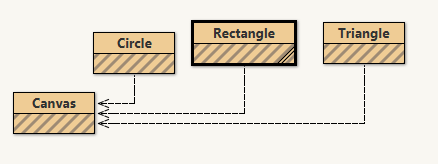
**Programación Orientada a Objetos – Lab No. 1**

**Federico Barrios -Cristian Ruíz**

**Shapes**

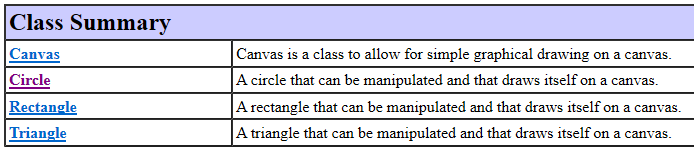
**2.** El diagrama de clases permite visualizar las clases de un artefacto software y las relaciones entre ellas. Considerando el diagrama de clases de “shapes”;

* **¿Qué clases ofrece?** El diagrama de clases de “shapes” ofrece cuatro diferentes clases que son:
  + Canvas
  + Rectangle
  + Triangle
  + Circle



* **¿Qué relaciones existen entre ellas?** Solo hay relación entre Circle-Canvas, Rectangle-Canvas y Triangle-Canvas

**3.** La documentación presenta las clases del proyecto y, en este caso, la especificación de sus componentes públicos. De acuerdo con la documentación generada:

* **¿Qué clases tiene el paquete shapes?** Según la documentación el paquete “shapes” ofrece las siguientes clases:
* **¿Qué atributos tiene la clase Circle?** Según la documentación la clase Circle tiene un único atributo que es PI
* **¿Cuáles métodos ofrece la clase Circle para que la figura cambie (incluya sólo el nombre)?** Según nuestro criterio, el hecho de que la figura cambie significa que la figura final es distinta a la figura inicial.
  + changeColor( )
  + changeSize( )

**4.** En el código de cada clase está el detalle de la implementación. Revisen el código de la clase Circle. Con respecto a los atributos:

* ¿Cuántos atributos realmente tiene?

Tiene 6 atributos:

* PI
* Diameter
* xPosition
* yPosition
* color
* isVisible
* ¿Cuáles son privados y cuáles públicos?

Los privados son:

* Diameter
* xPosition
* yPosition
* color
* isVisible

Los públicos son:

* PI

Con respecto a los métodos:

* ¿Cuántos métodos tiene en total?:

Tiene en total 14 métodos:

* changeColor()
* changeSize()
* makeVisible()
* makeInvisible()
* moveDown()
* moveHorizontal()
* moveLeft()
* moveRight()
* moveUp()
* moveVertical()
* slowMoveHorizontal()
* slowMoveVertical()
* draw()
* erase()
* ¿Cuáles son privados?
* draw()
* erase()
* ¿Quienes usan los componentes privados?
  + changeColor()
  + changeSize()
  + makeVisible()
  + makeInvisible()
  + moveHorizontal()
  + moveVertical()
  + slowMoveHorizontal()
  + slowMoveVertical()
  + draw()

**5.** ¿Qué no se ve en la documentación?

* No se ven los métodos privados, atributos privados ni tampoco el código

¿Por qué debe ser así?

* Según nuestro punto de vista, la documentación se recerva los atributos privados, métodos privados y el código ya que esto sería de uso o conocimiento exclusive del desarrollador.

**6.** En el código de la clase Circle revisen el detalle del atributo PI:

* ¿Qué se está indicándo? Se idica que la variable PI es estática, es decir que su valor no se va a modificar, también con double hace referencia a que es un decimal que ocupa 64 bits

**7.** ¿Cuál dirían es el propósito del proyecto “shapes”?

* Ver de manera aplicada cómo funciona una clase con sus métodos y atributos, y como interactuan para crear objetos.

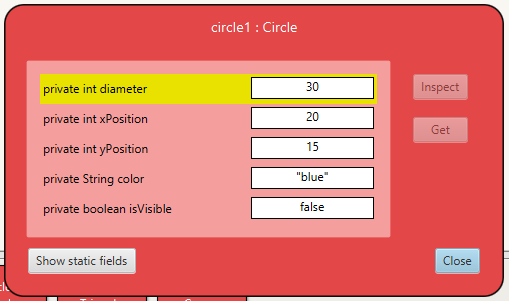
**Manipulando objetos. Usando opciones**

1. Creen un objeto de cada una de las clases que lo permitan.

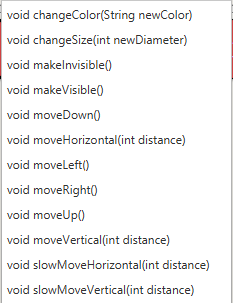


* + ¿Cuántas clases hay? Hay cuatro clases que nos permiten crear objetos, que son Canvas, Triangle, Rectangle y Circle.
  + ¿Cuántos objetos crearon? ¿Por qué? Cuatro, porque aunque los objetos de tipo Canvas son diferentes a los de las demás clases son entendidos como objetos por Java.

