

CLASS 11

DICTIONARIES (WORKSHEET)

1) MULTIPLE CHOICE QUESTIONS

- i) A dictionary is a collection of:
 - a) Sequence data
 - b) Ordered data
 - c) Unordered data
 - d) Sorted data

- ii) A set of key and its values is called:
 - a) key-data pair
 - b) key-value pair
 - c) data-value pair
 - d) string-numeric pair

- iii) Accessing of data in a dictionary is done through:
 - a) key
 - b) value
 - c) index
 - d) cell number

- iv) The function dict() is used to create a/an:
 - a) Mutable dictionary
 - b) Immutable dictionary
 - c) Ordered dictionary
 - d) Empty dictionary

- v) Which of the following is mutable?
 - a) "Python"
 - b) (4,5)
 - c) [4,5]
 - d) "45"

- vi) Given is a dictionary:
`d = {1:5, 2:{3:10, 4:15}, 5:20}`
Which of the following is a nested dictionary?
 - a) {1:5, 5:20}
 - b) {3:10, 4:15}
 - c) {1:5, 3:10}
 - d) {4:5, 5:20}

- vii) Given is a dictionary:
 $D = \{1:10, \text{"Name"}:\text{"Ashok"}, [3,5]:\text{"List"}, 4:20\}$
 Which of the following is an invalid key?
 a) 4
 b) 1
 c) "Name"
 d) [3,5]
- viii) Which of the following can be mutable in a dictionary?
 a) 4
 b) A value
 c) A key-value pair
 d) None of these

[V. K. PANDEY / D. K. DEY]

2) FILL IN THE BLANKS

- i) A dictionary is a _____ collection of key-value pairs.
 ii) In a dictionary, a key is associated with its _____.
 iii) All the keys of a dictionary can be accessed by using the _____ function.
 iv) Accessing the values of a dictionary one by one is said to be _____.
 v) The _____ is a function which returns the value when a key-value pair is deleted.
 vi) A dictionary used as the value of a key within another dictionary is known as a _____.
 vii) The operators **in** and **not** results in _____ type value.
 viii) A function that is used to delete all the elements of a dictionary and returns an empty dictionary is _____.

[V. K. PANDEY / D. K. DEY]

3) ASSERTION AND REASON BASED QUESTIONS:

- a) Both A and R are true and R is the correct explanation of A.
 b) Both A and R are true and R is not the correct explanation of A.
 c) A is true but R is false.
 d) A is false but R is true.
 e) Both A and R are false.

Based on the the assertion and reasoning below, choose an appropriate statement from the options given above.

- i) Assertion (A): You can also create a dictionary by using different nested lists as values.

Reason (R): It can be created using the dict() function. In this system, each inner list will contain a key and a value as its elements.

- ii) Assertion (A): You can't update the value of a key in the Python dictionary.
Reason (R): A dictionary in the Python contains keys and their values as immutable and mutable types respectively. Since the values being mutable, they can be updated in their places as per our requirement.
- iii) Assertion (A): You can delete a key-value pair from a Python dictionary.
Reason (R): Deleting a key-value pair from a dictionary can be performed by using `del` keyword and `pop()` function. The only limitation is that the `pop()` function will use the dictionary object rather than the dictionary name.
- iv) Assertion (A): The **in** and **not in** operators are used to check the existence of a key among all the keys available in the dictionary.
Reason (R): The **in** operator results in True, if the required key is available in the dictionary, otherwise False. The **not in** operator results in True, if the required key is not available in the dictionary, otherwise False.
- v) Assertion (A): The `items()` function returns all the key-value pairs of a dictionary as a sequence of tuples.
Reason (R): You can get the value with reference to the corresponding key. If the key is not available in the dictionary, then the dictionary will return None.

