# **Conceptos de POO**

* **Clase**: Una clase es una plantilla o molde que define la estructura y el comportamiento de un objeto.
* **Objeto**: Un objeto es una instancia de una clase. Representa una entidad específica del mundo real que tiene propiedades y puede realizar acciones. Por ejemplo, si creas un objeto de la clase "Perro", ese objeto será un perro individual con características como nombre, edad y color.
* **Atributo**: Un atributo es una propiedad o característica de un objeto.
* **Método**: Un método es una función o procedimiento asociado a un objeto de una clase. Representa las acciones que un objeto puede realizar.
* **Encapsulación**: Es un concepto que permite ocultar los detalles internos de un objeto y solo exponer una interfaz pública para interactuar con él. Esto se logra mediante el uso de modificadores de acceso como **public**, **private** y **protected**.
* **Herencia**: Es un mecanismo que permite crear nuevas clases basadas en clases existentes. La clase nueva hereda los atributos y métodos de la clase existente y puede agregar nuevos atributos y métodos o modificar los existentes.
* **Polimorfismo**: Es la capacidad de un objeto de tomar diferentes formas o comportarse de diferentes maneras según el contexto.
* **Abstracción**: Es un concepto que nos permite abstraer los detalles de las implementaciones que estamos haciendo en nuestras clases.

# **¿Qué es la programación orientada a objetos (POO)?**

Es un paradigma de programación (formas o estilos en los que podemos programar y estructurar nuestro código) que se basa en el concepto de **clases** y **objetos**.

## **Tipos de paradigmas de programación**

* Programación funcional.
* Programación estructurada.
* Programación reactiva.
* Programación orientada a aspectos.
* Programación orientada a objetos.

