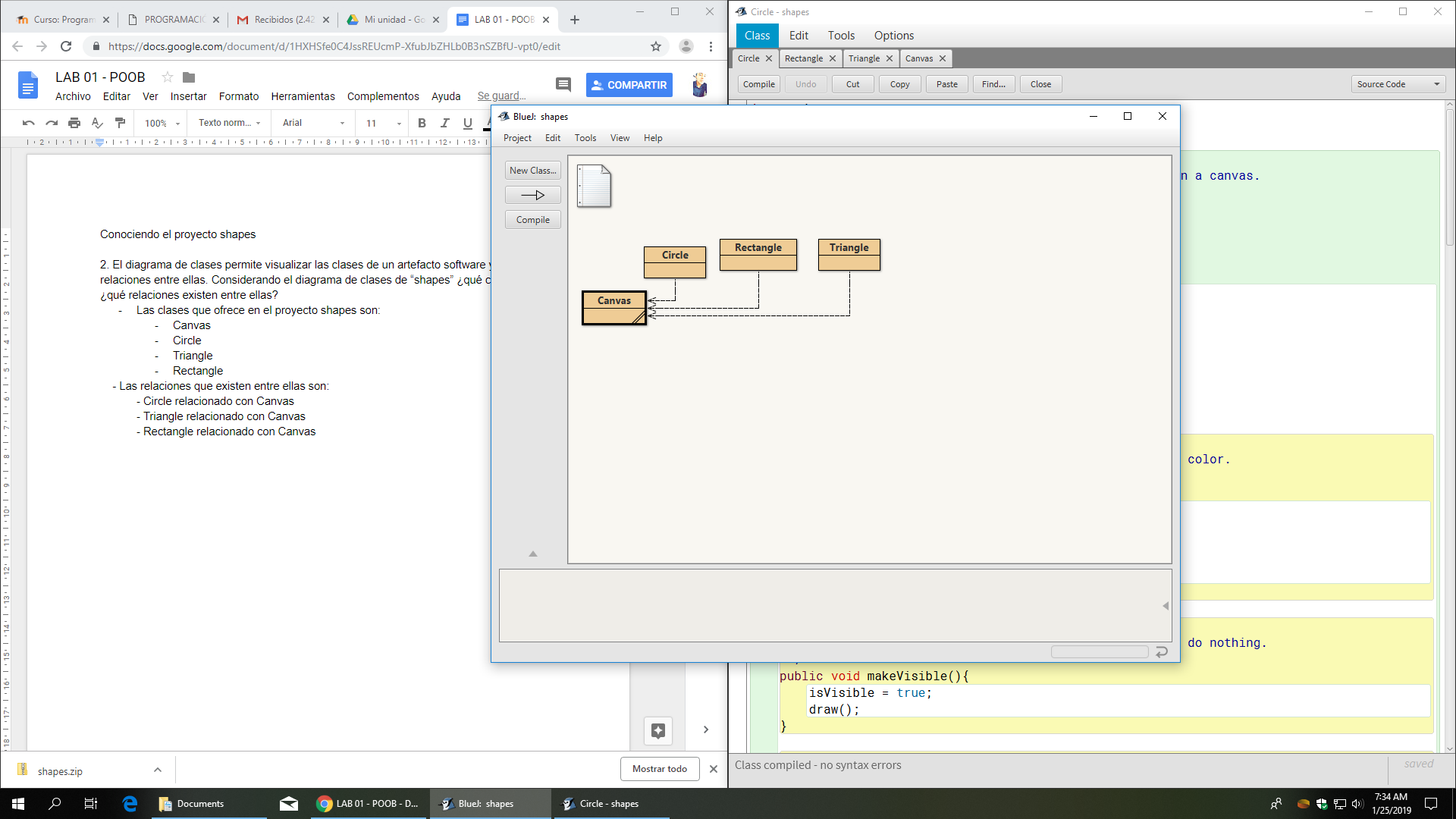
LAB 01 - POOB

* Conociendo el proyecto shapes

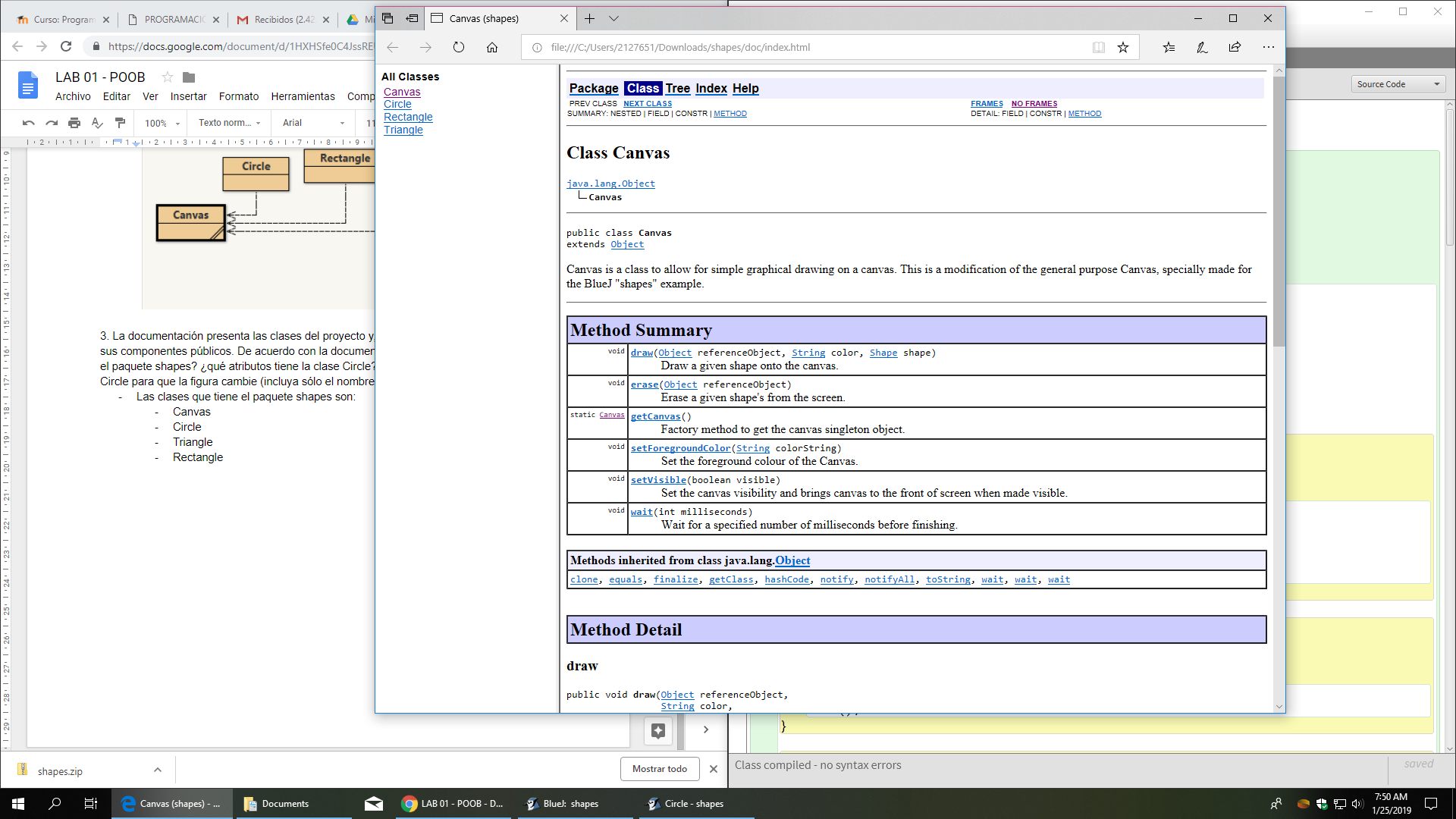
2. *El diagrama de clases permite visualizar las clases de un artefacto software y las relaciones entre ellas. Considerando el diagrama de clases de “shapes” ¿qué clases ofrece? ¿qué relaciones existen entre ellas?*

* Las clases que ofrece en el proyecto shapes son:
  + Canvas
  + Circle
  + Triangle
  + Rectangle
* Las relaciones que existen entre ellas son:
  + Circle relacionado con Canvas
  + Triangle relacionado con Canvas
  + Rectangle relacionado con 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? ¿qué atributos tiene la clase Circle? ¿cuáles métodos ofrece la clase Circle para que la figura cambie (incluya sólo el nombre)?*

