

# Variables and Assignment in Python

## Python Programming

Bindeshwar Singh Kushwaha

PostNetwork Academy

# What is a Variable?

- A variable is a name that refers to a value stored in memory.

# What is a Variable?

- A variable is a name that refers to a value stored in memory.
- It acts as a reference, not a storage container.

# What is a Variable?

- A variable is a name that refers to a value stored in memory.
- It acts as a reference, not a storage container.
- Unlike other languages, Python variables do not store values but rather point to objects in memory.

# What is a Variable?

- A variable is a name that refers to a value stored in memory.
- It acts as a reference, not a storage container.
- Unlike other languages, Python variables do not store values but rather point to objects in memory.
- Example:

# What is a Variable?

- A variable is a name that refers to a value stored in memory.
- It acts as a reference, not a storage container.
- Unlike other languages, Python variables do not store values but rather point to objects in memory.
- Example:

## Python Code

```
x = 10 # x refers to an object holding the value 10
```

# Variables Are Not Storage Containers

- In Python, variables are like labels attached to objects in memory.

# Variables Are Not Storage Containers

- In Python, variables are like labels attached to objects in memory.
- Assigning a variable does not copy data; it creates a reference.



# Variables Are Not Storage Containers

- In Python, variables are like labels attached to objects in memory.
- Assigning a variable does not copy data; it creates a reference.
- Example:

# Variables Are Not Storage Containers

- In Python, variables are like labels attached to objects in memory.
- Assigning a variable does not copy data; it creates a reference.
- Example:

## Python Code

```
a = [1, 2, 3] # a points to a list object  
b = a # b refers to the same list, not a copy
```

# Variables Are Not Storage Containers

- In Python, variables are like labels attached to objects in memory.
- Assigning a variable does not copy data; it creates a reference.
- Example:

## Python Code

```
a = [1, 2, 3] # a points to a list object  
b = a # b refers to the same list, not a copy
```

- Changing b affects a because both reference the same object.

# Variables Are Not Storage Containers

- In Python, variables are like labels attached to objects in memory.
- Assigning a variable does not copy data; it creates a reference.
- Example:

## Python Code

```
a = [1, 2, 3] # a points to a list object  
b = a # b refers to the same list, not a copy
```

- Changing b affects a because both reference the same object.
- Example:

# Variables Are Not Storage Containers

- In Python, variables are like labels attached to objects in memory.
- Assigning a variable does not copy data; it creates a reference.
- Example:

## Python Code

```
a = [1, 2, 3] # a points to a list object  
b = a # b refers to the same list, not a copy
```

- Changing b affects a because both reference the same object.
- Example:

## Python Code

```
b.append(4) # Modifies the list print(a) # Output: [1, 2, 3, 4]
```

# Variable Assignment in Python

- The assignment operator (=) is used to bind variables to values.

# Variable Assignment in Python

- The assignment operator (=) is used to bind variables to values.
- Python allows multiple assignments in a single statement.

# Variable Assignment in Python

- The assignment operator (=) is used to bind variables to values.
- Python allows multiple assignments in a single statement.
- Example:



# Variable Assignment in Python

- The assignment operator (=) is used to bind variables to values.
- Python allows multiple assignments in a single statement.
- Example:

## Python Code

```
a, b, c = 5, 10, 15  
name = "Alice"
```

# Multiple Assignment in Python

- Python allows assigning multiple values to multiple variables in one line.

# Multiple Assignment in Python

- Python allows assigning multiple values to multiple variables in one line.
- This feature makes code more concise and readable.

# Multiple Assignment in Python

- Python allows assigning multiple values to multiple variables in one line.
- This feature makes code more concise and readable.
- Example:

# Multiple Assignment in Python

- Python allows assigning multiple values to multiple variables in one line.
- This feature makes code more concise and readable.
- Example:

## Python Code

```
x, y, z = 1, 2, 3 # Assigning different values  
a = b = c = 100 # Assigning the same value to multiple variables
```

# Multiple Assignment in Python

- Python allows assigning multiple values to multiple variables in one line.
- This feature makes code more concise and readable.
- Example:

## Python Code

```
x, y, z = 1, 2, 3 # Assigning different values  
a = b = c = 100 # Assigning the same value to multiple variables
```

- Swapping values using multiple assignment:

# Multiple Assignment in Python

- Python allows assigning multiple values to multiple variables in one line.
- This feature makes code more concise and readable.
- Example:

## Python Code

```
x, y, z = 1, 2, 3 # Assigning different values  
a = b = c = 100 # Assigning the same value to multiple variables
```

- Swapping values using multiple assignment:

## Python Code

```
a, b = b, a # Swap values of a and b
```

# Dynamic Typing in Python

- Python is dynamically typed.



# Dynamic Typing in Python

- Python is dynamically typed.
- The type of a variable is determined at runtime.

# Dynamic Typing in Python

- Python is dynamically typed.
- The type of a variable is determined at runtime.
- You can reassign a variable to a different type.

# Dynamic Typing in Python

- Python is dynamically typed.
- The type of a variable is determined at runtime.
- You can reassign a variable to a different type.
- Example:

# Dynamic Typing in Python

- Python is dynamically typed.
- The type of a variable is determined at runtime.
- You can reassign a variable to a different type.
- Example:

## Python Code

```
x = 5 # Integer  
x = "Hello" # Now a string
```

# Best Practices for Variable Naming

- Use meaningful variable names.

# Best Practices for Variable Naming

- Use meaningful variable names.
- Follow the PEP 8 naming conventions.

# Best Practices for Variable Naming

- Use meaningful variable names.
- Follow the PEP 8 naming conventions.
- Avoid using Python keywords as variable names.

# Best Practices for Variable Naming

- Use meaningful variable names.
- Follow the PEP 8 naming conventions.
- Avoid using Python keywords as variable names.
- Example of valid names:



# Best Practices for Variable Naming

- Use meaningful variable names.
- Follow the PEP 8 naming conventions.
- Avoid using Python keywords as variable names.
- Example of valid names:

## Examples

```
student_name = "John"  
total_marks = 95
```

- Variables in Python are references, not storage containers.

# Summary

- Variables in Python are references, not storage containers.
- The assignment operator = creates a reference.

# Summary

- Variables in Python are references, not storage containers.
- The assignment operator = creates a reference.
- Python supports multiple assignments.

# Summary

- Variables in Python are references, not storage containers.
- The assignment operator = creates a reference.
- Python supports multiple assignments.
- Python supports dynamic typing.

# Summary

- Variables in Python are references, not storage containers.
- The assignment operator = creates a reference.
- Python supports multiple assignments.
- Python supports dynamic typing.
- Follow best practices for variable naming.

# Thank You!