

Multimedia Module

FLASH 5 PART II USER MANUAL



For information and permission to use these training modules, please contact:

Limell Lawson - limell@u.arizona.edu - 520.621.6576

or

Joe Brabant - jbrabant@u.arizona.edu - 520.621.9490

Modules available online at <http://mll.arizona.edu/workshops.shtml>

TABLE OF CONTENTS

Exercise 1: Creating the Clock.fla Flash File	3
Exercise 2: ActionScript to Loop to the First Frame of “mc, clock”	4
Exercise 3: Scripting and Action for a New Button Symbol	5
Exercise 4: Naming a Movie Clip Instance	6
Exercise 5: Controlling a Movie Clip Instance	6
Exercise 6: Dragging a Movie Clip	7
Exercise 7: New onClipEvent Method	8
Exercise 8: Make the Value of a Variable Visible	9
Exercise 9: More With Variables	9
Criterion Test	12

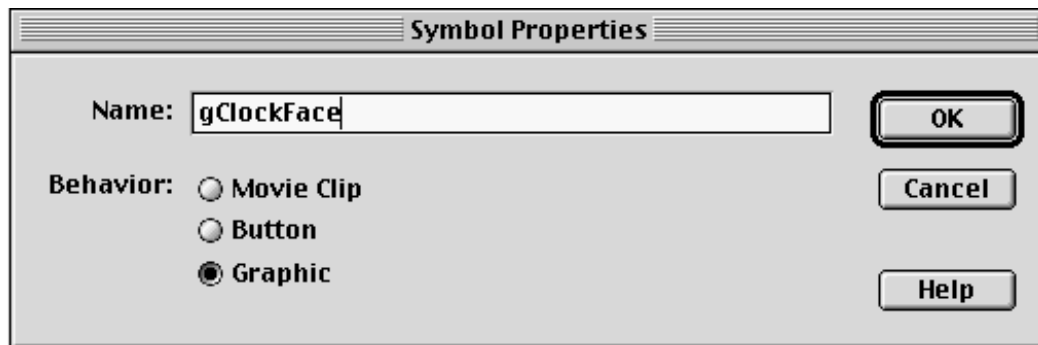
EXERCISE 1: CREATING THE CLOCK.FLA FLASH FILE

Demo [2 min], Follow Along [3 min], Practice [10 min]

Create a background clock face graphic and using the Rotation Palette:

Steps:

1. Create a new untitled movie, (**File > New**) if you do not already have one opened.
2. Choose **Window > Panel sets > Default layout**
3. **Insert > New Symbol**
4. Choose **Graphic** as behavior, and name it **gClockFace**. The “g” is for graphic, the appended “Clockface” is a general name that specific instance names can be based on later.

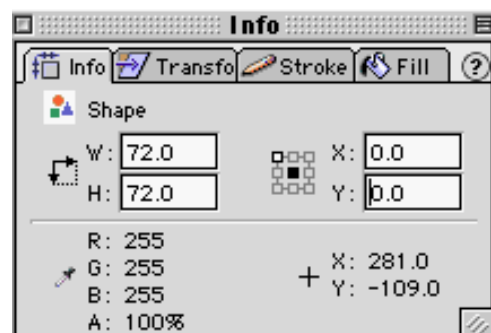


5. Draw a perfect **circle** shape (Hold down the shift key as you draw) as you are in **Edit Symbol** mode. Check the Address bar.
6. Give it a light color with a dark border.
7. **Center** it over the registration point.
8. Use the **Info and Transform Panels** to make the size appx. 72 pixels x 72 pixels and center it 0,0 at the registration cross. Notice it appears in the Library

Create a clock hand graphic:

Steps:

1. Click **Scene 1** in the Address bar to exit Edit Symbol mode.
2. **Insert > New Symbol**
3. Choose **Graphic** as behavior and name it **gClockHand**.
4. Draw a vertical box with the Rectangle tool. Solid black fill. Stroke of 1.
5. Make its width appx 3 pixels and a length of appx. 35 pixels and have the bottom of it touching the registration cross at 0,0.
6. Exit Edit Symbol Mode. Notice it appears in the Library.

