

## 07 Ejercicios POO

**1.** Cree una clase llamada Empleado que incluya tres datos como variables de instancia: un nombre (String), un apellido (String) y un salario mensual (double). La clase debe tener un constructor que inicialice las tres variables de instancia. Proporciona un método set y un método get para cada variable de instancia. Si el salario no es positivo, ponlo a 0.0. Escriba una aplicación de prueba llamada EmpleadoTest que demuestre las capacidades de la clase Empleado. Cree dos objetos Empleado y muestre el salario anual de cada objeto. A continuación empleado un aumento del 10% y muestre de nuevo el salario anual de cada empleado.

**2.** Crear la clase SavingsAccount. Utiliza una variable estática annualInterestRate para almacenar el tipo de interés anual de todos los titulares de la cuenta. Cada objeto de la clase contiene una variable de instancia privada savingsBalance que indica la cantidad que el ahorrador tiene actualmente en depósito. El método calculateMonthlyInterest calcula el interés mensual multiplicando el saldo de ahorro por el tipo de interés anual dividido por 12. Este interés se añadirá al savingsBalance. Proporcionar un método estático modifyInterestRate que establezca el annualInterestRate a un nuevo valor.

Escriba un programa para probar la clase SavingsAccount. Instancie dos objetos SavingsAccount, ahorrador1 y ahorrador2, con saldos de \$2000.00 y \$3000.00, respectivamente. Establezca AnnualInterestRate en 4%, calcule el interés mensual e imprima los nuevos saldos de ambos. e imprime los nuevos saldos de ambos ahorradores. A continuación, establezca la tasa de interés anual en 5%, calcule el interés del mes siguiente e imprima los nuevos saldos de ambos ahorradores. del mes siguiente e imprime los nuevos saldos de ambos ahorradores.

**3.**

**a.** Cree una superclase llamada Coche. La clase Coche tiene los siguientes campos y métodos.

- int speed;
- double regularPrice;
- String color;
- double getSalePrice();

**b.** Cree una subclase de la clase Coche y nómbrala como Camión. La clase Camión tiene los siguientes campos y métodos.

- int weight;
- double getSalePrice();//Si weight>2000,10% de descuento. De lo contrario, 20% de descuento.

**c.** Cree una subclase de la clase Car y nómbrala Ford. La clase Ford tiene los siguientes campos y métodos

- int year;
- int manufacturerDiscount;
- double getSalePrice();//Al precio de venta calculado a partir de la clase Coche, réstale el descuento del fabricante (manufacturerDiscount).

**d.** Crear una subclase de la clase Coche y llamarla Sedan. La clase Sedan tiene los siguientes campos y métodos.

- int length;
- double getSalePrice(); //si length>20pies, 5%descuento, si no, 10%descuento.

**e.** Cree la clase MyOwnAutoShop que contiene el método main(). Realice lo siguiente dentro del método main().

◦Cree una instancia de la clase Sedan e inicialice todos los campos con los valores apropiados.

Utilice el método super(...) en

el constructor para inicializar los campos de la superclase.

◦Crea dos instancias de la clase Ford e inicializa todos los campos con los valores adecuados.

Utilice el método super(...)

en el constructor para inicializar los campos de la superclase.

◦Crea una instancia de la clase Coche e inicializa todos los campos con los valores adecuados.