



## Modules:

In Python, a module is a file containing Python definitions and statements. The file name is the module name with the suffix `.py` added. Modules allow you to organize Python code into reusable units. These units can consist of functions, classes, and variables that can be used in other Python files by importing the module.

For example, if you have a file named `my_module.py` with the following content:

```
python Copy code

# my_module.py

def greet(name):
    print(f"Hello, {name}!")

def square(x):
    return x ** 2
```

You can use this module in another Python script by importing it:

```
python Copy code

# another_script.py
import my_module

my_module.greet("Alice")
result = my_module.square(5)
print(result)
```

In this example, the `another_script.py` file imports the `my_module` module and uses its `greet` and `square` functions. This helps in organizing code, promoting code reusability, and avoiding naming conflicts.

You can also use the `from ... import ...` syntax to import specific functions or variables from a module:

python

Copy code

```
# another_script.py
from my_module import greet, square

greet("Bob")
result = square(3)
print(result)
```

This way, you directly import only the elements you need from the module. Modules are a fundamental concept in Python programming, providing a way to structure and organize code in larger projects.

Pip stands for:- python package manager

Modules are of 2 types:

- 1) Build-in Module
- 2) External Module

C:\> pip install panda

```
Terminal Local x + v
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

(.venv) PS C:\Users\test\PycharmProjects\project_1> pip install panda
Collecting panda
  Downloading panda-0.3.1.tar.gz (5.8 kB)
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Preparing metadata (pyproject.toml) ... done
Collecting setuptools (from panda)
  Obtaining dependency information for setuptools from https://files.pythonhosted.org/packages/c0/7a/3da854f49c95d0cc0e9549a835b5818e0ae917e852ecc08e77550c8dc08b/setuptools-09.1.1-py3-none-any.whl.metadata
  Using cached setuptools-09.1.1-py3-none-any.whl.metadata (6.2 kB)
Collecting requests (from panda)
  Obtaining dependency information for requests from https://files.pythonhosted.org/packages/70/8e/0e2d847013cb52cd35b38c009bb167a1a26b2ce6cd965bf26b47bc0bf44/requests-2.31.0-py3-none-any.whl.metadata
  Downloading requests-2.31.0-py3-none-any.whl.metadata (4.6 kB)
Collecting charset-normalizer<4,>=2 (from requests>panda)
  Obtaining dependency information for charset-normalizer<4,>=2 from https://files.pythonhosted.org/packages/b0/7c/8debebb4f90174074b827c63242c23851bdf00a532489fba57fef3410e40/charset-normalizer-3.3.2-cp312-cp312-win_amd64.whl.metadata
  Downloading charset-normalizer-3.3.2-cp312-win_amd64.whl.metadata (34 kB)
Collecting idna<4,>=2.5 (from requests>panda)
  Obtaining dependency information for idna<4,>=2.5 from https://files.pythonhosted.org/packages/c2/e7/a82b05cfe3a03dfe0a8d5a6a8bf50a448a0718ea503360af4b54a9/idna-3.0-py3-none-any.whl.metadata
  Downloading idna-3.0-py3-none-any.whl.metadata (9.9 kB)
Collecting urllib3<3,>=1.21.1 (from requests>panda)
  Obtaining dependency information for urllib3<3,>=1.21.1 from https://files.pythonhosted.org/packages/a2/73/a08704750a7679d0bd3ed7a8d4a8e14e151ae82e6fee774e6e0d05ec9/urllib3-2.2.1-py3-none-any.whl.metadata
  Downloading urllib3-2.2.1-py3-none-any.whl.metadata (6.4 kB)
Collecting certifi>=2017.4.17 (from requests>panda)
  Obtaining dependency information for certifi>=2017.4.17 from https://files.pythonhosted.org/packages/ba/06/a07f09ac6a6aeb9f01a24f858c3add0a4e9136ac96257acbf4fce1e7de14/certifi-2024.2.2-py3-none-any.whl.metadata
  Downloading certifi-2024.2.2-py3-none-any.whl (2.2 kB)
  Downloading requests-2.31.0-py3-none-any.whl (62 kB)
    02.0/02.6 kB 3.3 MB/s eta 0:00:00
Using cached setuptools-09.1.1-py3-none-any.whl (819 kB)
Downloaded certifi-2024.2.2-py3-none-any.whl (163 kB)
```



```
Downloading certifi-2024.2.2-py3-none-any.whl (163 kB)
 163.8/163.8 kB 5.0 MB/s eta 0:00:00
Downloading charset_normalizer-3.3.2-cp312-cp312-win_amd64.whl (100 kB)
 100.4/100.4 kB ? eta 0:00:00
Downloading idna-3.6-py3-none-any.whl (61 kB)
 61.6/61.6 kB 3.4 MB/s eta 0:00:00
Downloading urllib3-2.2.1-py3-none-any.whl (121 kB)
 121.1/121.1 kB ? eta 0:00:00
Building wheels for collected packages: panda
  Building wheel for panda (pyproject.toml) ... done
  Created wheel for panda: filename=panda-0.3.1-py3-none-any.whl size=7245 sha256=2a0dfbc7a6b0e6c60ef2481f2ea2d477c04a100a3534156a4f0f9438550e0c93
  Stored in directory: c:\users\test\appdata\local\pip\cache\wheels\98\41\5b\6ca54e0b6a35e1b7248c12f56fcb753dfb7717fefaa0fb45f5
Successfully built panda
Installing collected packages: urllib3, setuptools, idna, charset-normalizer, certifi, requests, panda
Successfully installed certifi-2024.2.2 charset-normalizer-3.3.2 idna-3.6 panda-0.3.1 requests-2.31.0 setuptools-69.1.1 urllib3-2.2.1

[notice] A new release of pip is available: 23.2.1 -> 24.0
[notice] To update, run: python.exe -m pip install --upgrade pip
```

Simple Program to begin the python programming.

**Code:**

A screenshot of a code editor window titled 'Project\_1.py'. The code contains a single line: `print("Welcome in Lazy Programming Series")`.

```
Project_1.py x
1 print("Welcome in Lazy Programming Series")
```

**Output:**

A screenshot of a terminal window titled 'Run Project\_1'. It shows the command to run the script and the output. The command is: `C:\Users\test\PycharmProjects\project_1\.venv\Scripts\python.exe C:\Users\test\PycharmProjects\project_1\Project_1.py`. The output is: `Welcome in Lazy Programming Series`. The terminal also shows 'Process finished with exit code 0'.

```
Run Project_1 x
C:\Users\test\PycharmProjects\project_1\.venv\Scripts\python.exe C:\Users\test\PycharmProjects\project_1\Project_1.py
Welcome in Lazy Programming Series
Process finished with exit code 0
```