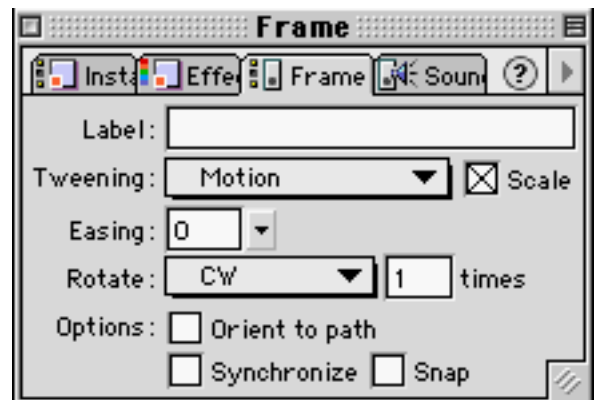


Create Movie Clip “mc, Clock” Tween:

The “mc” will identify it in the Library and allow us to sort by type of symbol later.

Steps:

1. **Insert > New Symbol.**
2. Choose **Movie Clip** as the behavior and name it “**mc, Clock.**”
3. Create new layers and name them in this order: Actions, Hand, Face
4. Drag **gClockFace** from the Library into the face layer and center it on the stage with use of the Info Panel.
5. Insert a keyframe at **frame 60** of the **face layer.**
6. Drag **gClockHand** from the Library into the hand layer and place it so the bottom is on the center of the face circle.
7. Insert a keyframe in **frame 60** of the **hand layer.**
8. Select the first keyframe in the hand layer and, using the **Frame Panel**, set Rotation to Motion Tweening, CW, one time.
9. From the Menu bar, Use the **Control > Play** command to check that it rotates correctly. Note: the use of only two keyframes to accomplish this decreases file size.
10. Save the file as **clock fla** to a place you will remember.
11. Do **NOT** save this next step. Check that you are in the scene 1 timeline and then drag an instance of the mc, Clock MovieClip to the main timeline and **test the movie.** Note that the clip seems to hesitate at the top as two exact frames play one after another. We will correct this in the next exercise with ActionScript.
12. **Undo** or revert to the last saved version of the file.

**EXERCISE 2:****ACTIONSCRIPT TO LOOP TO THE FIRST FRAME OF “MC,CLOCK”****Steps:**

1. Open the timeline of “mc, clock” by double clicking on it in the Library. In the Actions Layer, insert a **Blank Keyframe at frame 59.**
2. With the blank Keyframe in the actions layer selected, open the Frame Actions Palette, and add this script at frame 59 so that a frame doesn’t repeat: **gotoAndPlay (1);**
3. **Control > enable simple frame actions.**
4. **Control > Play** and note the animation is smoother when the hand reaches the top of the clock.

EXERCISE 3: SCRIPTING AN ACTION FOR A NEW BUTTON SYMBOL

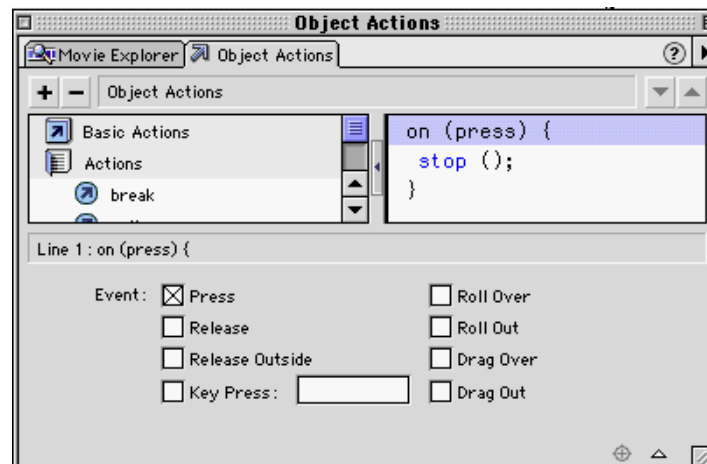
Steps:

1. **Insert > New Symbol.** Name it “**btn,SimpleButton**” and draw a rectangle for the button shape.
2. Insert a keyframe in the **hit** frame, and make it single color without texture etc., as you may have already done in the first keyframe. This will reduce file size.
3. Create a **new layer for text** and put in text that says “button” in at least the first keyframe. **Exit the symbol editing** mode by clicking the Scene 1 in upper left.
4. Select the “**mc, Clock**” in the Library, Duplicate it (**Library > Options > Duplicate ...**) and rename the copy to “**mc, Clock-button**”
5. Open the **timeline** of “mc, Clock-button”
6. Create and name a new layer called “**button,**” and then drag an instance of **simpleButton** to that layer in the Movie Clip.
7. Select this button in the “**button layer**” of Movie Clip “**mc, Clock-button**” and **open the Actionscript Panel** and add an ActionScript statement for STOP.
8. Return to the main (**_root**) timeline and drag an instance of the mc, Clock/button to the layer you now name “**layout.**”
9. **Test** your movie. The button action should stop the timeline that it is placed in from playing. Note that you can only stop it.
10. **Save** this Movie as **clockv2 fla.**

Edit the actions, using the parameter pane settings of the Object Actions Palette.

Steps:

1. Select the button in the “**mc, Clock-button**” **Movie Clip** and click the arrow for the **Actions Palette** in the lower right of the stage area.
2. Select the “**release**” condition in the script.
3. Change the event to “**press**” rather than “release.”



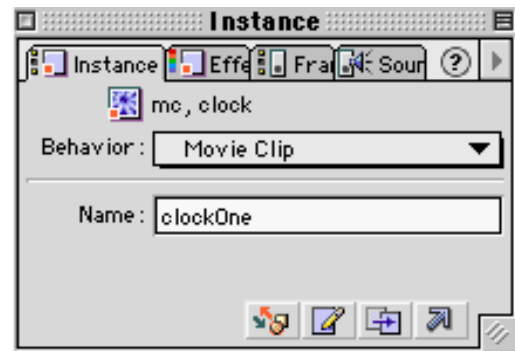
4. Select the last “**}**” and drag the play command over to it.
5. **Test the movie** and the button behavior. It should pause when you hold the mouse down and play when you let up.
6. **Save** the movie as **clock3 fla**

EXERCISE 4: NAMING A MOVIE CLIP INSTANCE

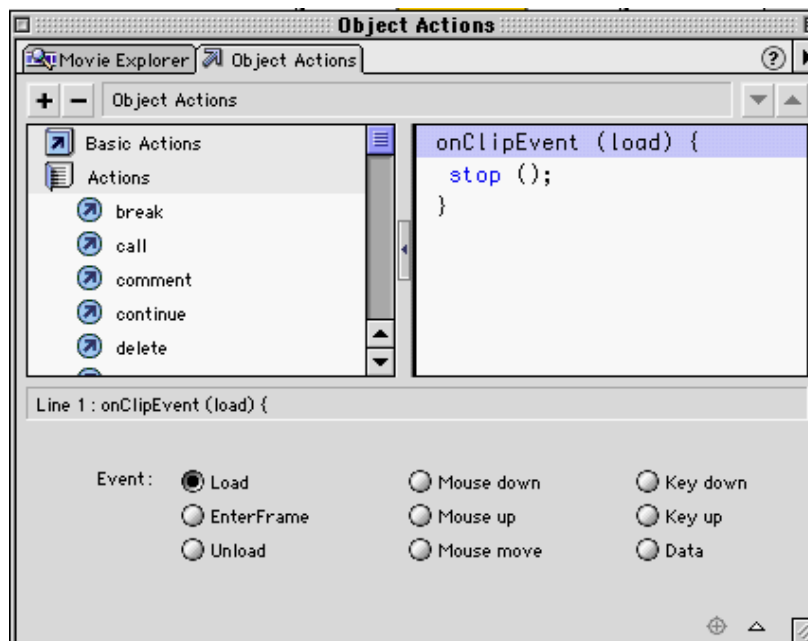
Instantiation of a movie-clip symbol involves two steps: placing an instance on the Stage and naming that instance.

