**MYMODULE.PY**

def greeting(name):

  print("Hello, " + name)

person1 = {

  "name": "Akasay",

  "age": 18,

  "country": "india"

}

def check\_number():

    num = int(input("Enter a number: "))

    if num > 0:

        print("The number is positive.")

    elif num < 0:

        print("The number is negative.")

    else:

        print("The number is zero.")

**PREWRITTENCODE.PY**

**## A module is a file that has code you can use in your program.**

import mymodule

**## Use a Module**

**## Now we can use the module we just created, by using the import statement:**

mymodule.greeting("Atharva")

a = mymodule.person1["name"]

print(a)

mymodule.check\_number()

**## Naming a Module**

**## You can name the module file whatever you like, but it must have the file extension .py**

**## Re-naming a Module**

**## You can give a module a short name when you import it by using the as keyword.**

import mymodule as mx

a = mx.person1["age"]

print(a)

**## Built-in Modules**

**## There are several built-in modules in Python, which you can import whenever you like**

**# platform.system() -   Returns the name of the OS (e.g., 'Windows', 'Linux')**

**# platform.release()    - Returns the OS release version (e.g., '10', '5.15.0')**

**# platform.version()    - Returns the OS version details**

**# platform.platform()   - Returns a single string with full OS info**

**# platform.machine()    - Returns the machine type (e.g., 'x86\_64', 'AMD64')**

**# platform.processor()  - Returns the processor name (e.g., 'Intel64 Family 6 Model')**

**# platform.architecture()   - Returns Python's bit architecture (e.g., ('64bit', ''))**

**# platform.node()   - Returns the computer's network name (hostname)**

**# platform.python\_version() - Returns the Python version (e.g., '3.11.2')**

**# platform.python\_build()   - Returns Python build number and date**

**# platform.python\_compiler()    - Returns the name of the compiler used to build Python**

**# platform.uname()  - Returns a tuple with full system info (system, node, release, version, machine, processor)**

import platform

x = platform.platform()

print(x)

**# Using the dir() Function**

**# There is a built-in function to list all the function names (or variable names) in a module. The dir() function:**

import mymodule

x = dir(mymodule)

print(x)

**## Import From Module**

**## You can choose to import only parts from a module, by using the from keyword.**

from mymodule import person1

print (person1["age"])