**Abhiram Basa**

**Python Assessment - 6 - 18/12/23**

Topics covered:

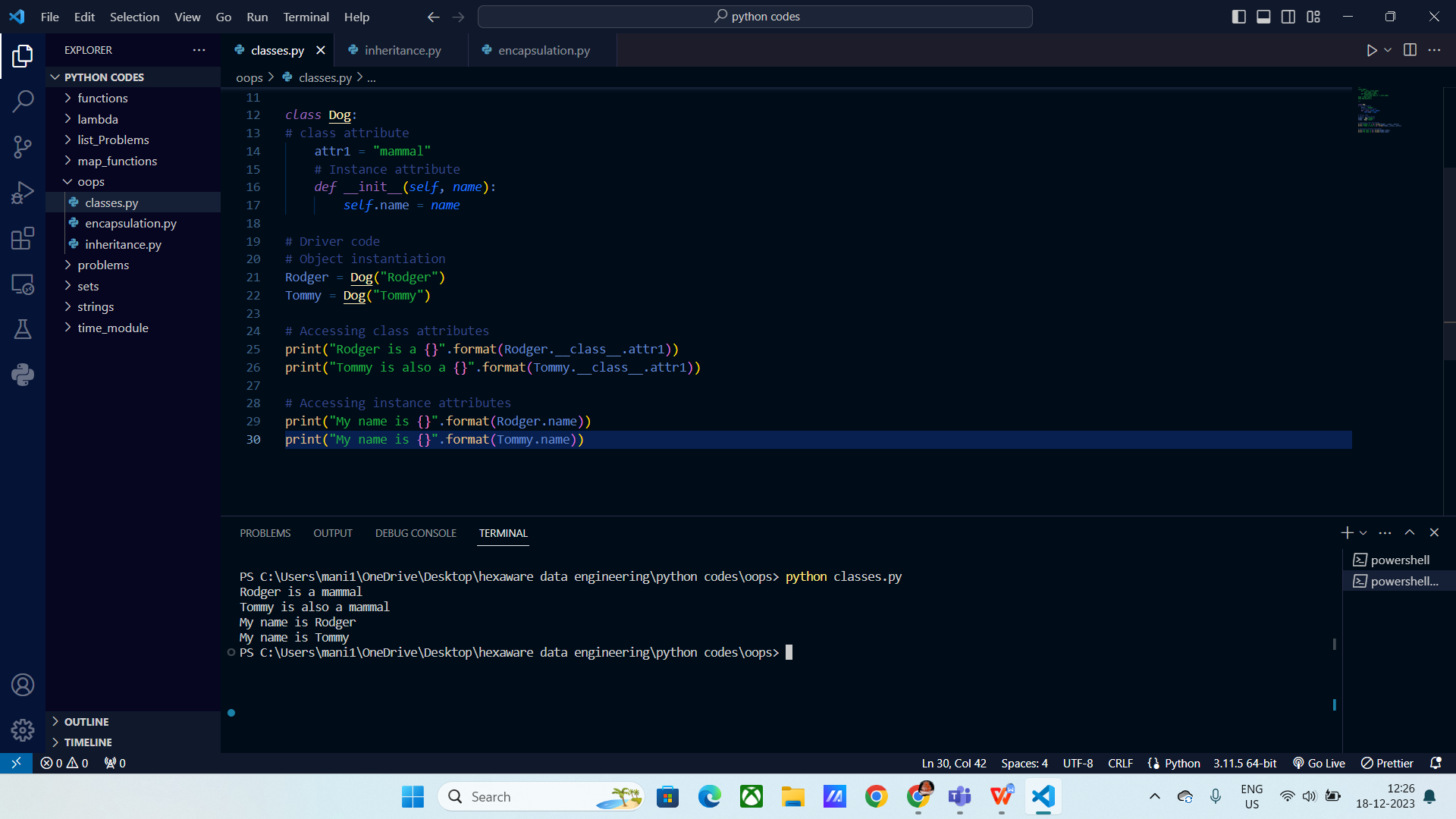
* Oops.
* File handling.
* Exceptions and Exceptions Handling.
* Python Modules.

