

G1962 - Programación

Grado en Ingeniería Civil
Problemas 5

Javier González Villa
(19 de diciembre de 2025)

Licencia: Creative Commons BY-NC-SA 4.0 Internacional 

Programación orientada a objetos

Ejercicio 1:

Implementar una clase llamada **Punto**, la cual recibe como argumentos en su método constructor las coordenadas X e Y y las guarda en dos atributos de la clase. También redefinir el método `__eq__(self, obj)` dentro de la clase para permitir comparar dos objetos del tipo **Punto** mediante el operador `==` y que retorne `True` si ambas coordenadas de los puntos son iguales o `False` en caso contrario. Posteriormente probar que el método redefinido funciona correctamente.

Ejercicio 2:

Se quiere plantear de manera teórica el desarrollo de un programa de gestión de un concesionario. Para ello se pide identificar qué clases sería necesario implementar, qué atributos o métodos podrían incluirse en estas y qué relaciones de herencia o polimorfismo existirían.

Ejercicio 3:

Mediante la utilización de la clase **Punto** antes definida, se pide implementar una nueva clase llamada **Cuadrado**. Esta clase recibirá como argumentos en su método constructor cuatro objetos de la clase **Punto** y estos serán guardados en una lista, atributo de la clase, previamente inicializada vacía en la definición de la clase **Cuadrado**.

Ejercicio 4:

Dentro de la clase **Cuadrado** antes definida, se pide implementar o redefinir los siguientes métodos:

- `__str__(self)`: redefinirlo de manera que retorne una cadena de texto indicando las coordenadas del cuadrado.
- `area(self)`: retorna el área del cuadrado.
- `perimetro(self)`: retorna el perímetro del cuadrado.

Posteriormente crear un objeto de la clase **Cuadrado** y probar los métodos implementados o redefinidos para comprobar su correcto funcionamiento.

Ejercicio 5:

Mediante el uso de las propiedades de herencia y polimorfismo, implementar las siguientes clases con los atributos indicados y relacionarlos de manera que se optimice su uso:

- **PersonalUniversitario:** tiene como atributos nombre, primer apellido, segundo apellido, DNI, fecha de nacimiento y fecha de inicio de vinculación con la universidad.
- **Estudiante:** tiene como atributos nombre, primer apellido, segundo apellido, DNI, fecha de nacimiento, nota media de selectividad y fecha de inicio de sus estudios en la universidad.
- **Profesor:** tiene como atributos nombre, primer apellido, segundo apellido, DNI, fecha de nacimiento, asignaturas impartidas, departamento y fecha de inicio de su contrato.
- **Tecnico:** tiene como atributos nombre, primer apellido, segundo apellido, DNI, fecha de nacimiento, especialidad y fecha de inicio de su contrato.
- **Catedratico:** tiene como atributos nombre, primer apellido, segundo apellido, DNI, fecha de nacimiento, asignaturas impartidas, departamento, fecha de inicio de su contrato, responsabilidades adicionales y fecha de obtención de la cátedra.

Ejercicio 6:

Piensa y plantea de manera teórica un problema sencillo de tu elección. Tras esto diseña e implementa un sistema que involucre entre tres y cinco clases abordando el problema inicialmente planteado. Implementa sus relaciones y su lógica incluyendo los atributos y métodos que consideres oportunos para resolver dicho problema.