**Introduction on Python and Data Types**

**What is Python?**

* Python is a high-level, interpreted, and versatile programming language.
* It is widely used in web development, data analysis, artificial intelligence, and more.

**Features of Python:**

* **Easy to Learn:** Beginner-friendly
* **Open Source:** Free to use and modify.
* **Interpreted:** Executes code line by line, making debugging easier.
* **Cross-Platform:** Runs on Windows, macOS, Linux, etc.
* **Large Libraries:** Includes modules for almost any task.

**Python Data Types**

Python data types are divided into several categories based on their functionality. Here’s a quick overview:

**1. Numeric Types**

* **int:** Represents integers (e.g., 10, -5).
* **float:** Represents decimal numbers (e.g., 3.14, -0.5).
* **complex:** Represents complex numbers (e.g., 3+4j).

**2. Sequence Types**

* **str:** Represents text (e.g., "Hello", 'Python').
* **list:** Represents an ordered collection of items (e.g., [1, 2, 3]).
* **tuple:** Represents an immutable ordered collection (e.g., (1, 2, 3)).

**3. Set Types**

* **set:** Represents an unordered collection of unique items (e.g., {1, 2, 3}).
* **frozen set:** Immutable version of a set.

**4. Mapping Type**

* **dict:** Represents a collection of key-value pairs (e.g., {'name': mani, 'age': 23}).

**5. Boolean Type**

* **bool:** Represents True or False values.

**6. None Type**

* **NoneType:** Represents the absence of a value (e.g., None).

**Common Operations on Data Types**

1. **Type Checking:**
   * Use type() to check the type of a variable.  
     Example:

**Ex:** x = 10

print(type(x)) #output = class < int >

**Type Conversion:**

* Convert one data type to another.  
  Example:

**Ex**: x = 10.5

y = int(x) # Converts float to int

print(y)