[V. K. PANDEY / D. K. DEY]

4) CASE STUDY BASED QUESTIONS:

- i) A code snippet using a dictionary is shown below:
- ```
mydict = {"Apple":50, "Orange":40, "Banana":30, "Mango":80}
print (len(mydict.clear())) # Statement 1
print (mydict.keys()) #Statement 2
print (mydict.items()) #Statement 3
print (mydict.popitem()) #Statement 4
```

With reference to the above code snippet, pick the appropriate answer for the questions given below:

- a) What will the statement 1 display?
1. Nil                      2. Null                      3. None                      4. 0
- b) In what form will the keys of the dictionary be displayed in statement 2?
1. In the form of a tuple

2. In the form of a list
  3. In the form of all the keys as a single string
  4. None of the above
- c) What will the statement 3 display?
1. Key-value pairs as the elements of a tuple
  2. Key-value pairs as the elements of a list
  3. Key-value pairs as different strings
  4. Key-value pairs in the forms of nested list
- d) What will the statement 4 display?
1. "Apple":50
  2. "Mango":80
  3. Any key-value pair randomly
  4. All the key-value pairs
- ii) A company has increased the salary by 20% of the employees whose age is 55 years and above. A code snippet to update the salary of the employees is shown below:
- ```
emp1 = {"Name": "Uday", "Age": 48, "Salary": 35000}
emp2 = {"Name": "Dinesh", "Age": 56, "Salary": 42000}
emp3 = {"Name": "Gaurav", "Age": 58, "Salary": 45000}
employee = {"E1": emp1, "E2": emp2, "E3": emp3}
```
- ```
for x in ?1?:
 if (employee[x][?2?] >= 55):
 incr = employee[x]["Salary"] * ?3?
 employee[x]["Salary"] = ?4? + incr
```

There are some places marked with ?1?, ?2?, ?3? and ?4? in the a above code snippet. With reference to the above code snippet, answer the following questions:

- a) What will be the statement / expression in place of ?1?
- b) What will be the statement / expression in place of ?2?
- c) What will be the statement / expression in place of ?3?
- d) What will be the statement / expression in place of ?4?

[V. K. PANDEY / D. K. DEY]

## 5) OUTPUT BASED QUESTIONS

|      |                                                                                                                                                                                               |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| i)   | <pre>d1 = {'a':10, 'b':20, 'c':30, 'd':40} sum = 0 for x in d1.values():     sum = sum + x print (sum)</pre>                                                                                  |
| ii)  | <pre>d1 = {1:5, 2:8, 3:12, 4:3, 5:6} x = y = d1[1] for k in d1:     if (d1[k] &gt; x):         x = d1[k]     else:         if (d1[k] &lt; y):             y = d1[k] print (x) print (y)</pre> |
| iii) | <pre>d1 = {1:5, 2:8, 3:5, 4:6, 5:5, 6:3, 7:5} n = 5 t = 0 for x in d1:     if (d1[x] == n):         t = t+1 print (t) <b>Name the function that performs the same task.</b></pre>             |
| iv)  | <pre>d1 = {1:[1, 2, 3], 2:[4, 5, 6], 3:[7, 8, 9]} print (d1[1].append(4)) print (d1[1] + d1[2]) print (d1[1] * 2) print (d1)</pre>                                                            |
| v)   | <pre>d1 = {"a":12, "e":18, "i":10, "o":15, "u":25} x = "i" if x in d1:     d1[x] = 20 print (d1)</pre>                                                                                        |
| vi)  | <pre>d1 = {"a":[1, 2, 3], "b":(1, 2, 3), "c":"Python"} print (d1.keys()) print (d1.values())</pre>                                                                                            |

|       |                                                                                                                                                             |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| vii)  | <pre> d1 = {"Amit":40, "Sumit":45} d2 = {"Amit":466, "Sumit":45} print (d1 == d2) </pre>                                                                    |
| viii) | <pre> d1 = {} d1[1] = 1 d1['1'] = 2 d1[1] = d1[1] + 1 count = 0 for i in d1:     count += d1[i] print (count) </pre>                                        |
| ix)   | <pre> d1 = {1:5, 2:8, 3:6, 4:3, 5:10} t = 0 for x in d1:     if (d1[x] % 2 == 0):         d1[x] += 2     else:         d1[x] *= 2 print (d1) </pre>         |
| x)    | <pre> d = {"Aman":20, "Sumit":30, "Dinesh":50, "Suresh":80} t = d["Aman"] d["Aman"] = d["Suresh"] d["Suresh"] = t d["Dinesh"] = d["Sumit"] print (d) </pre> |

