Flash {AS}^{3}

Session 2: An introduction to object-oriented design, the display tree and events

Shiny Yang (mootothemax@gmail.com), special thanks to Alex Miller

Syntax Review

```
package  {
    import flash.display.*;

    public class HelloWorld extends Sprite {

        public function HelloWorld():void {
            trace(returnTest());
        }

        public function returnTest():String {
            return “Hello, world!” ;
        }

    }
}
```

Make sure you know: making packages, classes, importing, extending(inheritance), constructors, methods, returns.

ActionScript Syntax

Inheritance

Given:

```
public class Pond {

    public function Pond() {
        trace(“POND CREATED!”);
    }

    public function swim() {
        trace(“splash!”);
    }

}

public class Ocean extends Pond {

    public override function swim() {
        trace(“drown!”);
    }

}
```

```
var lakeWashington:Pond = new Pond();
//pond created!
lakeWashington.swim(); //splash!

var chunnel:Ocean = new Ocean();
chunnel.swim(); //drown!
```

NOTE: override keyword needed, no overloading constructors and no nested classes

ActionScript Syntax

arrays

Works like a combination of ArrayList and Stack from java.
(or just a php/javascript array)

Make

```
var a:Array = [];
var b:Array = new Array();
var c:Array = [1, 2, 3];
var d:Array = new Array(1, 2, 3);
```

Set

```
a[0] = “look ma no bounds checking”;
b.push(“checkem”); //adds in like a stack
```

Get

```
var i:Number = c[0]
d.pop(); //removes like in a stack
```

(Kinda) different from java.

ActionScript Syntax

first class & anonymous functions

```
test(
    function(s:String){
        trace(s);
    }
);

function test(a:Function) {
    a.call(this, [”hey guys what’s going on here~”] );
    trace(”HOLY MERDE”);
}
```

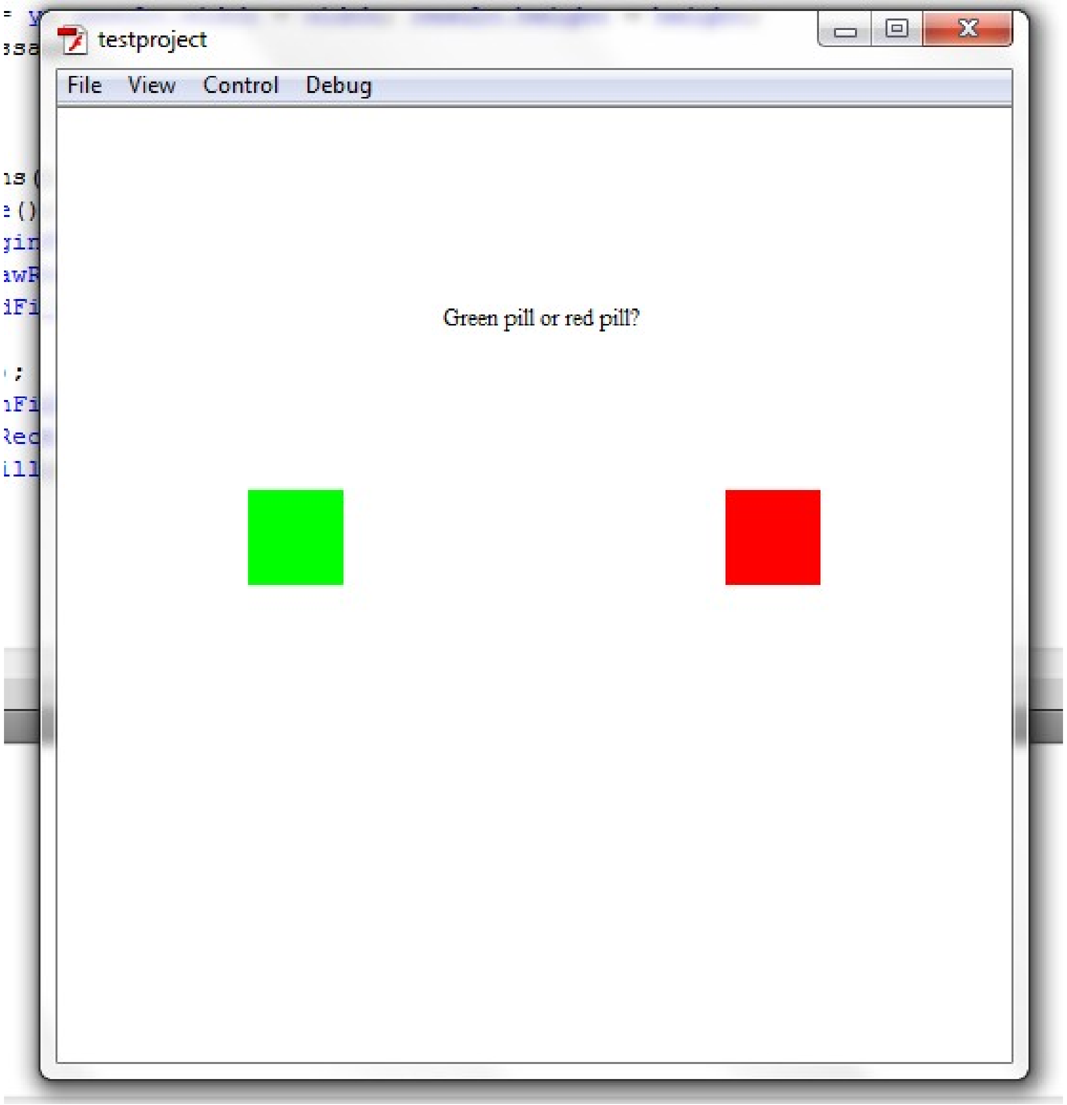
Passed methods can be invoked (used) in two ways, with the call method (shown above-see the actionscript API if you want more details)

