## Assignment 6C – Quiz Functionality with Dictionary Operations & Sequence Unpacking

**Author:** Praveen Kumar G (22MID0300)

**Date:** August 2, 2025

**Purpose:** Extend the PDS Management System by embedding a dictionary-based quiz to test understanding of dictionary operations, inspired by the GeeksforGeeks Python Dictionary Quiz. Demonstrates use of nested dictionaries, sequence unpacking, \*args/\*\*kwargs, and dictionary lookup.

Marks: 8/10

**Team mates: (Group-1)** 

- 1) G Praveen kumar (22MID0300)
- 2) Manaswitha (22MIC0127)
- 3) Vithun
- 4) Devi
- 5) Rohit

### Part-1:

### Concepts Used from GeeksforGeeks Dictionary Interview Questions

In this project, I used some of the ideas I learned from the GeeksforGeeks "Top 30 Python Dictionary Interview Questions" page.

### 1) Nested Dictionary

I learned how to store detailed info using dictionaries inside a dictionary. So I used that to create a quiz where each question has its own data like options and the correct answer.

2) Looping through Dictionary with .items()

I used .items() to loop through the quiz questions. This helped me get both the question number and its details easily in the loop.

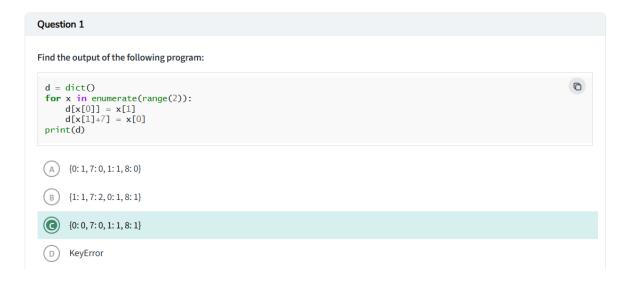
### 3) Unpacking Key and Value

From the examples, I saw how we can unpack the key and value directly in the for loop (like for key, value in dict.items()). I used this to neatly access question IDs and question info.

### 4) Dictionary Updates and Access

I understood how we can change values in a dictionary or get values safely. Even though I didn't use .get() here, learning about it helped me think about safe access.

### Part-B (Quiz)





# Prind the output of the following program: d = {1 : {'A' : {1 : "A"}, 2 : "B"}, 3 : "C", 'B' : "D", "D": 'E'} print(d[d[d[1][2]]], end = " ") print(d[d[1][1][2]]) A CB B E Key Error C BD D D B Discuss it Explanation Key-Value Indexing is used in the example above. D[1] = {'A' : {1 : "A"}, 2 : "B"}, D[1][2] = "B", D[D[1][2]] = D["B"] = "D" and d D["D"] = "E". D[1] = {'A' : {1 : "A"}, 2 : "B"}, D[1]["A"][2] doesn't exists, thus Key Error.

### Question 4

Find the output of the following program:

```
d = dict()
for i in range (3):
    for j in range(2):
        d[i] = j
print(d)
```

- (A) {0:0,1:0,2:0}
- B {0: 1, 1: 1, 2: 1}
- (0:0,1:0,2:0,0:1,1:1,2:1)
- D TypeError: Immutable object

### Discuss it

### Explanation

1st loop will give 3 values to i 0, 1 and 2. In the empty dictionary, valued are added and overwrited in j loop, for eg. D[0] = [0] becomes D[0] = 1, due to overwriting.

### Question 5

Question 5: Find the output of the following program:

```
d = {1 : [1, 2, 3], 2: (4, 6, 8)}
d[1].append(4)
print(d[1], end = " ")
li = [d[2]]
li.append(10)
d[2] = tuple(L)
print(d[2])
```

- (A) [1, 2, 3, 4] ((4, 6, 8), 10)
- (B) [1, 2, 3, 4] (4, 6, 8, 10)
- (c) [1, 2, 3, 4] TypeError: tuples are immutable
- (D) [1, 2, 3, 4] [4, 6, 8, 10]

### Discuss it

### Explanation

In the first part, key-value indexing is used and 4 is appended into the list. As tuples are immutable, in the second part the tuple is converted into a list, and value 10 is added finally then converted back to tuple.

### Question 6

What will be the output of the following code?

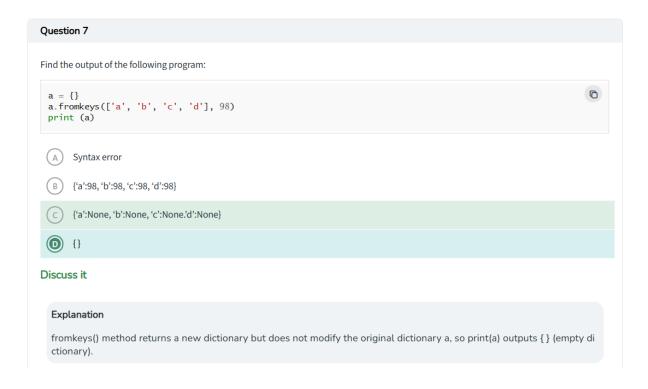
```
s = "GeeksforGeeks"
print(s[0], s[-1])

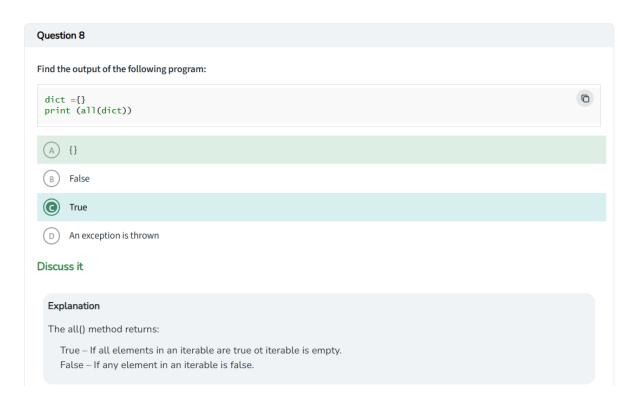
(A) GG
(B) Gs
(C) Gk
(D) ek
```

### Explanation

Discuss it

s[0] accesses the first character 'G' and s[-1] accesses the last character 's'.





## Prind the output of the following program: a = {'geeks' : 1, 'gfg' : 2} b = {'geeks' : 2, 'gfg' : 1} print (a == b) A True True True None Discuss it Explanation If two dictionary are the same it returns true, otherwise it returns false.

