Congratulations! You passed!

Grade Latest Submission received 100% Grade 100%

To pass 80% or higher

Go to next item

 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. 1/1 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"
     10
                         return hex_color
                print(color_translator("blue")) # Should be #0000ff
print(color_translator("yellow")) # Should be unknown
print(color_translator("red")) # Should be #ff0000
     13
                print(color_translator("black")) # Should be unknown
print(color_translator("green")) # Should be #00ff00
print(color_translator("")) # Should be unknown
     15
                                                                                                                                                                                    Run
     16
#0000ff
#ff0000
#00ff00
```

⊘ Correct

Well done! You're breezing through the if-else clauses!

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

1/1 point

- O True
- False
- O big
- O small

⊘ Correct

You nailed it! The conditional operator > checks if two values are equal. The result of that operation is a boolean: either True or False. Alphabetically, "big" is less than "small".

3. What is the elif keyword used for?

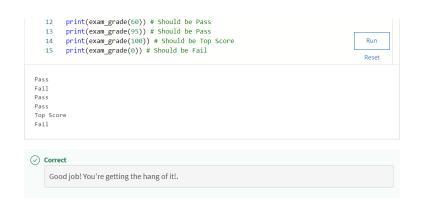
1/1 point

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

⊘ Correct

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 / 1 point



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

1/1 point

- O 2.2
- O 2
- 1
- 0 0

⊘ Correct

Excellent! "%" is the modulo operator, which returns the remainder of the integer division between two numbers. 11 divided by 5 equals 2 with remainder of 1.

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. 1/1 point

Specifically:

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

```
2 Name: Fitzgerald, Ella
```

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

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

or

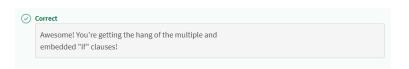
```
1 print(format_name("", "Einstein"))
2 Name: Einstein
```

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

```
1 print(format_name("", ""))
2
```

Implement below:

```
def format_name(first_name, last_name):
                  # code goes here
                  if first_name and last_name:
| string = "Name: " + last_name + ", " + first_name
elif first_name and last_name=="" :
                 string = "Name: " + first_name
elif first_name=="" and last_name ;
string = "Name: " + last_name
                  else :
                        string = ""
    10
    11
    12
                 return string
    13
           print(format_name("Ernest", "Hemingway"))
# Should return the string "Name: Hemingway, Ernest"
    16
           print(format_name("", "Madonna"))
# Should return the string "Name: Madonna"
    19
           print(format_name("Voltaire", ""))
# Should return the string "Name: Voltaire"
           print(format_name("", ""))
# Should return an empty string
                                                                                                                                        Run
    23
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.

1/1 point

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

chair
        beyond
        notebook
```

⊘ Correct

You got it! You've figured out how to use an elif clause, well done!

8. What's the output of this code?

1/1 point

10

⊘ Correct

You nailed it! We're calling the sum function 3 times: returning 3, then 7, then adding up 3 plus 7 for the total of 10.

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

1/1 point

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

True

O False

O 10 O 5*2

0 5 2

✓ Correc

Right on! When using the "and" operator, a statement is True if both parts of the conditional are True.

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.

1/1 point

```
def fractional_part(numerator, denominator):
                 # Operate with numerator and denominator to
            # keep just the fractional part of the quotient
                if denominator == 0 or numerator == 0 :
                      return 0
               return (numerator % denominator)/denominator
           print(fractional_part(5, 5)) # Should be 0
          print(fractional_part(5, 4)) # Should be 0.25

print(fractional_part(5, 3)) # Should be 0.25

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
   10
   11
   13
                                                                                                                            Run
   14
           print(fractional_part(0, 5)) # Should be 0
0.25
0.666666666666666
```

⊘ Correct

Well done! You're handling the math operations, as well as division by 0, perfectly!