**2.** Inspeccionen el estado del objeto: Circle. ¿Cuáles son los valores de inicio de todos sus atributos?



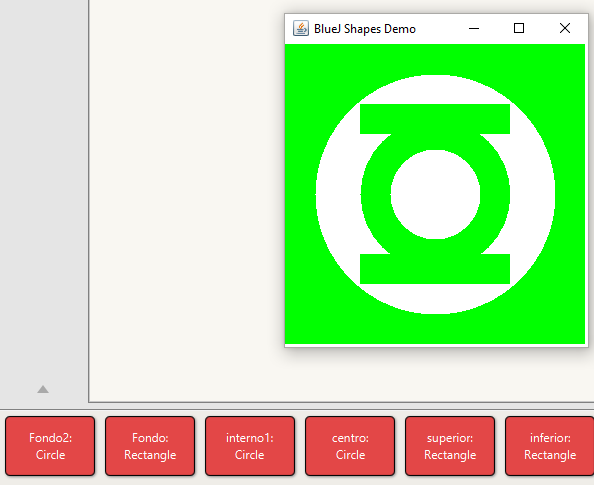
**3.** Inspeccionen el comportamiento que ofrece el objeto: Circle. Capturen la pantalla.

****

**¿Por qué no aparecen todos los que están en el código?**

Al inspeccionar el comportamiento del objeto faltan los métodos draw() y erase(), la razón por la cual no aparecen es que son métodos privados por lo cual no requieren interacción con el usuario.

**4.** Construyan, con “shapes” sin escribir código, una propuesta de la imagen de su comic favorito.





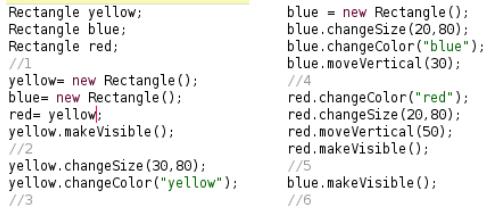
**¿Cuántas y cuáles clases se necesitan?**

Solo se necesitan las clases Circle y Rectangle

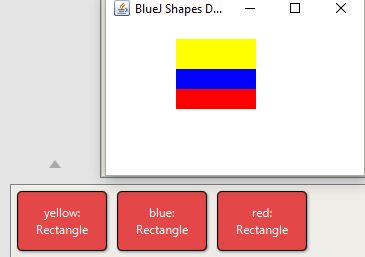
**¿Cuántos objetos se usan en total?**

En total se utilizaron 6 objetos

**Manipulando objetos. Analizando y escribiendo código.**



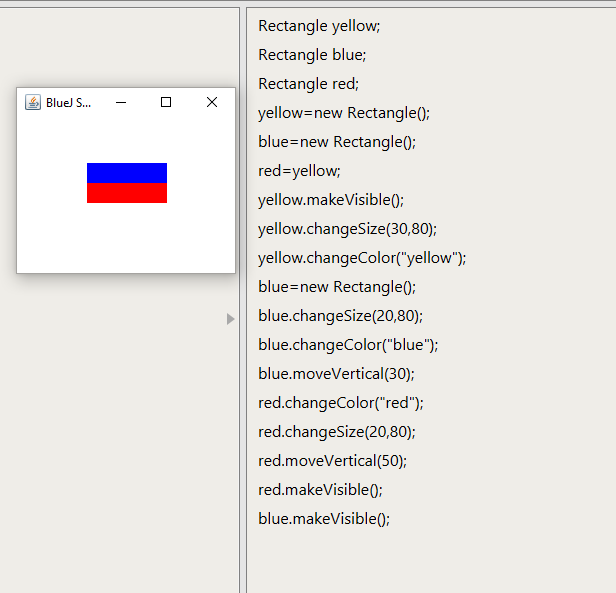
1. **Lean el código anterior ¿cuál es la figura resultante? Píntenla**



La figura resultante es la bandera de Colombia.