and through being passed as an “event method” (later in this powerpoint).

It's in java too (sorta).

ActionScript Example

How is this useful?



Green pill or red pill?

Make a general purpose menu that displays text and two buttons.

It takes 2 parameters:
-String, message to display
-Array of 2 methods, runs first method if click green else run second if click red.

Use the createwindow(String,Array) function in the MenuExample class.

ActionScript Mystery

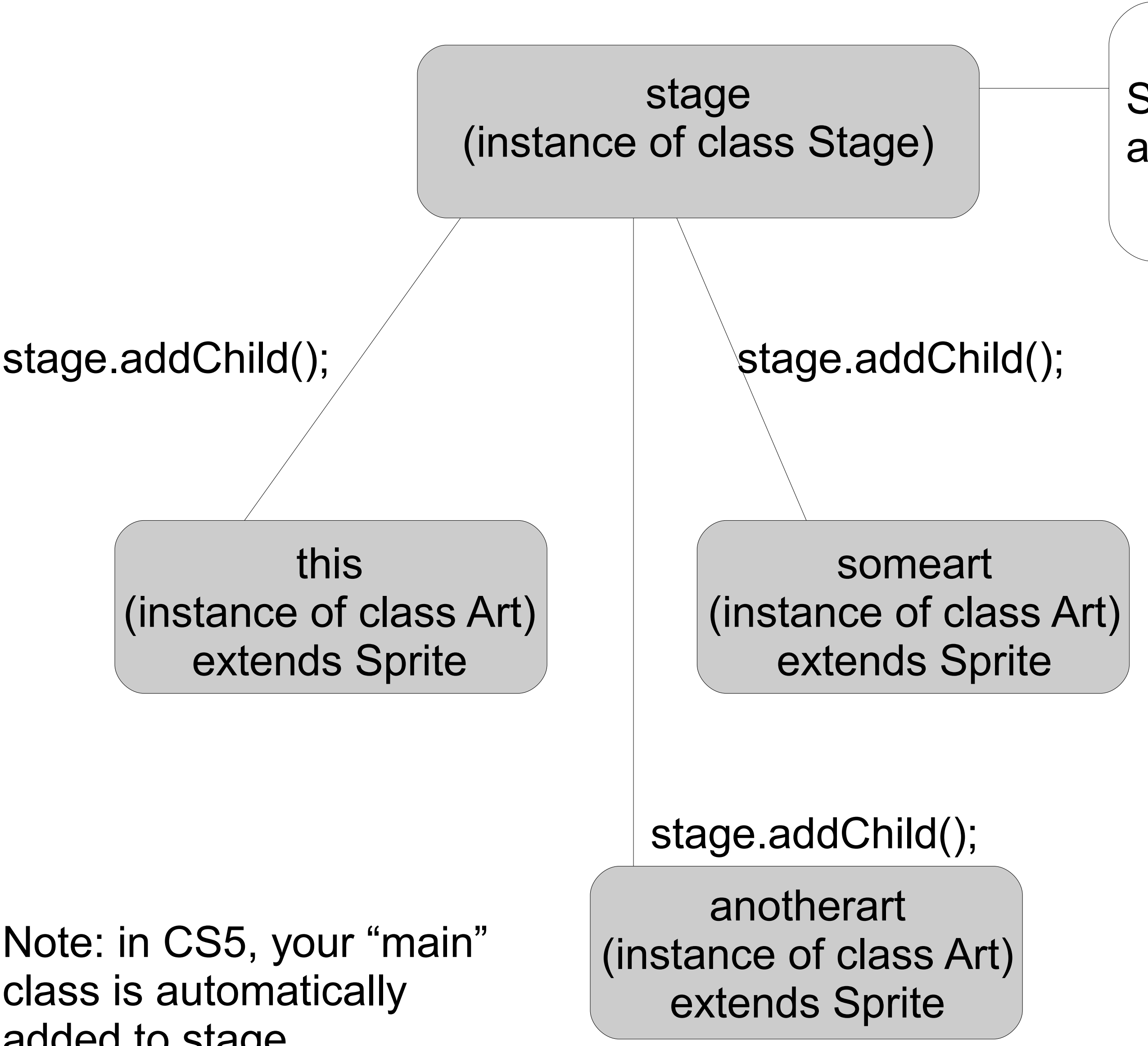
Why do we need to do these?

```
package  {
    import flash.display.*;

    public class Pix extends Sprite {


        public function Pix() {
            graphics.beginFill(0x0000FF);
            graphics.drawCircle(0,0,20);
            graphics.drawRect(100,100,20,20);
            graphics.endFill();

            stage.addChild(this);

        }

    }
}
```

