Module 2 Graded Assessment

TOTAL POINTS 10

1. Complete the function by filling in the missing parts. The color_translator function receives the name of a color, then prints its hexadecimal value. Currently, it only supports the three additive primary colors (red, green, blue), so it returns "unknown" for all other colors.

1 point

```
def color_translator(color):
    if color == "red":
        hex_color = "#ff0000"
    elif color == "green":
        hex_color = "#00ff00"
    elif color == "blue":
        hex_color = "#0000ff"
    else:
        hex_color = "unknown"
    return hex_color

print(color_translator("blue")) # Should be #0000ff
print(color_translator("yellow")) # Should be unknown
print(color_translator("red")) # Should be #ff0000
print(color_translator("black")) # Should be unknown
print(color_translator("green")) # Should be #00ff00
print(color_translator("")) # Should be unknown
```

Run

Reset

```
#0000ff
unknown
#ff0000
unknown
#00ff00
unknown
```

2. What's the value of this Python expression: "big" > "small"

1 point

- ○ True
- ◉ False
- ○ big
- ○ small

3. What is the elif keyword used for?

1 point

- ○ To mark the end of the if statement
- ◉ To handle more than two comparison cases
- ○ To replace the "or" clause in the if statement
- ○ Nothing - it's a misspelling of the else-if keyword

4. Students in a class receive their grades as Pass/Fail. Scores of 60 or more (out of 100) mean that the grade is "Pass". For lower scores, the grade is "Fail". In addition, scores above 95 (not included) are graded as "Top Score". Fill in this function so that it returns the proper grade.

1 point

```
def exam_grade(score):
    if score > 95:
        grade = "Top Score"
    elif score >= 60:
        grade = "Pass"
    else:
        grade = "Fail"
    return grade

print(exam_grade(65)) # Should be Pass
print(exam_grade(55)) # Should be Fail
print(exam_grade(60)) # Should be Pass
print(exam_grade(95)) # Should be Pass
print(exam_grade(100)) # Should be Top Score
print(exam_grade(0)) # Should be Fail
```

Run

Reset

```
Pass
Fail
Pass
Pass
Top Score
Fail
```

5. What's the value of this Python expression: 11 % 5?

1 point

- ○ 2.2

- ○ 2
- ◉ 1
- ○ 0

6. Complete the body of the ***format_name*** function. This function receives the ***first_name*** and ***last_name*** parameters and then returns a properly formatted string. 1 point

Specifically:

If both the ***last_name*** and the ***first_name*** parameters are supplied, the function should return like so:

```
print(format_name("Ella", "Fitzgerald"))
Name: Fitzgerald, Ella
```

If only ***one*** name parameter is supplied (either the first name *or* the last name) , the function should return like so:

```
print(format_name("Adele", ""))
Name: Adele
```

or

```
print(format_name("", "Einstein"))
Name: Einstein
```

Finally, if both names are blank, the function should return the empty string:

```
print(format_name("", ""))

```

Implement below:

```
def format_name(first_name, last_name):
    # code goes here
    string = ''
    if first_name and last_name:
        string = 'Name: ' + last_name + ', ' + first_name
    elif first_name or last_name:
        string = 'Name: ' + first_name + last_name
    #elif last_name:
    #   string = 'Name: ' + last_name
    else:
        string = ''

    return string

print(format_name("Ernest", "Hemingway"))
# Should return the string "Name: Hemingway, Ernest"

print(format_name("", "Madonna"))
# Should return the string "Name: Madonna"

print(format_name("Voltaire", ""))
# Should return the string "Name: Voltaire"

print(format_name("", ""))
# Should return an empty string
```

Run

Reset

```
Name: Hemingway, Ernest
Name: Madonna
Name: Voltaire
```

7. The longest_word function is used to compare 3 words. It should return the word with the most number of characters (and the first in the list when they have the same length). Fill in the blank to make this happen. 1 point

```
def longest_word(word1, word2, word3):
    if len(word1) >= len(word2) and len(word1) >= len(word3):
        word = word1
    elif len(word2) >= len(word1) and len(word2) >= len(word3):
        word = word2
    else:
        word = word3
    return(word)

print(longest_word("chair", "couch", "table"))
print(longest_word("bed", "bath", "beyond"))
print(longest_word("laptop", "notebook", "desktop"))
```

Run

Reset

```
chair
beyond
notebook
```

8. What's the output of this code? 1 point

```
def sum(x, y):
        return(x+y)
print(sum(sum(1,2), sum(3,4)))
```

10

9. What's the value of this Python expression? 1 point

((10 >= 5*2) and (10 <= 5*2))

- (•) True
- ( ) False
- ( ) 10
- ( ) 5*2

10. The fractional_part function divides the numerator by the denominator, and returns just the fractional part (a number between 0 and 1). Complete the body of the function so that it returns the right number. Note: Since division by 0 produces an error, if the denominator is 0, the function should return 0 instead of attempting the division. 1 point

```
def fractional_part(numerator, denominator):
    # Operate with numerator and denominator to
    # keep just the fractional part of the quotient
    if denominator == 0: return 0
    number = numerator / denominator
    return number-int(number)

print(fractional_part(5, 5)) # Should be 0
print(fractional_part(5, 4)) # Should be 0.25
print(fractional_part(5, 3)) # Should be 0.66...
print(fractional_part(5, 2)) # Should be 0.5
print(fractional_part(5, 0)) # Should be 0
print(fractional_part(0, 5)) # Should be 0
```

Run

Reset

```
0.0
0.25
0.6666666666666667
0.5
0
0.0
```

I, **Piyush Sambhi**, understand that submitting work that isn't my own may result in permanent failure of this course or deactivation of my Coursera account.

Learn more about Coursera's Honor Code

Save Submit