* Las clases que tiene el paquete shapes son:
  + Canvas
  + Circle
  + Triangle
  + Rectangle



* Los atributos que posee la clase Circle son:
  + PI
* Los métodos para que la figura cambie son:
  + 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? ¿cuáles son privados y cuáles públicos?. Con respecto a los métodos: ¿cuántos métodos tiene en total? ¿cuáles son privados?. ¿Quienes usan los componentes privados?*

* Posee 5 atributos:
  + diameter
  + xPosition
  + yPosition
  + color
  + isVisible
* Atributos privados:
  + diameter
  + xPosition
  + yPosition
  + color
  + isVisible
* Atributos públicos:
  + PI
* Hay 14 métodos en total, los cuales 12 de ellos son públicos y 2 son privados.
  + Métodos privados:

1. draw
2. erase

* Los componentes privados los utilizan los siguientes métodos:
  + makeVisible()
  + makeInvisible()
  + moveHorizontal()
  + moveVertical()
  + slowMoveHorizontal()
  + slowMoveVertical()
  + changeSize()
  + changeColor()

5. *¿Qué no se ve en la documentación? ¿por qué debe ser así?*

No se ven los atributos o métodos que son privados, porque solo los puede utilizar la clase y nadie más.

6*. En el código de la clase Circle revisen el detalle del atributo PI. ¿qué se está indicándo?*

Es un elemento static , eso significa que todos los objetos creados comparten este atributo y que se podría modificar.

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

El propósito de shapes es modificar, graficar y mover distintos elementos como lo son el circulo, triangulo y cuadrado. También cambiar su color,su tamaño y hacerlos visibles.

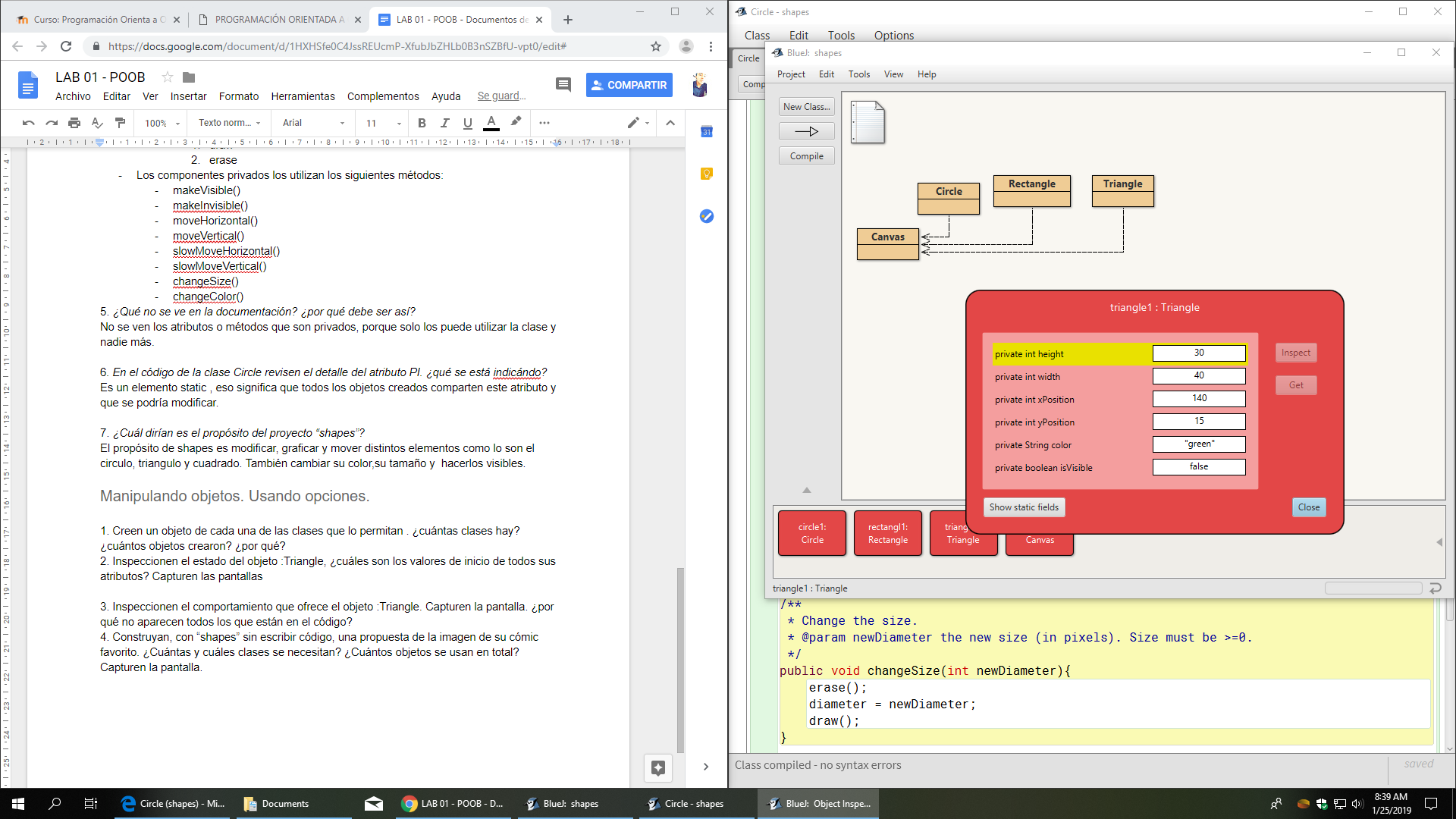
* Manipulando objetos. Usando opciones.

1. *Creen un objeto de cada una de las clases que lo permitan. ¿cuántas clases hay?*

