Flash 4 Exercise Guide

MMC 4341 Advanced Online Media Production

This guide is designed to help you manage the tasks required for the Flash 4 exercise.

1 / Understanding Masks and Movie Clips

Since we haven't covered masks yet, you're going to do two quick exercises using masks.

Mask Exercise 1
1. Open *ace.fla* in the folder named Masks.
2. Look in the Library. Nothing there but one playing card symbol (which is already on the Stage).
3. Animate the playing card on the Stage. Make it begin off the Stage on one side and move to the opposite side. Nothing fancy. Test the movie.
4. Lock the playing card layer and make a new layer above it.
5. Create a rectangle shape (any color) with NO stroke. Make it the same size as the playing card (150 x 200).
6. Position the rectangle at or near the center of the path of the card's animation. That is, the card will move under the rectangle.
7. **Right-click the name of the layer** holding the rectangle. Select "Mask" from the menu.
8. Save and test the movie.

Notice what happened to your layers after you designated the Mask layer: They are both locked, and the lower one is indented. If you unlock either layer, you will see the rectangle again. Any layer indented under a mask layer will be masked. Layers under a mask layer *but not indented* can be seen outside the mask. A mask can be any shape or a symbol.

If you don't feel certain about how the mask works, *play with it*. Make the mask rectangle a different size. Erase a hole in the middle of it. Change the direction of the playing card animation. Move the mask off center. Experiment!

Mask Exercise 2
1. Create a new Flash document, 600 x 300.
2. Create a rectangle shape (any color) on the Stage. After drawing the rectangle, use the Properties panel to change its dimensions to 1800 x 260. (Yes, 1800.)
3. Position it at X:0 and Y:20.
4. Convert it to a Movie Clip symbol. Name it *Content*.
5. Double-click to go inside the symbol.
6. Add a layer, and lock the bottom layer (the rectangle movie clip). Paint a blob or two on the upper layer, or add some text. Put some markings near the left edge, some more near the right edge, and more in the center. It doesn't matter what they are, as long as there are marks.

7. Exit from Symbol Editing Mode (the blue arrow!).
8. **Create a tween** in the movie clip's layer in the Main Timeline. For frame 1, keep it at X:0 and Y:20. Add frames up to frame 100. In the last frame, change the X to –1200 (note the minus sign). Keep Y:20.
9. Save and test the movie. **Save it in the Masks folder.**
10. **Here comes the mask part!** Lock the movie clip's layer. Make a new layer above it.
11. Create a new rectangle shape (any color) with NO stroke. Make the size 500 x 260.
12. Position the rectangle at X:50 and Y:20.
13. Right-click the name of the layer holding the new rectangle. Select "Mask" from the menu.
14. Save and test the movie.

Now you know how the Flash 4 exercise template is constructed.

In the template, there is a large horizontal rectangle movie clip (Content) under a mask rectangle movie clip (Rectangle). The instance name of the big movie clip is *content_mc,* and the mask is *mask_mc* (creative, yes?). This is what you just built here, except here you did not need to convert your mask rectangle to a movie clip symbol. In the Flash 4 exercise, the mask MUST be a movie clip BECAUSE it MUST have an instance name.

So right now, convert your mask into a Movie Clip symbol (unlock the layer). Name it Rectangle. Give it the instance name *mask_mc*. Now it's like the mask in the Flash 4 template. Save it!

Two things to understand about the big clip in the real Flash 4 exercise:

■ Inside the big clip, it can have *as many layers as you need*. To create the content for the assignment, you would be wise to make a new layer for each item you want to appear inside the big clip, and then scrupulously lock the layer after you have designed the item. Lots of locked layers prevent disasters and accidental deletions.

■ In this exercise, your big clip is animated on the Main Timeline. In the Flash 4 exercise, it WILL NOT BE ANIMATED or tweened. The motion will all be determined by ActionScript, which is provided for you. The Main Timeline in your Flash 4 exercise will have only 1 frame (but several layers) on the Main Timeline.

2 / Placing a Movie Clip Inside a Movie Clip

There is no mask in this exercise. Your task here is to place two movie clips *inside* the big clip (*content_mc*) that already exists in this FLA.