Why extend Sprite and why stage.addChild()?

ActionScript Concepts

The stage + display tree



stage
(instance of class Stage)

Stage functions()
and fields

stage.addChild();

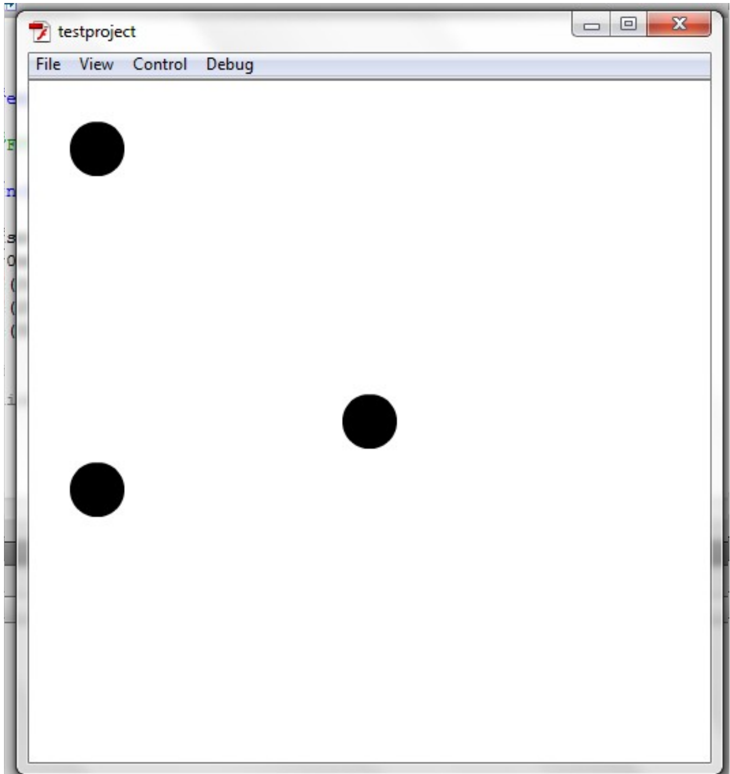
stage.addChild();

this
(instance of class Art)
extends Sprite

someart
(instance of class Art)
extends Sprite

stage.addChild();