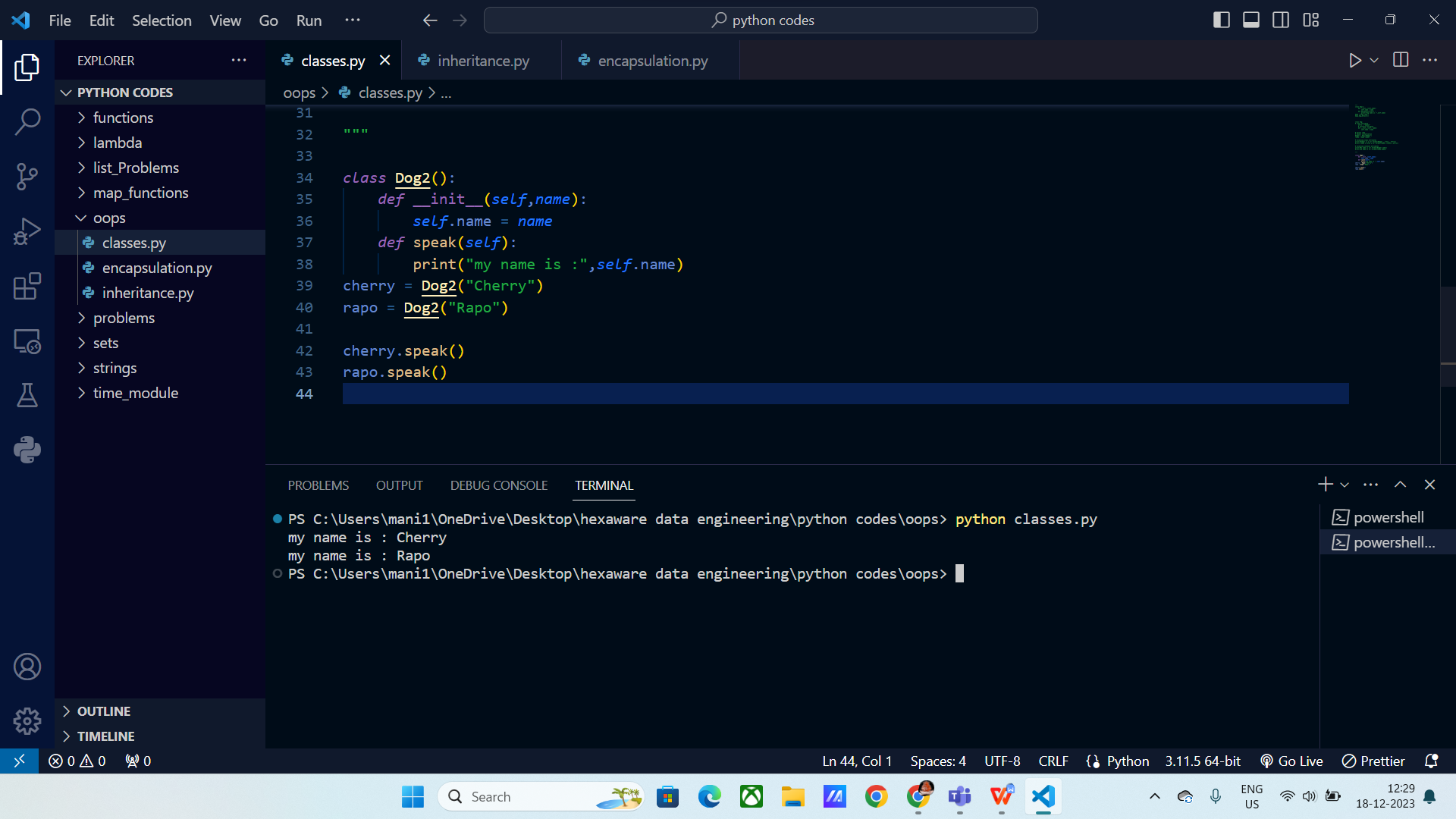
**OOP’S:**

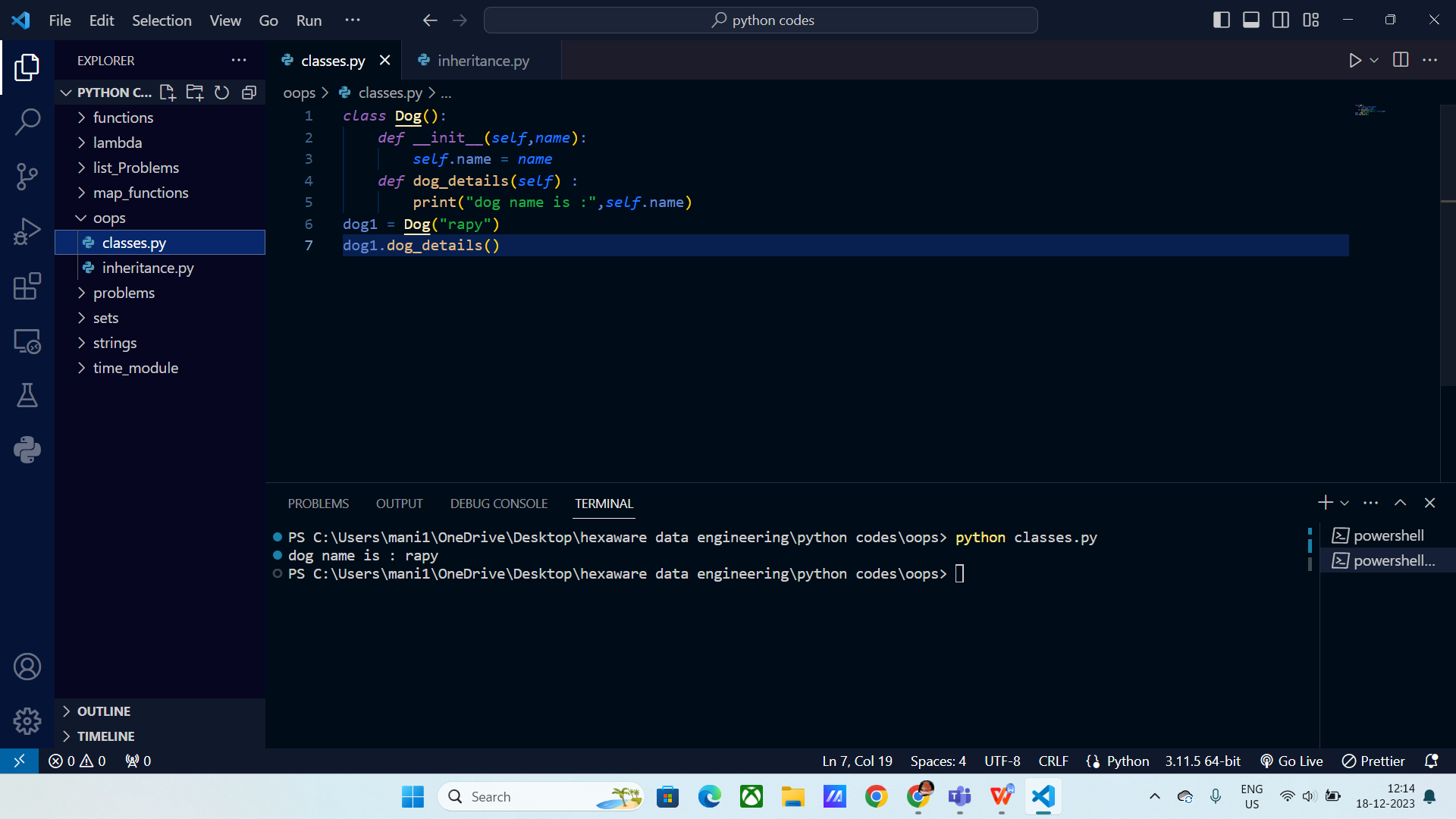
Python is considered as Object oriented language (OOP) as it contains classes and objects.

**Classes**: Classes are collection of objects.

**Objects**: Objects are real world entities.





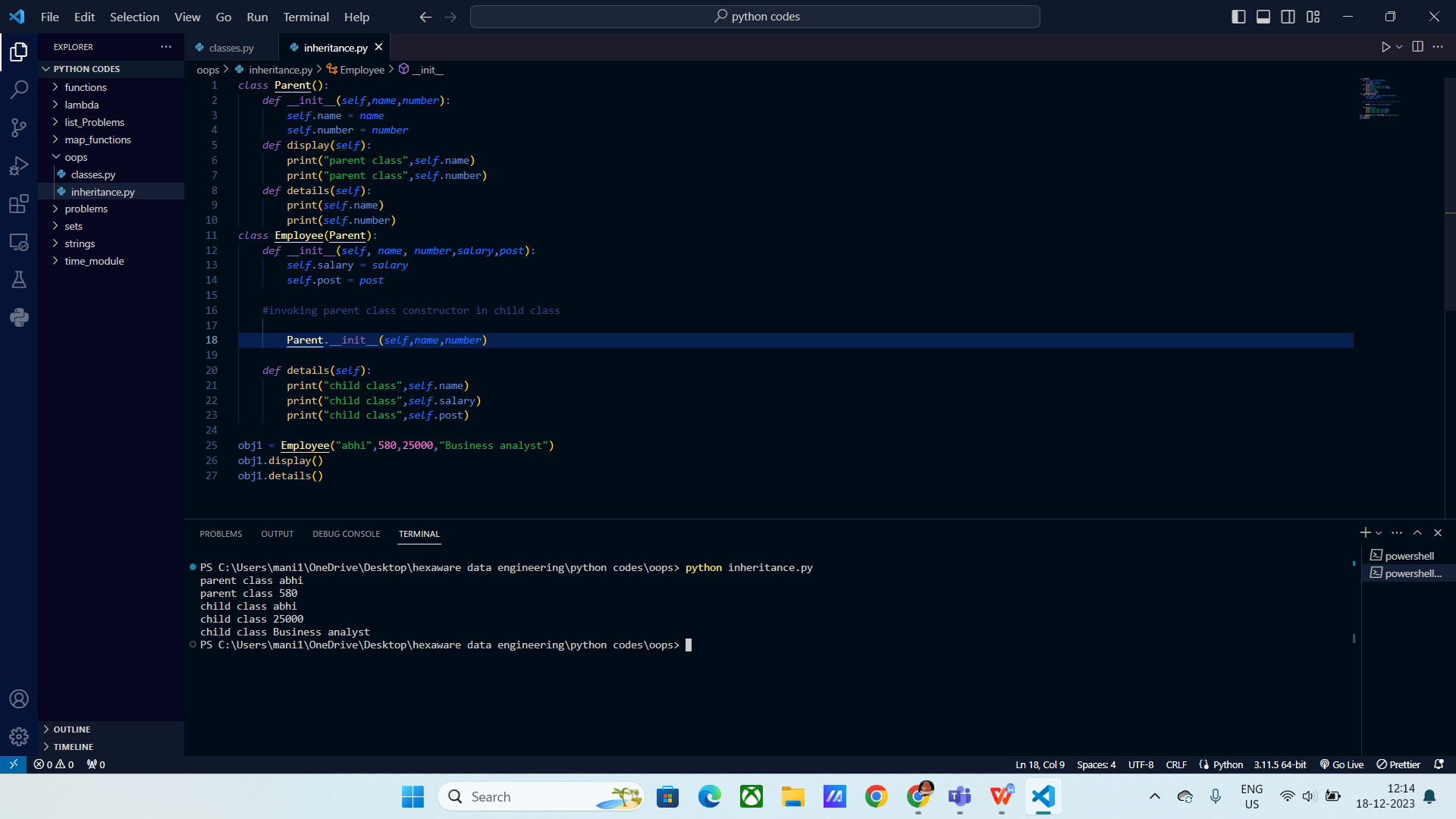


There are principles such as:

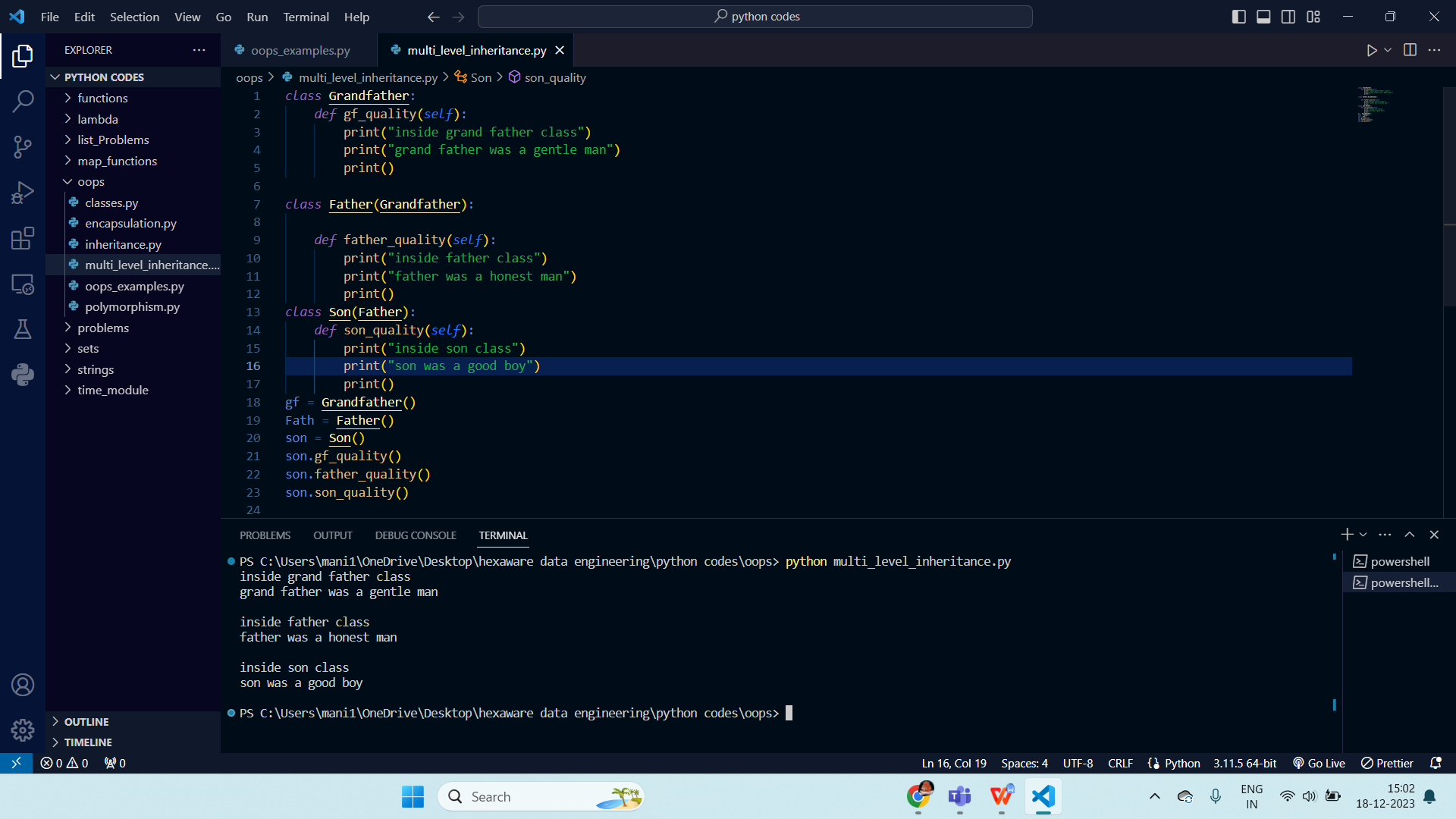
**Inheritance :** The properties from Parent class are inherited by child classes.