**[V. K. PANDEY / D. K. DEY]**

|       |                                                                                                                                                                                              |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| xi)   | <pre> d1 = {5 : [6, 7, 8], "a" : (1, 2, 3)} print (d1.keys()) print (d1.values()) </pre>                                                                                                     |
| xii)  | <pre> d1 = {5 : "number", "a" : "string", (1,2) : "tuple"} print ("Dictionary contents: ") for x in d1.keys():     print (x, ":", d1[x], end = " ")     print (d1[x] * 3)     print() </pre> |
| xiii) | <pre> d = dict() d["left"] = "&lt;" </pre>                                                                                                                                                   |

|      |                                                                                                                                                                                                                                                                                                               |
|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|      | <pre>d["right"] = "&gt;" print ("{left} and {right} or {right} and {left}" )</pre>                                                                                                                                                                                                                            |
| xiv) | <pre>d = dict() d["left"] = "&lt;" d["right"] = "&gt;" d["end"] = " " print (d["left"] and d["right"] or d["right"] and d["left"] ) print (d["left"] and d["right"] or d["right"] and d["left"] and d["end"]) print ((d["left"] and d["right"] or d["right"] and d["left"]) and d["end"]) print ("end")</pre> |
| xv)  | <pre>my_dict = {} my_dict [(1,2,4)] = 8 my_dict [(4,2,1)] = 10 my_dict [(4,2,1)] = 10 print (my_dict)</pre>                                                                                                                                                                                                   |

[SUMITA ARORA]

## 6) SHORT ANSWER QUESTIONS

- Why is dictionary termed as an unordered collection of objects?
- What type of objects can be used as keys in dictionaries?
- How is `del D` and `del [<key>]` different from one another if `D` is a dictionary?
- How is `clear()` function different from `del <dict>` statement?

[SUMITA ARORA]

- How will you compare a dictionary from other collections?
- Can a key be defined as a mutable type? Justify.
- What will happen if two keys are available with the same name in a dictionary?
- Differentiate between the following functions:
  - `keys()` and `values()`
  - `del()` and `clear()`
  - `max()` and `min()`
  - `pop()` and `popitem()`
- Write the purpose of the following functions when used on a Python dictionary:
  - `dict()`
  - `insert()`
  - `get()`
  - `items()`
  - `update()`

**7) APPLICATION BASED QUESTIONS**

- i) Which of the following will result in an error for given valid dictionary D?  
 a) `D + 3`    b) `D * 3`    c) `D + {3 : "3"}`    d) `D.update ({3 : "3"})`  
 e) `D.update ({"3" : 3})`    f) `D.update {"3" : 3}`
- ii) The following code is giving some error. Find out the error and correct it.  
`d1 = {"a" : 1, 1 : "a", [1, "a"] : "two"}`
- iii) The following code has two dictionaries with tuples as keys. While one of these dictionaries being successfully created, the other is giving some error. Find out which dictionary will be created successfully and which one will give error and correct it.  
`d1 = {(1,2) : [1,2], (3, 4) : [3,4]}`  
`d2 = {[1], [2]} : [1,2], ([3], [4]) : [3,4]}`
- iv) Nesting of dictionary allows you to store a dictionary inside another dictionary. Then why is following code raising error? What can you do to correct it?  
`d1 = {1:10, 2:20, 3:30}`  
`d2 = {1:40, 2:50, 3:60}`  
`d3 = {1:70, 2:80, 3:90}`  
`d4 = {d1:"a", d2:"b", d3:"c"}`
- v) Why is following code not giving correct output even when 25 is a member of the dictionary?  
`d1 = {"age":25, "name":"xyz", "salary":23450.5}`  
`v = d1["age"]`  
`if val in d1:`  
     `print ("This is member of dictionary")`  
`else:`  
     `print ("This is not a member of dictionary")`
- vi) Consider the following code and then answer the questions that follow:
- ```

myDict = {"a":27, "b":43, "c":25, "d":30}
valA = ""
for i in myDict:
    if i > valA:
        valA = i
        valB = myDict[i]

print (valA)           #Line 1
print (valB)           #Line 2

```



```

print (30 in myDict)           #Line 3
myLst = (myDict.items())
myLst.sort()                   #Line 4
print (myLst[-1])              #Line 5

```

- a) What output does Line 1 produce?
- b) What output does Line 2 produce?
- c) What output does Line 3 produce?
- d) What output does Line 5 produce?
- e) What is the return value from the list sort() function (Line 4) and what is its type?

