ASSIGNMENT

NAME- NIKHIL KUSHAWAHA

ROLL NO.- 34

UNIV. ROLL NO.- 2215001159

SECTION- U

SUBJECT- PYTHON PROGRAMMING

Q-1. What Will Be The Output Of The Following Code Snippet?

a = {(1,2):1,(2,3):2} print(a[1,2])

## ANS- A. KEY ERROR

-2. What Will Be The Output Of The Following Code Snippet? a = {'a':1,'b':2,'c':3} print (a['a','b']) print(a.get(‘a’,’b’))

## ANS- A. KEY ERROR

Q-3. What Will Be The Output Of The Following Code Snippet? fruit = {} def addone(index): if index in fruit: fruit[index] += 1 else: fruit[index] = 1 addone('Apple') addone('Banana') addone('apple') print (len(fruit))

ANS- C. 3

Q-4. What Will Be The Output Of The Following Code Snippet? arr = {} arr[1] = 1 arr['1'] = 2 arr[1] += 1 sum = 0

for k in arr: sum += arr[k] print (sum)

ANS- D. 4

Q-5. What Will Be The Output Of The Following Code Snippet?

my\_dict = {} my\_dict[1] = 1 my\_dict['1'] = 2 my\_dict[1.0] = 4 sum = 0 for k in my\_dict: sum += my\_dict[k] print (sum)

ANS- A. 7

Q-6. What Will Be The Output Of The Following Code Snippet?

my\_dict = {} my\_dict[(1,2,4)] = 8 my\_dict[(4,2,1)] = 10 my\_dict[(1,2)] = 12 sum = 0 for k in my\_dict: sum += my\_dict[k] print (sum) print(my\_dict)

## ANS- B. 30 {(1, 2): 12, (4, 2, 1): 10, (1, 2, 4): 8}

Q-7. What Will Be The Output Of The Following Code Snippet?

box = {} jars = {} crates = {} box['biscuit'] = 1 box['cake'] = 3 jars['jam'] = 4 crates['box'] = box crates['jars'] = jars

print (len(crates[box]))

ANS- D. TYPE ERROR

Q-8. What Will Be The Output Of The Following Code Snippet? dict = {'c': 97, 'a': 96, 'b': 98} for \_ in sorted(dict):

print (dict[\_])

## ANS- A. 96 98 97

Q-9. What Will Be The Output Of The Following Code Snippet? rec = {"Name" : "Python", "Age":"20"}

r = rec.copy()

print(id(r) == id(rec))

## ANS- B. FALSE

Q-10. What Will Be The Output Of The Following Code Snippet?

rec = {"Name" : "Python", "Age":"20", "Addr" : "NJ", "Country" : "USA"} id1 = id(rec) del rec

rec = {"Name" : "Python", "Age":"20", "Addr" : "NJ", "Country" : "USA"} id2 = id(rec)

print(id1 == id2)

ANS- B. FALSE

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

# ANS-

my\_dict = {"apple": 5, "banana": 10, "orange": 3, "pear": 8} asc\_dict = dict(sorted(my\_dict.items(), key=lambda x: x[1]))

desc\_dict = dict(sorted(my\_dict.items(), key=lambda x: x[1], reverse=True)) print("Ascending order:", asc\_dict)

print("Descending order:", desc\_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}

## ANS-

sample\_dict = {0: 10, 1: 20} key = 2 value = 30

sample\_dict[key] = 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 = {} for d in (dic1, dic2, dic3): result.update(d)

print(result)

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

## ANS-

my\_dict = {'apple': 2, 'banana': 4, 'orange': 6} key = 'banana' if key in my\_dict:

print(f"The key '{key}' exists in the dictionary.") else: print(f"The key '{key}' does not exist in the dictionary.")