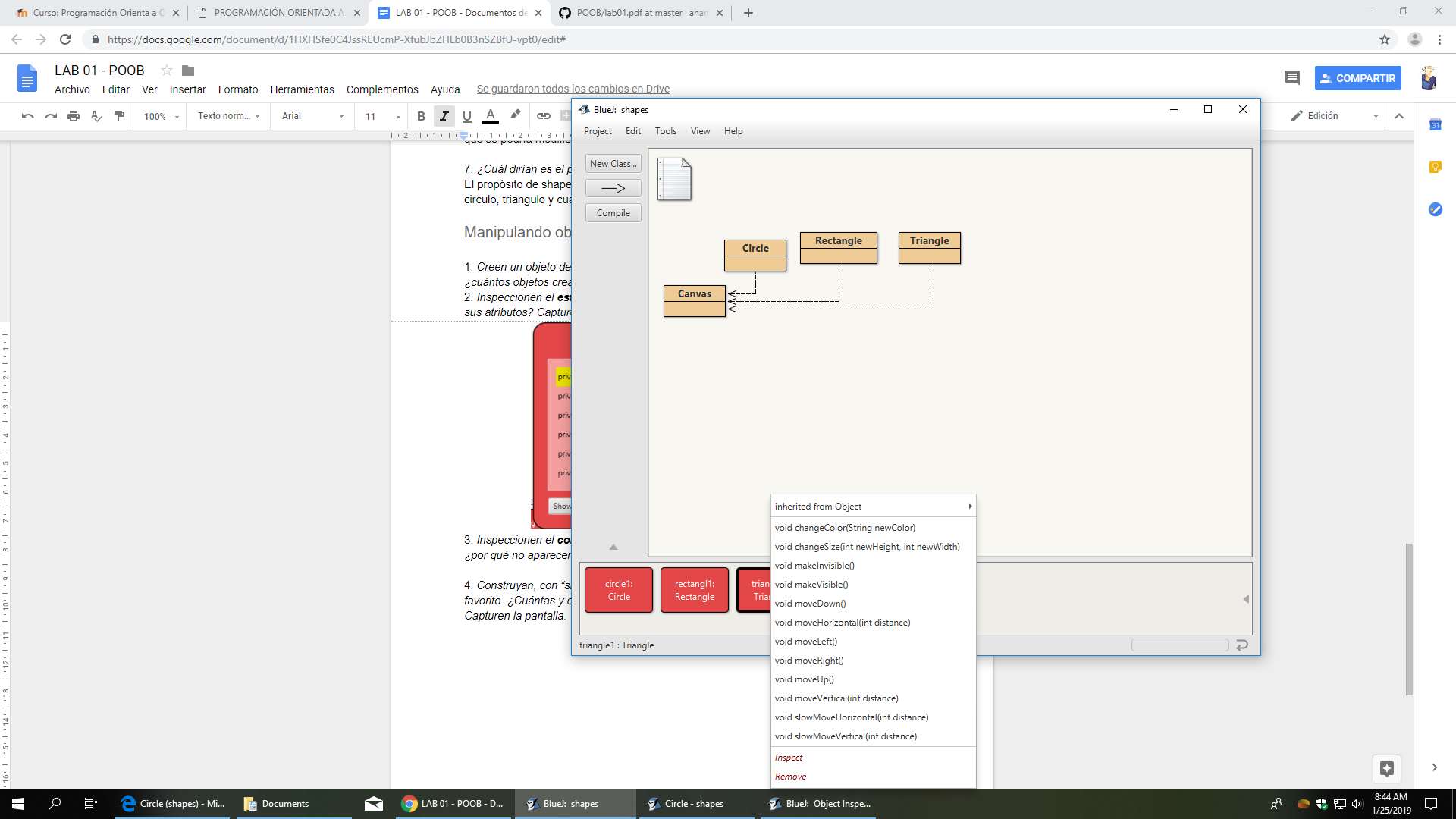
*¿cuántos objetos crearon? ¿por qué?*

Hay 4 clases, y nos dejaron crear solo 3 objetos. Ya que la clase canvas se usa para generar estos objetos.

2. *Inspeccionen el* ***estado*** *del objeto :Triangle, ¿cuáles son los valores de inicio de todos sus atributos? Capturen las pantallas*

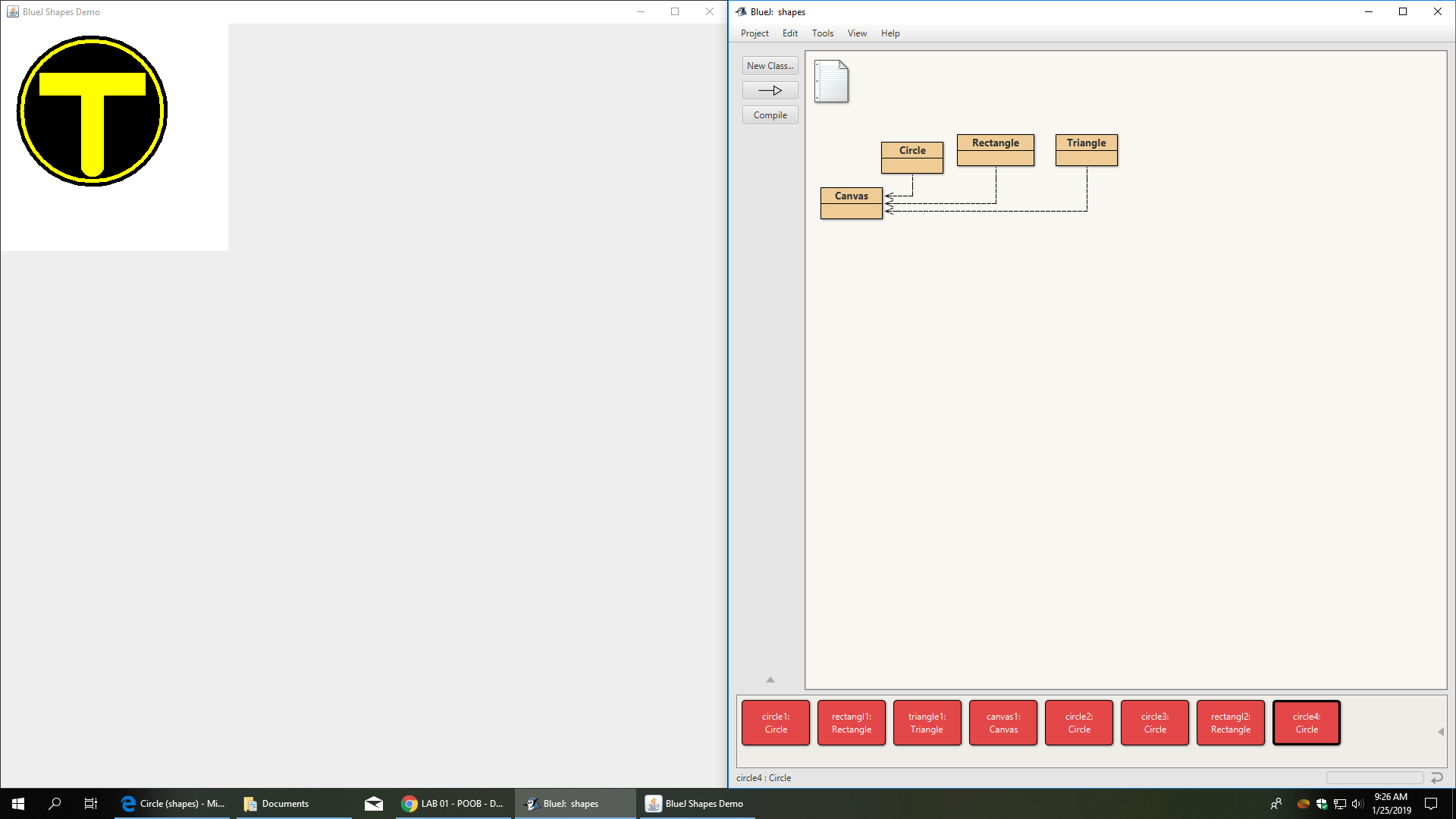


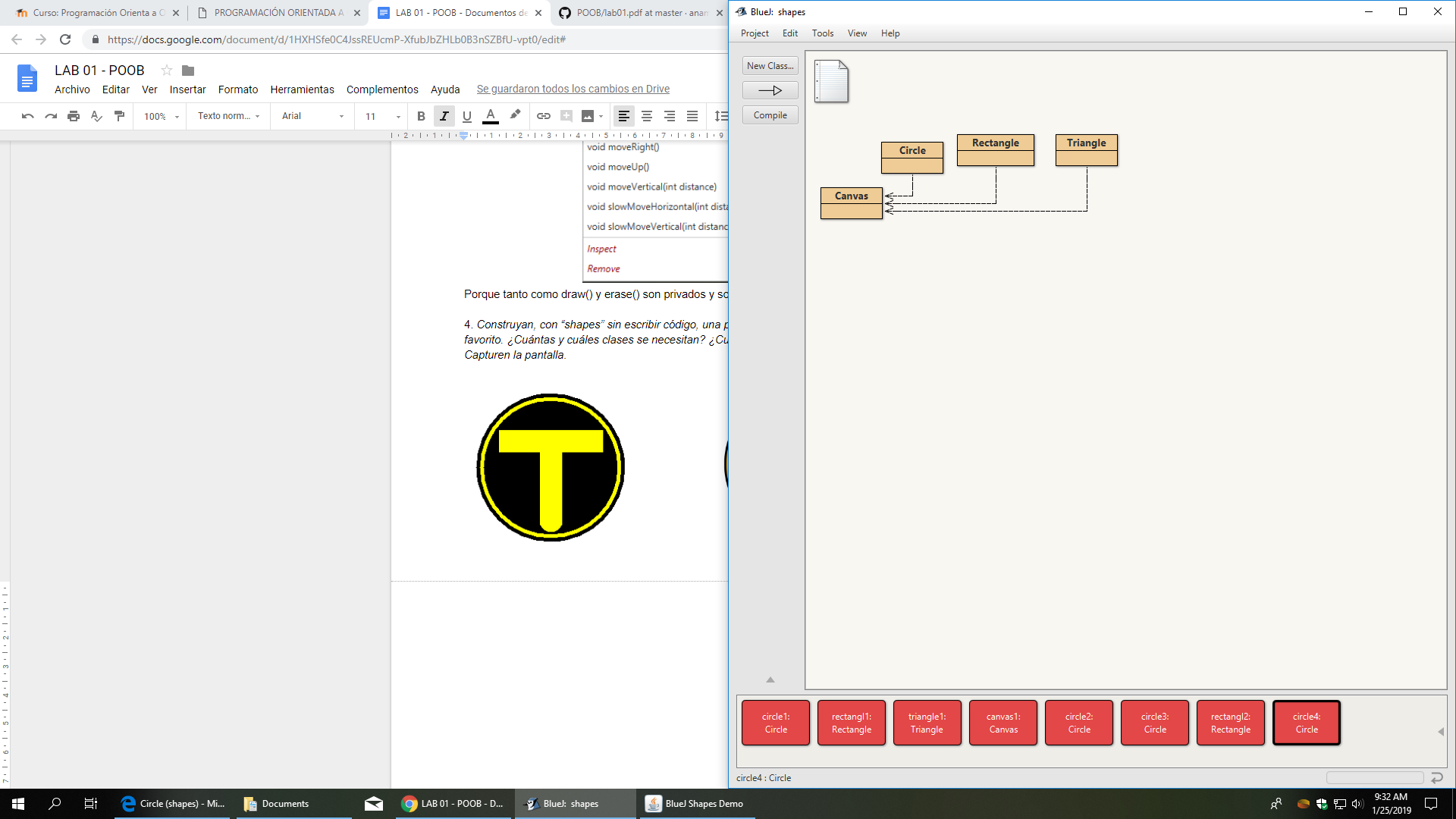
3. *Inspeccionen el* ***comportamiento*** *que ofrece el objeto :Triangle. Capturen la pantalla. ¿por qué no aparecen todos los que están en el código?*

