

Enunciado sobre Clases Abstractas en Python

Ejercicio:

Diseña una clase abstracta `FiguraGeometrica` con las siguientes características:

- Atributos:
 - `nombre`: nombre de la figura (string)
- Métodos abstractos:
 - `calcular_area()`: calcula el área de la figura
 - `calcular_perimetro()`: calcula el perímetro de la figura

Requisitos:

1. Definir una clase abstracta `FiguraGeometrica`:
 - La clase debe ser abstracta utilizando el decorador `@abc.abstractmethod`.
 - Se debe definir el atributo `nombre` como privado.
 - Se deben definir los métodos abstractos `calcular_area()` y `calcular_perimetro()`.
2. Implementar clases concretas que hereden de `FiguraGeometrica`:
 - **Cuadrado:**
 - Atributos:
 - `lado`: longitud del lado del cuadrado (float)
 - Métodos:
 - Implementar `calcular_area()` y `calcular_perimetro()` para un cuadrado.
 - **Triángulo:**
 - Atributos:
 - `base`: longitud de la base del triángulo (float)
 - `altura`: altura del triángulo (float)
 - Métodos:
 - Implementar `calcular_area()` y `calcular_perimetro()` para un triángulo.

Ayuda:

- Para definir una clase abstracta, utiliza el decorador `@abc.abstractmethod`.
- Para definir un método abstracto, no se implementa el código dentro de la clase abstracta.
- Las clases que hereden de una clase abstracta deben implementar los métodos abstractos.