These are further classified into:

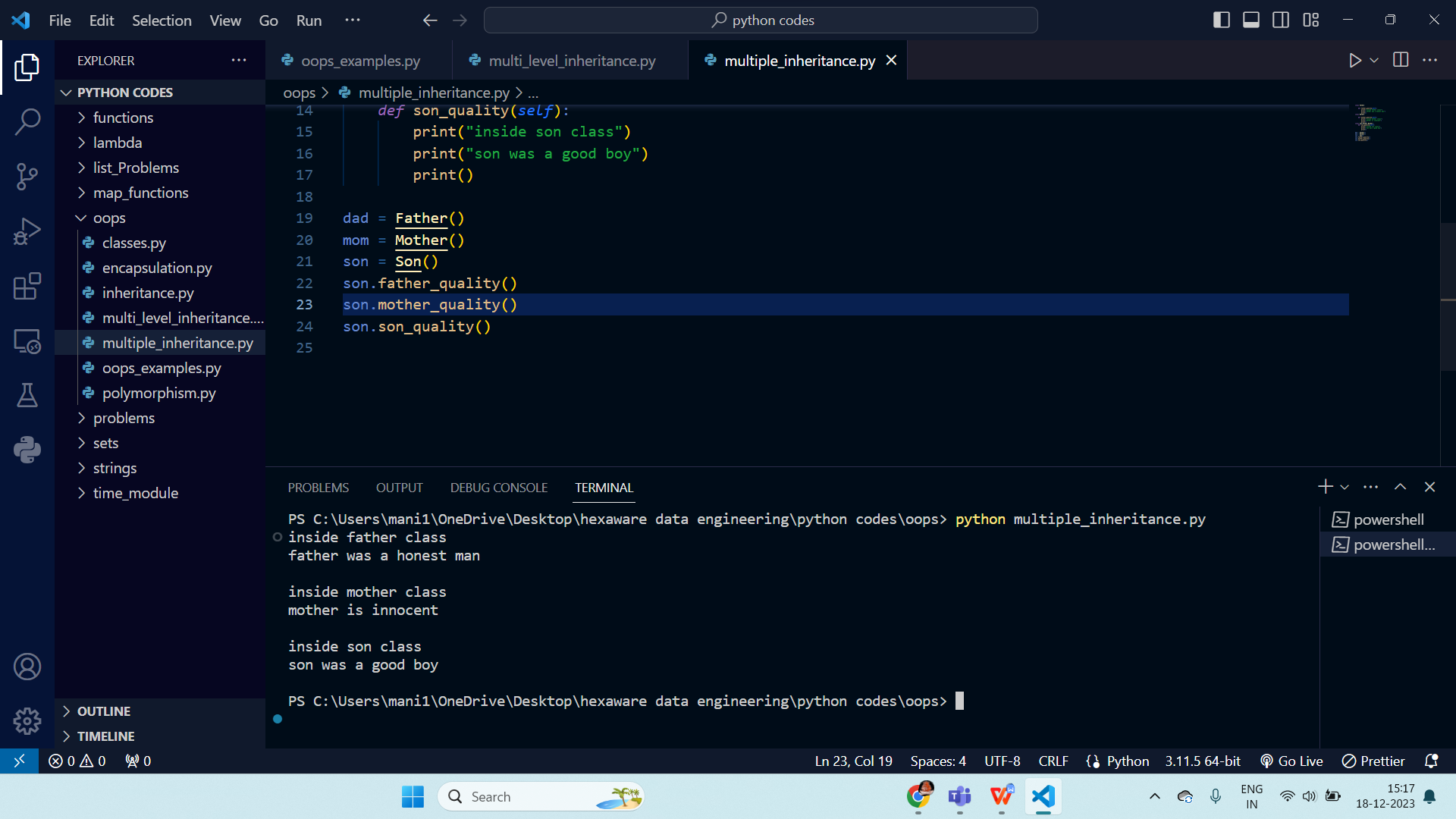
**Single Level :** One parent class properties are inherited by single child class.



**Multi level :** Base class properties are inherited by child classes.

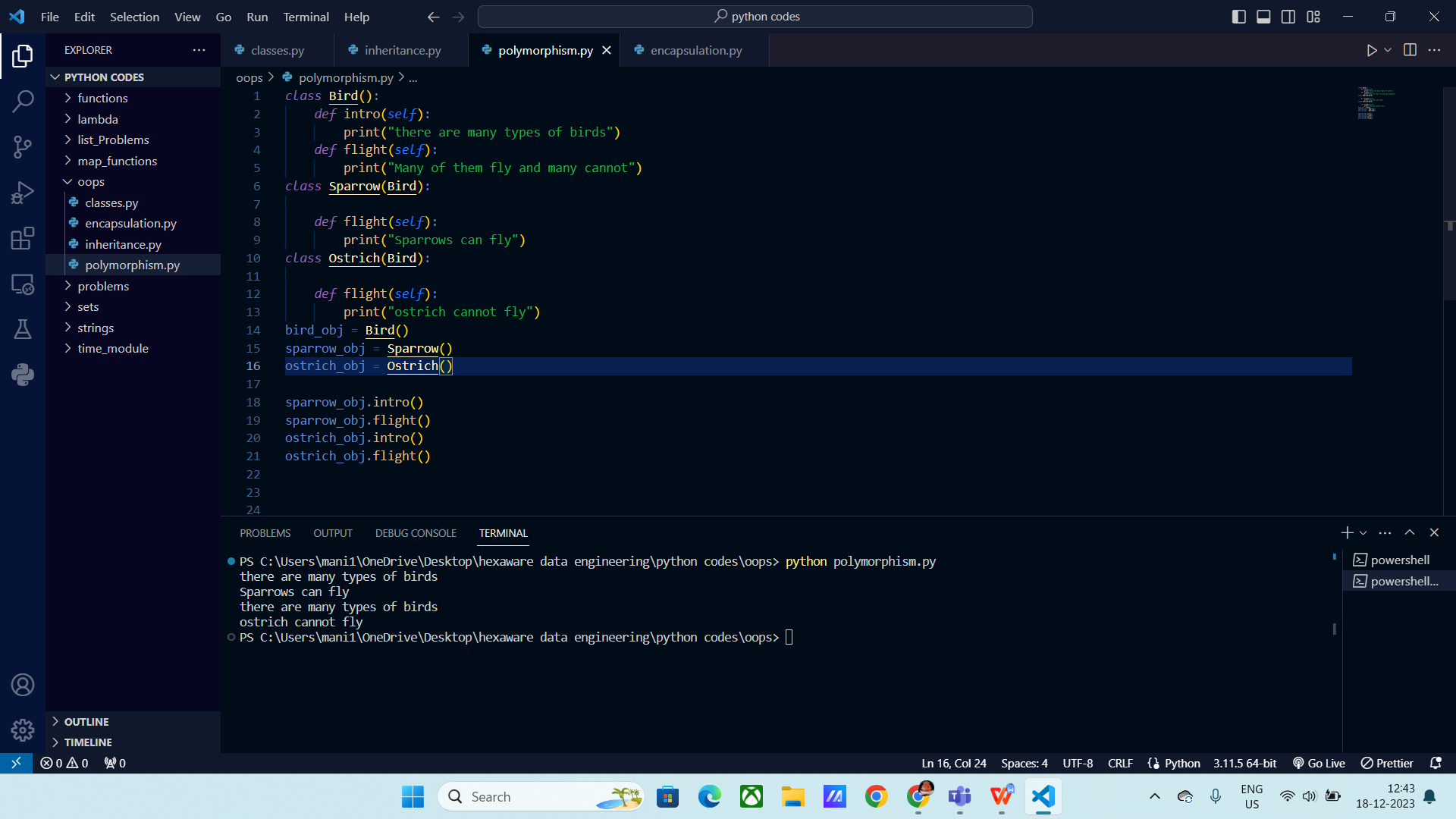


**Multiple level:** When a single child class inherits properties from two or more parent class.



**Polymorphism:** When one parameter has many form in one class and has different form in another class.

Poly means “ many forms ’.