**

Porque tanto como draw() y erase() son privados y solo los puede utilizar la clase Triangle.

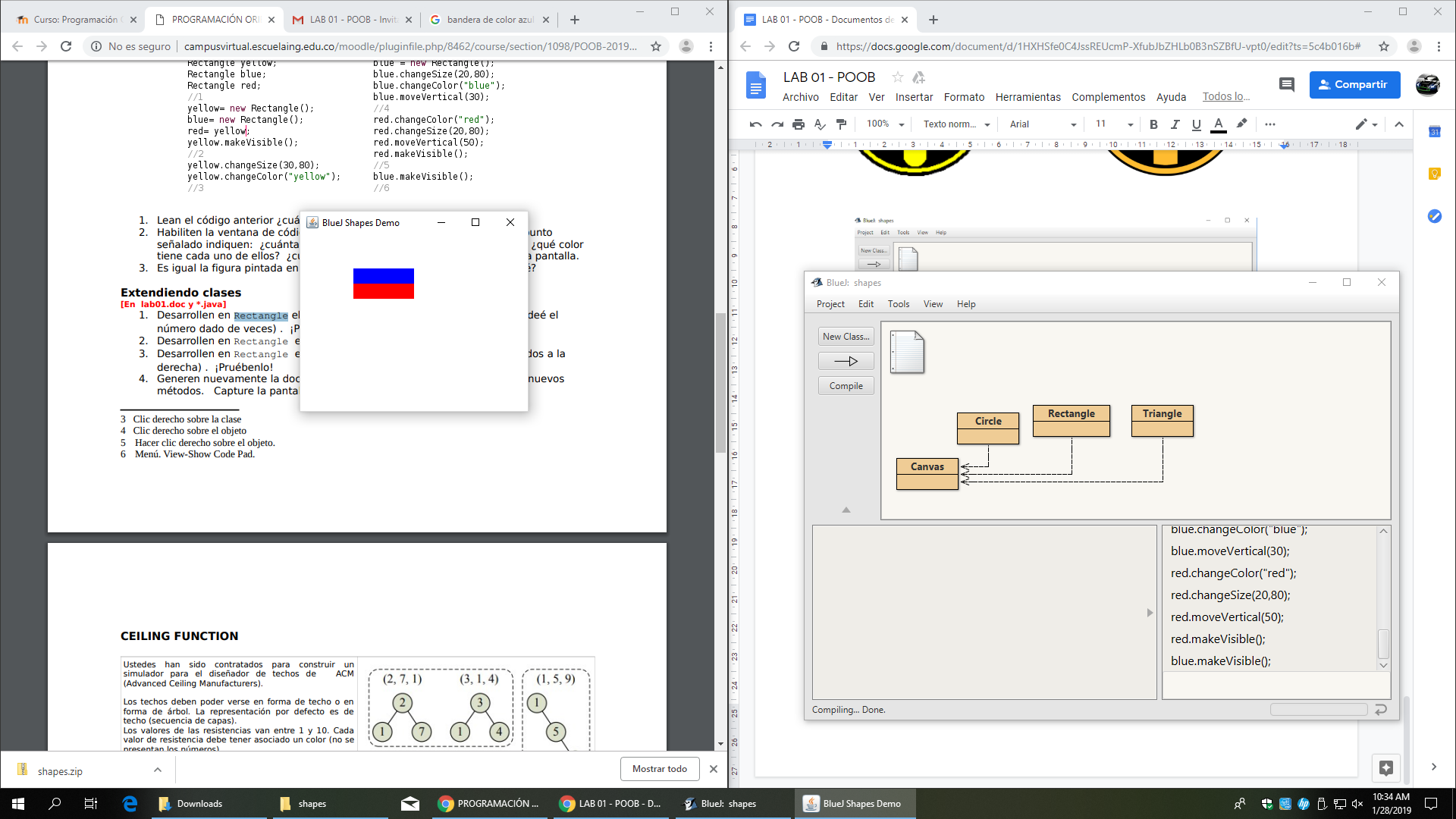
4. *Construyan, con “shapes” sin escribir código, una propuesta de la imagen de su cómic favorito. ¿Cuántas y cuáles clases se necesitan? ¿Cuántos objetos se usan en total? Capturen la pantalla.*



**

* Manipulando objetos. Usando opciones.

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



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

2.1

* ¿cuántas variable existen?

Son tres variables, las cuales las tres son de tipo rectángulo.

* ¿cuántos objetos existen?

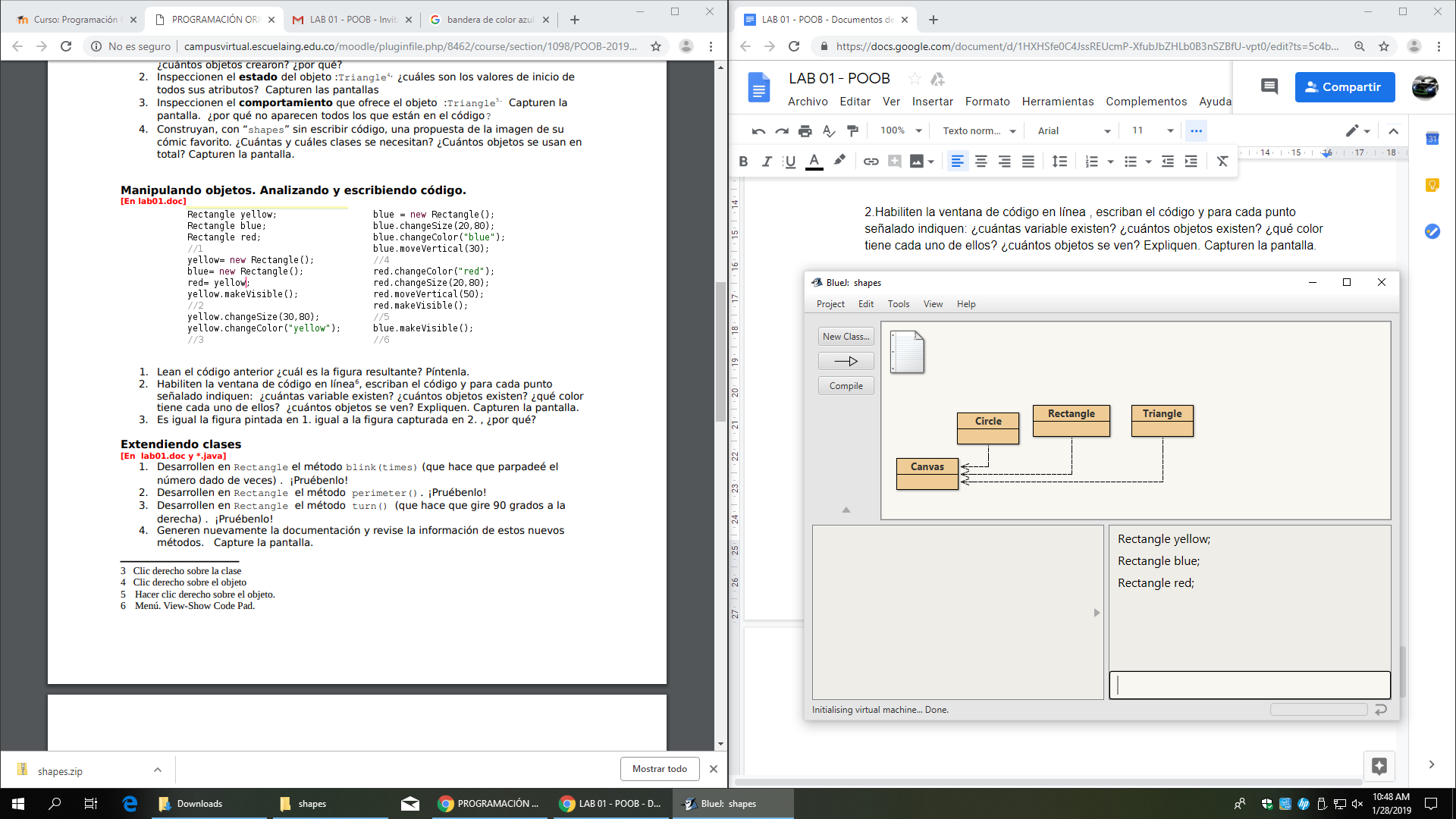
Ninguno.