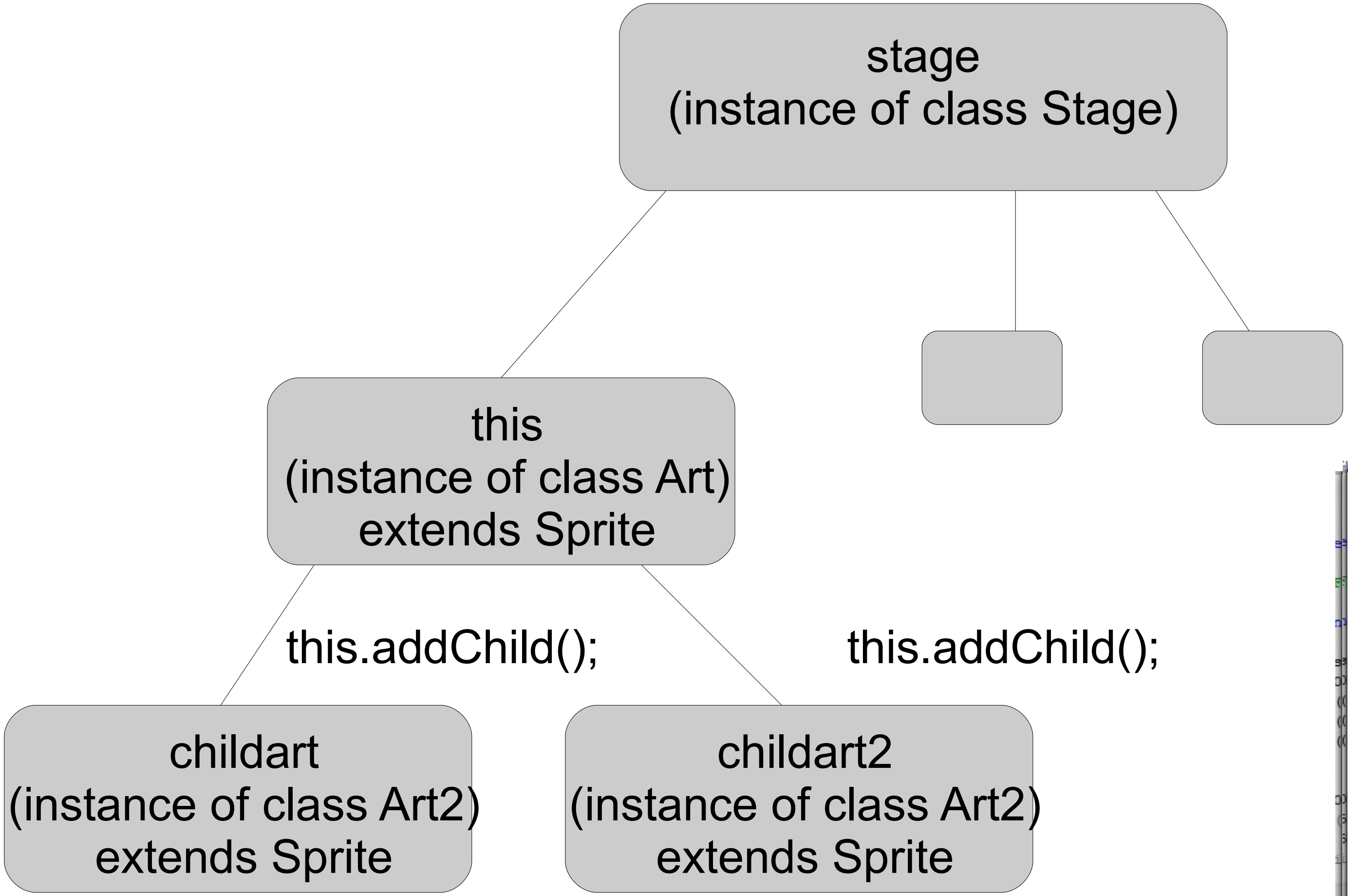
anotherart
(instance of class Art)
extends Sprite



Note: in CS5, your "main" class is automatically added to stage.