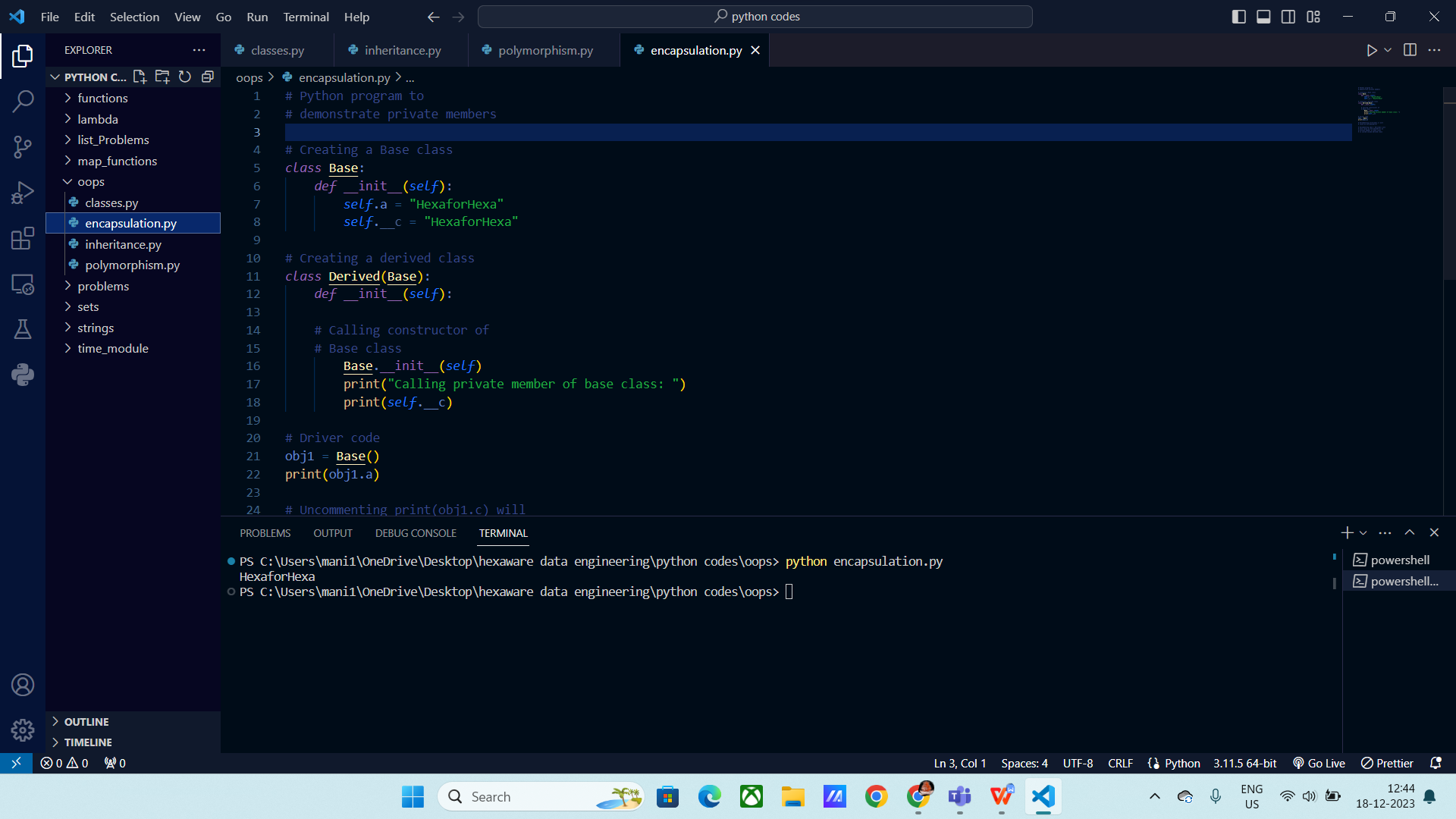
**Encapsulation:** Wrapping data and the mehtods in a single class, which can be accessed within the same class but not outside the class.

Encapsulation is achieved through Access specifiers :

Public members

Private members

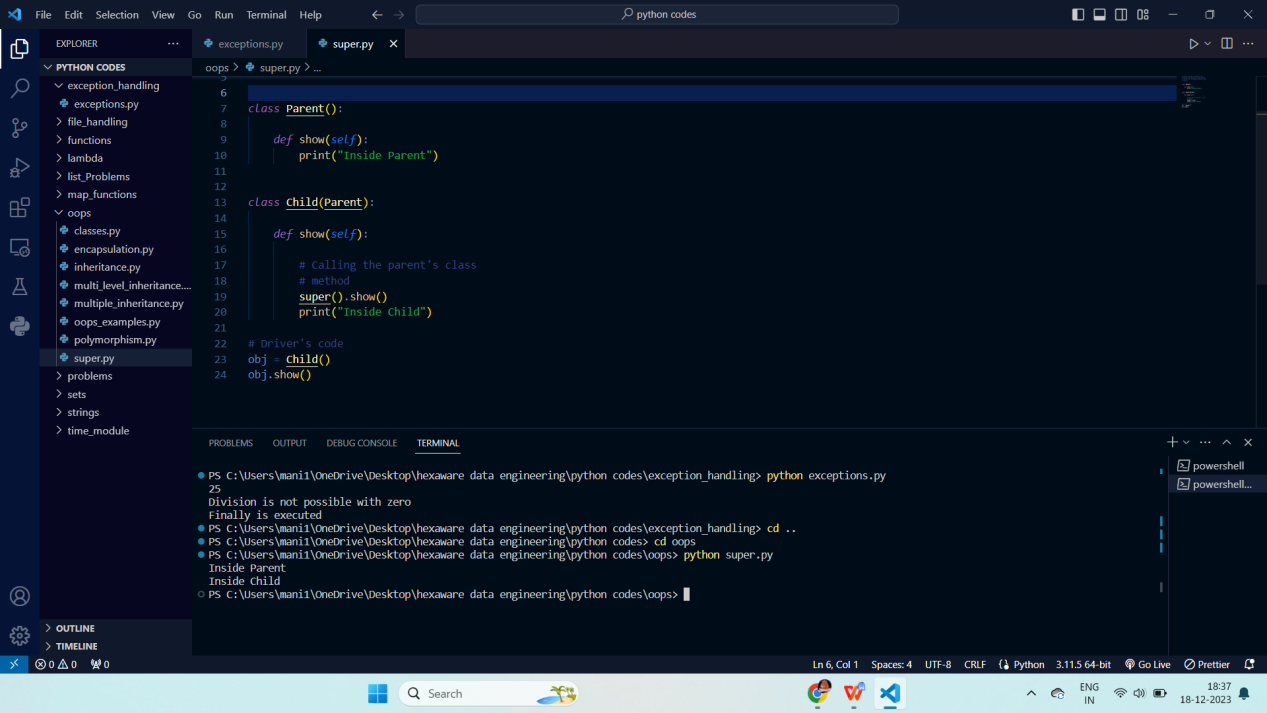
Protected members



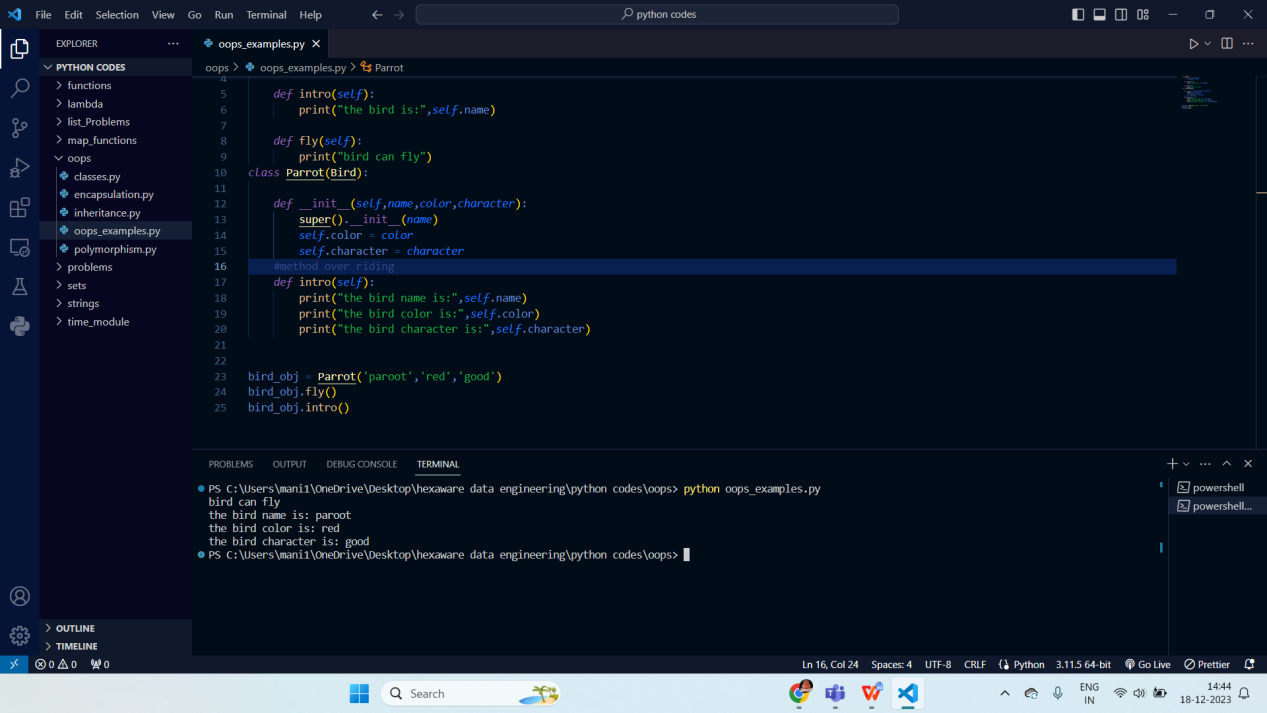
**Abstraction:** The process by which data and funcitons are defined in a way that only essential details are seen by user and unnecessary details are hidden.