* ¿qué color tiene cada uno de ellos?

Ninguno.

* ¿cuántos objetos se ven?

Ninguno.



2.2

* ¿cuántas variable existen?

Son tres variables, las cuales las tres son de tipo rectángulo.

* ¿cuántos objetos existen?

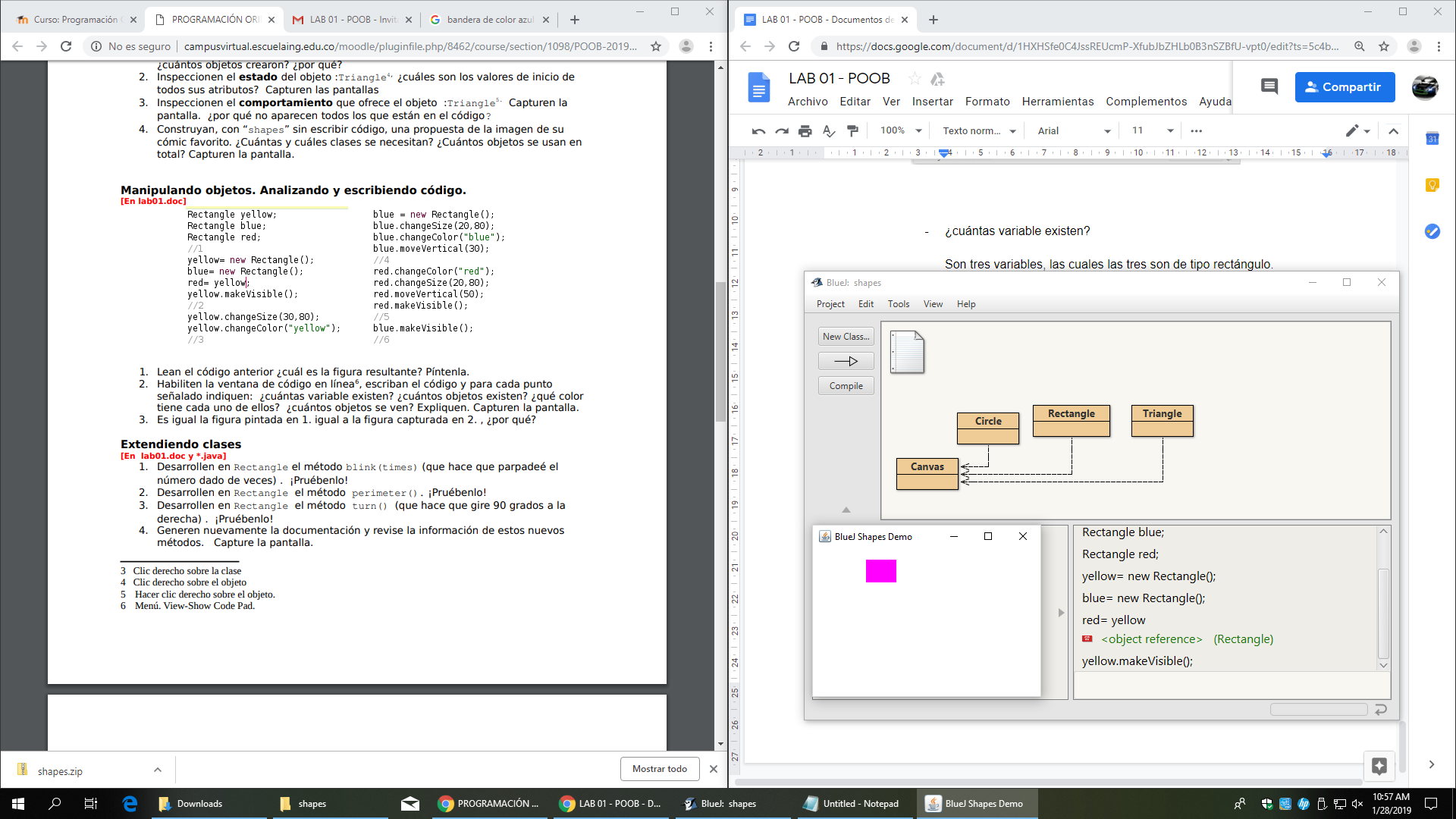
Dos objetos ya que red se considera que es uno solo con yellow y blue existe aunque no sea visible.

* ¿qué color tiene cada uno de ellos?

los dos objetos tienen color rosa.

* ¿cuántos objetos se ven?

Uno.



2.3

* ¿cuántas variable existen?

Son tres variables, las cuales las tres son de tipo rectángulo.

* ¿cuántos objetos existen?

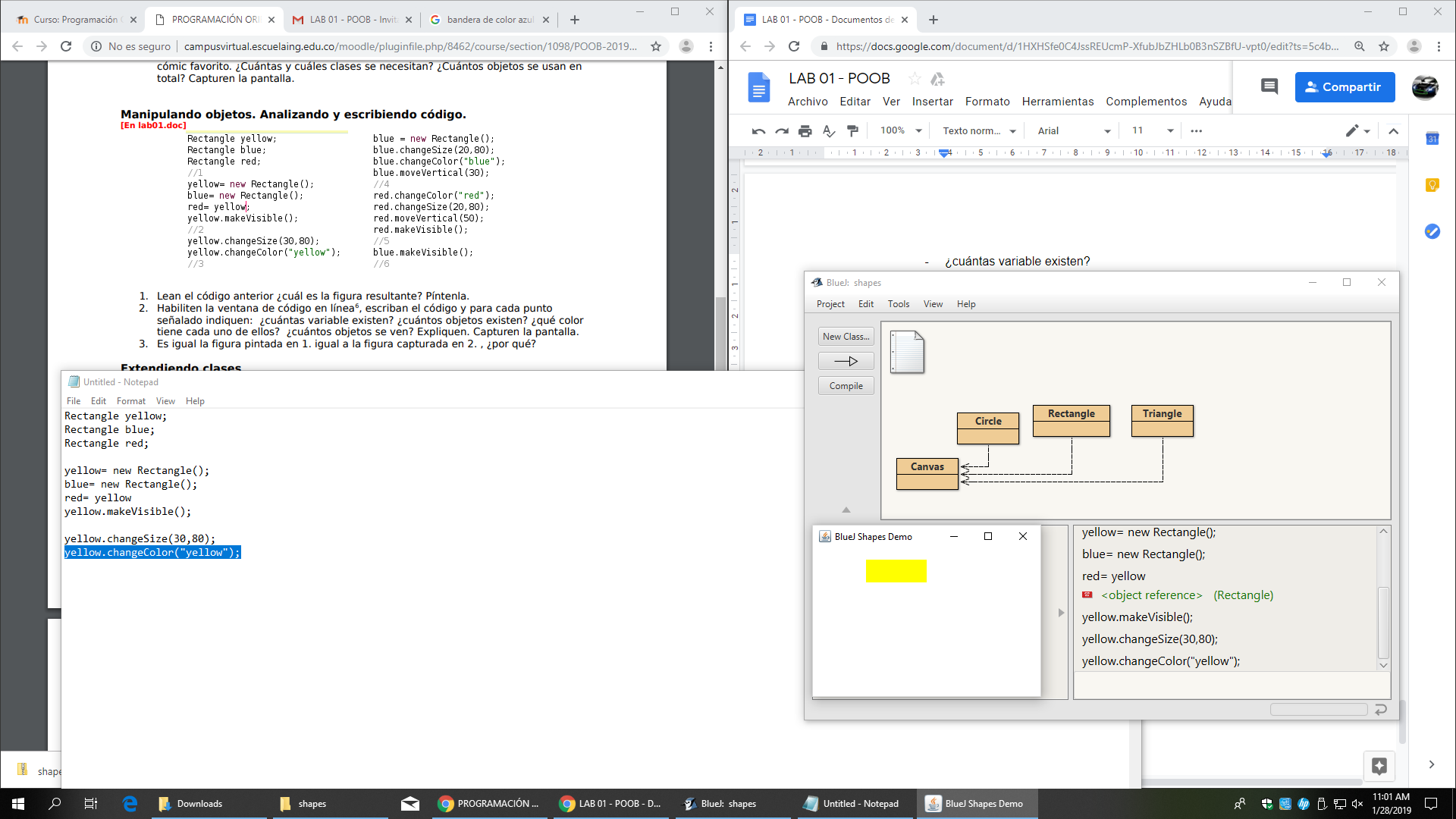
Dos objetos ya que red se considera que es uno solo con yellow y blue existe aunque no sea visible.

