

Laboratorio 2 8/11

8 DE NOVIEMBRE DEL 2023 A LAS 8:10, SALA LCOMP2EG

INTRODUCCIÓN

Este listado de ejercicios se centra en la programación orientada a objetos en Python, un paradigma esencial en el desarrollo de software. A través de ejercicios prácticos, desde la creación de clases y objetos hasta la implementación de herencia y polimorfismo, te sumergirás en conceptos clave de la POO. Ya sea un principiante o un programador experimentado, este listado te desafiará a aplicar estos conceptos en situaciones reales, fortaleciendo tus habilidades en POO y preparándote para proyectos más complejos en Python. ¡Es hora de explorar el emocionante mundo de la programación orientada a objetos!

OBJETIVOS

Familiarizarse con ámbitos de desarrollo

1. Como ámbito de desarrollo utilizaremos el programa Sublime Text 4.
2. Para correr nuestro código de Python utilizaremos el intérprete y la línea de comandos.

Resolución de problemas

1. Proponer y analizar solución
2. Generación de datos aleatorios.
3. Proponer soluciones de carácter recursivo.

EJERCICIOS

1. Cree una clase llamada Rectangulo que contenga información sobre un rectángulo, incluyendo el largo y el ancho. Implemente métodos para calcular el área y el perímetro
2. Implemente una clase llamada Empleado que contenga información de un empleado, tal como su nombre, la posición en la que trabaja y el salario mensual. Implemente un método para darle aumentos de sueldo y otro para calcular su salario anual.

3. Cree una clase `Persona` que contenga información sobre una persona, incluido su nombre, edad, género. Implemente métodos para poder configurar estos atributos, además de un método para calcular su edad en edad de perros.
4. Cree una clase `CuentaBancaria` que contenga información bancaria de un usuario. Entre ellos debe estar el número de cuenta, balance, tipo de cuenta (cuenta corriente o de ahorro) y un objeto de la clase `Persona` con los datos del usuario. Implemente métodos para depositar y retirar dinero de la cuenta.
5. Cree una clase `Punto2D` donde tenga como propiedades la posición en el espacio de este punto en las dimensiones de X e Y . Su clase debe incluir el método `DistanciaEuclidiana(otro_punto2d)`. Que devuelve la distancia Euclidiana entre el punto y otro punto entregado por argumento.
6. Cree una lista de `Puntos2D` de N elementos aleatorios entre $(0,0)$ y $(20,20)$. Imprima la lista ordenada de forma ascendente con respecto a su distancia con el origen.

Ahora, extienda la clase del ejercicio anterior (mediante herencia) en una subclase que se llama `Círculo`. Esta clase `Círculo` deberá tener el atributo `radio` e implementar los métodos de `Area()`, `Perimetro()` y `Contiene(punto2d)`. Estos métodos tienen que devolver el área, perímetro y `True` o `False` si un un objeto `punto2d` está contenido dentro del círculo o no.

7. Cree una lista de N elementos que serán `Círculos` o `Rectángulos` (con igual probabilidad de ser uno o el otro), calcule el área total de todos los objetos de la lista y muéstrellos por consola. Luego ordene los objetos en función de su área, de menor a mayor.
8. Utilizando la clase `Círculo`, genere una lista de N `Círculos` aleatorios con radios entre 1 y 3, con su centro entre $(0,0)$ y $(10,10)$. Luego genere una lista de M `Puntos2D` aleatorios, entre $(0,0)$ y $(10,10)$. Construya e imprima un diccionario que relaciona el índice i del círculo en la lista con una lista que contenga los índices de todos los puntos que son contenidos por el círculo i .
9. Cree una clase que contenga información sobre un estudiante. La información requerida deberá ser el nombre, apellido paterno, apellido materno, rut, fecha de nacimiento (año-mes-día) y una nota (que representa su promedio). Luego, cree una lista de N estudiantes con

datos aleatorios e imprima una vez la lista por cada uno de los siguiente ordenes: a) Orden alfabético apellido paterno-apellido materno, b) Orden descendente del promedio y c) Orden ascendente de las edades.