These can be achieved through Abstract Classes.

Super() method: It is used to initialize parent class \_\_init\_\_ method in child classes.



**Method over riding :**



**File Handling:**

File handling includes create, update, delete files data using built methods.

**Open () :** Used to open file.

Syntax:

Open(“file\_name”, “mode”)

There are few modes, how a file can be opened.

**Read (r) :** Read mode, you are opening file in only read mode.

**Append (a) :** Append mode, you are opening file in only Append mode.

**Write (w) :** Write mode, , you are opening file in only Write mode.

**Create (x) :** create mode, , you are opening file in only create mode. If the file does not exists then a new file is created.

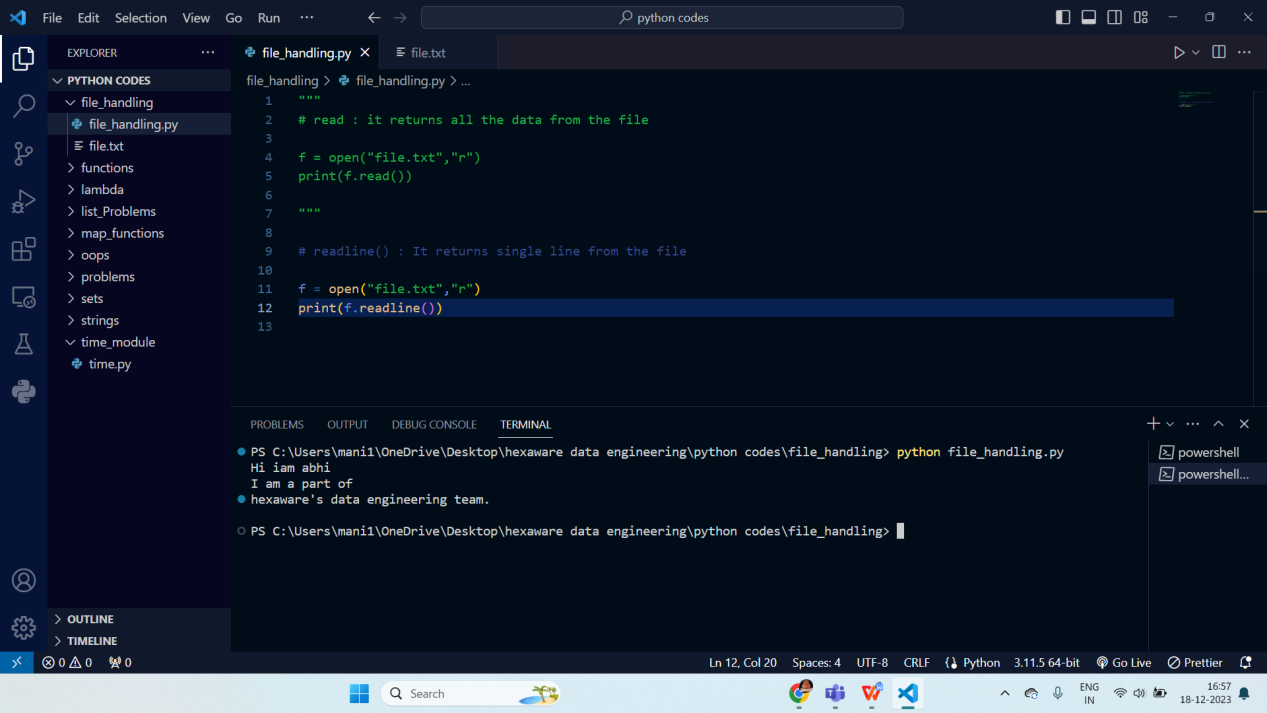
**Read():**

It returns all the data from the file.

Syntax:

f = open(“file”,”r”)

Print(f.read())

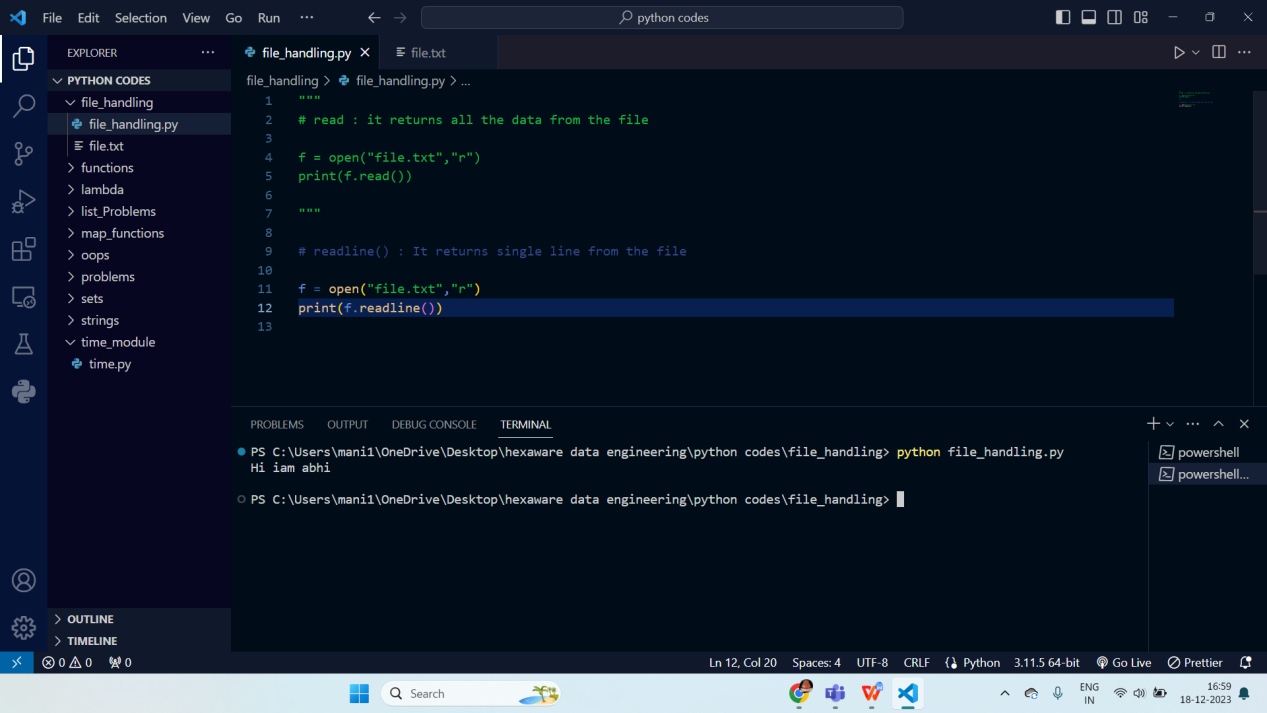


**Readline() :** It returns a single line from the file.

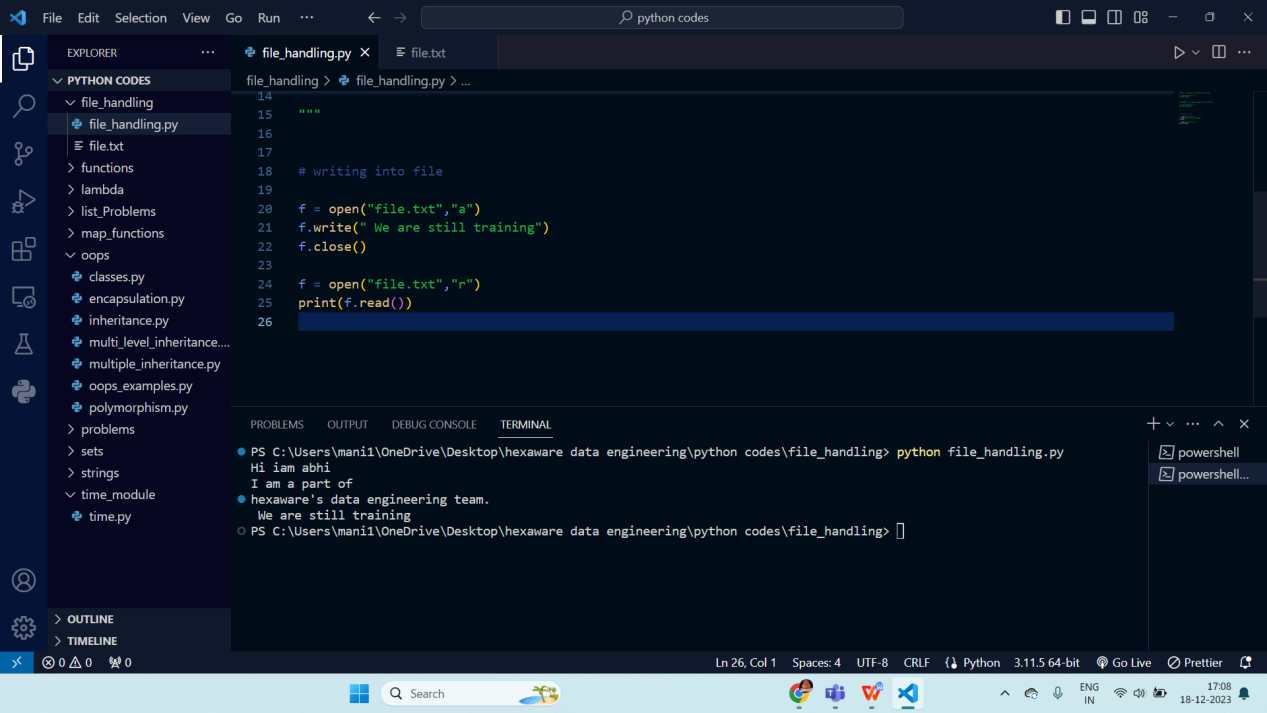
Syntax:

f = open(“file”,”r”)

