**Data Types in Python - Detailed Explanation with Examples**

In Python, a **data type** is a classification that tells the interpreter what kind of value a variable holds and what kind of operations can be performed on it.

Python is a dynamically typed language, so the type is determined automatically when a value is assigned.

**🔹 1. String (str)**

A **string** is a sequence of characters enclosed in either single quotes ' ' or double quotes " ". Strings are used to store text.

**✅ Example 1:**

name = "Alice"

**✅ Example 2:**

message = 'Hello, World!'

**🔹 2. Integer (int)**

An **integer** is a whole number, positive or negative, without decimals.

**✅ Example 1:**

age = 25

**✅ Example 2:**

year = -2024

**🔹 3. Float (float)**

A **float** is a number that contains a decimal point. Used for representing real numbers.

**✅ Example 1:**

price = 99.99

**✅ Example 2:**

height = -5.8

**🔹 4. Boolean (bool)**

A **boolean** data type represents one of two values: True or False. It's used in conditions and logic.

**✅ Example 1:**

is\_active = True

**✅ Example 2:**

has\_error = False

**🔹 5. List (list)**

A **list** is an ordered, mutable (changeable) collection of items. Lists are enclosed in square brackets [].

**✅ Example 1:**

fruits = ["apple", "banana", "cherry"]

**✅ Example 2:**

numbers = [1, 2, 3, 4, 5]

**🔹 6. Tuple (tuple)**

A **tuple** is like a list but immutable (unchangeable). Defined using parentheses ().

**✅ Example 1:**

coordinates = (10, 20)

**✅ Example 2:**

dimensions = (1920, 1080)

**🔹 7. Dictionary (dict)**

A **dictionary** is a collection of key-value pairs. Defined using curly braces {} and keys must be unique.

**✅ Example 1:**

student = {"name": "John", "age": 21}

**✅ Example 2:**

grades = {"Math": 90, "English": 85}

**🔹 8. Set (set)**

A **set** is an unordered collection of unique elements. Sets do not allow duplicate values.

**✅ Example 1:**

unique\_ids = {1, 2, 3, 4}

**✅ Example 2:**

vowels = {'a', 'e', 'i', 'o', 'u'}

**🔹 9. NoneType**

The **NoneType** represents the absence of a value or a null value. It's commonly used to initialize variables before assigning actual data.

**✅ Example 1:**

result = None

**✅ Example 2:**

data = None

**What is type() in Python?**

The type() function is a built-in Python function that returns the **type (class)** of the given object. It is used to determine the **data type** of variables and values in Python.

**🔹 Syntax:**

type(object)

* **object**: Any Python variable, value, or expression whose data type you want to identify.

**✅ Why Use type()?**

* To check what kind of data is stored in a variable.
* To debug and validate data dynamically.
* To perform operations conditionally based on data type.

**Examples of type() with Different Data Types**

# Example 1: String

name = "Python"

print("Type of name:", type(name)) # Output: <class 'str'>

# Example 2: Integer

number = 42

print("Type of number:", type(number)) # Output: <class 'int'>

# Example 3: Float

pi = 3.14

print("Type of pi:", type(pi)) # Output: <class 'float'>

# Example 4: Boolean

status = True

print("Type of status:", type(status)) # Output: <class 'bool'>

# Example 5: List

items = [1, 2, 3]

print("Type of items:", type(items)) # Output: <class 'list'>

# Example 6: Dictionary

data = {"name": "Alice", "age": 22}

print("Type of data:", type(data)) # Output: <class 'dict'>

# Example 7: NoneType

result = None

print("Type of result:", type(result)) # Output: <class 'NoneType'>