Python Programming Basis

Presented By:

Suman Kumar Gochhayat

What is Python Full-Stack Development?

Full-stack development involves building both:

- **Front-end: User interface (UI/UX)**
- **Back-end: Server logic, databases, server functions**

Python frameworks like Django or Flask streamline the entire process.

Why Python Full-Stack?

- **Simplifies development with modular frameworks**
- **Faster application deployment**
- **Enhances collaboration between teams**

History of Python Programming

1

1980s: Foundation

The conceptual groundwork for Python was laid.

2

1989: Implementation Begins

Guido van Rossum started working on Python at CWI in the Netherlands.

3

Feb 20, 1991: Official Release

Python 0.9.0 was officially released to the public.

4

Today: PSF Management

Managed by the non-commercial Python Software Foundation (PSF).

www.python.org

Created by Guido van Rossum, the "Father of Python."

Functions in Python

Functions are essential for performing specific operations and promoting code re-usability.

Definition

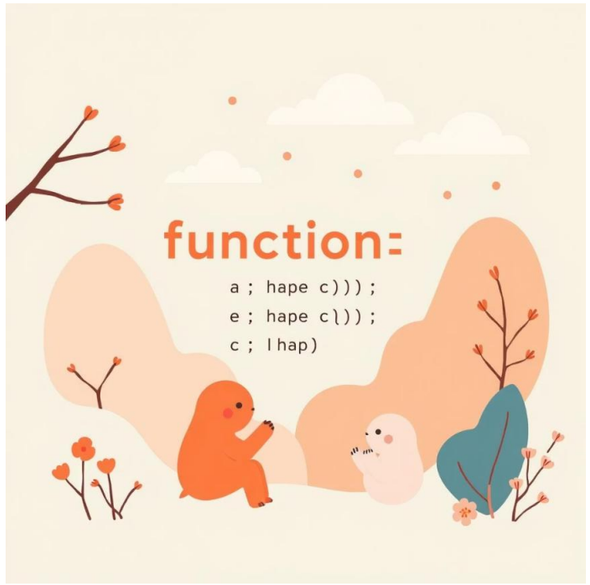
A sub-program or a part of the main program.

Advantages

- Reduces development time
- Minimizes memory usage
- Speeds up execution
- Enhances application performance
- Reduces code redundancy

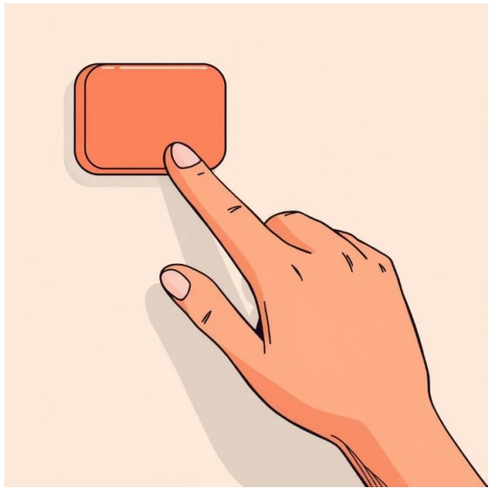
Parts of a Function

Every function requires two key parts for proper execution:



Function Definition

- Exists only once.
- Contains the code to be executed.
- PVM executes only when called.

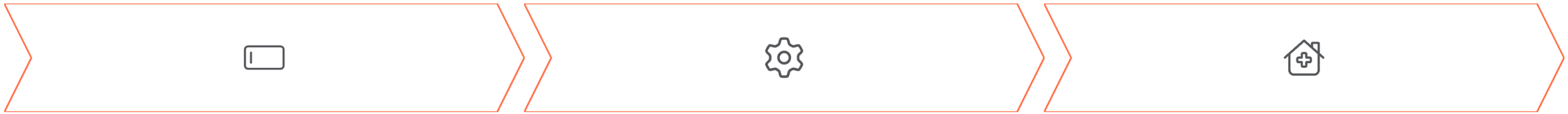


Function Calls

- Triggers the function definition.
- Must correspond to an existing definition (prevents NameError).

Phases of a Function

When defining functions, consider these three essential phases:



Input

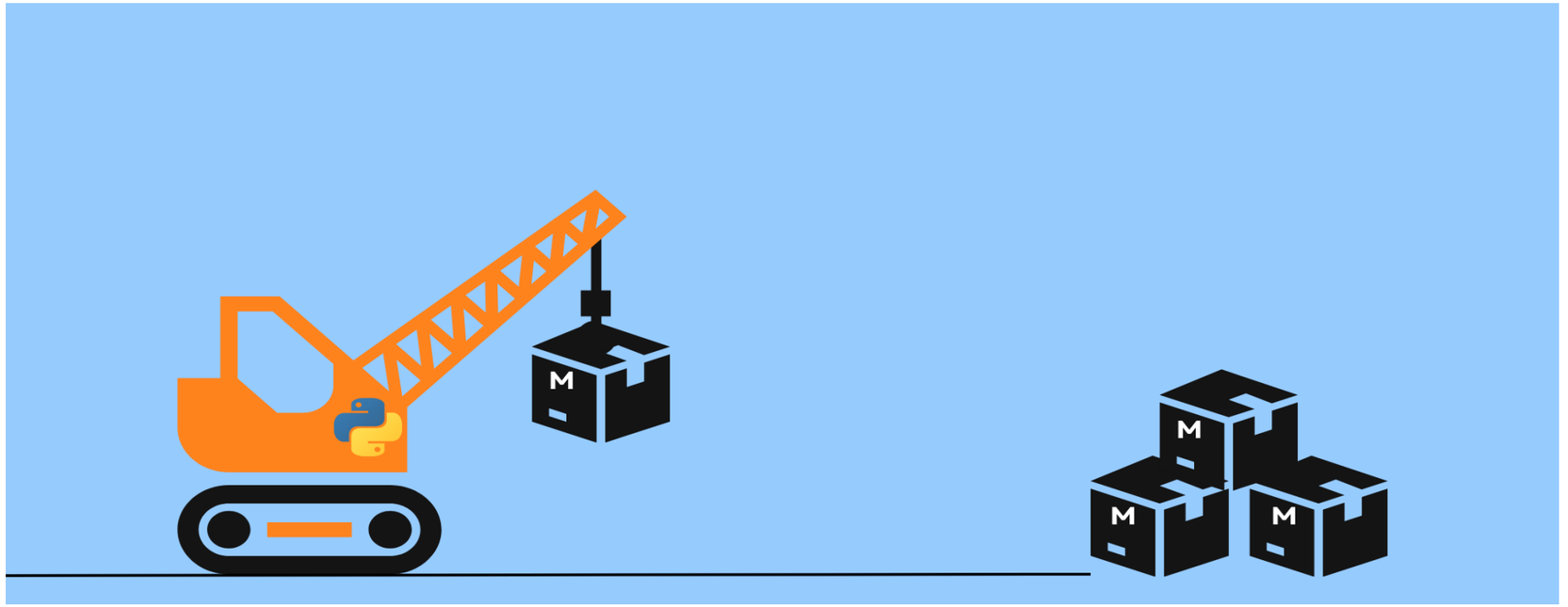
Every function must take an input to operate on.

Process

The function processes the given input according to its logic.

Output

The function provides an output or result based on its processing.



Modules in Python

While functions provide code re-usability within a single program, modules extend this capability across multiple programs in the same folder.

> ## Definition of a Module:
>
> A module is a collection of global variable names, functions, and class names.

Types of Modules

Python supports two primary types of modules to organize and reuse code:

1

Pre-Defined Modules

Built-in modules provided by Python developers.

2

Programmer-Defined Modules

Custom modules created by users.

Pre-Defined Modules

These modules are readily available within the Python software. They address universal requirements and are used by all Python programmers.

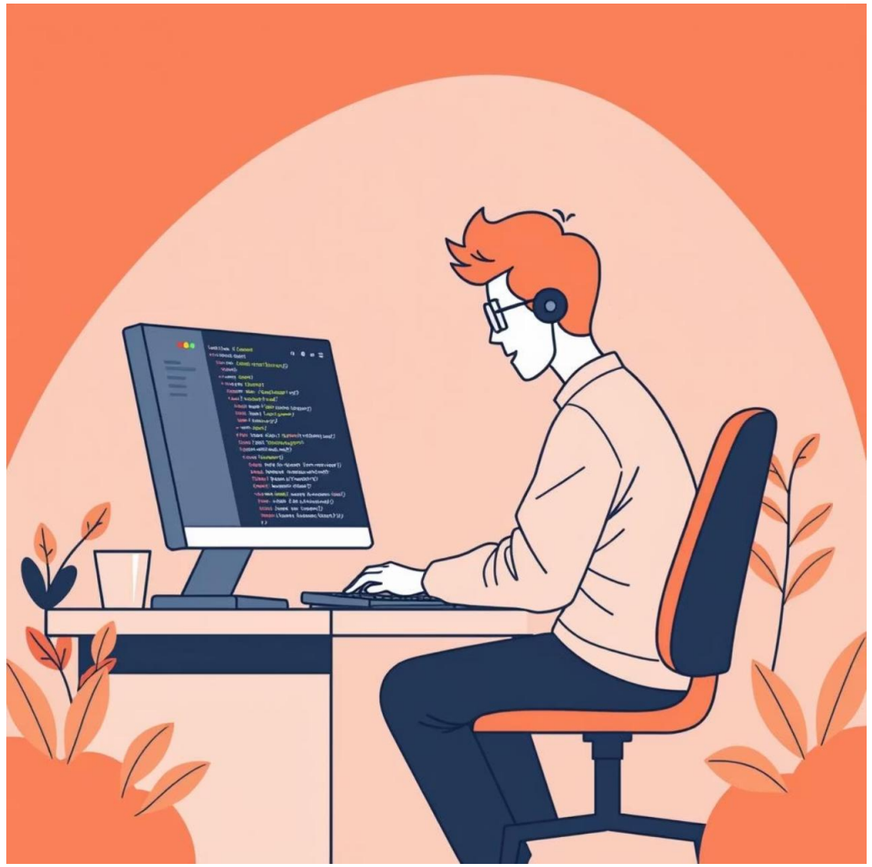
Examples include:

The "`builtins`" module is automatically imported into every Python program. It's the default imported module!

Programmer-Defined Modules

- Created by developers for specific project needs.
- Allow for custom functions, classes, and variables.
- Promote code organization and reusability within a project.
- Can be shared and imported into other custom programs.

These modules are essential for building complex, scalable applications.




Thank You!