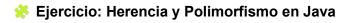


Consejería de Educación, Cultura y Deportes IES VIRREY MORCILLO

AVDA. MENÉNDEZ PELAYO S/N 967 140 881 02003120.ies@edu.jccm.es







© Objetivo

Demostrar la comprensión de los conceptos de **herencia** y **polimorfismo** mediante la creación de una jerarquía de clases relacionada con la gestión de empleados en una empresa.

Descripción del ejercicio

La empresa **TechSolutions S.A.** quiere desarrollar una pequeña aplicación en Java para **gestionar los empleados**.

Para ello, deberás definir una jerarquía de clases que represente diferentes tipos de empleados y sus comportamientos.

1. Clase base Empleado

Elemento Descripción

Atributos nombre (String), salarioBase (double)

Constructor Inicializa ambos atributos.

MétodosIncluye métodos para obtener los valores de los atributos y calcular y mostrar el salario base del empleado.

🔅 2. Subclases que heredan de Empleado

Crea las siguientes subclases aplicando el principio de herencia:

Clase EmpleadoTiempoCompleto

- Añade un atributo adicional para representar las horas extra trabajadas.
- Redefine el método que calcula el salario para incluir el pago por horas extra.
- Redefine también el método que muestra la información, aprovechando el método de la clase base mediante super.

Clase EmpleadoPorHoras

- Incluye atributos para las horas trabajadas y la tarifa por hora.
- Redefine el método que calcula el salario para obtener el pago según las horas trabajadas.



Consejería de Educación, Cultura y Deportes IES VIRREY MORCILLO

AVDA. MENÉNDEZ PELAYO S/N 967 140 881 02003120.ies@edu.jccm.es





Adapta el método que muestra la información a los datos de esta clase.

3. Clase principal MainEmpresa

Crea una clase que contenga el método main() y que:

- 1. Cree un array de tipo Empleado con distintos tipos de empleados.
- 2. Recorra el array y muestre la información de cada empleado.
- 3. Comprueba cómo se ejecuta el método correspondiente de cada subclase, demostrando el **polimorfismo**.

4. Estructura del proyecto
HerenciaEmpleados/
—— model
├— Empleado.java
├— EmpleadoTiempoCompleto.java
│
L— main
L— MainEmpresa.java

5. Entrega

- El código deberá entregarse a través del **repositorio de GitHub** utilizado en los ejercicios anteriores.
- Crea una nueva carpeta con el nombre:
- 03_Inheritance
- Incluye un archivo README.md donde expliques brevemente:
 - o La jerarquía de clases creada.
 - o Cómo se aplica el **polimorfismo** en tu programa.



Consejería de Educación, Cultura y Deportes IES VIRREY MORCILLO

AVDA. MENÉNDEZ PELAYO S/N 967 140 881 02003120.ies@edu.jccm.es





7. Preguntas de reflexión

- 1. ¿Por qué es útil definir métodos con el mismo nombre en distintas clases?
- 2. ¿Qué ventaja aporta el uso de super frente a duplicar código?
- 3. ¿Cómo mejora el polimorfismo la escalabilidad del programa?
- 4. Si añadieras un nuevo tipo de empleado, ¿qué cambios deberías realizar en la jerarquía?

La respuesta a estas preguntas de reflexión debe añadirse en el documento README, creado dentro del repositorio.