1. Open *exercise2.fla* in the folder named Embedded Clips. Test the movie: Ctrl-Enter (Win) or Command-Return (Mac). Let it play all the way until it repeats. Then try the Stop and Play buttons.

2. Close the SWF window and examine the FLA. This is very similar to what you made in Mask Exercise 2 above (well ... without a mask, and with two buttons added). The main similarity is the large rectangle movie clip (*Content*), which is moving.
3. Open the Library. You'll see three movie clips there. Double-click the movie clip **Photos** in the Library and examine it. You can play its timeline within Flash, but you can't test it as a SWF *because it is a Movie Clip symbol*.
4. Play it by pressing Enter or Return and study the clip's construction. Notice how this is similar to Lesson 1 in the book—it is just a simple photo slideshow with NO tweens.
5. Now double-click the movie clip **Bobo Moving** in the Library and examine it. It's constructed very differently from the *Photos* movie clip. Note in particular the two tweened layers. You can put multiple tweens inside a movie clip symbol.
6. Click the blue arrow to exit from Symbol Editing Mode.
7. Double-click the Content clip *on the Stage* to go inside it. Up beside the blue arrow, the text should be: Scene 1 > Content
8. Check the timeline. There are several layers in the timeline of this movie clip. One is unlocked. It is named "clips."
9. Select the "clips" layer and drag the **Photos** clip from the Library to the Stage. Position it at Y:0 and somewhere to the right of the number 2 in the clip.
10. Slide the scrollbar so you can see the right end of the big Content clip.
11. Drag the **Bobo Moving** clip from the Library to the Stage. Position it at Y:0 and somewhere to the right of the number 3 in the clip.
12. Save and test the movie. Make sure you try all the buttons.

The purpose of this exercise was to show you how to put multiple movie clips *inside* another movie clip. Sure, it was not difficult, because the movie clips were already made for you. When you are working on your Flash 4 exercise, however, you might get into a mess if you are not thinking clearly about how this exercise worked. So, please consider:

■ The movie clip symbols **Photos** and **Bobo Moving** are completely self-contained. Their buttons and their ActionScript are all inside their own timelines.

■ Both movie clips can be dropped into *the same layer* inside the big Content clip. Movie clips that are self-contained do not interfere with one another.

■ There is no tweening in the "clips" layer. The motion is all *inside* the movie clips.

## 3 / Building a Movie Clip Inside a Movie Clip

There is no mask in this exercise. Your task here is to build two completely new movie clips *inside* the big clip (content_mc) that already exists in this FLA.

1. Open *exercise3.fla* in the folder named Embedded Clips. Test the movie: Ctrl-Enter (Win) or Command-Return (Mac). Let it play all the way until it repeats. Then try the Stop and Play buttons.

