**Parte A**

1. Tareas a realizar en el laboratorio
2. Abre el proyecto **s*hapes*** con BlueJ
3. Pulsa el botón derecho sobre la caja **Circle**, selecciona **new Circle()** en el menú desplegable y pulsa ***Ok***. Acabas de crear (***instanciar***) tu primer ***objeto.*** La referencia al objeto se guarda en la variable *circle1* y se coloca en el banco de objetos (*object bench*) de BlueJ para su uso posterior.

* Ahora, pulsa el botón derecho del ratón sobre el objeto circulo creado (*circle1*, en el banco de objetos) y selecciona ***Inspect*** para invocar al ***Inspector de Objetos (Object Inspector)***. Los campos mostrados son los ***atributos.*** Los atributos describen las características de un objeto. El conjunto de valores que tienen asignados los atributos de un objeto determinan su ***estado***.
* Cuando pulsas el botón derecho del ratón sobre un objeto, además del ***Inspect*** se muestran todos sus ***métodos (comportamientos)***. Los métodos permiten ejecutar acciones (p.ej. mover) y/o modificar el estado del objeto:
  + ¿Qué valor tiene isVisible? \_\_\_\_\_
  + Pulsa el botón derecho sobre el objeto y selecciona makeVisible(). ¿Qué sucede?
  + ¿Cuál es la posición (x, y) del círculo? \_\_\_\_\_\_\_
  + Pulsa el botón derecho sobre el objeto y selecciona moveDown().
  + ¿Cuál es la nueva posición (x, y) del círculo? \_\_\_\_\_\_\_
  + ¿Qué crees que pasará si ***invocas*** el método (es decir, ***pase de mensaje***) moveDown() de nuevo? Indica el resultado sin ejecutarlo \_\_\_\_\_
  + ¿Has acertado? \_\_\_\_\_
  + Invoca el método moveRight() del círculo. ¿Cuál es la nueva posición (x, y) del círculo? \_\_\_\_\_\_\_

1. Crea un nuevo objeto de la clase **Circle** (sigue las instrucciones del ejercicio anterior).

* Pulsa el botón derecho del ratón sobre el objeto y selecciona *Inspect* para invocar al *Inspector de Objetos*.
* ¿Por qué no puedes ver el círculo en el dibujo? \_\_\_\_\_\_\_
* Haz que sea visible
* Cambia el color del círculo (p.ej. “*red”*).
* ¿Qué ocurre si invocas el método changeColor(), y escribes un color *sin* comillas en el cuadro de texto? \_\_\_\_\_\_\_
* ¿Qué ocurre si especificas un color desconocido? \_\_\_\_\_\_\_
* Cambia el tamaño del diámetro a 10.
* Pásale el mensaje de moveVertical el valor de 10 pixeles. ¿Qué ha ocurrido con su *estado*, exactamente con el atributo yPosition? \_\_\_\_\_\_\_
* ¿Cómo puedes hacer que el círculo se desplace hacia arriba 25 píxeles? \_\_\_\_\_\_\_
* Mueve el círculo de manera que quede en la esquina superior izquierda del lienzo (*canvas*).

1. Crea un nuevo objeto de la clase **Square**. Invoca al *Inspector de objetos,* haz visible el objeto y cambia su color a azul

* Muévelo a la esquina superior derecha del lienzo, ¿cómo? \_\_\_\_\_\_\_
* Pon el cuadrado en el centro del lienzo, ¿cómo? \_\_\_\_\_\_\_
* Mueve lentamente el cuadrado 50 píxeles a la derecha con el método slowMoveHorizontal().

1. Crea un objeto de la clase **Triangle**. Abre el *Inspector de Objetos* para cada figura.

* Escribe los nombres de los atributos que los 3 tipos de objetos tienen en común: \_\_\_\_\_\_\_
* Escribe los nombres de los atributos propios de cada figura
* ¿Qué métodos distinguen el comportamiento del triángulo de las otras figuras?

1. Utilizando el proyecto ***shapes*** y el menú contextual de *BlueJ* para el **paso de mensajes** a los objetos:
   1. Abre la ventana de terminal de BlueJ (*Ver→Mostrar Terminal*) y activa la opción de registro (Opciones*→* Registro de llamadas a método). Si es necesario, limpia lo que tengamos de ejecuciones previas (*Opciones→ Limpiar*)
   2. Crea un **triángulo** y modifícalo de manera que la base cubra el borde inferior completo del lienzo y el vértice superior esté en el borde superior del lienzo. Anota tanto la **instanciación** **de los objetos** (es decir, creación de las figuras) como el **pase de mensajes** a los objetos (es decir, la invocación de sus métodos)

|  |
| --- |
| **Objetos Instanciados** |
| *Formato: TipoDatos variable= new Constructora(parámetros);*  *Ejemplo:* ***Circle circle1= new Circle();*** |
| **Pase de Mensajes** |
| *Formato: referenciaObjeto.método(parámetros);*  *Ejemplo :* ***circle1.moveRight();*** |

* 1. Abre la clase **Demo** con el editor de código. Copia en el método *exercise6b()* las instrucciones ejecutadas en el ejercicio 6b. Compílalo y pruébalo.

**Tareas complementarias:**

1. Utilizando el proyecto ***shapes*** y el menú contextual de *BlueJ* para el **paso de mensajes** a los objetos:
   1. Limpia lo que tengas de ejecuciones previas en la ventana de terminal de BlueJ (*Opciones→ Limpiar*)
   2. Pon un cuadrado (**Square**) de tamaño 10x10 en el origen de coordenadas y desplázalo a lo largo del borde superior del lienzo. Cuando llegue a la esquina superior derecha, desplázalo a la esquina inferior derecha del lienzo. Anota tanto la **instanciación** **de los objetos** (es decir, creación de las figuras) como el **pase de mensajes** a los objetos (es decir, la invocación de sus métodos)

|  |
| --- |
| **Objetos Instanciados** |
|  |
| **Pase de Mensajes** |
|  |

* 1. Abre la clase **Demo** con el editor de código. Copia en el método *exercise7b()* las instrucciones ejecutadas en el ejercicio 7b. Pon en el método main una llamada al método *exercise7b()*. Compílalo y pruébalo.