Steps:

1. Create a **new layer** and name it “**layout2.**” Drag an instance of the movie-clip symbol “mc, Clock” from the Library onto the Stage into the “layout2” layer.
2. Create a **new layer** and name it “**Controls.**”
3. Create a **new layer** and name it **ActionScripts.** Drag it to the top if need be.
4. Set the File/Publish settings .../Flash/to **Top down as load order.**
5. Select the instance of the “mc, Clock” in the “layout2” layer.
6. In the Instance panel of the Instance Panel, type in the **Name Field** a unique descriptive name for your movie-clip instance. Name it “**clockOne.**” This is case Sensitive! Your movie-clip instance can now be modified or addressed by actionscript, using this name to identify it. It is case sensitive!
7. **Test the Movie.** Note that both clock instances work.



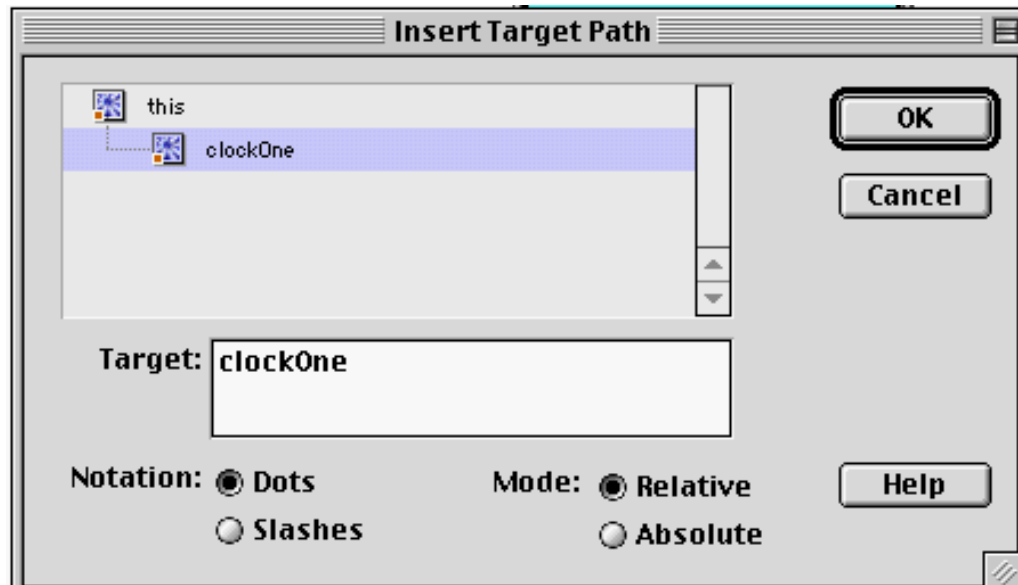
EXERCISE 5: CONTROLLING A MOVIE CLIP INSTANCE



Steps:

1. Select the instance of “**clockOne**” check to confirm that the name is still in the Instance Panel.
2. Open the ActionScript Panel another way via **Windows > Panels > Instance.**
3. **Add the Stop statement** to the script window.
4. **Test the Movie.** Note that clockOne doesn’t play.
5. Open the script for the “clockOne” instance again and select the conditional statement “**load.**” Note the different events and names options in the Object Actions panel. Flash has an overlapping but separate set of events for the Button and Movie clip instances.

6. **Close** the Object Actions Panel.
7. Select the “**Controls**” Layer in the main timeline and **drag an instance of “simpleButton”** from the Library to it.
8. Open the Actions Panel and with the plus button, choose **Actions > Evaluate**.
9. In the Parameters area, click the **Insert Target Path** button.
10. Select **Relative mode**, and choose the movie-clip instance “clockOne” from the display list. The target path appears in the Target field. Click OK.
11. The target path appears in the Expression field.



