9. Mutable and Immutable Variables

Immutable Variables

Immutable variables are variables whose values cannot be changed after they are created. In other words, once a value is assigned to an immutable variable, it cannot be modified or reassigned. Any attempt to change the value of an immutable variable will result in creating a new variable with the new value.

Here's an example in Python:

```
x = 5  # x is an immutable integer
print(x)  # Output: 5

x = 10  # Reassigning x creates a new variable with the value 10
print(x)  # Output: 10
```

In the above example, we first assign the value `5` to the variable `x`. However, when we try to change the value of `x` to `10`, Python creates a new variable `x` with the new value `10` instead of modifying the existing value.

Immutable types in Python include:

- Numbers (integers, floats, complex numbers)
- Strings
- Tuples
- Frozen sets

These data types are immutable because their internal state cannot be changed after they are created. Any operation that appears to modify an immutable object actually creates a new object with the desired changes.

Mutable Variables

Mutable variables are variables whose values can be changed after they are created. In other words, the content of a mutable variable can be modified directly without creating a new variable.

Here's an example in Python:

```python
my_list = [1, 2, 3]  # my_list is a mutable list
print(my_list)  # Output: [1, 2, 3]

my_list[0] = 4  # Modifying the first element of the list
print(my_list)  # Output: [4, 2, 3]
```

In the above example, we create a list `my_list` with the values `[1, 2, 3]`. We then modify the first element of the list by assigning the value `4` to `my_list[0]`. The output shows that the list has been successfully modified.

Mutable types in Python include:

- Lists
- Dictionaries
- Sets
- User-defined classes (objects)

These data types are mutable because their internal state can be changed after they are created. You can add, remove, or modify elements within mutable objects without creating a new object.

See Also

10. Random Module