Game and Animation Loop

Let's setup a JFrame and JPanel as in previous tutorial

```
In [ ]: import java.awt.*;
        import javax.swing.*;
```

```
In [ ]: JFrame frame = new JFrame("My Frame");
        frame.setPreferredSize(new Dimension(500, 500));
        frame.setResizable(false);
        frame.pack();
        frame.setVisible(true);

        JPanel panel = new JPanel();
        panel.setPreferredSize(new Dimension(500, 500));

        frame.add(panel);
        frame.pack();

        frame.setIgnoreRepaint(true);
        panel.setIgnoreRepaint(true);
```

Currently our JFrame's content is rendered once and thus static.

We need a way for the program to **repeatedly** update its state and then draw (or render) the new state to the screen (graphics object of frame/panel). We thus idealy want an `update()` function and a `draw(Graphics2D g2)` function.

```
Create a loop that wil repeatedly run and then sleep for 50ms.
```

```
In [ ]: while (true) {

            // code goes here

            Thread.sleep(50);
        }
```

Now add a functions for updating and drawing. For the moment we will be accessing these method from a static main method, therefore these functions need to be static as well. Implement them in the game loop.

```
In [ ]: static void update() {

            // updates to game state goes here

        }
```

```
In [ ]: static void draw(Graphics2D g2) {

            // drawing now code goes here

        }
```

```
In [ ]: while (true) {

            // update game parameters
            update();

            // draw current state to graphics object
            Graphics2D g2 = (Graphics2D) panel.getGraphics();
            draw(g2);
            g2.dispose();

            Thread.sleep(50);
        }
```

Now lets animate something. e.g. A circle moving from top left to bottom right of screen.

Note the coordinate system used by java.awt and javax.swing libraries
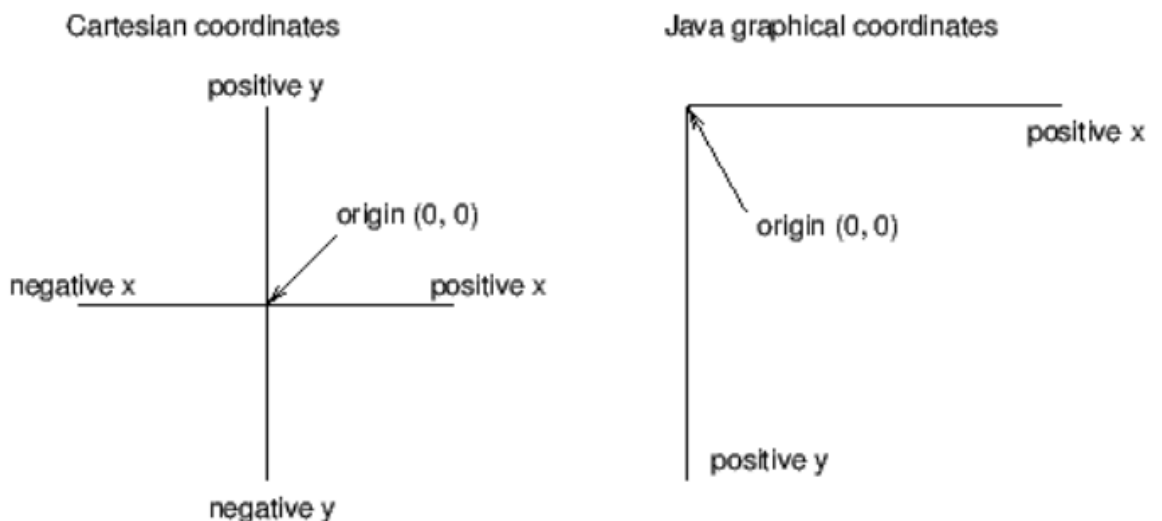

Image source: http://www.greenteapress.com/thinkapjava/html/thinkjava019.html (http://www.greenteapress.com/thinkapjava/html/thinkjava019.html)

