**Flash ActionScript – Ball Follows Mouse**

**Question of the Day:**
- How does Flash know the difference between copies of the same movie clip on the stage?
- Provide a list of **events** that are frequently used with movie clips.
- Open a new Flash file > save as "**Ball Follows Mouse**" > create a movie clip on the stage > give it an instance name of "**ball**" > rename Layer1 to "**Graphics**" > create a new layer and name it "**Actions**"

**Objectives:**
- Answer Question of the Day
- Create animation with ActionScript  (demo & tutorial)
- **ASSIGNMENT DUE AT THE END OF CLASS → 3pts**
  - Add a second movie clip to "Ball Follows Mouse" file.
  - Add ActionScript code to make the second movie clip move, change shape, or anything else that you would like it to do.

**Answer to Question of the Day:**

**How does Flash know the difference between copies of the same movie clip?**
- An **Instance** is a copy of any symbol from your Flash file's library which resides on the stage.
- **Instance Names** are used by ActionScript to identify each instance.  This is what distinguishes copies of the same movie clip.

**Frequently Used Movie Clip Events:**

| Event | Event | Event |
| --- | --- | --- |
| onEnterFrame | onPress | onKeyUp |
| onMouseDown | onMouseMove | onRelease |
| onMouseUp | onKeyDown | onRollover |
| onLoad | | |

**Demo:**  Creating animation with ActionScript
- Create a new file → save as "**Ball Follows Mouse**"
- Add a circle symbol
- Make the circle a moviclip symbol → **right-click** and choose **Convert to Symbol**; name the symbol **mc_ball,** check **"Export for ActionScript",** give the symbol on the stage an instance name of **ball** (ActionScript needs to know what to call the symbol before it can tell it what to do).
- Insert a new layer, name it **Actions**
- Open the **Action** window at the bottom of the screen
- Type the following code in frame 1 of the **Actions** layer:

```
ball.onMouseMove = function( ) {
        ball._x =  _xmouse
        ball._y =  _ymouse
}   // end onMouseMove
```

- Now we'll change the width of the ball when the user left-clicks the mouse and return the ball to original size when the mouse is released.

```
ball.onMouseDown = function( ) {
        this._xscale = 500
}
ball.onMouseUp = function( ) {
        this._xscale = 100
}
```