

## Actividad 3: POO - Herencia y polimorfismo

### Objetivo:

Implementar los conceptos de herencia y polimorfismo de la POO usando diagrama de clases UML y Java.

### Instrucciones:

Desarrolla un sistema para representar figuras geométricas como círculos, rectángulos y triángulos, con una clase base **FiguraGeometrica** y clases derivadas específicas para cada figura. Dibuje el diagrama UML que se proporciona abajo y agregue una nueva clase Triangulo con base en las indicaciones que se describen para esta clase. El sistema debe cumplir con lo siguiente:

- **Clase Base: Figura Geométrica**
  - **Atributos:** color (String, "blanco" por defecto), relleno (boolean, false por defecto), fechaCreacion (Date).
  - **Métodos:** Constructor para inicializar el color y relleno, getters y setters para color, relleno, y fechaCreacion, y un método toString que describa la figura.
- **Círculo**
  - **Atributo:** radio (double).
  - **Métodos:** Constructores para crear un círculo, métodos para calcular el área, perímetro y un toString que describa el círculo.
- **Rectángulo**
  - **Atributos:** base y altura (double).
  - **Métodos:** Constructores, getters/setters para base y altura, métodos para calcular el área, perímetro y un toString para describir el rectángulo.
- **Triángulo.** Agregue a su diagrama UML la clase Triangulo que extienda FiguraGeometrica. La clase contiene:
  - Tres **atributos** de datos **doubles** denominados **lado1**, **lado2** y **lado3** con valores predeterminados **1,0** para denotar tres lados de un triángulo.
  - Un constructor sin argumentos que crea un triángulo predeterminado.
  - Un constructor que crea un triángulo con los **lados1**, **lado2** y **lado3**.
  - Los métodos de acceso (getters) para los tres campos de datos.
  - Un método llamado **getArea()** que devuelve el área de este triángulo.
  - Un método llamado **getPerimeter()** que devuelve el perímetro de este triángulo.
  - Un método llamado **toString()** que devuelve una descripción de cadena para el triángulo.
- Escriba un programa de prueba con menú de opciones - 1) Círculo 2) Rectángulo 3) Triángulo 4) Salir – para realizar los calculos o acciones que considere. Por ejemplo: para la opción Triángulo: solicite al usuario que introduzca tres lados del triángulo, un color y un valor booleano para indicar si el triángulo está relleno. El programa debe crear un objeto *triangulo* con estos lados y establecer las propiedades de color y relleno usando la entrada. El programa debe mostrar el área, el perímetro, el color y verdadero o falso para indicar si está lleno o no.

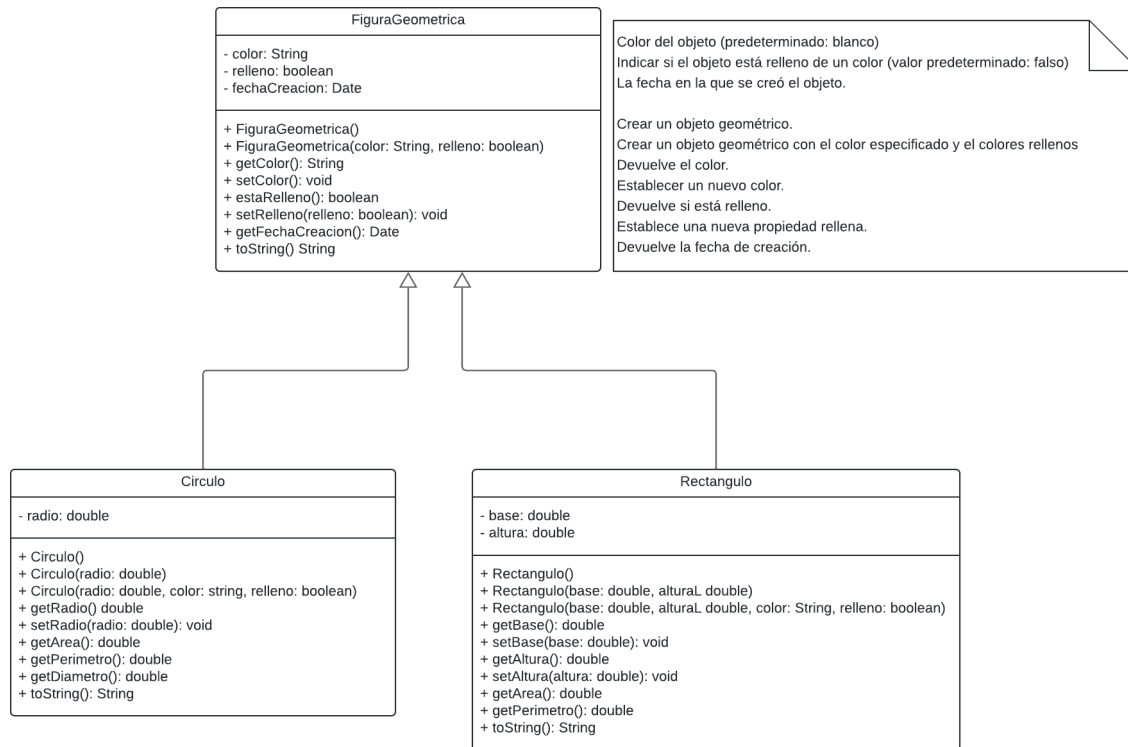


Figura 1. Diagrama UML, figuras geométricas

### Forma de entrega:

- Envíe el diagrama UML en formato PDF o imagen
- Comparta el enlace GitHub con la evidencia de la actividad
- Recuerden incluir los nombres de los estudiantes que participaron en la elaboración de esta Actividad 3