1. What is the difference between module and package?
2. How do you create and use a module in python?
3. What is the significance of the. \_\_init\_\_.py file in python?

|  |  |
| --- | --- |
| MODULE | PACKAGE |
| A single Python file .py containing functions, classes, and variables. | A directory containing multiple modules and a special \_\_init\_\_.py file |
| Organizes code into reusable files | Organizes multiple related modules |
| A standalone file | A folder with multiple .py files and an \_\_init\_\_.py file |
| import module | import package.mymodule |

**CREATING MODULE :**

Step 1: Create a Module

Create a file module.py with the following content:

def greet(name):

return f"Hello, {name}!"

pi = 3.14159

**USING MODULE:**

Step 2: Use the Module in Another Python File

You can import and use module in another script:

import module

print(module.greet("Bharat"))

print(module.pi)

**Significance of \_\_init\_\_.py File in Python**

The \_\_init\_\_.py file is used to define a package in Python. It signals to Python that a directory should be treated as a package.

Defines a Package – Without \_\_init\_\_.py, Python treats the directory as a normal folder, not a package.

Controls Package Imports –> It can include initialization code or import specific modules.

Improves Code Organization –> Allows grouping related modules together.