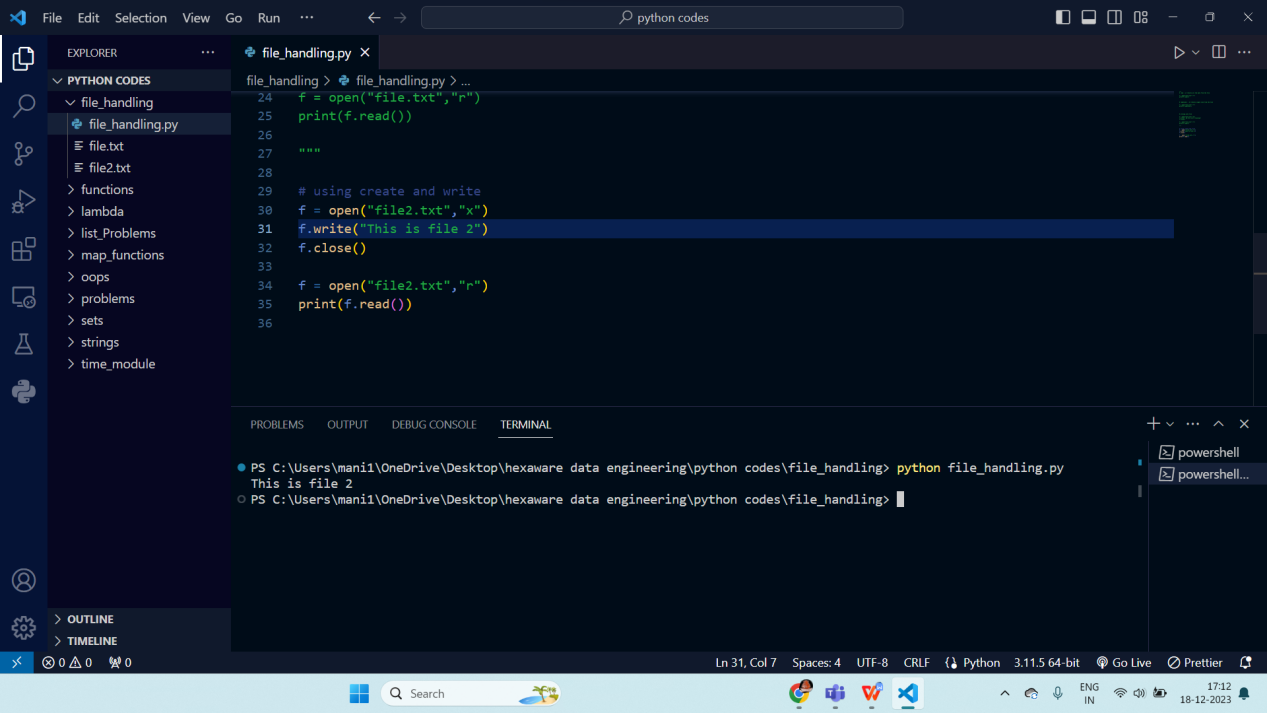
Print(f.readline())



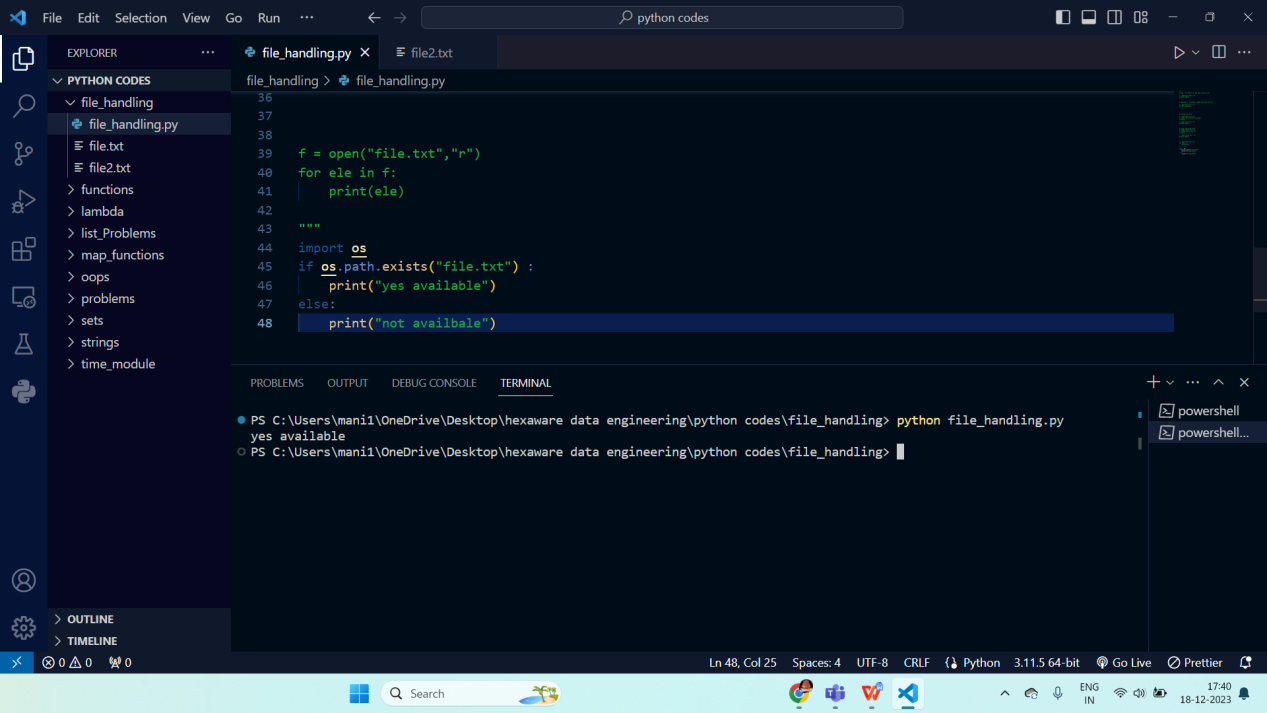
**Writing into file :**



**Using create and write mode :**



**Checking if the file exists in our local system or not.**



**Exception Handling:**

These are the errors which can be handled are known as Exceptions.

There are many kind of error which can be handled such as

* **Syntax Error:** This exception is raised when the interpreter encounters a syntax error in the code, such as a misspelled keyword, a missing colon, or an unbalanced parenthesis.
* **Type Error**: This exception is raised when there is a TYPO in code.
* **Name Error**: This exception is raised when a variable or function name is not found in the code.
* **Index Error**: This exception is raised when an index is out of range.

We usually see this error in list, dictionary.

* **Key Error**: This exception is raised when a key is not found in a dictionary.
* **Value Error**: This exception is raised when a function or method is called with an invalid argument.

When there are not enough arguments provided to the function argument list.

* **Attribute Error**: This exception is raised when an attribute or method is not found on an object.
* **IO Error**: This exception is raised when an I/O operation, such as reading or writing a file, fails due to an input/output error.
* **Zero Division Error**: This exception is raised when an attempt is made to divide a number by zero.
* **Import Error**: This exception is raised when an import statement fails to find or load a module.

Exceptions can be handled by

**Try :**

This block catches the errors.

Syntax:

Try:

// block of code

**Except:**

This block will be used to resolve errors.

Syntax :

Except Exception\_Name :

//block of code

**Finally:**

This block will run irrespective of the code.

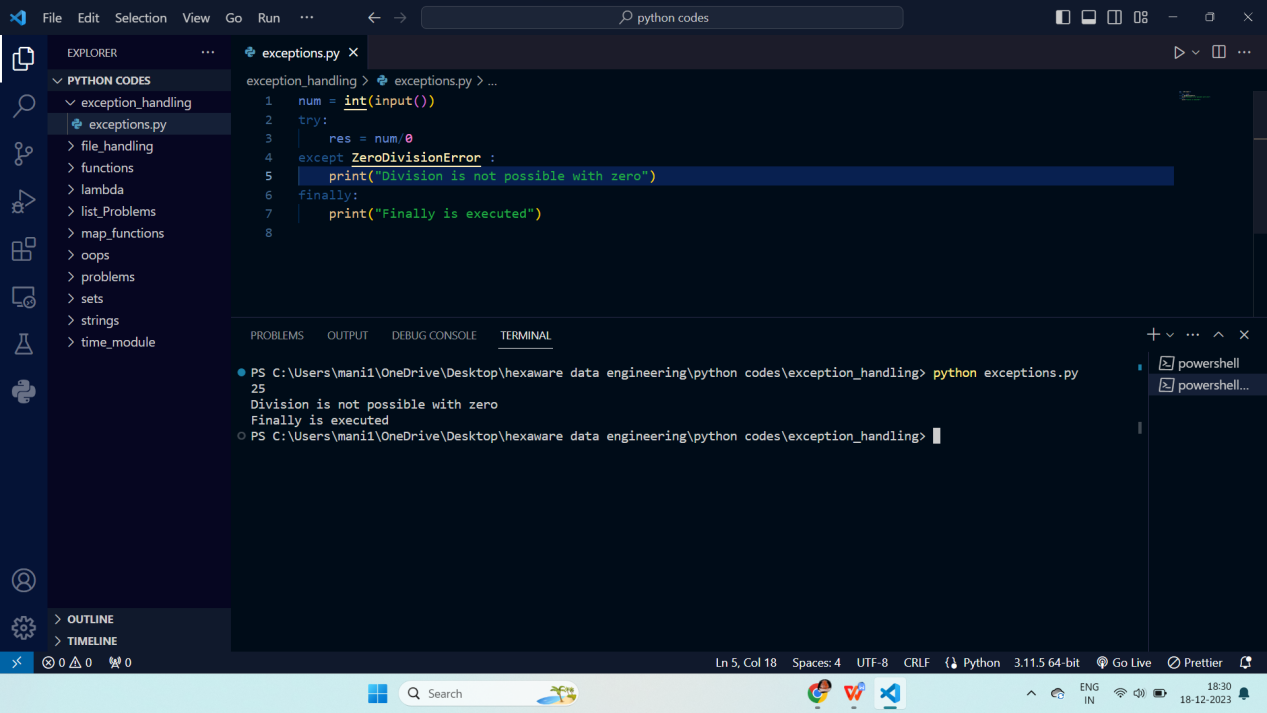
Syntax:

Finally :

//block of code

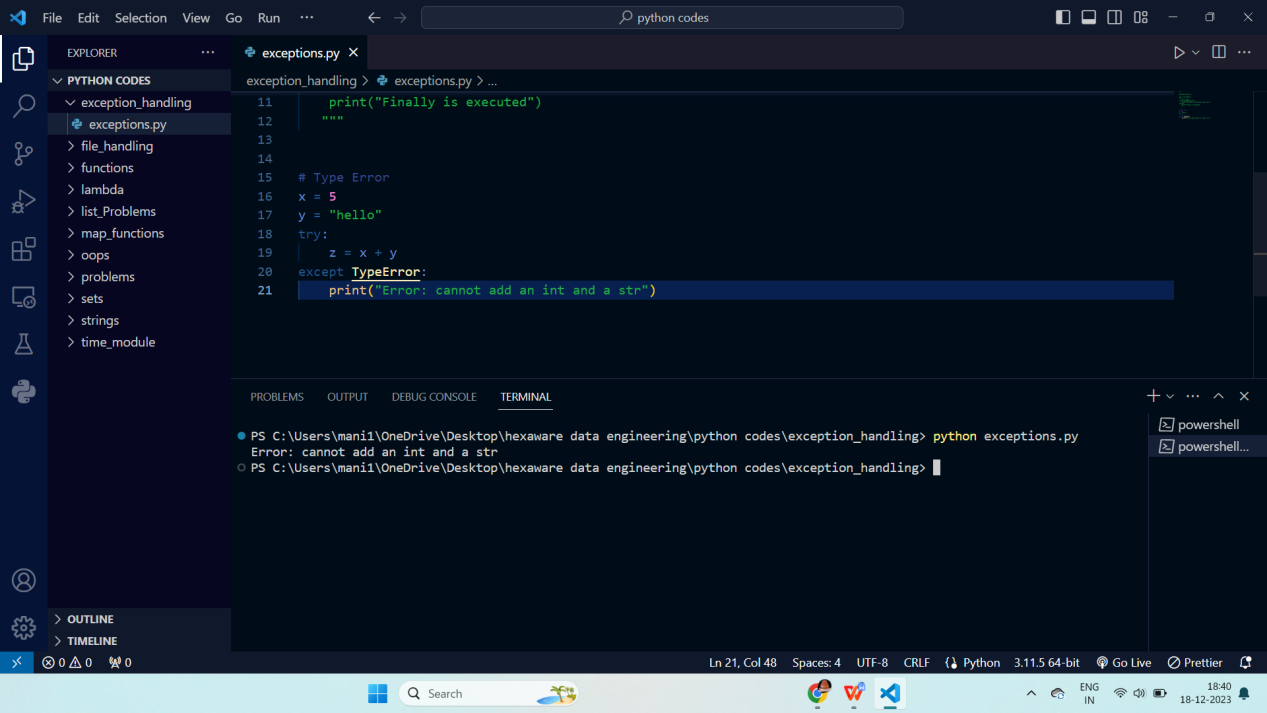
Here I’ve written the code which can give me error in the try block and handled the exception in Except block.

Here num input gives the **Zero Division Error,** So I handled with Except block.



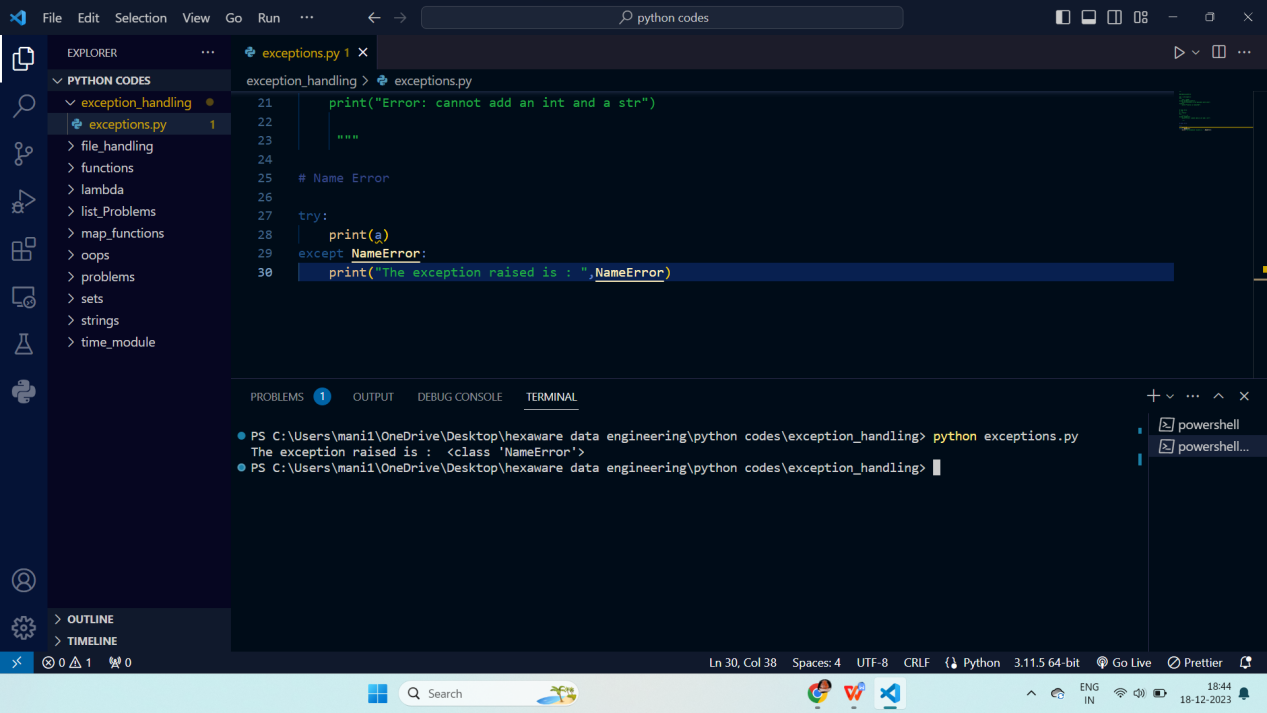
**Type Error:**

As we know that integer and string cannot be added or concatinaed, so it throws an error.



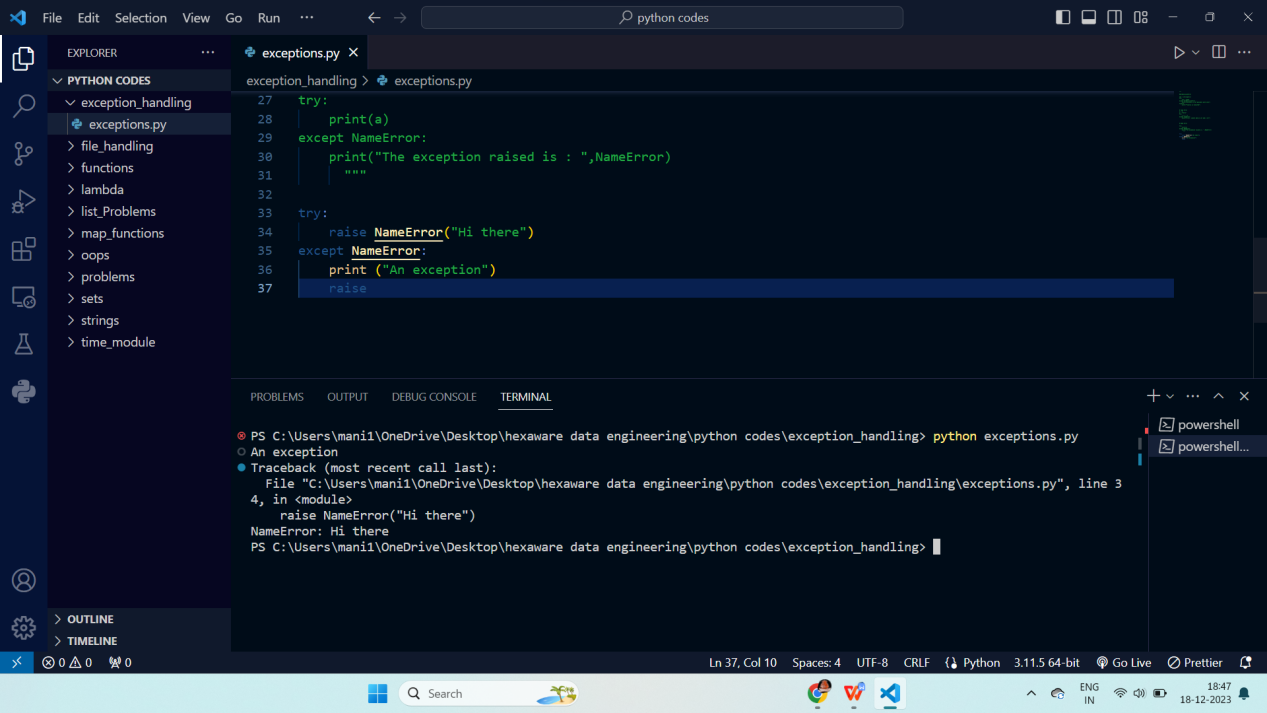
**Name Error:**

Here variable “a” is not there in the scope so it throws “ Name Error”.