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

## ANS-

my\_dict = {"name": "John", "age": 30, "city": "New York"} for key in my\_dict: print(key, ":", my\_dict[key])

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-

n = 5 d = {} for x in range(1, n+1): d[x] = x\*x

print(d)

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}

## ANS-

d = {} for i in range(1, 16):

d[i] = i \*\* 2

print(d)

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

## ANS-

dict1 = {'a': 1, 'b': 2} dict2 = {'c': 3, 'd': 4}

dict1.update(dict2)

print(dict1) # {'a': 1, 'b': 2, 'c': 3, 'd': 4}

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

## ANS-

dict = {'name': 'John', 'age': 25, 'country': 'USA'} for key in dict: print(key, ':', dict[key])

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

## ANS-

my\_dict = {'a': 10, 'b': 20, 'c': 30} total = sum(my\_dict.values())

print(total)

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

## ANS-

def multiply\_dict\_items(dict):

result = 1 for key in dict: result \*= dict[key] return result

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

# ANS-

my\_dict = {'apple': 2, 'banana': 3, 'orange': 4}

print("Original dictionary:", my\_dict)

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

## ANS-

keys = ['name', 'age', 'gender'] values = ['Alice', 25, 'Female'] my\_dict = dict(zip(keys, values)) print(my\_dict)

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

## ANS-

my\_dict = {'b': 2, 'c': 3, 'a': 1}

sorted\_dict = {k: v for k, v in sorted(my\_dict.items(), key=lambda x: x[0])} print(sorted\_dict)

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

## ANS-

my\_dict = {'a': 10, 'b': 5, 'c': 20, 'd': 30} max\_val = max(my\_dict.values()) min\_val = min(my\_dict.values()) print('Maximum value:', max\_val)

print('Minimum value:', min\_val)

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

## ANS-

class Person: def \_\_init\_\_(self, name, age): self.name = name

self.age = age

person = Person("John", 30)

person\_dict = vars(person)

print(person\_dict)

17. Write a Python program to remove duplicates from Dictionary

## ANS-

original\_dict = {"a": 1, "b": 2, "c": 1, "d": 3, "e": 2}

new\_dict = {value:key for key, value in original\_dict.items()}

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

## ANS-

my\_dict = {}

if len(my\_dict) == 0:

print("The dictionary is empty") else:

print("The dictionary is not empty")

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})

## ANS-

from collections import Counter

d1 = {'a': 100, 'b': 200, 'c': 300}

d2 = {'a': 300, 'b': 200, 'd': 400}

result = Counter(d1) + Counter(d2)

print(result)

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'}

## ANS-

data = [{"V":"S001"}, {"V": "S002"}, {"VI": "S001"}, {"VI": "S005"}, {"VII":"S005"}, {"V":"S009"},{"VIII":"S007"}]

unique\_values = set() for d in data: for value in d.values():

unique\_values.add(value)

print("Unique Values:", unique\_values)

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

## ANS-

def generate\_combinations(data):

combinations = [] keys = data.keys() values = [data[key] for key in keys] for i in range(len(values[0])): for j in range(len(values[1])): combination = values[0][i] + values[1][j] combinations.append(combination) return combinations

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

## ANS-

def find\_highest\_values(data):

sorted\_values = sorted(data.values(), reverse=True) return sorted\_values[:3]

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})

## ANS-

from collections import Counter def combine\_values(data):

result = Counter() for d in data: result[d['item']] += d['amount'] return result

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: {'3': 1, 's': 1, 'r': 2, 'u': 1,

'w': 1, 'c': 1, 'e': 2, 'o': 1}

## ANS-

def create\_dict\_from\_string(s): counts = {} for letter in s: if letter in counts:

counts[letter] += 1 else:

counts[letter] = 1

return counts

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

## ANS-

def print\_dict\_table(data): keys = list(data.keys()) values = list(data.values()) max\_key\_length = max(len(str(key)) for key in keys) max\_value\_length = max(len(str(value)) for value in values) print('+' + '-' \* (max\_key\_length + 2) + '+' + '-' \* (max\_value\_length + 2) + '+') print('| {:<{}} | {:>{}} |'.format('Key', max\_key\_length, 'Value', max\_value\_length)) print('+' + '-' \* (max\_key\_length + 2) + '+' + '-' \* (max\_value\_length + 2) + '+') for key, value in data.items(): print('| {:<{}} | {:>{}} |'.format(key, max\_key\_length, value, max\_value\_length)) print('+' + '-' \* (max\_key\_length + 2) + '+' + '-' \* (max\_value\_length + 2) + '+')