ActionScript Concepts

The stage + display tree



stage
(instance of class Stage)

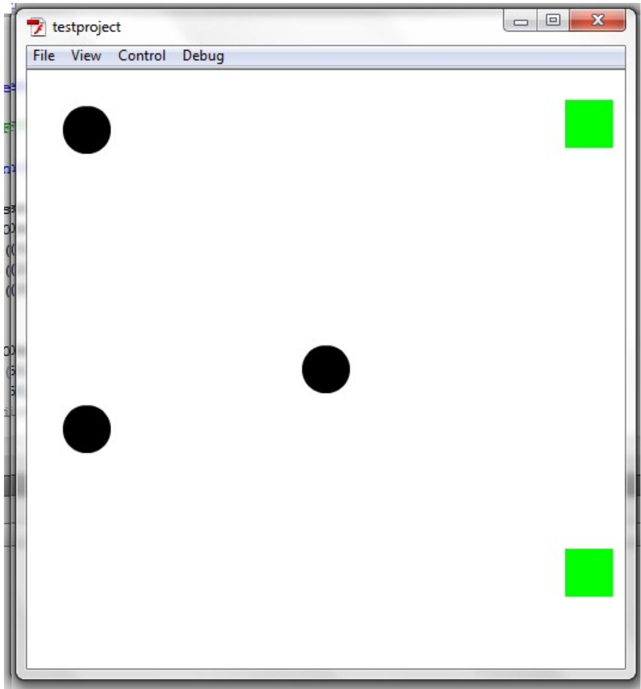
this
(instance of class Art)
extends Sprite

this.addChild();

this.addChild();

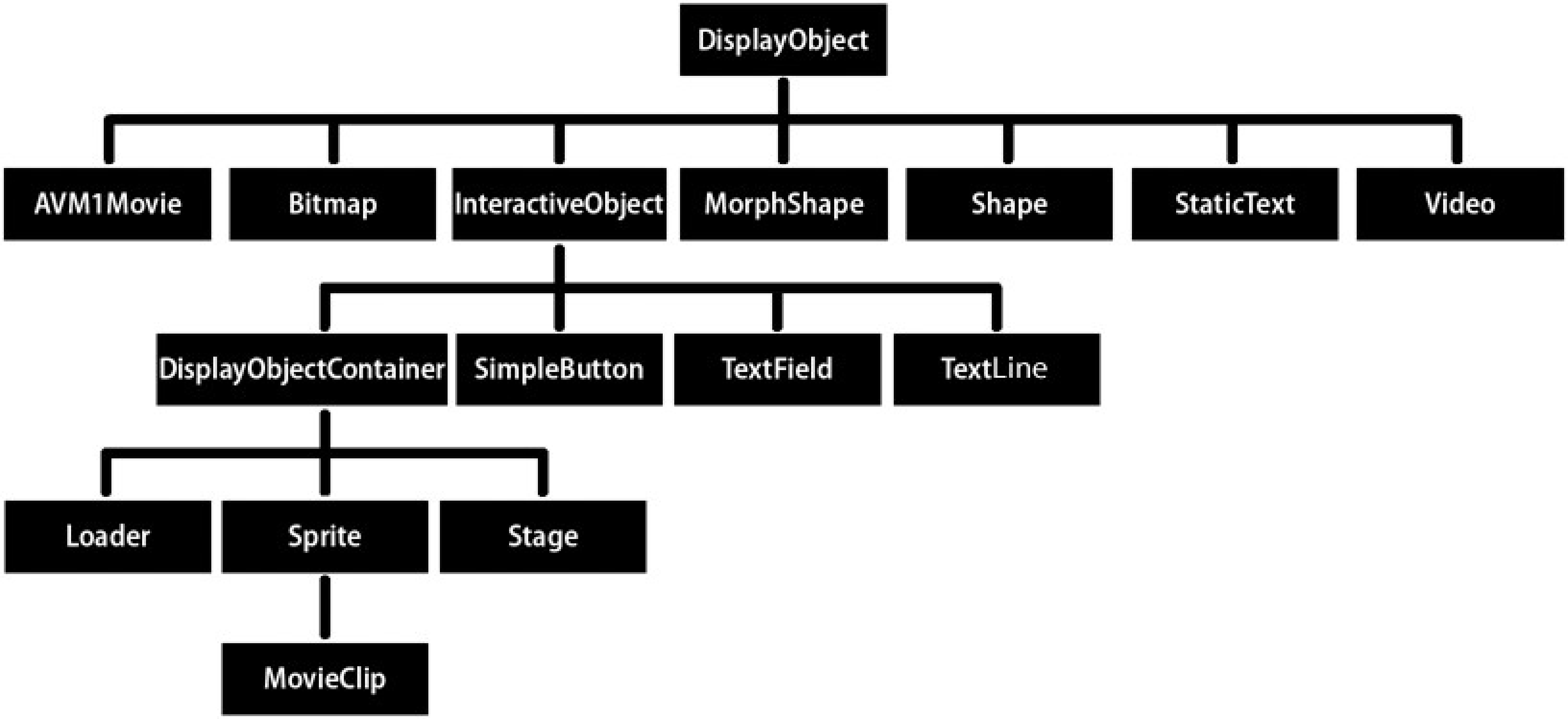
childart
(instance of class Art2)
extends Sprite