12. In the Expression field after the Target Path, enter a dot, then the desired method with any applicable arguments. In this case type “**this.clockOne.play()**” (without the quotes).
13. **Test your movie**. The button should get the clockOne instance to play. Your button instance controls the Timeline of the targeted movie clip.
14. **Save the movie as clock4 fla**

EXERCISE 6: DRAGGING A MOVIE CLIP

Steps:

1. In the file you just made file, select the instance of the movie clip symbol **clockOne** on the stage. **Open its timeline**.
2. Create an **invisible button**. Do this by; **Insert > New Symbol**, name it “invisible button” and then leave empty keyframes in all states except for the hit state. In the last keyframe of the button put a graphic about the same size as the clockface graphic (you may copy the graphic face symbol). An instance of this button will have a transparent blue shape when on the stage). **Place an instance** of your button inside the movie clip timeline in its own named layer.
3. Select this button instance, and **open the actions panel**.
4. Choose **Actions > On**. The on (release) statement appears in the script window.
5. In the Parameters pane, uncheck the box next to Release, and check the box next to Press. The on (press) statement changes to on (release).
6. Choose the **Actions > startDrag**. The startDrag action appears under the on (press) mouse event.

7. In the parameters pane, in the Target field, enter “**this**” and check the expression box. The keyword **this** refers to the present movie clip. Checking the Expression box removes the quotation marks around **this** so that Flash recognizes the word as a **keyword** rather than as a **string literal**.
8. **Delete the instance** of the movie clip “mc,clock” on the stage in the main timeline and pull a new instance from the Library to the stage in the same layer. Since the button now inside the movie clip uses the relative term, **this**, it wasn’t necessary to name your movie clip instance.
9. **Test your movie**. When you press your mouse button over the movie clip, you can drag it around.
10. Re-open the script and add a **stopDrag action** to the script.
11. **Save the movie as clock5 fla**.

(You could also have set drag parameters for the object if it was to be part of a slider “widget.”)

EXERCISE 7: NEW onClipEvent METHOD

Steps:

1. **Create** a movie clip, or use the previously created movie tutorial .fla file.
2. Select it on the stage.
3. Open the **Actions Panel**.
4. Choose **Expert Mode**, then write or paste this script into the actionscript window for the **mc, clock instance**:

(Note the optional comments preceded by “//” marks.)

```
//this script checks to see if the mouse is over the object,  
//and then sets its property  
// to draggable if the cursor is also over it.  
  
onClipEvent (mouseDown) {  
    if (this.hitTest(_root._xmouse,_root._ymouse, true)) {  
        startDrag (this);  
    }  
}
```

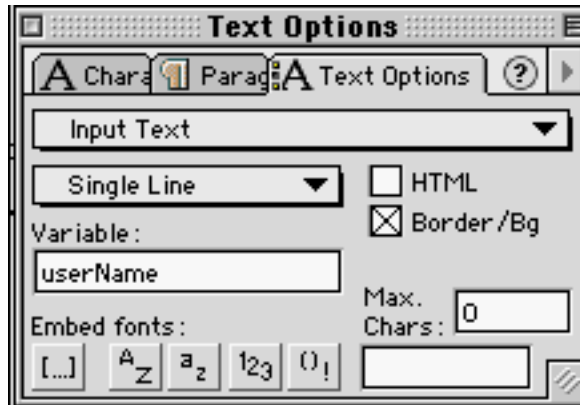
The underscore (as in `_underscore`), is how Flash distinguishes built in properties and variables in actionscript.

Since the `onClipEvent (mouseDown)` will respond to a mousedown anywhere on the stage, we must pre-qualify the `startDrag` with a conditional statement – is the cursor over the instance before executing the `startDrag do` statement. So we have the `if(this.hitTest(_root._xmouse,_root._ymouse, true))` conditional.

EXERCISE 8: MAKE THE VALUE OF A VARIABLE VISIBLE

Steps:

1. **Create** a new file. Name the layer "Layout."
2. Select the keyframe in the layout layer.
3. In the Toolbox, select the text tool and draw a text field on the stage. Access the Text Options panel (**Window > Panels > Text Options**).
4. From the Text Type Panel Menu, choose **Input Text**. Check border/background. Change the default value of the Variable associated with the text field from "TextField1" to "userName." Without the quotes. Case Sensitive!



5. Type in **enterNameHere** into the text field.
6. Test the Movie (**Control > Test Movie**).
7. While the file is playing, Type in another name in the user editable input text field.

