# **PYTHON BASICS**

DAY-1

1. **WHAT IS PYTHON -**

* Scripting language
* High Level Language
* Interpreter Language
* Interactive Language
* Procedure Oriented Language
* Functional Oriented Language
* Object Oriented Language
* Purely Object Oriented Language
* Modular Programming Language
* General Purpose Language

1. **Scripting Language –**

All interpreter-based languages are called scripting languages . Since Python is an interpreter-based language, it is called a scripting language . Scripting languages are open-source and can be modified at any time . An interpreter does not generate an executable file; it always executes the source code directly.

1. **High Level Language –**

Python is a user-friendly language, similar to English, making it easy to read and write. It abstracts complex hardware-level details, making development simpler and faster.

1. **Interpreter based Language –**

Python code is processed at runtime by the interpreter — there is no need to compile the program before execution. This behavior is similar to scripting languages like Linux Shell Scripting, Perl, and PHP.

1. **Interactive Language –**

Python allows you to write and test code directly in the command line or shell, just like you do in a Linux terminal. This makes Python ideal for quick testing, learning, and debugging .

1. **Procedure-Oriented Language –**

A procedure-oriented language focuses on functions (also called procedures) to perform tasks. The program is divided into smaller parts called functions, making it easier to manage and reuse code.

1. **Functional-Oriented Language –**

A functional-oriented language focuses on functions as the core building blocks of a program.

It emphasizes:

* Pure functions (no side effects)
* Immutability (no changing state)
* Function composition (building complex logic by combining simple functions)

1. **Object-Oriented Language –**

An object-oriented language (OOP) focuses on objects and classes to organize code.

It supports key OOP concepts:

* Encapsulation – hiding internal details
* Abstraction – showing only essential features
* Inheritance – reusing code from existing classes
* Polymorphism – using a single interface for different data types

1. **Purely Object-Oriented Language –**

In a purely object-oriented language, everything is treated as an object, including basic data types like numbers and functions. You cannot write procedural code separately — everything must be inside a class or object.

1. **Modular Programming Language –**

A modular programming language allows you to break a large program into smaller, independent modules. Each module performs a specific task and can be developed, tested, and reused separately.

This approach improves:

* Code organization
* Reusability
* Debugging and maintenance .

1. **General Purpose Language –**

A general-purpose language is designed to build a wide variety of applications, not limited to a specific domain.

It can be used for:

* Web development
* Data analysis
* Automation
* Game development
* Machine learning
* Scientific computing

1. **Applications developed In Python –**
2. **Web Development -**

* Python is widely used in web development thanks to powerful frameworks like **Django** and **Flask**.
* These frameworks offer built-in tools for:
  + Routing
  + Templating
  + Database management
* **Popular companies** using Python for web development:  
  **Instagram, Pinterest, Spotify**

1. **Scientific & Numeric Computing -**

* Python is a top choice in **scientific computing** and **mathematics**.
* Key libraries:
  + NumPy – numerical computing
  + SciPy – scientific calculations
  + Pandas – data analysis and manipulation
* Used for:
  + Statistical analysis
  + Complex computations
  + Data visualization

1. **Data Science, Machine Learning & AI**

* Python dominates the world of **Data Science**, **ML**, and **AI**.
* Top libraries include:
  + TensorFlow, Keras, PyTorch – for deep learning and model training
  + Scikit-learn – for machine learning algorithms
* Applications:
  + Image recognition
  + Predictive modeling
  + Natural language processing (NLP)

1. **Business Applications -**

* Python is great for building **business tools**, such as:
  + Financial applications
  + Inventory management systems
  + Reporting & dashboards
* Also used to **automate repetitive tasks** and handle **large-scale data processing**.

1. **Game Development -**

* Python can be used for **basic 2D games** using libraries like:
  + Pygame
* Ideal for **learning and prototyping**, though less common in high-end game development.

1. **Scripting & Automation -**

* Python is a powerful **scripting language** for:
  + Automating repetitive tasks
  + File system operations
  + System administration
* Helps boost **productivity and efficiency**.

1. **Desktop Applications -**

* Python supports **GUI-based application development**.
* Popular frameworks:
  + Tkinter (built-in GUI toolkit)
  + PyQt, Kivy – for more complex apps
* Can be used to build calculators, media players, editors, etc.

1. **Other Applications**

* Python is also used in:
  + **Blockchain** development
  + **Audio/Video** processing
  + **Web scraping** (using BeautifulSoup, Selenium)
  + **Chatbots** and **virtual assistants**
  + **Embedded systems** and **IoT**