Define initial position x, y of circle as static global variables in Tut2 class

```
In [ ]: static int x = 0;
        static int y = 0;
```

```
In [ ]: static void update() {

            x += 5;
            y += 5;
        }
```

```
In [ ]:  static void draw(Graphics2D g2) {

             g2.drawOval(x,y,20,20);
         }
```

Try and compile and the program will probably have an issue with the Thread.sleep(50) not being wrapped in a try ... catch

```
error: unreported exception InterruptedException; must be c
aught or declared to be thrown
    Thread.sleep(50);
```

For now just add `throws Exception` to your main method definition to take care of the error.

```
public static void main(String[] args) throws Exception {
    ...
```
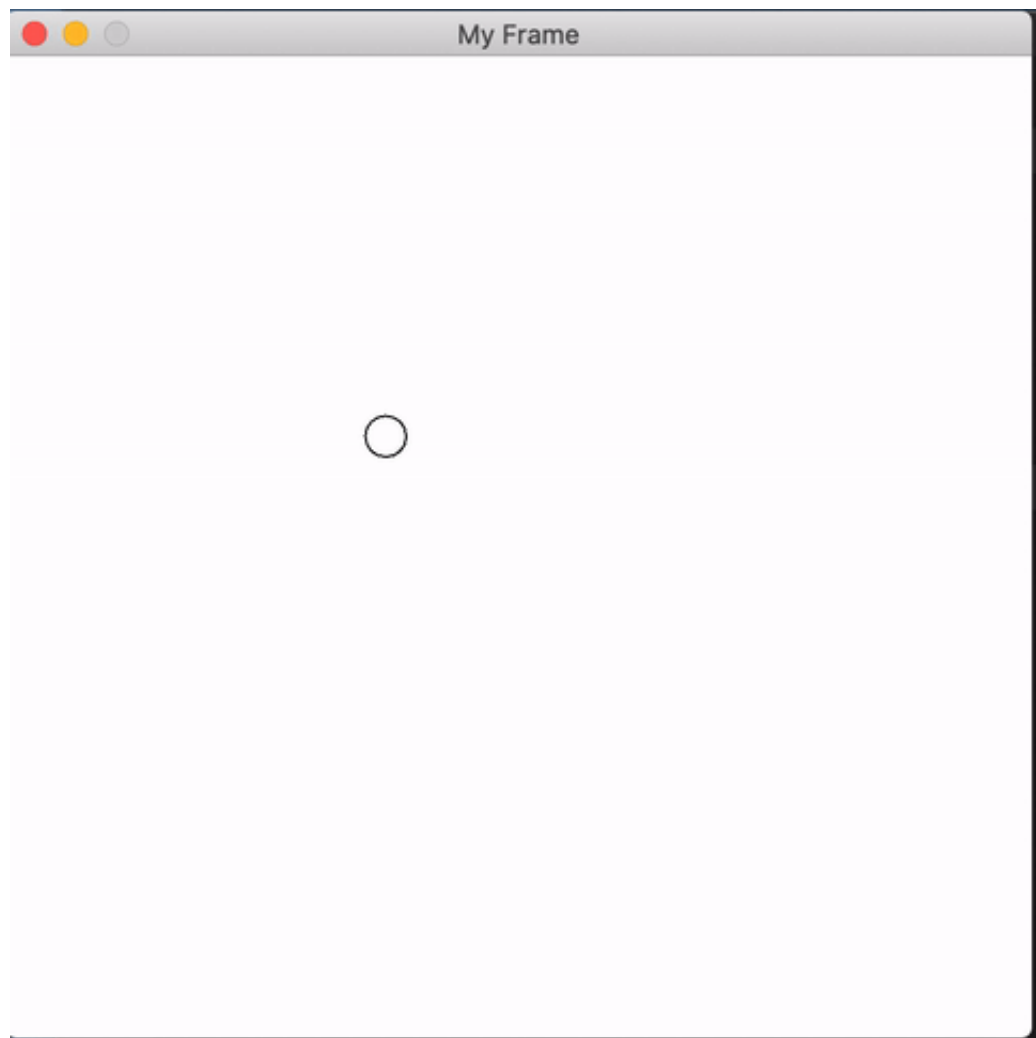
Run again and you should see this:
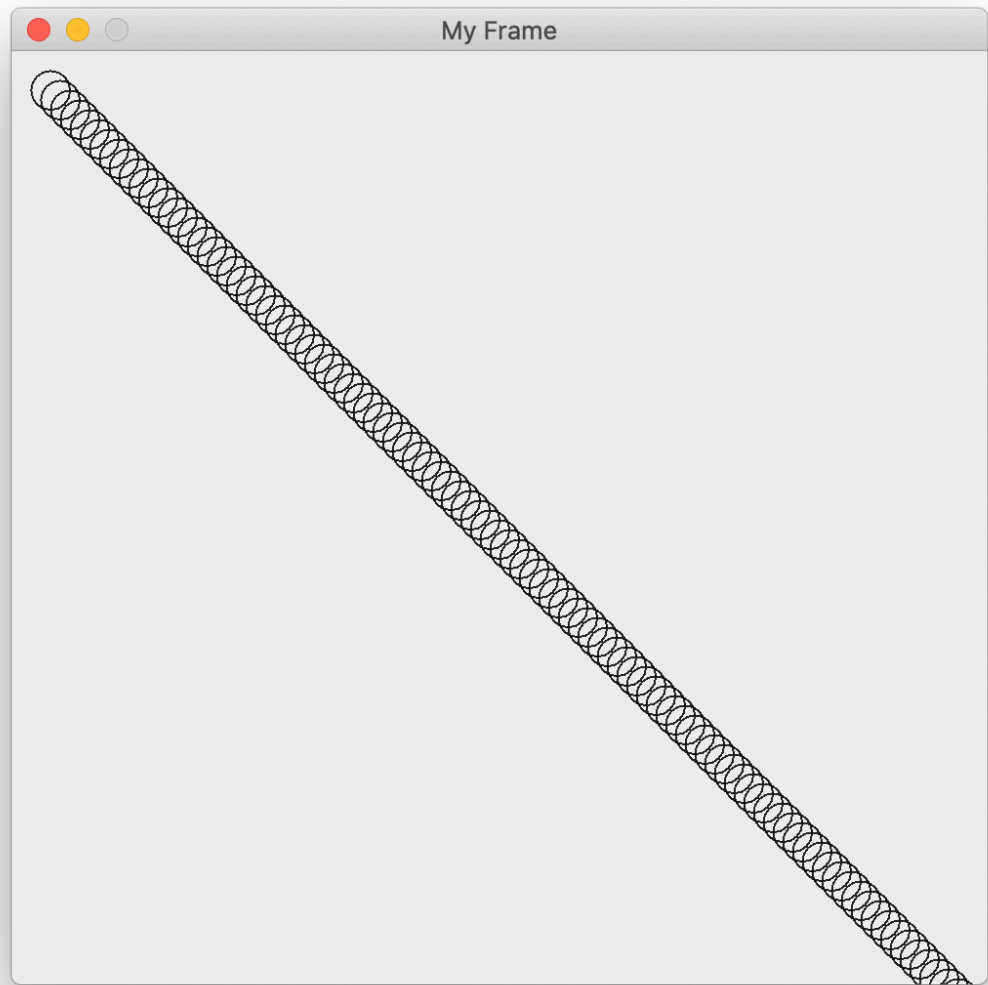
Add a solid rectangle to act as background and to 'clear' any drawing of previous frame before drawing next

In [ ]:
```
static void draw(Graphics2D g2) {

    g2.setColor(Color.WHITE);
    g2.fillRect(0, 0, 500, 500);

    g2.setColor(Color.BLACK);
    g2.drawOval(x, y, 20, 20);

}
```

Run again, and you will see more desirable results.

This loop with update() and draw(Graphics2D g2) method forms the basis to our game and is critical to understand. Almost all other classes (e.g. Shooter, Missile, Bunker) will have their own version of a update and draw function that will be somehow called from this game loop. More on that in the next tut.