EXERCISE 9: MORE WITH VARIABLES

Initializing another variable for a greeting message:

Steps:

1. Create a **new layer** and name it button.
2. Select the keyframe in the button layer and then Drag a pre-made button from the Library (**Windows > Common Libraries > Buttons** and then put the "Play" button from "(circle)Button set" on the stage.
3. Create a **new layer** and name it scripts. Select the first keyframe in the "script" layer on this root Timeline, and open the Actions Panel.
4. Choose **Actions > Set Variable**.
5. In the Variable field of the Actions panel, enter the name of your variable, which we will call greetingMessage. In the **Value field enter your value** ("Hey there ... "). This is the value that the variable greetingMessage will hold. Leave the Expression box unchecked.



6. **Test the movie** to see what works.

Just because you assigned a value to a variable doesn't mean it was the right one in the right place to get used. There is a way to check these as you construct your file.

Check the value of a variable with the Trace function when the movie plays:

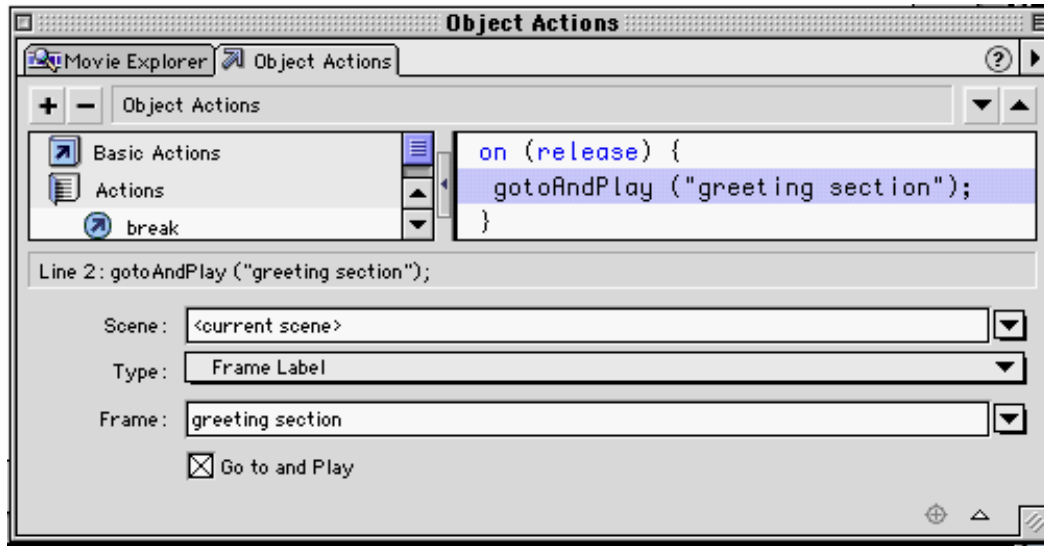
Steps:

1. Select the script in the script layer keyframe, and open the Actions panel (**Window > Actions**). Note that it is labeled "Frame Actions"
2. **Add the Trace Function** to the script by selecting the last line in the script and then MouseDown on the "+" button and select **Actions > Trace**. Trace (""); appears.
3. Type **greetingMessage** in the ()'s and check the expression box.
4. **Test the Movie**. Note that the value of greetingMessage should now appear in the OutPut window that opens.

We want the user to go to another section after entering their name, so lets construct part two of this file.

Steps:

1. Open the script layer script in the first keyframe, and add a **Stop statement** by selecting the last line in the script window and then selecting Stop from Frame Actions commands.
2. **Insert** a keyframe in frame 10 of the script layer
3. Add a **Stop command** in keyframe 10 of the script layer.
4. With the Frame Panel **add a label** to the frame 10 of the script layer. Name it Greeting Section.
5. Select the button keyframe and add the script "**Go to**"
6. Select framelabel vs frame number in the Type menu and "Greeting section" should appear as the value for Frame.



7. **Test the Movie.** The button should disappear as we are on frame 10.

CRITERION TEST

20 min:

This is the criterion test for the Flash 5 Part II module. It is going to test your comprehension of the workshop you just participated in or allow you to test out of the Flash 5 Part II module.

Design and create a very simple Flash file that:

- ◆ uses a variable associated with a text field and
- ◆ uses one button to control another MC, or is draggable.