26. Write a Python program to count the values associated with key in a dictionary. Sample data: = [{'id': 1, 'success': True, 'name': 'Lary'}, {'id': 2, 'success': False, 'name':

'Rabi'}, {'id': 3, 'success': True, 'name':

'Alex'}]

Expected result: Count of how many

dictionaries have success as True

## ANS-

def count\_dicts\_with\_value(data, key, value):

count = 0 for dictionary in data: if dictionary.get(key) == value:

count += 1

return count

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

## ANS-

def list\_to\_nested\_dict(lst):

nested\_dict = {} for item in reversed(lst): nested\_dict = {item: nested\_dict} return nested\_dict

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

## ANS-

my\_dict = {'fruit': ['apple', 'orange', 'banana', 'kiwi']} my\_dict['fruit'].sort()

print(my\_dict)

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

## ANS-

my\_dict = {'my key': 1, 'another key': 2, 'third key': 3}

new\_dict = {key.replace(' ', ''): value for key, value in my\_dict.items()} print('Original dictionary:', my\_dict)

print('New dictionary:', new\_dict)

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 ANS-

shop\_inventory = {'item1': 45.50, 'item2': 35, 'item3': 41.30, 'item4': 55, 'item5': 24} top\_three = sorted(shop\_inventory.items(), key=lambda x: x[1], reverse=True)[:3] for item, price in top\_three:

print(item, price)

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

ANS-

my\_dict = {'apple': 2, 'banana': 3, 'orange': 4} for key, value in my\_dict.items():

item = (key, value) print(f"Key: {key}, Value: {value}, Item: {item}")

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

ANS-

my\_dict = {'apple': 2, 'banana': 3, 'orange': 4} for key, value in my\_dict.items():

print(f"{key}: {value}")

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

ANS-

my\_dict = {'apple': 2, 'banana': 3, 'orange': 4, 'pear': 5} keys\_to\_check = ['apple', 'banana', 'peach'] if all(key in my\_dict for key in keys\_to\_check):

print("All keys exist in the dictionary") else:

print("At least one key does not exist in the dictionary")

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

ANS-

my\_dict = {'fruits': ['apple', 'banana', 'orange'], 'vegetables': ['carrot', 'celery']} for key, value in my\_dict.items():

num\_items = len(value) print(f"{key} has {num\_items} items")

1. 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)]

ANS-

from collections import Counter

my\_counter = Counter({'Math':81, 'Physics':83, 'Chemistry':87})

sorted\_counter = sorted(my\_counter.items(), key=lambda x: x[1], reverse=True) print(sorted\_counter)

1. 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-VII': {2}, 'Class-VI': {2}, 'Class-VIII':

{3}, 'Class-V': {1}})

ANS-

from collections import defaultdict keys = ['Class-V', 'Class-VI', 'Class-VII', 'Class-VIII'] values = [1, 2, 2, 3] result = defaultdict(set) for k, v in zip(keys, values):

result[k].add(v) print(result)

1. Write a Python program to replace dictionary values with their sum.

ANS-

data = {'a': 10, 'b': 20, 'c': 30, 'd': 40} total = sum(data.values()) for key in data: data[key] = total print(data)

1. 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

ANS-

x = {'key1': 1, 'key2': 3, 'key3': 2} y = {'key1': 1, 'key2': 2} for key in x: if key in y: if x[key] == y[key]:

print(key + ": " + str(x[key]) + " is present in both x and y")