* ¿qué color tiene cada uno de ellos?

Los el objeto yellow y red son de color amarillo y el objeto blue de color rosa.

* ¿cuántos objetos se ven?

Uno.



2.4

* ¿cuántas variable existen?

Son tres variables, las cuales las tres son de tipo rectángulo.

* ¿cuántos objetos existen?

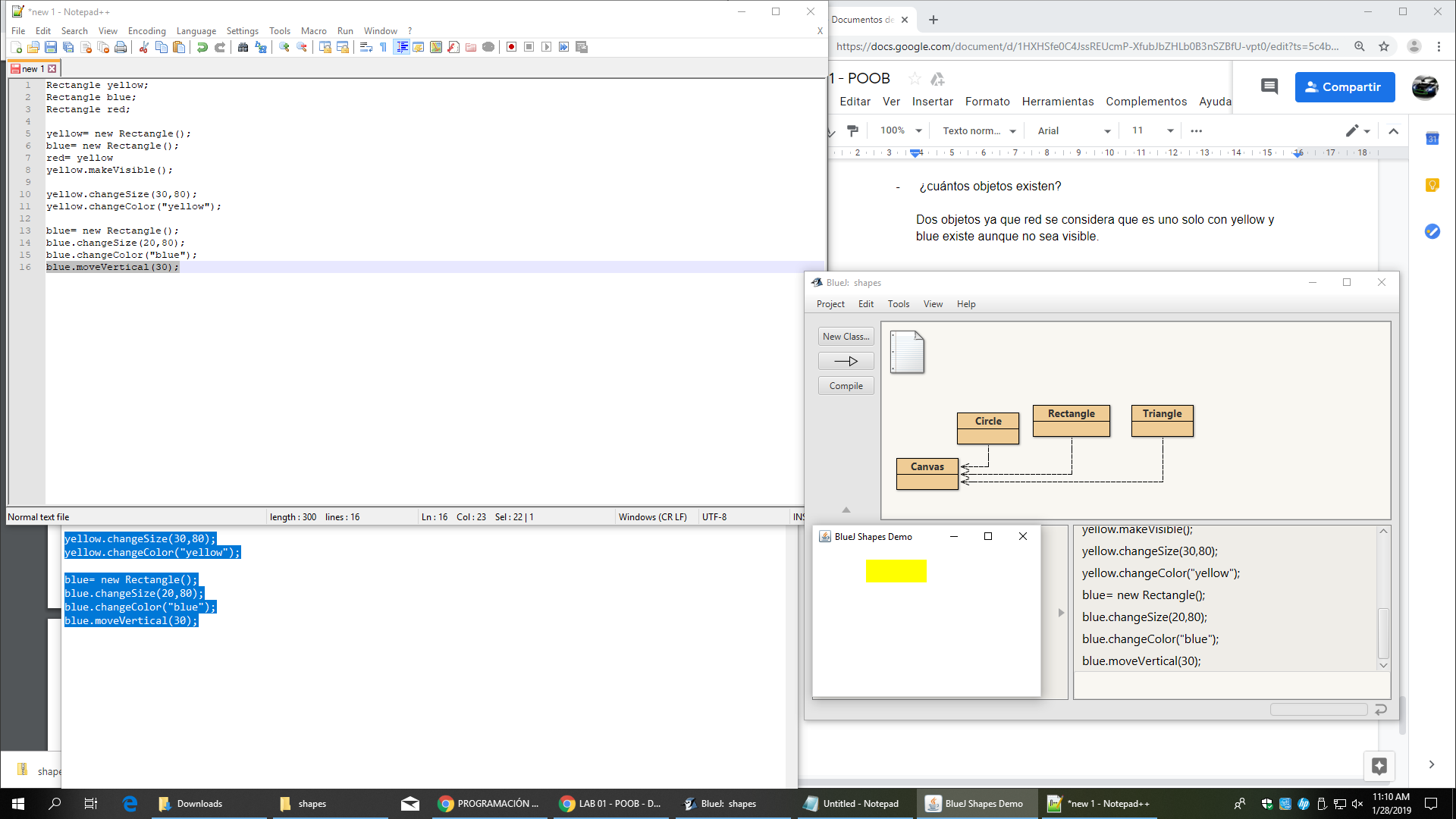
Dos objetos ya que red se considera que es uno solo con yellow y blue existe aunque no sea visible.

* ¿qué color tiene cada uno de ellos?

Los el objeto yellow y red son de color amarillo y el objeto blue de color azul.

* ¿cuántos objetos se ven?

Uno.



2.5

* ¿cuántas variable existen?

Son tres variables, las cuales las tres son de tipo rectángulo.

* ¿cuántos objetos existen?

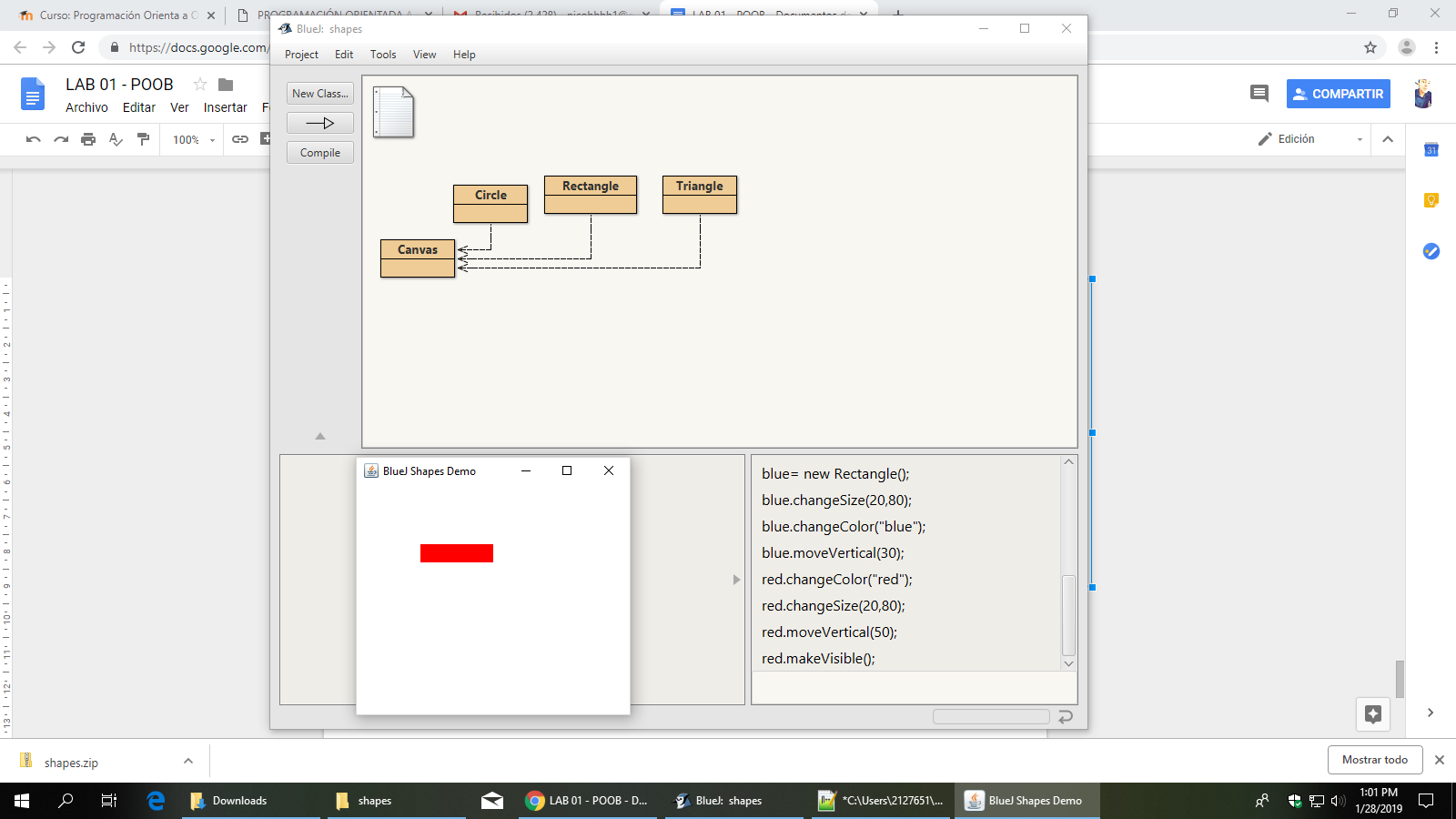
Dos objetos ya que red se considera que es uno solo con yellow y blue existe aunque no sea visible.

