chapter 1 :- Basics

Python is a **high-level, general-purpose programming language** that is widely used for web development, data science, automation, artificial intelligence, and more. It is known for its **simplicity** and **readability** due to its clean syntax and indentation-based structure.

Key Characteristics of Python

1. **Interpreted Language**
   - Python is an **interpreted** language, meaning that code is executed line by line by the Python interpreter rather than being compiled into machine code beforehand.
   - This allows for rapid testing and debugging but may result in slower execution compared to compiled languages like C or C++.
2. **Dynamically Typed**
   - Python is **dynamically typed**, meaning that you don't need to declare variable types explicitly.
   - Example:

     ```
     x = 10                  # x is an integer
     x = "Hello"               # x is now a string
     ```

   - The type of `x` changes at runtime based on the assigned value.
3. **Strongly Typed**
   - Python is **strongly typed**, which means you cannot implicitly convert one type to another without explicit conversion.
   - Example:

     ```
     x = "10"
     y = 5
     print(x + y)  # TypeError: can only concatenate str (not "int") to str
     ```

   - To fix this, you need to explicitly convert `x`:

     ```
     print(int(x) + y)  # Output: 15
     ```

4. **Garbage Collection & Memory Management**
   - Python has **automatic memory management** and **garbage collection** via reference counting and cyclic garbage collection.
5. **Multi-Paradigm**
   - Python supports multiple programming paradigms:
     - **Procedural Programming** (writing functions and procedures)
     - **Object-Oriented Programming (OOP)** (using classes and objects)
     - **Functional Programming** (using higher-order functions, map, reduce, etc.)
6. **Platform Independent**
   - Python is **cross-platform**, meaning the same code can run on Windows, macOS, and Linux without modification.

7. **Extensible & Embeddable**
    - Python can be extended with C/C++ and embedded into other applications.
8. **Interpreters & Compilers**
    - Python has multiple implementations:
        - **CPython** (the standard implementation written in C)
        - **PyPy** (a faster JIT-compiled version)
        - **Jython** (Python running on the Java Virtual Machine)
        - **IronPython** (Python running on .NET)
9. **Concurrency & Multi-threading**
    - Python has **multi-threading** but is limited by the **Global Interpreter Lock (GIL)** in CPython.
    - For true parallel execution, **multiprocessing** or external tools (like Dask or Ray) are often used.
10. **Standard Library & Third-Party Ecosystem**
    - Python has a **rich standard library** with modules for file handling, networking, database access, etc.
    - A vast **third-party ecosystem** (via PyPI) provides tools for web development, data science, machine learning, etc.