

PDF generated at: 17 Jun 2024 17:08:54 UTC

View this report on HackerRank □

Score

91.7% • 55 / 60

scored in CodePath TIP101: Unit 2 Assessment, Version A - Summer 2024 in 36 min 23 sec on 17 Jun 2024 09:30:26 PDT

Candidate Information

Email concepting@protonmail.com

Test CodePath TIP101: Unit 2 Assessment, Version A - Summer 2024

Candidate Packet View ℃

Taken on 17 Jun 2024 09:30:26 PDT

Time taken 36 min 23 sec/ 90 min

Work Experience < 1 years

Invited by CodePath

Suspicious Activity detected

Code similarity



Code similarity

1 question

Skill Distribution

Candidate Report Page 1 of 12



There is no associated skills data that can be shown for this assessment

Tags Distribution



There is no associated tags data that can be shown for this assessment

Questions

Status	No.	Question	Time Taken	Skill	Score
8	1	What will be the output of the following Python code? Multiple Choice	7 sec	-	5/5
⊗	2	What will be the output of the following Python code? Multiple Choice	8 sec	-	0/5

Candidate Report Page 2 of 12

⊗	3	What will be the output of the following Python code? Multiple Choice	8 sec -	5/5
©	4	Get Top Player Multiple Choice	46 sec	5/5
\otimes	5	Contains Duplicate Coding	25 min 6 - sec	20/20
⊗	6	Element Frequency Greater than N Coding	8 min 57 - sec	20/20 🏳

1. What will be the output of the following Python code?

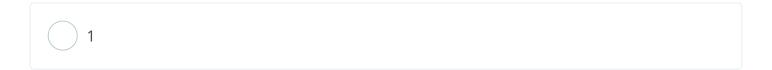
Multiple Choice

Question description

```
my_dict = {'a': 1, 'b': 2, 'c': 3}
print(my_dict['b'])
```

Candidate's Solution

Options: (Expected answer indicated with a tick)



Candidate Report Page 3 of 12

2	\otimes
{'b': 2}	
KeyError	
① No comments.	
2. What will be the output of the following Python code? Multiple Choice Question description	⊗ Incorrect
my_list = ['a', 'b', 'c', 'd'] print(my_list[1:3])	
Candidate's Solution Options: (Expected answer indicated with a tick)	
['a', 'b']	
['b', 'c', 'd']	

Candidate Report Page 4 of 12

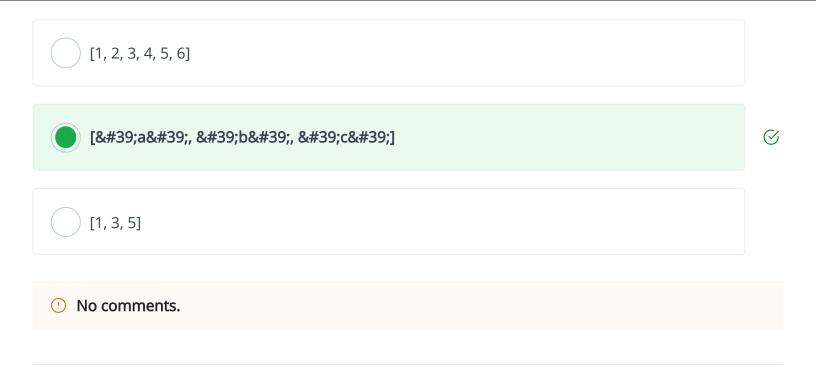
['b', 'c']	\otimes
['c', 'd']	
No comments.	
3. What will be the output of the following Python code? Multiple Choice	⊘ Correct
	⊘ Correct
Multiple Choice Question description	⊘ Correct
<pre>Multiple Choice Question description my_dict = {'a': [1, 2], 'b': [3, 4], 'c': [5, 6]}</pre>	⊘ Correct

Candidate's Solution

Options: (Expected answer indicated with a tick)



Candidate Report Page 5 of 12



4. Get Top Player

Multiple Choice

Question description

The following function accepts a dictionary which maps player names to their score and wants to return the name of the highest scoring player. However, it's not working as intended. How do we modify this function to actually return the player with highest score.

```
dictionary = {"Audrey": 90, "Char": 60, "Mario": 95, "Kyra": 12}
# Expected Output: "Mario"

def get_top_player(dictionary):
    high_score = 0
    top_player = ""
    for name, score in dictionary.items():
        if score > high_score:
            high_score += score
            top_player = name
    return top_player
```

Candidate Report Page 6 of 12

Candidate's Solution

Options: (Expected answer indicated with a tick)



Replace high_score += score with high_score = score



Replace top_player = "" with top_player = dictionary[0]

Replace return top_player with return high_score

Replace top_player = name with name = top_player

() No comments.

