

DUMPS ARENA

Foundations of Programming (Python)E010
JIV1

WGU Foundations-of-Programming-Python

Version Demo

Total Demo Questions: 4

Total Premium Questions: 44

Buy Premium PDF

<https://dumpsarena.co>

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

QUESTION NO: 1

Which punctuation mark must appear at the end of an if statement line?

- A. ; semicolon
- B. : colon
- C. . period
- D. , comma

ANSWER: B

Explanation:

The correct answer is : **colon**. In Python, an `if` statement begins a compound statement: it has a header that states a condition and an indented suite of code that runs when that condition evaluates to `True`. Python requires a colon at the end of the header to mark the transition from the condition to the indented block. For example, `if score >= 70:` must end with a colon before the following indented statements. This same header-and-colon pattern is used for other compound statements, including `elif`, `else`, `for`, `while`, `def`, `class`, `try`, and `with`. The colon is therefore a required part of Python's syntax, not merely a style convention. Python's official language reference defines compound statements as having a header ending in a colon followed by a suite, and its tutorial shows this syntax in conditional examples. See the [Python Language Reference: Compound statements](#) and the [Python Tutorial: More Control Flow Tools](#).

QUESTION NO: 2 - (SIMULATION)

Write a complete function `calculate_discount(price, discount_percent)` that calculates and returns the final price after applying a discount percentage.

For example, `calculate_discount(75, 20)` should return `60.0`.

```
def calculate_discount(price, discount_percent):  
    # TODO: Calculate and return the final price after discount  
    pass
```

ANSWER: Check the explanation for the answer

Explanation:

Replace `pass` with a `return` statement that multiplies the price by the percentage remaining after the discount.

```
def calculate_discount(price, discount_percent):  
    return price * (1 - discount_percent / 100)
```

For example, `calculate_discount(75, 20)` calculates `75 * (1 - 20 / 100)`, which returns `60.0`.

QUESTION NO: 3 - (SIMULATION)

Fix the logic error in this function that should return the larger of two numbers.

```
def find_maximum(a, b):  
    if a < b:  
        return a  
    else:
```

return b

ANSWER: Check the explanation for the answer

Explanation:

Correct the return values so the function returns the larger number:

```
def find_maximum(a, b):  
    if a < b:  
        return b  
    else:  
        return a
```

When $a < b$, b is the larger value, so return b . Otherwise, return a (including when both values are equal).

QUESTION NO: 4

Which data type is the value 3.14 in Python?

- A. Integer
- B. Float
- C. String
- D. Boolean

ANSWER: B

Explanation:

Float is correct because the literal `3.14` includes a fractional component represented with a decimal point. In Python, numeric literals written with a decimal point are created as floating-point values, whose built-in type is `float`. For example, evaluating `type(3.14)` produces `<class 'float'>`. Floating-point values are used for numbers that may include a fractional part, such as measurements, percentages, rates, and decimal calculations. Python's standard documentation identifies `float` as the built-in type for floating-point numbers and describes these values as machine-level double-precision floating-point numbers. The Python tutorial also shows decimal numeric literals, such as `5.0`, as floating-point values. Thus, because `3.14` is written with a decimal point, Python classifies it as a `float`. See the [Python documentation for float](#) and the [Python tutorial's numeric-types overview](#).