1. **Habiliten la ventana de código en línea, escriban el código y para cada punto señalado indiquen: ¿cuántas variables existen? ¿cuántos objetos existen? ¿qué color tiene cada uno de ellos? ¿cuántos objetos se ven? Expliquen. Capturen la pantalla.**

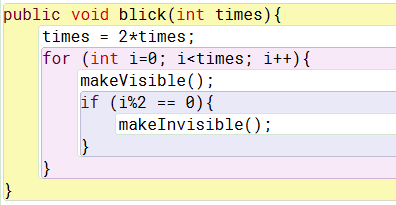
* ¿Cuántas variables existen? Existen tres variables que son yellow, blue y red.
* ¿Cuántos objetos existen? Existen tres objetos, pero solo dos de ellos están referenciados.
* ¿Qué color tiene cada uno de ellos? Los objetos que están referenciados son de color rojo y azul, y el objeto que perdió la referencia es de color magenta aunque no está visible.
* ¿Cuántos objetos se ven? Sólo se ven dos objetos que son el rectángulo rojo y el rectánguo azul. Esto se debe a que el rectángulo magenta no está visible.



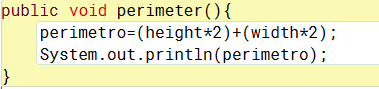
1. **Es la figura pintada en 1 igual a la figura capturada en 2 ¿por qué?**
   * No, porque en la figura 1 las variables yellow y red no referenciaban al mismo objeto, por lo que las operaciones realizadas en yellow no se vieron afectadas por las figuras realizadas sobre red.

**Extendiendo clases**

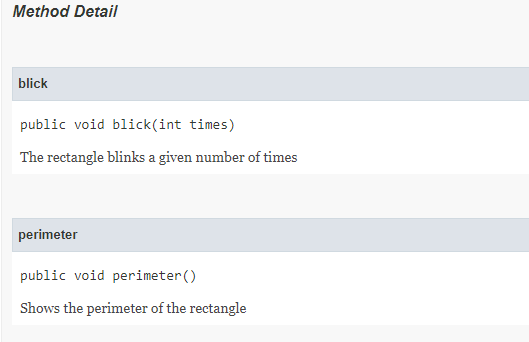
1. **. Desarrollen en Rectangle el método blick(times) (parpadea el número dado de veces). ¡Pruébenlo!**



1. **Desarrollen en Rectangle el método perimeter(). ¡Pruébenlo!**



1. **Desarrollen en Rectangle el método rotate() (que hace que rote a la derecha transladándose) . ¡Pruébenlo!**
2. **Generen nuevamente la documentación y revise la información de estos nuevos métodos. Capture la pantalla.**



**Retrospectiva:**

1. **¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes? (Horas/ Hombre)**

Cada uno de nosotros invirtió 8 horas.

1. **¿Cuál es el estado actual del laboratorio? ¿Por qué?**

Llegamos a desarrollar aproximadamente el 80% del laboratorio, esto se debe a que nos demoramos mucho planeando y diseñando la manera en la que íbamos a desarrollar la calculadora Maya y no pudimos ejecutar nuestras ideas.

1. **Considerando las prácticas XP del laboratorio. ¿cuál fue la más útil? ¿por qué?**

Programación por pares, porque nos ayudó a comprender la manera en la que vamos a trabajar el resto del semestre y en nuestra vida profesional.

1. **¿Cuál consideran fue el mayor logro? ¿Por qué?**

El mayor logro que conseguimos fue entender como se manejan las clases y objetos desde la práctica, reforzando los conceptos vistos en la clase teórica. Consideramos que este fue nuestro mayor logro ya que ninguno de los dos había tenido experiencia previa con la programación orientada a objetos y empezar a entender como funciona este paradigma es un gran logro.

1. **¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?**

Que en ocasiones no sabíamos el porqué de algunas cosas, para resolverlo consultamos en internet y buscamos la lógica que hay detrás de lo que no entendíamos.

1. **¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?**

Lo que hicimos bien como equipo fue complementar nuestras ideas y conocimientos ya que cuando uno tenía falencias en algo el otro podía ayudar y aclarar las dudad. Para mejorar nuestros resultados nos comprometemos a organizar mejor nuestro tiempo, ya que durante los días anteriores no estábamos muy ordenados y era difícil vernos para trabajar.