

Ejercicios Adicionales de Polimorfismo en Java

Ejercicio 6: Herramientas y uso

Objetivo: Aplicar polimorfismo en herramientas con diferentes funcionalidades.

Instrucciones:

1. Crea una clase abstracta `Herramienta` con un método abstracto `usar()`.
 2. Implementa dos subclases:
 - `Martillo`, que sobrescriba `usar()` para imprimir "Usando el martillo para clavar un clavo".
 - `Destornillador`, que sobrescriba `usar()` para imprimir "Usando el destornillador para ajustar un tornillo".
 3. Implementa una clase de prueba que cree instancias de `Martillo` y `Destornillador` y llame a `usar()` en cada una.
-

Ejercicio 7: Tipos de mensajes

Objetivo: Implementar polimorfismo en diferentes tipos de comunicación.

Instrucciones:

1. Crea una clase abstracta `Mensaje` con un método abstracto `enviar()`.
 2. Implementa dos subclases:
 - `Email`, que sobrescriba `enviar()` para imprimir "Enviando un correo electrónico".
 - `SMS`, que sobrescriba `enviar()` para imprimir "Enviando un mensaje de texto".
 3. Implementa una clase de prueba que cree instancias de `Email` y `SMS` y llame a `enviar()` en cada una.
-

Ejercicio 8: Dispositivos inteligentes

Objetivo: Usar polimorfismo para representar dispositivos con distintas funciones.

Instrucciones:

1. Crea una clase abstracta `DispositivoInteligente` con un método abstracto `realizarFuncion()`.
 2. Implementa dos subclases:
 - `AsistenteVirtual`, que sobrescriba `realizarFuncion()` para imprimir "Respondiendo preguntas y controlando dispositivos".
 - `SmartWatch`, que sobrescriba `realizarFuncion()` para imprimir "Mostrando la hora y registrando actividad física".
 3. Implementa una clase de prueba que cree instancias de `AsistenteVirtual` y `SmartWatch` y llame a `realizarFuncion()` en cada una.
-

Ejercicio 9: Formas de pago

Objetivo: Implementar polimorfismo en métodos de pago.

Instrucciones:

1. Crea una clase abstracta `FormaPago` con un método abstracto `procesarPago(double monto)`.
 2. Implementa dos subclases:
 - `TarjetaCredito`, que sobrescriba `procesarPago()` para imprimir "Procesando pago con tarjeta de crédito por [monto]".
 - `Efectivo`, que sobrescriba `procesarPago()` para imprimir "Procesando pago en efectivo por [monto]".
 3. Implementa una clase de prueba que cree instancias de `TarjetaCredito` y `Efectivo` y llame a `procesarPago()` en cada una.
-

Ejercicio 10: Videojuegos y personajes

Objetivo: Aplicar polimorfismo en un sistema de personajes de un videojuego.

Instrucciones:

1. Crea una clase abstracta `PersonajeJuego` con un método abstracto `atacar()`.
 2. Implementa dos subclases:
 - `Guerrero`, que sobrescriba `atacar()` para imprimir "El guerrero ataca con su espada".
 - `Arquero`, que sobrescriba `atacar()` para imprimir "El arquero dispara una flecha".
 3. Implementa una clase de prueba que cree instancias de `Guerrero` y `Arquero` y llame a `atacar()` en cada una.
-

Estos ejercicios reforzarán el uso de polimorfismo en diferentes contextos. ¡Manos a la obra! 🚀