Python Programming Lab Assignment-3

Name:- Dev Agnihotri

Part A: Dictionary Basics (Q1–Q10)

1. Write a program to create a dictionary of 5 students with their names as keys and marks as values. Print the dictionary.

Answer 1:

```
students = {'Dev': 85, 'Raj': 78, 'Indra': 92, 'ki': 67, 'Jai': 90}
print(students)
```

Output:

```
{'Dev': 85, 'Raj': 78, 'Indra': 92, 'ki': 67, 'Jai': 90}
```

2. Write a function that takes a dictionary and prints all its keys and values separately.

Answer 2:

```
def print_dict(d):
    for k, v in d.items():
        print(f"Key: {k}, Value: {v}")
```

3. Write a function to find the length (number of items) of a dictionary without using len().

Answer 3:

```
def dict_length(d):
    count = 0
    for _ in d:
        count += 1
    return count
```

4. Write a program to check whether a given key exists in a dictionary or not.

Answer 4:

```
students = {"A": 5, "B": 7, "C": 2, "D": 6}

key = input("Enter the student name to search: ")

if key in students:

    print(f"Yes, '{key}' exists in the dictionary with marks {students[key]}")

else:

    print(f"No, '{key}' does not exist in the dictionary")
```

5. Write a program to update the marks of a specific student in a dictionary using a function.

Answer 5:

```
students = {"A": 85, "B": 78, "C": 92, "D": 67}

print("Original dictionary:", students)
```

```
name = input("Enter the student name whose marks you want to update: ")
new_marks = int(input("Enter the new marks: "))
if name in students:
   students[name] = new_marks
   print("Marks updated successfully!")
else:
   print("Student not found in the dictionary.")
print("Updated dictionary:", students)
```

6. Write a function to merge two dictionaries into one.

Answer 6:

```
def merge_dicts(d1, d2):
   d = d1.copy()
   d.update(d2)
   return d
```

7. Write a program to sort a dictionary by its values (ascending and descending).

Answer 7:

```
def sort_dict_by_values(d):
   asc = dict(sorted(d.items(), key=lambda x: x[1]))
   desc = dict(sorted(d.items(), key=lambda x: x[1], reverse=True))
   return asc, desc
```

Output:

```
Ascending: {'David': 67, 'Bob': 78, 'Alice': 85, 'Eva': 90, 'Charlie': 92}
Descending: {'Charlie': 92, 'Eva': 90, 'Alice': 85, 'Bob': 78, 'David': 67}
```

8. Write a function that takes a dictionary and returns the key with the maximum value.

Answer 8:

```
def max_value_key(d):
   return max(d, key=d.get)
```

9. Write a program to remove a given key from a dictionary using a function.

Answer 9:

```
def remove_key(d, key):
   if key in d:
      del d[key]
   return d
```

10. Write a function that counts how many times each character appears in a given string using a dictionary.

Answer 10:

```
def char_count(s):
    d = {}
    for ch in s:
        d[ch] = d.get(ch, 0) + 1
    return d
```

Example Input: 'hello'

```
Output: {'h':1, 'e':1, 'l':2, 'o':1}
```

Part B: Functions with Dictionary Applications (Q11–Q20)

11. Write a function that accepts a dictionary of students with marks and returns the average marks.

Answer 11:

```
def avg_marks(d):
    return sum(d.values()) / len(d)
```

12. Write a function to invert a dictionary (swap keys and values).

Answer 12:

```
def invert_dict(d):
    return {v: k for k, v in d.items()}
```

13. Write a function that takes a dictionary and returns a new dictionary with only those items where the value is greater than 50.

Answer 13:

```
def filter_values(d):
    return {k: v for k, v in d.items() if v > 50}
```

14. Write a program to create a dictionary of squares of numbers from 1 to n, where n is input from the user.

Answer 14:

```
n = int(input("Enter n: "))
squares = {i: i**2 for i in range(1, n+1)}
print(squares)
```

Example Input: n=5

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

15. Write a function to count the frequency of words in a given sentence using a dictionary.

Answer 15:

```
def word_freq(sentence):
    words = sentence.split()
    d = {}
    for word in words:
        d[word] = d.get(word, 0) + 1
    return d
```

Example Input: 'this is a test this'
Output: {'this':2, 'is':1, 'a':1, 'test':1}

16. Write a function to combine two dictionaries by adding values of common keys. Example: d1 = {'a':10, 'b':20}, d2 = {'a':30, 'c':40} → Output: {'a':40, 'b':20, 'c':40}

Answer 16:

```
def combine_dicts(d1, d2):
    d = d1.copy()
    for k, v in d2.items():
        d[k] = d.get(k, 0) + v
    return d
```

Example Input: d1={'a':10,'b':20}, d2={'a':30,'c':40}
Output: {'a':40,'b':20,'c':40}

17. Write a program to create a nested dictionary to store employee details (id, name, salary) and print them neatly using a function.

Answer 17:

```
employees = {
    1: {'name': 'Alice', 'salary': 50000},
    2: {'name': 'Bob', 'salary': 60000},
    3: {'name': 'Charlie', 'salary': 55000}
}

def print_employees(emp):
    for id, details in emp.items():
        print(f"ID: {id}, Name: {details['name']}, Salary: {details['salary']}")
```

Output:

```
ID:1, Name: Alice, Salary: 50000
ID:2, Name: Bob, Salary: 60000
ID:3, Name: Charlie, Salary: 55000
```

18. Write a function to check if two dictionaries are equal (same keys and values).

Answer 18:

```
def dicts_equal(d1, d2):
    return d1 == d2
```

19. Write a function that takes a dictionary of items with prices and returns the total bill amount.

Answer 19:

```
def total_bill(d):
    return sum(d.values())
```

Example Input: {'apple':50,'banana':30,'milk':20}
Output: 100

20. Write a program to create a dictionary from two lists: one containing keys and the other containing values.

Answer 20:

```
keys = ['a', 'b', 'c']
values = [1, 2, 3]
d = dict(zip(keys, values))
print(d)
```

Output:

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