childart2
(instance of class Art2)
extends Sprite



ActionScript Concept

What can you display?



Anything that is a child of DisplayObject, though not DisplayObject itself.

ActionScript Code Snippets

Nifty Code for Clearing Screen

```
while(main.numChildren > 0) {
    main.removeChildAt(0);
}
```

How and why does this work?

NOTE: Be sure that you have access to stage.
Stage is NOT a global variable, but your “main” class will have access to it.

ActionScript Syntax

What is the Sprite class, and why do we use it?

```
Sprite.x
Sprite.y
Sprite.graphics
Sprite.addChild();
```

NOTE the sprite x and y, the graphics drawn will move relative to the sprite's x & y position.

And some other nice methods (we'll be using these later):

```
Sprite.addEventListener();
Sprite.hitTestObject();
Sprite.hitTestPoint()
```

To learn more, check out the API

Lets go back and see if we can make more sense of MenuExample.as

ActionScript Syntax

TextField

```
var displayfield:TextField = new TextField();

(import flash.text.*;)
```

```
TextField.x
TextField.y
TextField.width
TextField.height
TextField.text
```

Part of the DisplayObject family, can be directly addChild()

Some more cool fields/methods we will be using:

```
TextField.selectable
TextField.addEventListener()
TextField.setTextFormat()
Textfield.scrollable()
```

ActionScript Syntax

Your introduction to Events (to be continued next time)

```
Sprite.addEventListener(MouseEvent.CLICK, function(){});
```

```
(import flash.events.MouseEvent;)
```

Inherited from DisplayObject (so Stage can do it too!)

