In Python, external modules (also called third-party libraries) are not included in the standard library and need to be installed separately. These modules provide additional functionality, such as web scraping, machine learning, data analysis, or database interaction.

**Installing External Modules**

Python uses the package manager **pip** (Python Package Installer) to install external modules. Below are the steps to install and manage external modules.

**1. Installing a Module Using pip**

To install an external module, open your terminal (Command Prompt, PowerShell, or macOS/Linux Terminal) and use:

pip install module\_name

For example, to install the **requests** module (used for making HTTP requests):

pip install requests

**2. Installing a Specific Version**

If you need a specific version of a module, specify it using ==:

pip install module\_name==version\_number

Example:

pip install numpy==1.21.0

**3. Upgrading an Installed Module**

To upgrade an installed module to the latest version:

pip install --upgrade module\_name

Example:

pip install --upgrade pandas

**4. Installing Multiple Modules at Once**

If you have a list of required modules, you can put them in a file (e.g., requirements.txt) and install them all using:

pip install -r requirements.txt

**5. Checking Installed Modules**

To see a list of installed modules and their versions:

pip list

To check details about a specific module:

pip show module\_name

Example:

pip show flask

**6. Uninstalling a Module**

To remove an installed module:

pip uninstall module\_name

Example:

pip uninstall matplotlib

**7. Using Virtual Environments**

It is recommended to use virtual environments to manage dependencies separately for different projects. To create and activate a virtual environment:

* **Windows:**
* python -m venv myenv
* myenv\Scripts\activate
* **Mac/Linux:**
* python3 -m venv myenv
* source myenv/bin/activate

To deactivate the virtual environment:

deactivate

**Conclusion**

Installing external modules in Python is simple using pip, and using virtual environments helps manage dependencies efficiently. If you encounter issues, ensure pip is updated:

pip install --upgrade pip