

Programación Orientada a Objetos

INTERMEDIATE

Pilares de la Programación Orientada a Objetos



ABSTRACCIÓN

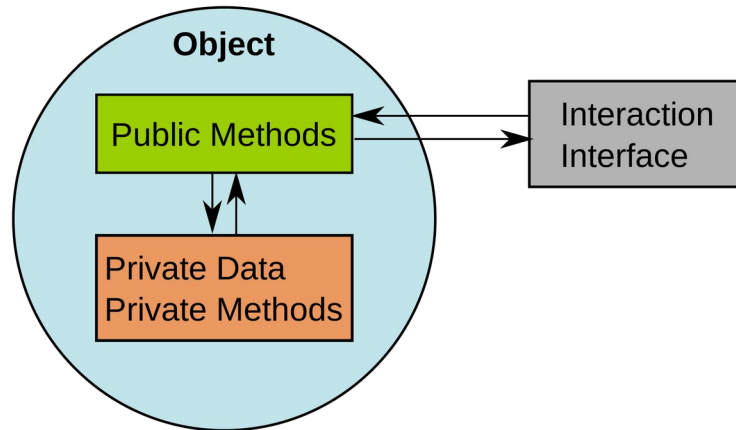
Proceso mediante el cual se definen las características de un objeto, eliminando los detalles innecesarios, para solo enfocarnos en los elementos que son útiles para el contexto o sistema que estamos desarrollando.



ENCAPSULAMIENTO

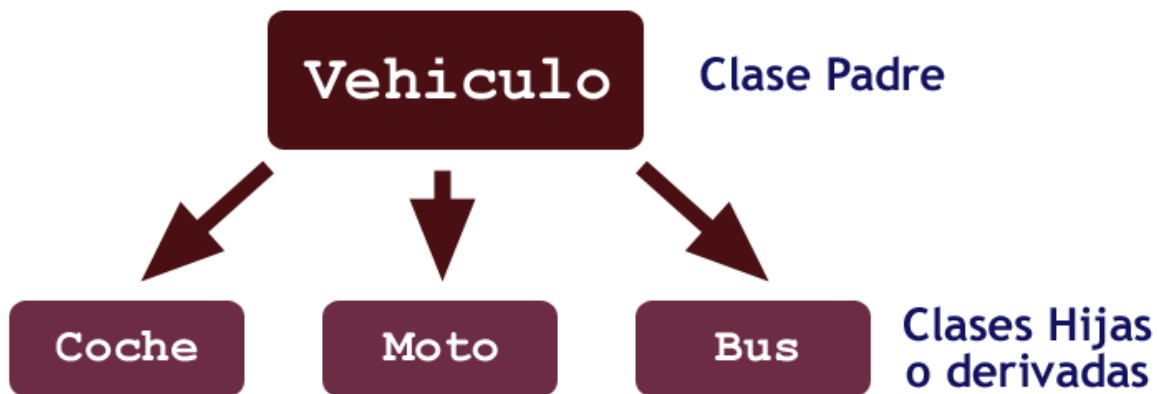
Consiste en ocultar los detalles que no son relevantes para el exterior, esto quiere decir que se deben agrupar los datos y métodos que actúan sobre esas propiedades de manera que el acceso a esos datos este restringido desde afuera.

Aquí entran en juego los famosos métodos **GETTERS** y **SETTERS**.



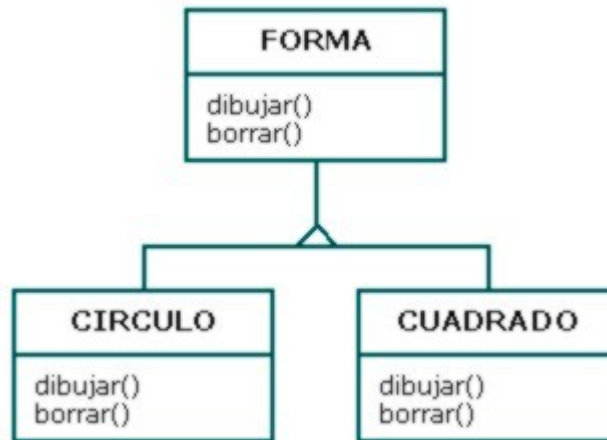
HERENCIA

Consiste en que por medio de una clase 'Padre' una clase llamada 'hija' hereda las propiedades y los atributos de su 'padre'.



POLIMORFISMO

Deriva del latín *poli*: **muchos**, *morfo*: **forma**. Y su aplicación en la POO, consiste en que una clase que hereda de otra puede redefinir sus métodos, para aplicar una funcionalidad propia.





Hands On

RETOS DE PILARES POO

RETO 1

Crear una clase **Persona** con atributos *nombre*, *apellido*, *edad*, y métodos mostrarInfo que retorne la información de la persona.

Crear las clases **Profesor** y **Alumno** que hereden de la clase **Persona**, **Profesor**, debe tener atributos *especialidad* (que indica en que es especialista) e *institución* que define la institución donde labora, también un atributo *salario*.

La clase **Alumno** debe tener propiedades *curso* y *grado*, que especifican el curso y grado que cursa el alumno.

En ambas clases se debe reescribir el método mostrarInfo, para que se indique la información propia de cada objeto.

RETOS DE PILARES POO

RETO 2

Modifique las clases del reto anterior, y añada los métodos necesarios para encapsular los atributos de cada clase.

RETO 3

Modifique las clases del reto anterior, donde se agregue un método propio de cada clase que hereda y que muestre que tipo de objeto es con su respectiva información

RETOS DE PILARES POO

RETO 4

Cree una clase **Forma**, que contenga los atributos *nombre*, *color* y *posición*, entre las acciones de la clase están:

imprimirForma: imprime la información de la forma.

cambiarColor: me permite cambiar el color de la forma.

moverForma: mediante el cual movemos la forma en una unidad.

Defina una clase **Rectángulo** que herede de **Forma**, con atributos *largo* y *ancho*, y con funcionalidad calcular área, que me permita calcular el área del rectángulo y calcular perímetro.

RETOS DE PILARES POO

RETO 5

Cree una clase **Elipse** que herede de Forma, con atributos radio mayor y radio menor, establezca un método que permita calcular el área de la elipse.

RETO 6

Cree una clase **Cuadrado** que herede de Forma, y defina lado como propiedad, establezca un método que permita calcular el área y perímetro del cuadrado.

Happy Coding...