2. Close the SWF window and examine the FLA. This Main Timeline is very different—here the movement of the Content movie clip is controlled entirely by ActionScript. *This is similar to the template for Flash 4.*
Do not concern yourself with the script. The purpose of Flash 4 is to give you practice with movie clips—not to teach complex scripting.
3. Here's a good procedure for creating a new complex movie clip inside another movie clip. Read it first, then follow it in Step 4 below.
   a. Double-click the Content clip *on the Stage* to go inside it. Up beside the blue arrow, the text should be: Scene 1 > Content
   b. Create a new layer above all the others. Name it "clips."
   c. Lock all the other layers.
   d. In the "clips" layer, begin by placing or positioning the largest item that will appear inside this new movie clip. (This is especially good if the content will extend to the boundaries of the outer movie clip.)
   e. Select that item (click it once) and Convert to Symbol. Type: Movie Clip. Name it very clearly—so that later you do not become confused by too-similar names in your Library! Example: Photo Animation.
   f. Double-click to go inside the new movie clip. Up beside the blue arrow, the text should be: Scene 1 > Content > Photo Animation (or whatever name you gave to your new movie clip).
   g. Immediately name and lock Layer 1. You might edit or delete this layer later, but not until *after* you have built most or all of the movie clip's functionality.
4. Follow the instructions given in Step 3 above and create a photo animation that begins with a small photo and then tweens to its full size. You are making this *inside* the new movie clip.
   a. Select any photo from the BMPs folder in the Library. Follow Steps 3a through 3g above. Use the photo in Step 3d.
   b. After Step 3g: Create a new layer. Drag the *same* photo out of the Library onto this new layer. Convert it to a symbol (because you are going to tween it).
   c. Consider the locked photo layer as *your guide for positioning* as you Transform the photo symbol to about 30 percent of its original size. Then tween it to 100 percent. Position the 100 percent version to completely cover the photo on the locked layer. That should put the symbol at X:0 and Y:0. (Note that a movie clip has *its own* internal X and Y coordinates!)
   d. Now add a new layer above the photo. Add some text (Classic Static) and write a caption for the photo.
   e. Drag the text keyframe to the *last* frame of the movie clip timeline:
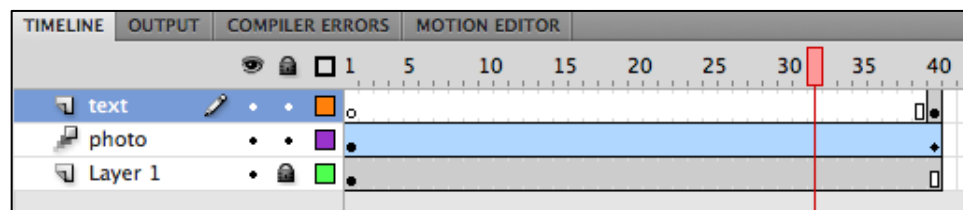
f. Add a layer for buttons and another layer for actions. The movie clip will be self-contained, remember?
g. Create an invisible button to cover ONLY the small version of the photo on frame 1. The user will be clicking on what he or she sees—not on the whole animation, which cannot be seen until AFTER the click.
h. Use the Close button in the Library. It must appear ONLY on the last frame (after the animation completes).
i. Write a listener and a function for *each* button. HINT: They are *the same* as those found inside the movie clips in the previous FLA you were using in this exercise. Copy them! Note that the listener for the Close button must NOT appear until the Close button itself is present on the Stage.
j. When you are finished, your movie clip's timeline should look similar to this:
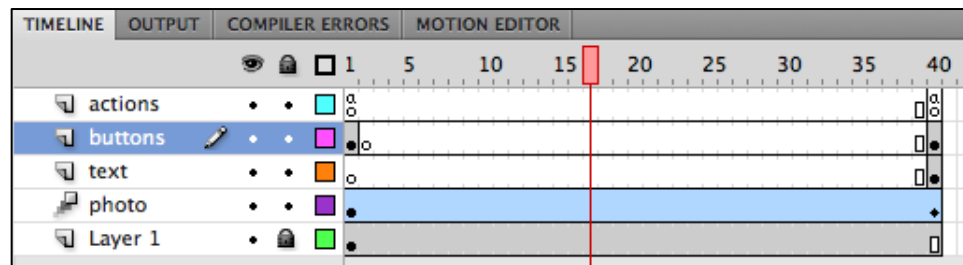
k. If you do not need the locked layer, delete it now.
l. Save and test the movie.

5. **Now you will make the second movie clip.** Go into your Library and *duplicate* the Photo Animation movie clip. A good name might be Photo 2 Animation.
6. You need to convert *a different photo* to a symbol. To do so, you will need to drag the photo out onto the Stage. I recommend that you do it in a new layer, and then simply DELETE the new layer. The new photo symbol will be in your Library.
7. Double-click in the Library to open the *duplicate* animation movie clip.
8. **Here is where all that hard work pays off.** Click once on the tiny photo on the Stage (frame 1 of the duplicate movie clip). Is the layer unlocked? If so, you should see the **Swap** button in the Properties panel.
9. Click the Swap button, and select your new photo symbol.
10. Instantly you have a whole different animation! You will need to change the caption text, of course. You may also need to reposition that text because of the difference in the photos.
11. Open up the big Content clip and position your new animated photo clip on the "clips" layer, somewhere to the right of the first animated photo clip.
12. Save and test your movie.

What you have done in *this* exercise is the objective of the Flash 4 assignment: Design and build self-contained movie clips that work properly.