Raising an Error:

Any error can be raised using Raise keyword in the Try block only.



**Python Modules:**

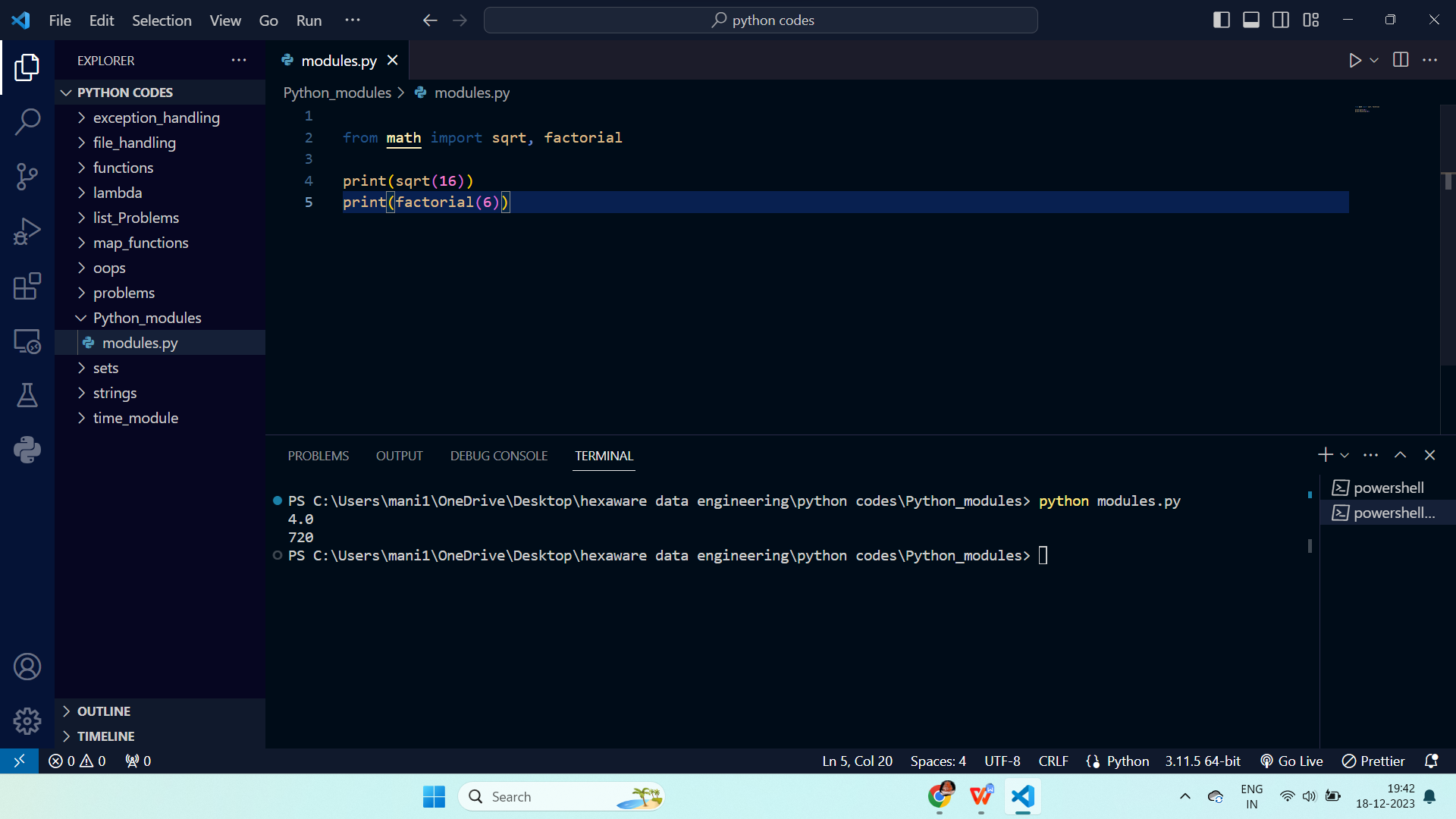
Modules in Python is a file containing Python definitions and statements.

**Syntax of Python Import:**

import module

**Math Module:**

It contains all the mathematical functions like log, sqrt, factorial, etc.



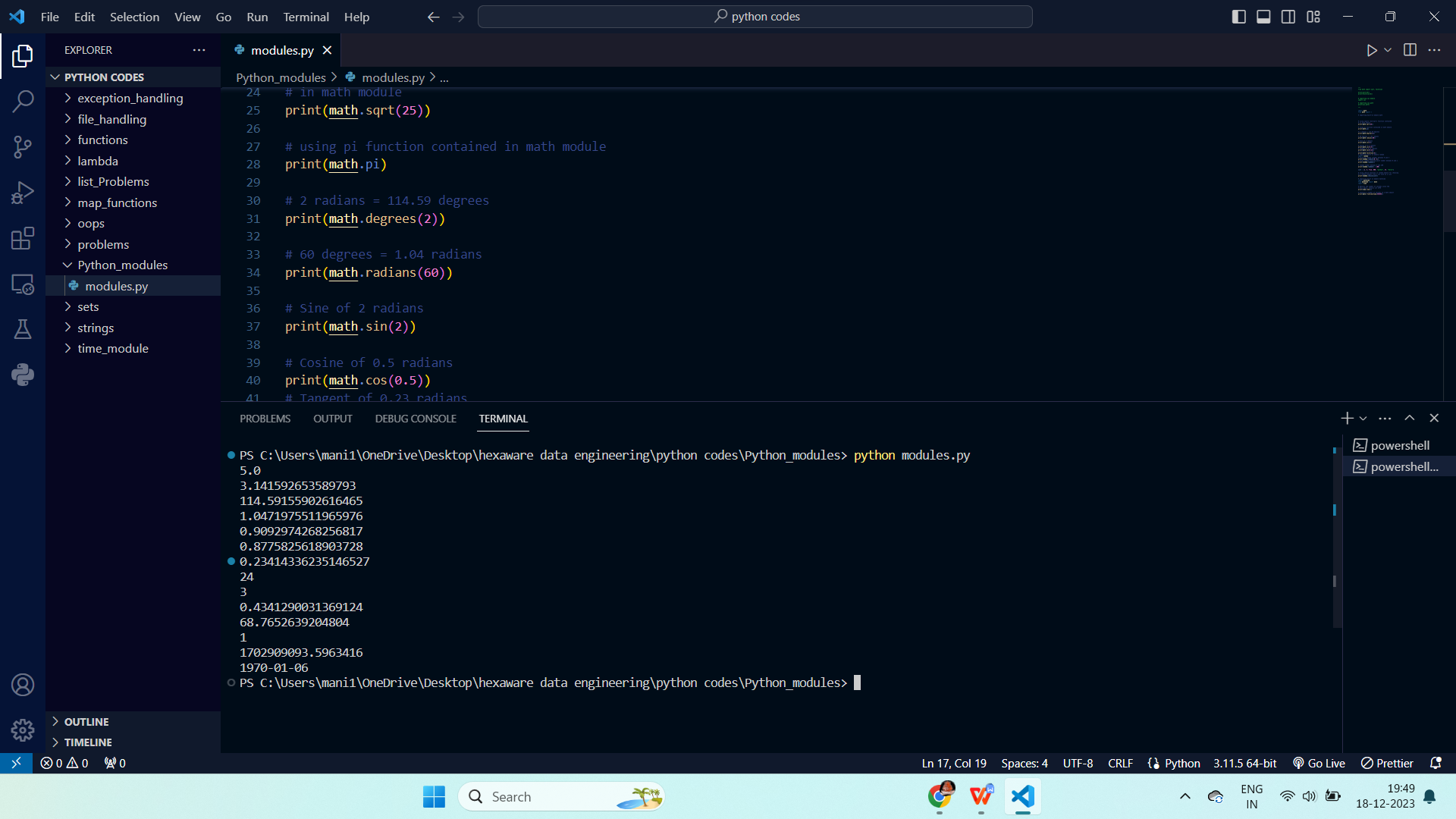
Here we used math module to calculate sqrt and factorial of the inputs.

Sys module:

We used Sys module to find out all the modules of python in the local system.



We used math module to find sin, cos, log, factorial, et.



Importing module and using the module

Here we can also create own modules and import them in other python files.

Syntax: Import file\_name