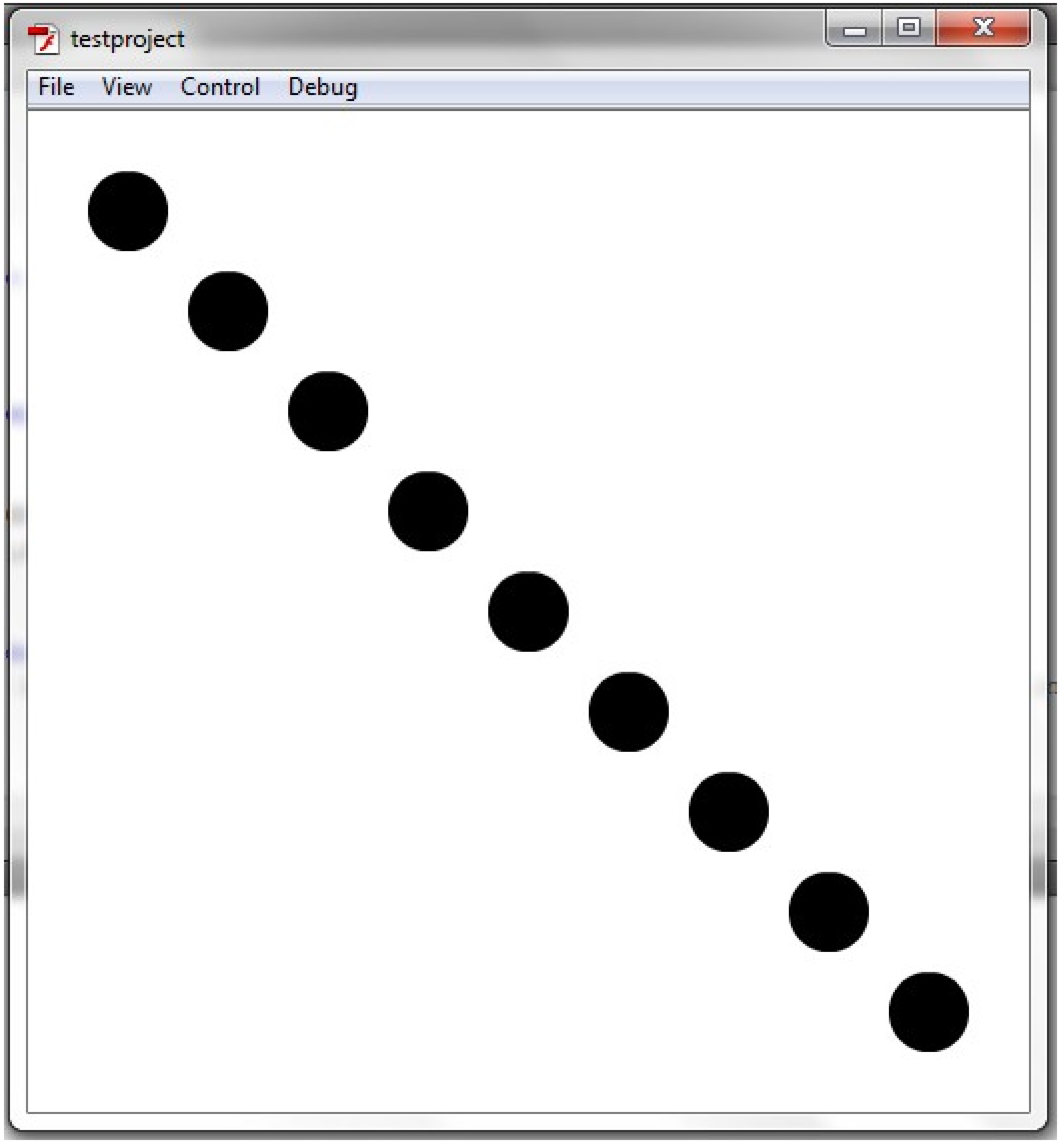
First parameter is event type, second is function to run when the event happens.

MouseEvent.CLICK happens when the sprite (or DisplayObject) is clicked.

NOTE--the 'THIS' keyword inside the parameter function will not refer to the object the event was called upon (unlike javascript), but a relatively useless “object global”. Thus, place the event function where you want your scope to be.

ActionScript Example

Clickballs



Make a class that extends Sprite and draws a 20radius circle.

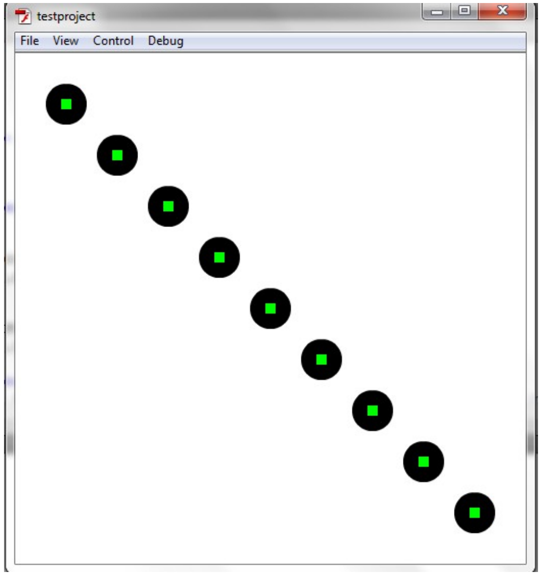
Make a main for the program that draws 10 of these in a row like on the left.

