In Python, data types can be classified as **mutable** or **immutable** based on whether their value can be changed after they are created.

### **Immutable Data Types:**

Immutable types cannot be changed after their creation. Any operation that modifies the object will result in a new object.

#### **1. String (str):**

Strings are immutable in Python. Once a string is created, it cannot be modified.

**Example:**

s = "Hello"

# Trying to change a character in the string will raise an error

# s[0] = "h" # Error: 'str' object does not support item assignment

# To modify, a new string must be created

s = "h" + s[1:]

print(s) # Output: "hello"

#### **2. Tuple (tuple):**

Tuples are also immutable. You cannot change their elements after creation.

**Example:**

t = (1, 2, 3)

# Trying to change an element in a tuple will raise an error

# t[0] = 10 # Error: 'tuple' object does not support item assignment

# To modify, you would need to create a new tuple

t = (10,) + t[1:]

print(t) # Output: (10, 2, 3)

### **Mutable Data Types:**

Mutable types can have their values changed after their creation.

#### **1. List (list):**

Lists are mutable, so you can change their elements or modify their structure.

**Example:**

l = [1, 2, 3]

l[0] = 10 # Change the first element

print(l) # Output: [10, 2, 3]

l.append(4) # Add an element to the list

print(l) # Output: [10, 2, 3, 4]

#### **2. Dictionary (dict):**

Dictionaries are mutable. You can change the value associated with a key or add new key-value pairs.

**Example:**

d = {"name": "Alice", "age": 25}

d["age"] = 26 # Change the value of the 'age' key

print(d) # Output: {"name": "Alice", "age": 26}

d["location"] = "New York" # Add a new key-value pair

print(d) # Output: {"name": "Alice", "age": 26, "location": "New York"}

#### **3. Set (set):**

Sets are mutable. You can add or remove elements from a set.

**Example:**

s = {1, 2, 3}

s.add(4) # Add an element to the set

print(s) # Output: {1, 2, 3, 4}

s.remove(2) # Remove an element from the set

print(s) # Output: {1, 3, 4}

### **Summary:**

* **Immutable**: str, tuple
* **Mutable**: list, dict, set

Mutable data types allow changes in place, whereas immutable data types require creating new objects for changes.