

Course
&
Test Series

Creating and Using a Module in Python

Package Creation and Importing in Python

Package creation means organizing related Python modules (files) into a directory (folder) that Python can recognize and import from. It's basically a way to group related code together so you can reuse, maintain, and import it easily.

Think of it like this:

Package → contains → Modules → contains → Functions, classes, variables

Why to create a Package?

- Organize your code better (modular programming)
- Reuse code across multiple projects
- Avoid naming conflicts
- Make your code shareable or installable via pip

Why we use Packages in Python?

- Organize code – group related modules (files) into one logical folder
- Reuse code – write functions once and import them anywhere
- Avoid name conflicts – separate namespaces prevent clashes between modules with the same name
- Improve maintainability – makes large projects easier to manage and update
- Enhance scalability – easy to add new modules or features without breaking existing code
- Simplify testing – test each module or part of the package independently
- Improve readability – keeps code structured, clear, and easier to understand

Structure of a Package

Let's create a simple package called **mathpackage**.

```
mathpackage/  
  __init__.py  
  add_module.py  
  subtract_module.py
```

Here:

- **__init__.py** → tells Python that this folder is a **package**. (It can be an empty file or contain initialization code.)
- **add_module.py** → contains addition function.
- **subtract_module.py** → contains subtraction function.

Creating Package Files

```
mathpackage/add_module.py
```

```
def add(a, b):  
    return a + b
```

CBSE

ICSE

NTSE

Banking & Insurance

Central Govt. Service

State Govt. Services

LAW Entrance

MBA Entrance

Railways & Metro Services

...many more

abhyasonline.in

Course
&
Test Series

Creating and Using a Module in Python

 CBSE

 ICSE

 NTSE

 Banking & Insurance

 Central Govt. Service

 State Govt. Services


 LAW Entrance

 MBA Entrance

 Railways & Metro Services

...many more

abhyasonline.in

 mathpackage/subtract_module.py

```
def subtract(a, b):
    return a - b
```

 mathpackage/__init__.py

```
# This can be empty or can import key functions for easy access
from .add_module import add
from .subtract_module import subtract
```

Using the Package

Now create a new Python file main.py in the parent folder:

```
# Importing the package and its modules
import mathpackage.add_module
import mathpackage.subtract_module
```

```
# Using functions
```

```
print("Addition:", mathpackage.add_module.add(10, 5))
print("Subtraction:", mathpackage.subtract_module.subtract(10, 5))
```

Output:

```
Addition: 15
Subtraction: 5
```

Shortcut Import Using __init__.py

If you used __init__.py like this:

```
from .add_module import add
from .subtract_module import subtract
```

Then you can import functions **directly** like this 🙌

```
from mathpackage import add, subtract

print("Addition:", add(12, 8))
print("Subtraction:", subtract(12, 8))
```

Output:

```
Addition: 20
Subtraction: 4
```

Built-in Python Packages

Python comes with many ready-made packages like:

- os - For operating system operations
- math - For mathematical operations
- datetime - For date and time
- json - For handling JSON data

**Course
&
Test Series**

Creating and Using a Module in Python

 **CBSE**

 **ICSE**

 **NTSE**

 **Banking & Insurance**

 **Central Govt. Service**

 **State Govt. Services**

 **LAW Entrance**

 **MBA Entrance**

 **Railways & Metro Services**

...many more

abhyasonline.in

Example:

```
import os
print(os.getcwd()) # Prints current working directory
```

In Summary:

Concept	Description
Module	A single .py file containing code
Package	A folder containing multiple modules
__init__.py	Tells Python the folder is a package
Importing	from packagename import modulename
Use Case	For organizing big projects and reusing code

Solved Example – Creating and Using a Package

Ques: Create a package called **greetpackage** that contains a module to greet users.

Folder Structure:

```
greetpackage/
  __init__.py
  greeter.py
  main.py
```

greetpackage/greeter.py

```
def say_hello(name):
    return f"Hello, {name}! Welcome to Python Packages."
```

greetpackage/init.py

```
# This makes the package easier to use by importing the function directly
from .greeter import say_hello
```

main.py

```
# Importing from our package
from greetpackage import say_hello

# Using the function
message = say_hello("Megha")
print(message)
```

Output:

Hello, Megha! Welcome to Python Packages.

Course
&
Test Series

Creating and Using a Module in Python

Step-by-Step: Creating a Package (greetpackage)

Step 1: Create a Project Folder

You can do this anywhere on your computer (for example, on Desktop or in Documents).

Let's call it:
MyPythonProject

Step 2: Inside it, Create a Folder for the Package

Inside MyPythonProject, make a new folder named:
greetpackage

- Folder structure now looks like this:

MyPythonProject/
greetpackage/

Step 3: Add an Empty `__init__.py` File

Inside greetpackage, create a new file named:
`__init__.py`

- This file tells Python that **this folder is a package.**
- It can even be empty, but if you want to make importing easier, you can add code inside it later.

Step 4: Add a Module File

Inside greetpackage, create another Python file – this will be your **module**.

Name it:

`greeter.py`

Now your folder looks like this:

MyPythonProject/
greetpackage/
`__init__.py`
`greeter.py`

Step 5: Write Some Code

In `greeter.py`:

```
def say_hello(name):
```

```
    return f"Hello, {name}! Welcome to Python Packages."
```

In `__init__.py`:

```
from .greeter import say_hello
```

Step 6: Create a Main Program

Now, outside the package folder (still inside MyPythonProject), create another file:

`main.py`

And write:

 CBSE

 ICSE

 NTSE

 Banking & Insurance

 Central Govt. Service

 State Govt. Services

 LAW Entrance

 MBA Entrance

 Railways & Metro Services

...many more

abhyasonline.in

Course
&
Test Series

Creating and Using a Module in Python

```
from greetpackage import say_hello
```

```
message = say_hello("Megha")  
print(message)
```

Step 7: Run the Program

Output:

Hello, Megha! Welcome to Python Packages

Assignment

Ques: Create a package named **mathoperations** with a module **square.py** that defines a function to calculate the square of a number. Then write a main program to import and display the square of 9.

Hint: return num ** 2

 CBSE

 ICSE

 NTSE

 Banking & Insurance

 Central Govt. Service

 State Govt. Services

 LAW Entrance

 MBA Entrance

 Railways & Metro Services

...many more

abhyasonline.in