Place event listeners so that when you click on the ball, it should tell you what number it was in creation. (So top left is 1, next down is 2, etc)

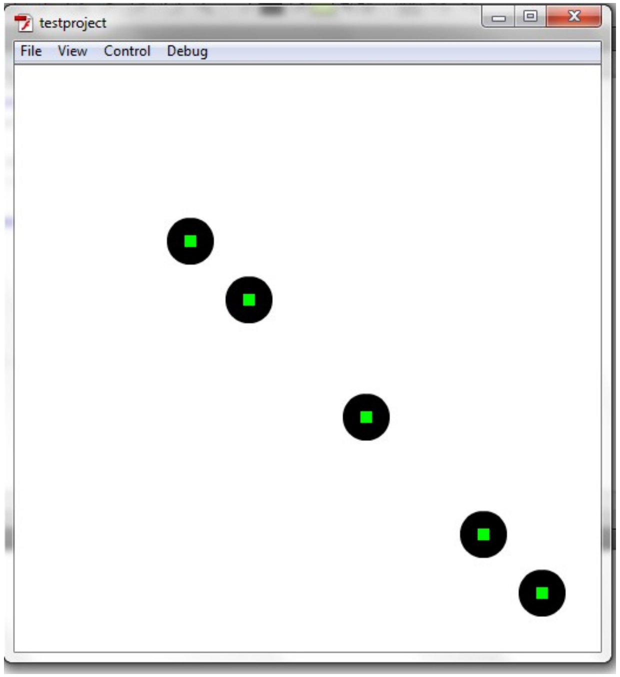
(HINT-the event listener needs to go inside the ball class)

ActionScript Example

Clickballs EXTRA- if there is time



Give every ball a green (0x00FF00) center rectangle
(-5,-5,10,10)



Make it so when you click on a ball, it dissapears. Do with graphics.clear(), then try actually removing from display tree.

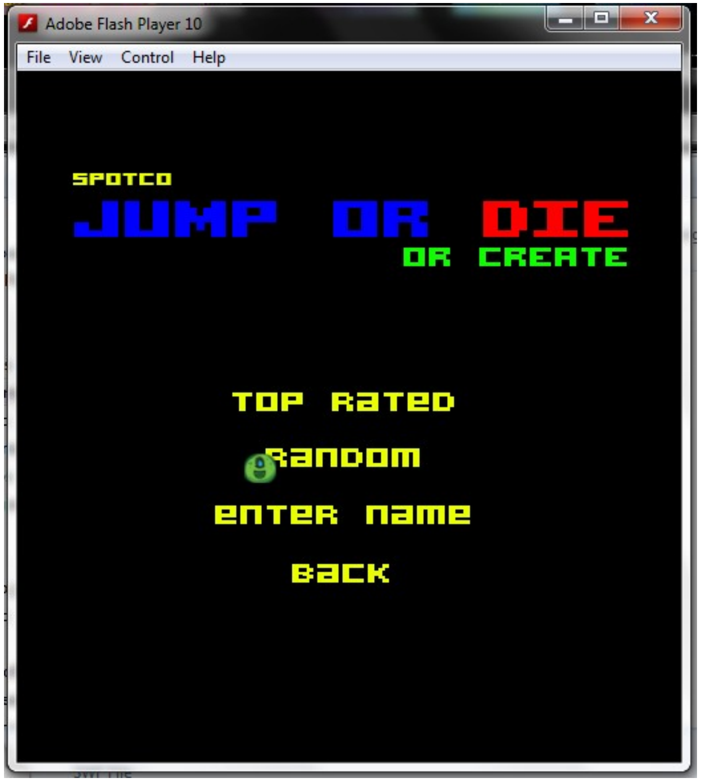
HINTS:
-pass main into the balls as a field
-make another reference in the balls to “this” (you can't use “this” in the anon function)
-main.stage.removeChild(thisball);

Homework

Requirements:

Make a Menu.as that includes at least ONE class that extends sprite and ONE textfield.
Add a working click event listener.
Make this click event listener do something significant that can be seen from the stage.
(DO MORE THAN JUST TRACE)

And above all - **be creative!**
(turnin coming soon)



Ideas:
-A menu with buttons that show pictures?
-A GUI that moves around a sprite on stage? (the buttons change that sprite's x and y)
-A button that makes a random string of text appear in a TextField?