

Congratulations! You passed!

Grade
received **100%**

Latest Submission
Grade **100%**

To pass 80% or
higher

[Go to next item](#)

1. Complete the code to output the statement, "192.168.1.10 is the IP address of Printer Server 1". Remember that precise syntax must be used to receive credit.

1 / 1 point

```

1 IP_address = "192.168.1.10"
2 host_name = "Printer Server 1"
3 print(IP_address + " is the IP address of " + host_name)
4 # Should print "192.168.1.10 is the IP address of Printer Server 1"

```

Run
Reset

192.168.1.10 is the IP address of Printer Server 1

 **Correct**

Correct.

2. What is the value of this Python expression: `"blue" == "Blue"`?

1 / 1 point

- ☐ orange
- ☐ True
- ☐ blue
- ☒ False

 **Correct**

3. What directly follows the `elif` keyword in an `elif` statement?

1 / 1 point

- ☐ A logical operator
- ☐ A colon
- ☐ A function definition
- ☒ A comparison

 **Correct**

4. Consider the following scenario about using `if-elif-else` statements:

1 / 1 point

The fall weather is unpredictable. If the temperature is below 32 degrees Fahrenheit, a heavy coat should be worn. If it is above 32 degrees but not above 50 degrees, then a jacket should be sufficient. If it is above 50 but not above 65 degrees, a sweatshirt is appropriate, and above 65 degrees a t-shirt can be worn.

Fill in the blanks in the function below so it returns the proper clothing type for the temperature.

```

1 def clothing_type(temp):
2     if temp > 65:
3         clothing = "T-Shirt"
4     elif temp > 50:
5         clothing = "Sweatshirt"
6     elif temp > 32:
7         clothing = "Jacket"
8     else:
9         clothing = "Heavy Coat"
10    return clothing
11
12
13 print(clothing_type(72)) # Should print T-Shirt
14 print(clothing_type(55)) # Should print Sweatshirt
15 print(clothing_type(65)) # Should print Sweatshirt
16 print(clothing_type(50)) # Should print Jacket
17 print(clothing_type(45)) # Should print Jacket
18 print(clothing_type(32)) # Should print Heavy Coat
19 print(clothing_type(0)) # Should print Heavy Coat

```

Run
Reset

T-Shirt
Sweatshirt
Sweatshirt
Jacket
Jacket
Heavy Coat
Heavy Coat

 **Correct**

Correct.

5. In the following code, what would be the output:

1 / 1 point

```
1 test_num = 12
2 if test_num > 15:
3     print(test_num / 4)
4 else:
5     print(test_num + 3)
6
```

- ☒ 15
- ☐ 4
- ☐ 3
- ☐ 12

✓ Correct

6. Fill in the blanks to complete the function. The "complementary_color" function receives a primary color name in all lower case, then prints its complementary color. Currently, the function only supports the primary colors of red, yellow, and blue. It returns "unknown" for all other colors or if the word has any uppercase characters.

1 / 1 point

```
1 def complementary_color(color):
2     if color == "blue":
3         complement = "orange"
4     elif color == "yellow":
5         complement = "purple"
6     elif color == "red":
7         complement = "green"
8     else:
9         complement = "unknown"
10    return complement
11
12 print(complementary_color("blue")) # Should print orange
13 print(complementary_color("yellow")) # Should print purple
14 print(complementary_color("red")) # Should print green
15 print(complementary_color("black")) # Should print unknown
16 print(complementary_color("Blue")) # Should print unknown
17 print(complementary_color("")) # Should print unknown
```

Run

Reset

orange
purple
green
unknown
unknown
unknown

✓ Correct

Correct

7. Can you calculate the output of this code?

1 / 1 point

```
1 def greater_value(x, y):
2     if x > y:
3         return x
4     else:
5         return y
6
7
8 print(greater_value(10,3*5))
```

15

✓ Correct

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

1 / 1 point

x = 5*2

((10 != x) or (10 > x))

- ☐ True
- ☒ False
- ☐ 15
- ☐ 10

✓ Correct

9. Fill in the blanks to complete the "safe_division" function. The function accepts two numeric variables through the function parameters and divides the "numerator" by the "denominator". The function's main purpose is to prevent a ZeroDivisionError by checking if the "denominator" is 0. If it is 0, the function should return 0 instead of attempting the division. Otherwise all other numbers will be part of the division equation. Complete the body of the function so that the function completes its purpose.

1 / 1 point

```
1 def safe_division(numerator, denominator):
2     # Complete the if block to catch any "denominator" variables
3     # that are equal to 0.
4     if denominator == 0:
5         result = 0
6     else:
7         # Complete the division equation.
8         result = numerator/denominator
9     return result
10
11
12 print(safe_division(5, 5)) # Should print 1.0
13 print(safe_division(5, 4)) # Should print 1.25
14 print(safe_division(5, 0)) # Should print 0
15 print(safe_division(0, 5)) # Should print 0.0
```

Run
Reset

1.0
1.25
0
0.0

✓ Correct

Correct.

10. Which of the following are good coding-style habits? Select all that apply.

1 / 1 point

☒ Cleaning up duplicate code by creating a function that can be reused

✓ Correct

☐ Writing code using the least amount of characters as possible

☒ Adding comments

✓ Correct

☒ Refactoring the code

✓ Correct