

Guía 3 Ejercicios Programación II: Polimorfismo

1. Juegos:

Implementa una clase `Juego` con un método `iniciar()`. Luego, crea dos clases que hereden de `Juego`: `Ajedrez` y `Futbol`, cada una con su propia implementación de `iniciar()`. Muestra cómo utilizar polimorfismo para iniciar distintos juegos.

2. Almacenables:

Define una interfaz `Almacenable` con un método `guardar()`. Luego, crea las clases `Archivo` y `BaseDeDatos` que implementen esta interfaz y cada una tenga su propia lógica para guardar información. Usa polimorfismo para guardar diferentes tipos de datos.

3. Robot:

Crea una clase `Robot` con un método `trabajar()`. Luego, crea dos clases que hereden de `Robot`: `RobotLimpiador` y `RobotCocinero`. Cada clase debe implementar el método `trabajar()` con su propia lógica. Utiliza polimorfismo para hacer trabajar a los robots.

4. Mensajes:

Crea una clase base `Empleado` con atributos `nombre` y `salario`. Luego crea dos subclases: `Gerente` y `Desarrollador`. En la clase `Gerente`, añade un atributo para el número de empleados supervisados. En la clase `Desarrollador`, añade un atributo para el lenguaje de programación que utiliza.

5. Transportes:

Crea una clase `Transporte` con un método `viajar()`. Luego, crea dos subclases `Avion` y `Barco`, cada una con su propia implementación del método `viajar()`. Usa polimorfismo para manejar diferentes medios de transporte.

6. Imprimibles:

Define una interfaz `Imprimible` con un método `imprimir()`. Implementa dos clases que implementen esta interfaz: `Informe` y `Factura`, cada una con su propio método de impresión. Muestra cómo usar polimorfismo para imprimir distintos documentos.

7. DispositivoElectronico:

Crea una clase `DispositivoElectronico` con un método `encender()`. Luego, crea dos subclases `TelefonoMovil` y `Televisor`, cada una con su propio comportamiento al encenderse. Utiliza polimorfismo para encender diferentes dispositivos electrónicos.

8. Cobrable:

Crea una interfaz `Cobrable` con un método `cobrar()`. Luego, crea las clases `Servicio` y `Producto`, que implementen esta interfaz y tengan su propio método para cobrar. Utiliza polimorfismo para cobrar tanto servicios como productos.

9. Instrumentos:

Crea una clase abstracta `Instrumento` con un método `afinar()`. Implementa dos subclases `Violin` y `Trompeta`, cada una con su propia lógica para afinar el instrumento. Utiliza polimorfismo para afinar diferentes instrumentos musicales.

10. Operables:

Define una interfaz `Operable` con un método `operar()`. Luego, implementa dos clases `Calculadora` y `RobotQuirurgico`, que implementen esta interfaz y definan su propia lógica de operación. Usa polimorfismo para realizar operaciones con distintos dispositivos.