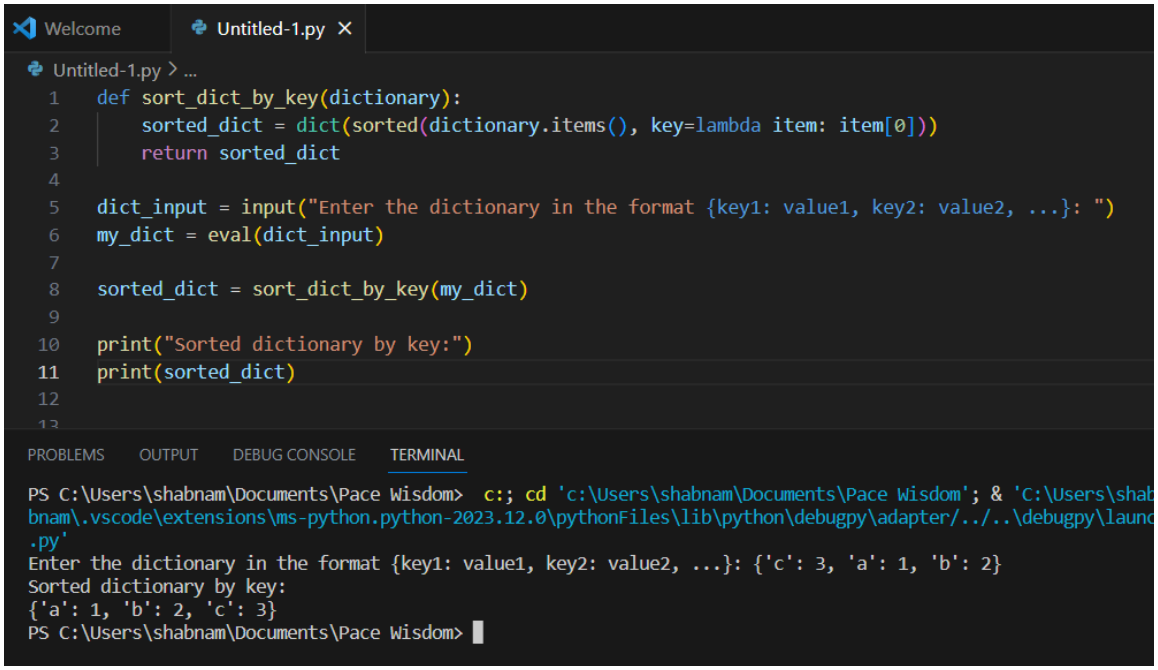
```python
dict1_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}: ")
dict1 = eval(dict1_input)

key_to_remove = input("Enter the key to remove from the dictionary: ")

if key_to_remove in dict1:
    del dict1[key_to_remove]
    print("Key removed successfully.")
    print("Updated dictionary:", dict1)
else:
    print("The key '{}' does not exist in the dictionary.".format(key_to_remove))
```

```
PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Users\shabnam\Documents\Pace Wisdom'; & 'C:\Users\shab
bnam\.vscode\extensions\ms-python.python-2023.12.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launc
.py'
Enter the dictionary in the format {key1: value1, key2: value2, ...}: {'Fruit': 'apple', 'animal': 'goat'}
Enter the key to remove from the dictionary: Fruit
Key removed successfully.
Updated dictionary: {'animal': 'goat'}
PS C:\Users\shabnam\Documents\Pace Wisdom>
```

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

ANS:

```python
def sort_dict_by_key(dictionary):
    sorted_dict = dict(sorted(dictionary.items(), key=lambda item: item[0]))
    return sorted_dict

dict_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}: ")
my_dict = eval(dict_input)

sorted_dict = sort_dict_by_key(my_dict)

print("Sorted dictionary by key:")
print(sorted_dict)
```



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

```
def get_max_min_values(dictionary):
    max_value = max(dictionary.values(), default=None)
    min_value = min(dictionary.values(), default=None)
    return max_value, min_value

dict_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}:")
my_dict = eval(dict_input)

max_value, min_value = get_max_min_values(my_dict)

print("Maximum value in the dictionary:", max_value)
print("Minimum value in the dictionary:", min_value)
```

```python
def get_max_min_values(dictionary):
    max_value = max(dictionary.values(), default=None)
    min_value = min(dictionary.values(), default=None)
    return max_value, min_value

dict_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}:")
my_dict = eval(dict_input)

max_value, min_value = get_max_min_values(my_dict)

print("Maximum value in the dictionary:", max_value)
print("Minimum value in the dictionary:", min_value)
```

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Users\shabnam\Documents\Pace Wisdom'; & 'C:\Users\shab
bnam\.vscode\extensions\ms-python.python-2023.12.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launc
.py'
Enter the dictionary in the format {key1: value1, key2: value2, ...}: {'a': 10, 'b': 20, 'c': 5, 'd': 30}
Maximum value in the dictionary: 30
Minimum value in the dictionary: 5
PS C:\Users\shabnam\Documents\Pace Wisdom>
```

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

ANS:

```python
def remove_duplicates(dictionary):
    unique_dict = {}
    for key, value in dictionary.items():
        if value not in unique_dict.values():
            unique_dict[key] = value
    return unique_dict

dict_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}: ")
my_dict = eval(dict_input)

unique_dict = remove_duplicates(my_dict)

print("Dictionary without duplicates:")
print(unique_dict)
```

Welcome Untitled-1.py

Untitled-1.py > ...

