

Ejercicios de PROGRAMACION ORIENTADA A OBJETOS

Tema: Herencia, Clases Abstractas, Interfaces y Polimorfismo

1. **CASO PACIENTE:** Diseñe un programa que ingrese los datos de los pacientes de una clínica: código del paciente, apellidos y nombres, sexo, edad, especialidad en la que se atiende (puede ser cirugía, nefrología o gastroenterología), cantidad de días de internamiento, costo por día, costo por rehabilitación solo para cirugía, costo por uso de equipos solo para nefrología, costo por resonancia magnética solo para gastroenterología, descuento del 10% para todos los pacientes de cirugía, descuento del 15% para pacientes de nefrología menores de 12 años y descuento del 12% para pacientes de gastroenterología que sean mujeres.

Se pide aplicar **herencia, clase abstracta e interface**. La clase abstracta tendrá como método abstracto el importe bruto a pagar, la interface reporte tendrá como método `toString()` y la interface apoyo tendrá como método `descuento`.

2. **CASO UN ESTUDIANTE:** Construya una **clase abstracta** de nombre **Estudiante** que tiene los siguientes atributos: código, apellidos, nombres, semestre de ingreso (2023-I o 2023-II) e incluya sus constructores. Además, construya sus métodos getter y setter (encapsulamiento) y también declare los **métodos abstractos** como pago parcial de pensión y descuentos, además construya **métodos no abstractos** como el cálculo del pago final de pensión que será igual a: pago parcial de pensión - descuentos.

Luego construya una subclase **Estudiante de PreGrado** de la clase Estudiante que tendrá los atributos como: Categoría (A o B), Promedio ponderado, colegio de Procedencia (Estatad o Particular) y contador de atributo estático. Construya su constructor que actualizará los datos de la clase padre, así como actualizará el atributo código, heredado de la clase Estudiante con un código autogenerado y sus métodos getter y setter. Para esta clase desarrolle los métodos abstractos heredados y que permitan hacer lo siguiente:

Calcule el pago parcial de pensiones en función a la categoría y el colegio de procedencia tal y como se muestra en la tabla:

Categoría	Pago parcial de pensiones	
	Estatad	Particular
A	320	450
B	300	420

Además, calcular el descuento sobre el pago parcial de pensiones de acuerdo al promedio ponderado como sigue a continuación:

Rango del Promedio Ponderado	% de descuento sobre el pago de pensión
Menor a 13	0%
Mayor a 13	8%

Finalmente, tendrá su método propio que permita generar un código con la siguiente característica: EPRE0001, EPRE0002...

3. **CASO ASISTENTE** Construya una clase abstracta de nombre **Asistente** que tiene los siguientes atributos: código, apellidos, nombres, especialidad (Recursos Humanos o Comportamiento Organizacional) incluya sus constructores. Además, construya sus métodos getter y setter (encapsulamiento) y también declare los **métodos abstractos** como sueldo base, descuentos y bonificación, además construya **métodos no abstractos** como cálculo del sueldo bruto: sueldo base + bonificación y del sueldo neto que se calcula de: sueldo bruto – descuentos., además de un método que permita visualizar la información.

Asimismo, construya una interface que permita manejar las siguientes constantes el descuento de porcentajes del 17% para el AFP y 5% para Essalud que se implementara sobre la clase Asistente.

Luego construya la Clase **Asistente de Gerencia** de la clase Asistente que tiene los siguientes atributos: categoría (A o B) y contador con atributo. Construya su constructor que actualizará los datos de la clase padre, así como actualizara el atributo código, heredado de la Clase Asistente con un código autogenerado y sus métodos getter y setter. Para esta clase desarrolle los métodos abstractos heredados y que permitan hacer lo siguiente:

El sueldo base se calcula de la siguiente tabla:

Categoría	SUELDO BASE	
	Recursos Humanos	Comportamiento Organizacional
A	9000	11000
B	8000	10000

La bonificación será de 600 soles cualquiera que sea su categoría. Los descuentos se aplican de acuerdo a las constantes establecidas en la interface. Esta clase tendrá un método propio que asignará un monto de movilidad de acuerdo al cargo

Categoría	Monto
A	850
B	550

Finalmente, otro método propio que permita generar un código con la siguiente característica: AGE00001, AGE00002...

7. **CASO VENDEDOR** Implemente una **clase padre abstracta** de nombre **Empleado** con los siguientes atributos: DNI, apellidos, nombres, así como su constructor correspondiente que actualice su información y considere un método para mostrar la información de los atributos, así como sus métodos getter y setter (encapsulamiento).

Así como también defina **métodos abstractos** para los ingresos, bonificación y descuentos, y también métodos no abstractos para el sueldo y la impresión de la información del empleado.

Luego construya una **clase hija** de nombre **EmpleadoVendedor** con los siguientes atributos: monto vendido y la tasa de comisión y su respectivo constructor y los métodos setter y getter (encapsulamiento). Para esta clase desarrolle los métodos abstractos heredados con las siguientes características y que permita hacer lo siguiente:

- Calcular sus ingresos basados en $\text{montovendido} * \text{tasacomision}$.
- Calcular su bonificación de la siguiente manera: si el monto vendido es menor que 1000 no tendrá bonificación, si está entre 1000 y 5000 será el 5% de sus ingresos, si supera los 5000 su bonificación será del 10% de sus ingresos.
- Calcular su descuento de la siguiente manera: si sus ingresos son menores a 1000 su descuento será del 11% de sus ingresos sino será el 15% de sus ingresos.
- Calcular su sueldo neto como: $\text{ingresos} + \text{bonificaciones} - \text{descuentos}$

Luego construya otra **clase hija** de nombre **EmpleadoPermanente** con los siguientes atributos: sueldo base y afiliación (AFP y SNP) e incluya su constructor y los métodos setter y getter (encapsulamiento). Para esta clase desarrolle los **métodos abstractos heredados** con las siguientes características y que permita hacer lo siguiente:

- Retornar el sueldo base, método que llamara ingresos
- Calcular el descuento basado en: si la afiliación es AFP será el 15% del sueldo
- base sino será el 11% del sueldo base.
- No hay bonificación es decir la bonificación es 0.
- El sueldo neto se calculará como: $\text{ingresos} - \text{descuentos}$
- Además, recibirá asignación de movilidad si sus ingresos son menores de 1000 soles recibirá 50 soles sino recibirá 40 soles.

Considere los objetos necesarios para utilizar la funcionalidad de la clase.