5. Contains Duplicate

Coding

Question description

Given an integer list **nums**, return **True** if any value appears **at least twice** in the list, and return **False** if every element is distinct.

Candidate Report Page 7 of 12

HackerRank ryan

```
Example 1:
Input: [1, 2, 3, 1]
Output: True

Example 2:
Input: [1, 2, 3, 4]
Output: False
```

Candidate's Solution

Language used: Python 3

```
1 #!/bin/python3
 2
 3 import math
4 import os
 5 import random
6 import re
7 import sys
8
9
10
11 #
12 # Complete the 'contains_duplicate' function below.
13 #
14 # The function is expected to return a BOOLEAN.
15 # The function accepts INTEGER_ARRAY nums as parameter.
16 #
17
18 def contains duplicate(nums):
       # Write your code here
19
20
       new set = set()
21
22
       for i in range(len(nums)):
23
           if nums[i] in new set:
24
               return True
25
           new set.add(nums[i])
26
       return False
27
28 if name == ' main ':
29
       fptr = open(os.environ['OUTPUT PATH'], 'w')
30
31
       temp = input()
32
```

Candidate Report Page 8 of 12

```
33
       if len(temp) > 40:
34
           input string = temp
           chunks = input_string.split(", ")
35
            list_of_lists = [list(map(int, chunk.split())) for chunk in chunks]
36
            result = [contains duplicate(lst) for lst in list of lists]
37
38
       else:
            result = contains_duplicate([int(n) for n in temp.split()])
39
40
       fptr.write(str(result) + '\n')
41
42
       fptr.close()
43
44
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample	Success	0	0.0326 sec	10.4 KB
Testcase 1	Easy	Sample	Success	0	0.0326 sec	10.3 KB
Testcase 2	Easy	Hidden	Success	0	0.0295 sec	10.4 KB
Testcase 3	Easy	Hidden	Success	0	0.0295 sec	10.2 KB
Testcase 4	Easy	Hidden	Success	20	0.0307 sec	10.2 KB

No comments.

6. Element Frequency Greater than N

Correct

Candidate Report Page 9 of 12

HackerRank ryan

Coding

Question description

Given a list of integers nums and an integer **n**, return a dictionary with elements as keys and their frequencies as values, but only include elements whose frequency is greater than **n**.

```
Example 1:
Input: nums = [1, 1, 2, 3, 3, 4], n = 1
Output: {1: 2, 3: 3}

Example 2:
Input: nums = [1, 2, 3, 4, 5], n = 0
Output: {1: 1, 2: 1, 3: 1, 4: 1, 5: 1}
```

Language used: Python 3

Candidate's Solution

```
1 #!/bin/python3
 2
3 import math
 4 import os
 5 import random
6 import re
7 import sys
8
9
10
11 #
12 # Complete the 'frequency greater than n' function below.
13 #
14 # The function is expected to return a DICTIONARY.
15 # The function accepts following parameters:
16 # 1. INTEGER ARRAY nums
17 #
      2. INTEGER n
18 #
19
20 def frequency_greater_than_n(nums, n):
       # Write your code here
21
22
       frequency = {}
23
```

Candidate Report Page 10 of 12

```
24
        for num in nums:
25
            if num in frequency:
                frequency[num] += 1
26
27
            else:
28
                frequency[num] = 1
29
30
        result = {}
31
32
        for key, value in frequency.items():
33
            if value > n:
                result[key] = value
34
35
36
        return result
37
38
   if name == ' main ':
39
        fptr = open(os.environ['OUTPUT PATH'], 'w')
40
41
       t = input()
42
43
       if len(t) > 65:
44
            chunks in range = t.split(", ")
45
            list of lists in range = [list(map(int, chunk.split())) for chunk in
   chunks in range]
46
            result = [frequency_greater_than_n(lst[1:], lst[0]) for lst in
   list_of_lists_in_range]
       else:
47
48
            temp = ([int(n) for n in t.split()])
49
            result = frequency greater than n(temp[1:], temp[0])
50
51
        fptr.write(str(result) + '\n')
52
       fptr.close()
53
54
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 0	Easy	Sample	Success	0	0.0261 sec	10.2 KB
Testcase 1	Easy	Sample	Success	0	0.043 sec	10.3 KB

Candidate Report Page 11 of 12

Testcase 2	Easy	Hidden	Success	0	0.0325 sec	10.3 KB
Testcase 3	Easy	Hidden	Success	0	0.039 sec	10.3 KB
Testcase 4	Easy	Hidden	Success	20	0.0403 sec	10.1 KB

No comments.

Candidate Report Page 12 of 12