[SUMITA ARORA]

8) PROGRAMS

- i) Repeatedly ask the user to enter a team name and how many games the team has won and how many they lost. Store this information in a dictionary where the keys are the team names and the values are the lists of the form [wins, losses].
 - a) Using the dictionary created above, allow the user to enter a team name and print out the team's winning percentage.
 - b) Using the dictionary, create a list whose entries are the number of wins of each team
 - c) Using the dictionary, create a list of all those teams that have winning records.
- ii) Create a dictionary whose keys are month names and whose values are the number of days in the corresponding months.
 - a) Ask the user to enter a month name and use the dictionary to tell how many days are in the month.
 - b) Print out all the keys in alphabetical order.
 - c) Print out all the months with 31 days.
 - d) Print out the (key-value) pairs sorted by the number of days in each month.
- iii) Given the dictionary $x = \{'k1': 'v1', 'k2': 'v2', 'k3': 'v3'\}$, create a dictionary with the opposite mapping i.e., write a program to create the dictionary as:
`inverted_x = {'v1': 'k1', 'v2': 'k2', 'v3': 'k3'}`
- iv) Given two dictionaries say d1 and d2. Write a program that lists the overlapping keys of the two dictionaries, i.e., if a key of d1 is also a key of d2, then list it.
- v) Write a program that checks if two same values in a dictionary have different , keys. That is for

dictionary d1 = {'a':10, 'b':20, 'c':10}, the program should print “**2 keys have same values**” and for dictionary d2 = {'a':10, 'b':20, 'c':30}, the program should print “**No keys have same values**”

[SUMITA ARORA]

- vi) Write a Python code to create a dictionary having names of your friends as Values associated with suitable keys. Display the names of the friends whose names contain more than three characters.
- vii) Write a Python code to create a dictionary containing some values as numbers by using suitable keys. Display all the values which are perfect numbers.
- viii) Write a Python code to create a dictionary containing three key-value pairs such that each value contains a list of integers. Find and display the maximum of the numbers among each list.
- ix) Write a Python code to create two dictionaries (say d1 and d2) containing some key-value pairs. Find the frequency of each value of dictionary d1 in dictionary d2.
- x) Write a Python code to input the names of the employees with their salaries as key-value pairs of a dictionary. Display minimum and maximum salary along with the names of the employees.

[V. K. PANDEY / D. K. DEY]

HOME WORK

1) What will be the output of following code snippets:

a) `D = dict()`

`for i in range (3):`

`for j in range(2):`

`D[i] = j`

`print(D)`

b) `a={}`

`a[2]=1`

`a[1]=[2,3,4]`

`print(a[1][1])`

c) `dictionary = {1:'1', 2:'2', 3:'3'}`

`del dictionary[1]`

`dictionary[1] = '10'`

`del dictionary[2]`

`print(len(dictionary))`

d) `dict1 = {"name": "Mike", "salary": 8000}`

`temp = dict1.pop("age")`

`print(temp)`

e) `dict1 = {"key1":1, "key2":2}`

`dict2 = {"key2":2, "key1":1}`

`print(dict1 == dict2)`

f) `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)`

2) Write a program to count the frequency of each character of a string input by the user using dictionary.

3) Write a Python Program to create a dictionary with key as first character and value as words starting with that character.

- 4) Python Program to Generate a Dictionary that Contains Numbers (between 1 and n) in the Form (x,x*x).
- 5) Write a Python script to concatenate following dictionaries to create a new one.
dic1 = {1:10,2:20}
dic2 = {3:30,4:40}
dic3 = {5:50,6:60}
- 6) Write a Python program to remove duplicates from Dictionary.
- 7) Write a Python program to get the maximum and minimum value in a dictionary.