Python: Variables

Connect with me: Youtube | LinkedIn | WhatsApp Channel | Web | Facebook | Twitter

- Download PDF
- To access the updated handouts, please click on the following link: https://yasirbhutta.github.io/python/docs/variables.html

Variables

- Storage containers for data (numbers, text, etc.).

What is a variable

- A variable is a named storage location in a computer's memory that is used to hold data or values. It allows programmers to store and manipulate data within a program.

Purpose: Variables provide a way to store and manage data that can be used and manipulated throughout a program. They make programs more flexible and allow for dynamic data storage.

Assignment statement: in Python is used to assign a value to a variable. Its primary purpose is to store and manage data within a program.

Imagine variables as labeled boxes:

- You have boxes for storing different things (numbers, words, etc.).
- Each box has a name (label) to identify what's inside.
- You can put things in, take them out, and change what's inside.

Example #1: Storing a name

```
name = "Muhammad Hamza"
print(name)
```

Example #2: Tracking a score:

```
score = 0
score = score + 10 # adds 10 to the score
print(score)
```

Example #3: Remembering a favorite color

```
favorite_color = "blue" #stores "blue"  in variable
print(favorite_color)
```

Example #4: Calculating the area of a rectangle

```
length = 10
width = 5

# calculates the area
area = length * width
print(area)
```

Understanding Dynamic Variables in Python with Examples

Important: In Python, variables are dynamic, meaning they can change types during the execution of a program. This flexibility allows you to assign a value of any type to a variable and later reassign it to a value of a different type without any issues. This dynamic nature of variables is due to Python being a dynamically typed language.

Example #5: Dynamic Variables in Python

```
# Initial assignment of an integer value
x = 10
print(x)  # Output: 10
print(type(x))  # Output: <class 'int'>

# Reassigning a string value to the same variable
x = "Hello, World!"
print(x)  # Output: Hello, World!
print(type(x))  # Output: <class 'str'>

# Reassigning a list to the same variable
x = [1, 2, 3]
print(x)  # Output: [1, 2, 3]
print(type(x))  # Output: <class 'list'>

# Reassigning a float value to the same variable
x = 3.14
print(x)  # Output: 3.14
print(type(x))  # Output: <class 'float'>
```

In this example:

1. `x` is initially assigned an integer value of `10`.
2. `x` is then reassigned a string value `"Hello, World!"`.
3. `x` is later reassigned a list `[1, 2, 3]`.
4. Finally, `x` is reassigned a float value `3.14`.

Each time, the type of `x` changes dynamically to match the type of the value assigned to it. This flexibility is one of the powerful features of Python, allowing for more concise and adaptable code.

Key Points:

- **Choose meaningful names:** Use names that describe what the variable stores (e.g., pizza_slices instead of x).
  - video: Meaningful Variable Names | Python Best Practices
- **Assign values using =:** The equals sign is used to put a value into a variable.
- **Change values:** You can update a variable's value later in your code.
- **Use variables in calculations and operations:** Variables can be used just like regular numbers or text in expressions.
- **Think of variables as placeholders:** They hold information that can change as your program runs.

Key Terms

True/False (Mark T for True and F for False)

1. Variable names in Python are case-sensitive.
2. In Python, variables must be declared with a specific data type before they can be used.
3. The statement x = 5 both creates the variable x and assigns it the value 5.

Answer Key (True/False):

1. True
2. False
3. True

Multiple Choice (Select the best answer)

1. **What is a variable in Python?**

   - A) A reserved word in Python
   - B) A placeholder for storing data values
   - C) A function that prints data
   - D) A built-in library in Python

   Answer: B) A placeholder for storing data values

2. **Which statement best describes a variable in Python?**

   - A) A variable can hold multiple values at once.
   - B) A variable must be declared with a data type.
   - C) A variable is a name that refers to a value.
   - D) A variable is used only in loops.

   Answer: C) A variable is a name that refers to a value.

3. **What is the output of the following code?**

```
x = 10
print(x)
```

- A) `10`
- B) `x`
- C) `Error`
- D) `None`

Answer: A) `10`

4. **Which of the following is not true about variables in Python?**

   - A) Variables can be reassigned to different data types.
   - B) Variables must start with a letter or an underscore.
   - C) Variables are case-sensitive.
   - D) Variables must be declared before use.

   Answer: D) Variables must be declared before use.

5. **What will be the output of the following code?**

```
x = 5
y = x
x = 7
print(y)
```

   - A) `7`
   - B) `5`
   - C) `0`
   - D) `None`

   Answer: B) `5`

6. **Why is it important to use meaningful variable names?**

   - A) It is required by the Python interpreter.
   - B) It helps make the code more readable and maintainable.
   - C) It increases the execution speed of the program.
   - D) It is necessary for the code to run without errors.

   Answer: B) It helps make the code more readable and maintainable.

7. **What will be the output of the following code?**

```
a = 1
b = a
a = a + 1
print(a, b)
```

   - A) `1 1`
   - B) `2 1`

- C) `1 2`
- D) `2 2`

Answer: B) `2 1`

8. **Which of the following is a valid variable name in Python?**

   - A) `2ndValue`
   - B) `value#2`
   - C) `_value2`
   - D) `value-2`

   Answer: C) `_value2`

9. **Which of the following is a correct way to declare a variable in Python?**

- A) `int x = 5`
- B) `x = 5`
- C) `declare x = 5`
- D) `var x = 5`

Answer: B) `x = 5`

10. **What is the output of the following code?**

```
x = 5
y = "Hello"
print(x + y)
```

- A) `5Hello`
- B) `Hello5`
- C) `TypeError`
- D) `Hello 5`

Answer: C) `TypeError`

11. **Which of the following is not a valid variable name in Python?**

- A) `my_var`
- B) `_var`
- C) `2var`
- D) `var2`

Answer: C) `2var`

12. **Which of the following statements is true about variable assignment in Python?**

- A) Variables must be declared before they are assigned a value.
- B) Variables are created when they are first assigned a value.
- C) Variable names must begin with a number.

- D) Python variables must be declared with a type.

Answer: B) Variables are created when they are first assigned a value.

13. **What will be the output of the following code?**

```
x = 5
y = x
x = 10
print(y)
```

- A) `5`
- B) `10`
- C) `0`
- D) `5 10`

Answer: A) `5`

Fill in the Blanks

1. Variable names in Python must start with a letter or an _____.
2. Variables in Python are _____, meaning they can change type when assigned a new value.
3. The assignment operator in Python is the _____ symbol.

Answer Key (Fill in the Blanks):

1. underscore (_)
2. dynamic
3. equals (=)

Exercises

Review Questions

1. What is a variable in computer programming? Answer: A variable is a named storage location in a computer's memory that is used to hold data or values. It allows programmers to store and manipulate data within a program.

2. What is the purpose of using variables in programming? Answer: Variables provide a way to store and manage data that can be used and manipulated throughout a program. They make programs more flexible and allow for dynamic data storage.

3. What is the difference between declaring and initializing a variable? Answer: Declaring a variable involves specifying its name and data type, while initializing a variable means giving it an initial value. Initialization usually follows declaration but is not always required.

References and Bibliography