

Q1: What is a variable?

A variable is a named container in the computer memory with a meaningful name which can hold a value or data that can be changed.

Q2: How do you initialize, assign, and reassign variable values in Python?

In Python, variables are assigned value using an assignment operator (=) (Eg: myNo = 13)

Initialization means creating a variable for the first time and giving it an initial value.

Reassignment happens when you change the value of an existing variable. When you reassign a variable, you simply overwrite the new value to the variable. (Eg: myNo = 12)

Q3: What are data types in programming?

Data types tell the computer what kind of data it is processing. Computers need to know if data is a word, a whole number, or a decimal number to process it correctly.

Q4: What are the three main data types we use in Python?

Commonly used data types in Python are:

1. Integer (int): Whole numbers without decimals. (e.g., `5`, `-10`, `100`).
2. Real / Float (float): Numbers with decimal points. (e.g., `2.5`, `3.14`, `-0.01`).
3. String (str): Text, words, or characters enclosed in quotes. (e.g., `"Hello"`, `"123"`, `"A"`).

Q5: Why is it important to use the correct data type?

A: Because computers perform different operations on different types.

If we add two Integers (5 + 5), the computer does Math: result is 10.

If we add two Strings ("5" + "5"), the computer joins them: result is "55".

Using the wrong type can cause errors or incorrect results.

Q6: How do I change data from one type to another (Casting)?

A: You can "cast" (convert) data using special functions:

`int("5")` converts the string "5" into the integer 5.

`str(10)` converts the integer 10 into the string "10".

`float(5)` converts the integer 5 into the decimal 5.0.

Q7: What are the rules for naming variables in Python?

1. Must start with a letter or underscore _.
2. Cannot start with a number.
3. No spaces allowed (use _ instead, like my_score).
4. Case sensitive (Score and score are different variables).

Q8: How do I read data from a user in Python?

A: We use the `input()` command to read data from the user in Python.

Syntax: `variable_name = input("Prompt for user: ")`

Example: `name = input("Enter your name: ")``

Note: `input()` always reads data as a String by default.

Q9: How do I store user input as a Number (Integer)?

A: Since `input()` collects Strings, you must cast it if you want to do math.

Syntax: `age = int(input("Enter your age: "))`

Q10: How do I output data to the screen?

A: Use the `print()` command to output data to the screen.

Syntax: `print("Hello", name)` or `print(variable_name)`

Q11: What are Arithmetic Operations?

A: These are symbols used to perform math.

+ (Addition)

- (Subtraction)

* (Multiplication)

/ (Division)

% (Modular)

Q12: What is BIDMAS and how is it implemented in Python?

BIDMAS is a rule that tells you the correct order to solve math problems in Python:

Brackets, Indices, Division, Multiplication, Addition, and Subtraction.

- **B** — Brackets () are solved first.
- **I** — Indices (powers or exponents) like `2**3` are solved second.
- **D & M** — Division / and Multiplication * come next, working from left to right.
- **A & S** — Addition + and Subtraction - are done last, also from left to right.

Example:

Without brackets

`result_1 = 2 + 3 * 4` # Python does `3 * 4` first (12), then adds 2

`print(result_1)` Output: 14

With brackets

`result_2 = (2 + 3) * 4` # Brackets force Python to do `2 + 3` first (5)

`print(result_2)` Output: 20

With indices (powers)

```
result_3 = 2 + 3 ** 2 # Python calculates 3 ** 2 first (9), then adds 2  
print(result_3)      Output: 11
```

Q13: What are Comparison Operations?

A: These symbols compare two values and return either True or False.

- == (Equal to)
- != (Not equal to)
- > (Greater than)
- < (Less than)
- >= (Greater than or equal to)
- <= (Less than or equal to)

Q14: How do I write a simple IF-ELSE statement?

A: This is used when there are two possible paths.

Syntax:

```
if condition:  
    do these steps if the condition is true  
else:  
    do these steps if the condition is false
```

Example:

```
if age >= 18:  
    print("You are an adult.")  
else:  
    print("You are a child.")
```