**1. Module**

**1.1 Definition**

* A module is typically the smallest unit of code organization in a program. It often corresponds to a single file containing functions, classes, or variables that serve a particular purpose.
* In Python, any file with the .py extension (e.g., utils.py) can be considered a module.

1.2 Purpose

* Breaks down large codebases into smaller, more manageable components.
* Prevents naming conflicts for variables, functions, or classes.
* Encourages code reuse across different parts of the application or in other projects.

1.3 Example

def add(a, b):

return a + b

**2. Package**

**2.1 Definition**

* A package is a collection of related modules grouped under a common namespace or directory structure. This helps organize code logically and prevent naming collisions.
* In Python, a folder containing multiple .py files and an \_\_init\_\_.py file is recognized as a package.

**2.2 Purpose**

* Organizes modules that share similar functionality into a structured hierarchy.
* Simplifies project navigation and maintenance.
* Avoids conflicts by providing a distinct namespace.

**2.3 Example**

my\_app/

│── \_\_init\_\_.py

│── models.py

│── controllers.py

│── views.py

Usage: from my\_app import models, controllers

**3. Library**

**3.1 Definition**

* A library is a collection of modules, packages, or functionalities bundled together for reuse across multiple projects. Libraries are often shared in binary form or as source code, making it easy for developers to incorporate them into their applications.
* Python libraries are often installed via pip (e.g., requests, numpy).

**3.2 Purpose**

* Offers ready-made solutions to common problems (e.g., networking, database access, mathematical computations).
* Saves development time by providing well-tested and reusable components.
* Encourages consistent and maintainable code across different projects.

**3.3 Example**

pip install requests

Usage:

import requests

response = requests.get("https://example.com")

print(response.status\_code)