* ¿qué color tiene cada uno de ellos?

Los del objeto yellow y red son de color rojo ya que se realiza el metodo red.changeColor(“red”) y el objeto blue de color azul.

* ¿cuántos objetos se ven?

Uno.



2.6

* ¿cuántas variable existen?

Son tres variables, las cuales las tres son de tipo rectángulo.

* ¿cuántos objetos existen?

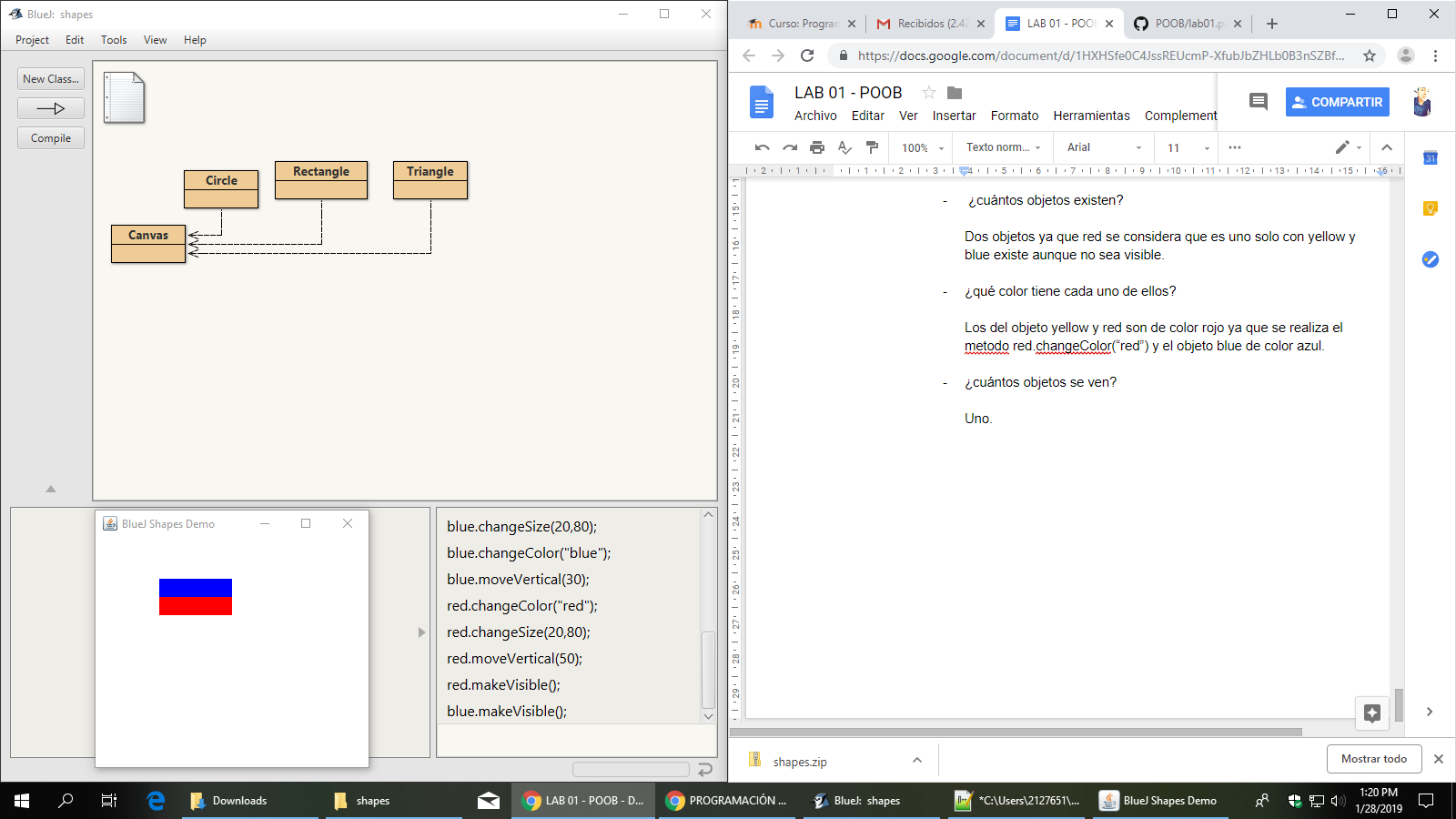
Dos objetos ya que red se considera que es uno solo con yellow, blue es un objeto más que ahora es visible.

* ¿qué color tiene cada uno de ellos?

Los del objeto yellow y red son de color rojo ya que se realiza el metodo red.changeColor(“red”) y el objeto blue de color azul.

* ¿cuántos objetos se ven?

Dos.

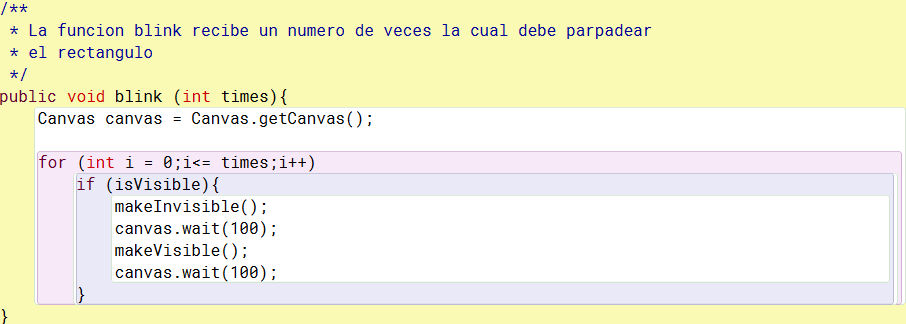


3. Es igual la figura pintada en 1. igual a la figura capturada en 2. , ¿por qué?

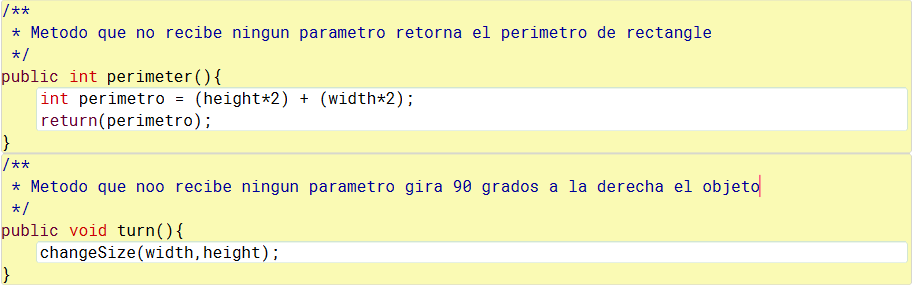
Si , ya que la igualdad red=yellow nos indica, que lo que le suceda al objeto red le sucederá al objeto yellow, ademas tenemos nuestro objeto blue que se ubica en la parte de arriba de red, tal cual vimos en la figura 1.