```python
def remove_duplicates(dictionary):
    unique_dict = {}
    for key, value in dictionary.items():
        if value not in unique_dict.values():
            unique_dict[key] = value
    return unique_dict

dict_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}: ")
my_dict = eval(dict_input)

unique_dict = remove_duplicates(my_dict)

print("Dictionary without duplicates:")
print(unique_dict)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Users\shabnam\Documents\Pace Wisdom'; & 'C:\Users\shabnam\AppData\Local\Prog
bnam\.vscode\extensions\ms-python.python-2023.12.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launcher' '59413' '--' 'c:\
.py'
Enter the dictionary in the format {key1: value1, key2: value2, ...}: {'c': 5, 'a': 10, 'b': 20, 'c': 5, 'd': 30, 'a': 10}
Dictionary without duplicates:
{'c': 5, 'a': 10, 'b': 20, 'd': 30}
PS C:\Users\shabnam\Documents\Pace Wisdom>
```

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

ANS:

```python
def is_dict_empty(dictionary):
    return not bool(dictionary)

dict_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}: ")
my_dict = eval(dict_input)

if is_dict_empty(my_dict):
    print("The dictionary is empty.")
else:
    print("The dictionary is not empty.")
```

```python
def is_dict_empty(dictionary):
    return not bool(dictionary)

dict_input = input("Enter the dictionary in the format {key1: value1, key2: value2, ...}: ")
my_dict = eval(dict_input)

if is_dict_empty(my_dict):
    print("The dictionary is empty.")
else:
    print("The dictionary is not empty.")
```

```
PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Users\shabnam\Documents\Pace Wisdom'; & 'C:\Users\shabn
bnam\.vscode\extensions\ms-python.python-2023.12.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launch
.py'
Enter the dictionary in the format {key1: value1, key2: value2, ...}: {'a': 10, 'b': 20, 'c': 5, 'd': 30}
The dictionary is not empty.
PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Users\shabnam\Documents\Pace Wisdom'; & 'C:\Users\shabn
bnam\.vscode\extensions\ms-python.python-2023.12.0\pythonFiles\lib\python\debugpy\adapter/../..\debugpy\launch
.py'
Enter the dictionary in the format {key1: value1, key2: value2, ...}: {}
The dictionary is empty.
PS C:\Users\shabnam\Documents\Pace Wisdom>
```

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})

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

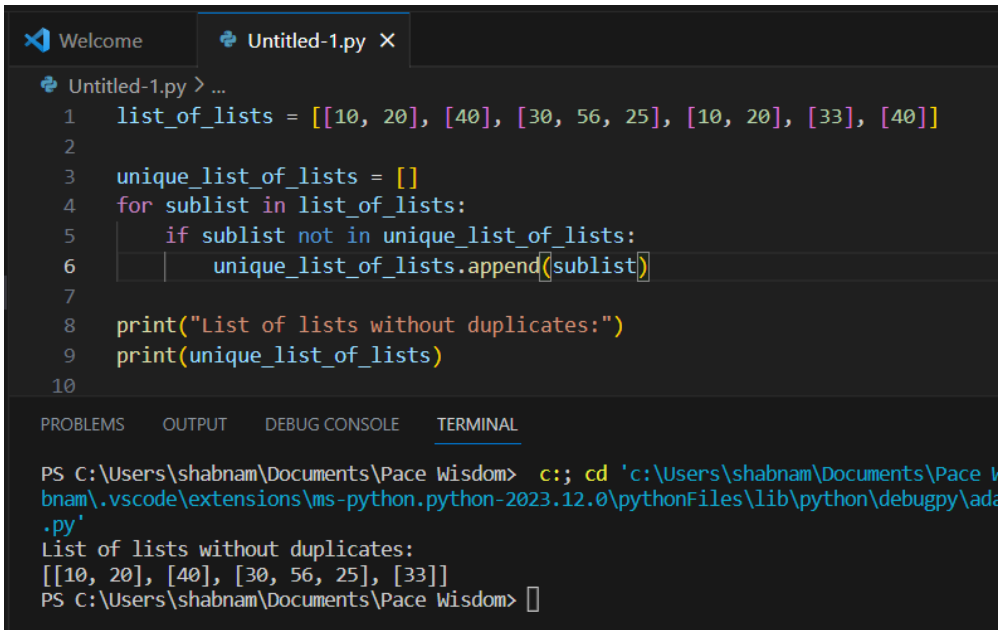
17. Write a Python program to match key values in two dictionaries.
Sample dictionary: {'key1': 1, 'key2': 3, 'key3': 2}, {'key1': 1, 'key2': 2}
Expected output: key1: 1 is present in both x and y

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

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]]
ANS:

```
list_of_lists = [[10, 20], [40], [30, 56, 25], [10, 20], [33], [40]]

unique_list_of_lists = []
for sublist in list_of_lists:
    if sublist not in unique_list_of_lists:
        unique_list_of_lists.append(sublist)

print("List of lists without duplicates:")
print(unique_list_of_lists)
```



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]

ANS:

```
list1 = [10, 20, 30]
list2 = [40, 50, 60]

list1 += list2

print("Extended list:", list1)
```

Welcome | Untitled-1.py

Untitled-1.py > ...

```python
list1 = [10, 20, 30]
list2 = [40, 50, 60]

list1 += list2

print("Extended list:", list1)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
PS C:\Users\shabnam\Documents\Pace Wisdom>  c:; cd 'c:\Use
bnam\.vscode\extensions\ms-python.python-2023.12.0\pythonF
.py'
Extended list: [10, 20, 30, 40, 50, 60]
PS C:\Users\shabnam\Documents\Pace Wisdom>
```