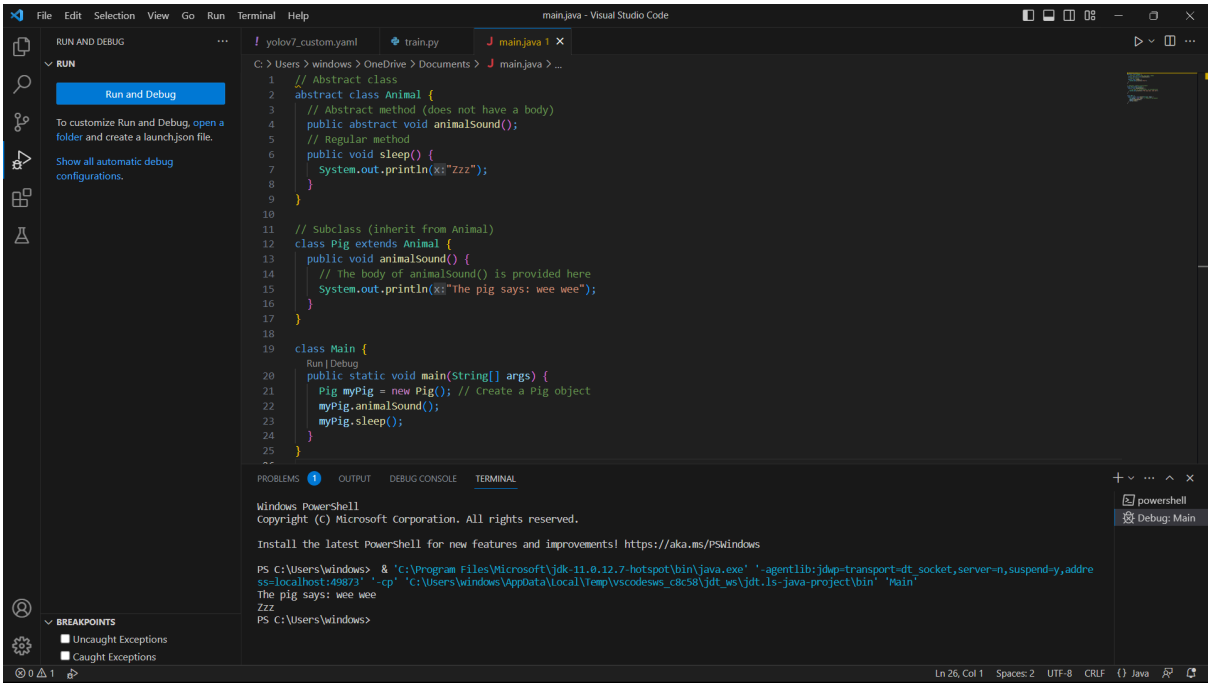
Abstraction

Abstraction in Object-Oriented Programming (OOP) is a fundamental principle that allows you to create models of real-world objects as classes. It focuses on representing essential features and behaviours of an object while hiding unnecessary details.

Abstraction helps in managing complexity by breaking down a system into smaller, more manageable parts. It allows you to create abstract classes or interfaces that define the common properties and methods that a group of objects should have, without specifying the implementation details. These abstract classes serve as blueprints for creating concrete classes, which provide the actual implementation.



Encapsulation in Java

Encapsulation is a fundamental concept in object-oriented programming (OOP) that refers to the bundling of data and methods that operate on that data within a single unit, which is called a class in Java. Encapsulation is a way of hiding the implementation details of a class from outside access and only exposing a public interface that can be used to interact with the class.

In Java, encapsulation is achieved by declaring the instance variables of a class as private, which means they can only be accessed within the class. To allow outside access to the instance variables, public methods called getters and setters are defined, which are used to retrieve and modify the values of the instance variables, respectively. By using getters and setters, the class can enforce its own data validation rules and ensure that its internal state remains consistent.