* Extendiendo clases.

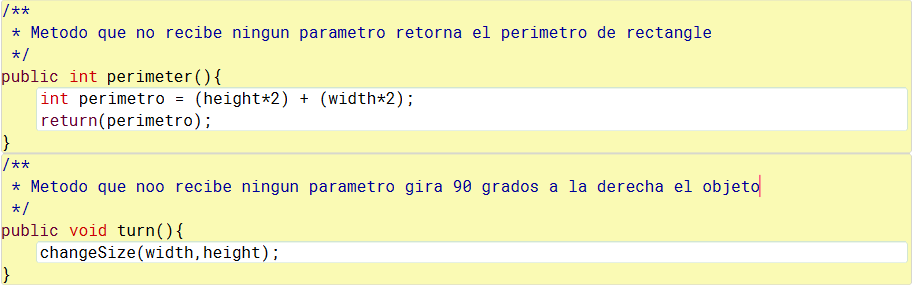
1. Desarrollen en Rectangle el método blink(times) (que hace que parpadeé el número dado de veces) . ¡Pruébenlo!



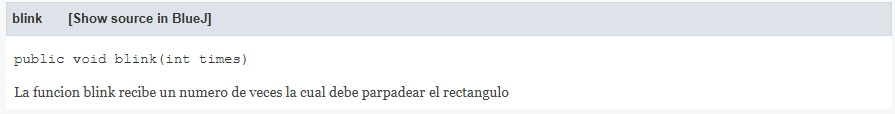
2. Desarrollen en Rectangle el método perimeter(). ¡Pruébenlo!

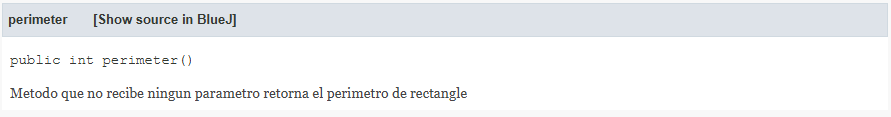


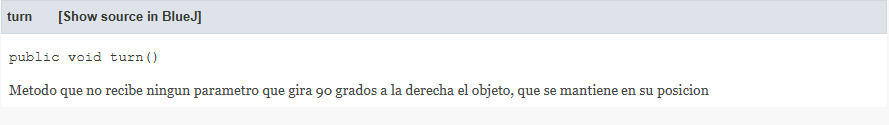
3. Desarrollen en Rectangle el método turn() (que hace que gire 90 grados a la derecha) . ¡Pruébenlo!



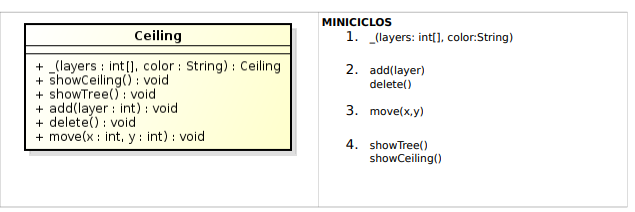
4. Generen nuevamente la documentación y revise la información de estos nuevos métodos. Capture la pantalla.







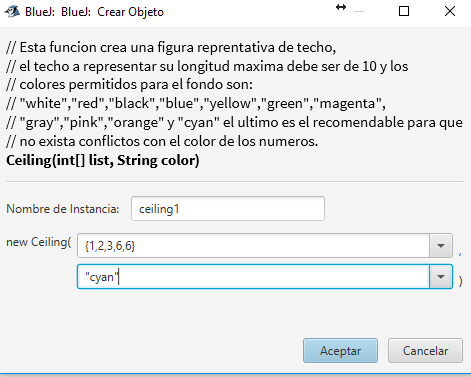
* Implementando una nueva clase. Ceiling.

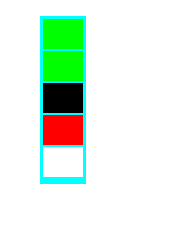


1. Revisen el diseño y clasifiquen los métodos en: constructores, analizadores y modificadores.

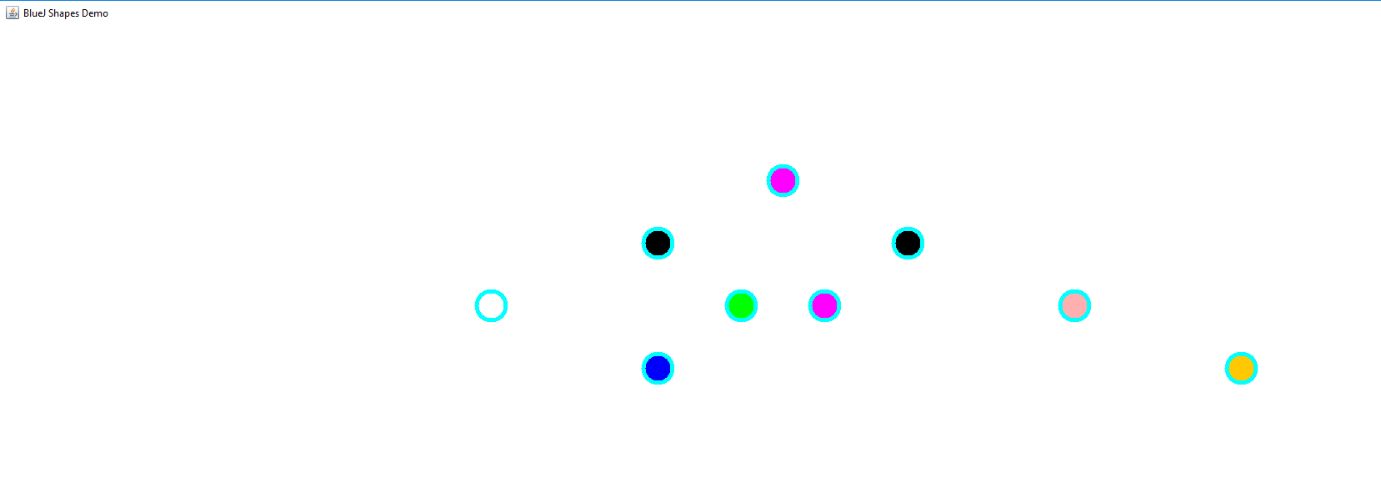
El constructor el (layers: int[], color, String), los analizadores son showCeiling() y showTree(); y los modificadores son add(layer:int), delete() y move(x :int y: int).

2. Desarrollen la clase Ceiling considerando los mini-ciclos. No olviden la documentación. Al final de cada mini-ciclo realicen una prueba indicando su propósito. Capturen las pantallas relevantes.





showCeiling()

showTree()

* Retrospectiva

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

30 horas de trabajo en total.

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

Incompleto, porque se llego hasta la parte de CeilingManager y se comenzó a tener errores de tipos en cuanto a la manipulación de datos. Algunos puntos fueron mas complejos de entender y otros de realizar.

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

Nos parece que la más útil es la programación en parejas, ya que de esta forma se pueden evitar muchos errores y hay una mejor comunicación y fluidez en cuanto al desarrollo de ideas para la solución del problema.

4. ¿Cuál consideran fue el mayor logro? ¿Por qué?

El mayor logro fue poder establecer el árbol con la jerarquía que se pedía, que depende al dato tomara un orden diferente.

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

Uno de los problemas técnicos fue el ordenamiento de las posiciones en el árbol binario ya que cada una debe cumplir un patrón que muestre un orden acorde a la imagen de padre e hijo, y también la creación de líneas que conectan a cada nodo del árbol. Se pudo solucionar el ordenamiento y la creación del árbol lo cual tomo mucho tiempo en su elaboración incluyendo la unión de este método con el ordenamiento del árbol, pero la representación de las líneas no pudimos realizarlas por falta de conocimiento de líneas en 2d.

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

Nosotros diríamos que la comunicación fue clave para la elaboración de este laboratorio, desempeñamos un buen papel como equipo ayudando al otro en cuando a sus debilidades para la elaboración del programa. Nos comprometemos a dedicarle más tiempo al laboratorio y mejorar el nivel de conocimiento del lenguaje (JAVA).