Your movie clips in Flash 4 do not need to start small and grow big. You can design *any kind of animation* that makes sense with your subject matter. The button script should be the same as it was here. *You do not need frame labels.* You need only `stop;` and `play();`

4 / Embedded Movie Clips Using Functions Outside Themselves

There is an example FLA (*special_script.fla*) for you in the folder named Examples. If you look at the ActionScript on frame 1 inside the timeline for the photo movie clips (it is the same in each movie clip), you'll see something new:

`MovieClip(root).stopMove(e);`

Not new: `stopMove();` is a **function** in frame 1 on the Main Timeline. (It could be *any* function *anywhere* on the Main Timeline.) Normally a movie clip does not have access to any functions that exist outside the movie clip timeline.

New: The way to provide access to that outside function is to precede the function name (inside the embedded movie clip) with `MovieClip(root).` —this tells ActionScript to go all the way "up" to the Main Timeline to find the function. It tells ActionScript where to look: at the "root" level of the movie.

5 / Understanding a Scrubber (Controller Bar)

Now you will build a very simple controller bar with a draggable "handle." Don't worry about making it look beautiful now. Just use simple shapes.

The controller (or scrubber) will have two pieces: a handle and a track. Each one will be a movie clip (so both of them can have instance names!) The two will be inside a third movie clip. Below is an extremely simple design for a scrubber:

Scrubber Exercise, Part 1
1. Open *exercise5.fla* in the folder named Masks.
2. In the "scrubber" layer, draw a wide, skinny rectangle like the one above. Be sure to use a FILL and NOT a stroke. (Strokes have added pixels for width and height, and that would mess up the script.)
3. Convert it to a Movie Clip symbol, name it *Track*, and give it the instance name *trackbar_mc*. **The instance name is important; use this exact name.**
4. Now select it (maybe it is already selected) and Convert to Symbol *again*. This is one way to put a movie clip inside a movie clip. Name it Scrubber and give it the instance name *scrub_mc*. **The instance name is important; use this exact name.**

5. Double-click to go inside *scrub_mc*. Up beside the blue arrow you should see this if you have done everything correctly:
Scene 1 > Scrubber (if you see > *Track,* you went too far in)
6. Name the layer "track" and lock it. Add a new layer on top. Name it "handle."
7. Draw a little shape to be dragged on the track.
8. Convert it to a Movie Clip symbol, name it *Handle*, and give it the instance name *handle_mc*. **The instance name is important; use this exact name.**
9. Save the movie.

The scrubber cannot do anything yet, because it needs ActionScript to work. But you should understand that what you have made is a controller that (depending on what a script says) could change or move things when someone drags the handle left or right.

There is an example of a scrubber (*scrubber.fla*) in the Examples folder.

Scrubber Exercise, Part 2
If a script has been written to be *portable,* it should work with any set of movie clips (and/or buttons) that have instance names that match the script.

The scrubber you just made should work perfectly with the script for Flash 4.

1. Open *slider_template.fla* in the folder named Flash 4 Template.
2. Save and test the movie. Make sure you roll over the Content movie clip and also drag the Handle on the Scrubber.
3. Inspect the Library. The names of the movie clips should be familiar to you!
4. Open the Actions panel and copy ALL the script on frame 1.
5. Go to the *exercise5.fla* you saved (containing the scrubber that YOU made). Paste the script into frame 1 in the "actions" layer.
6. CHECK CAREFULLY to ensure that the big movie clip has the instance name *content_mc*; the mask is *mask_mc*; the scrubber is *scrub_mc*; the track inside is *trackbar_mc*, and the handle is *handle_mc*. (Any typos, and the script will not work.)
7. Do not change anything in the script.
8. Save and test the movie.

You have now done everything necessary to complete the Flash 4 assignment … except ONE thing. Notice that up until now, you have not written or changed ANY script, except when you when creating movie clips to go INSIDE the big clip. This is very important. The Flash 4 assignment is not about writing script; it is about movie clips!

6 / Special Script for the Flash 4 Exercise

1. Go inside the *content_mc* movie clip.
2. Add a new layer and name it "clips."
3. **File** menu > **Import** > Open External Library
4. Find the Examples folder and select *special_script.fla*.

5. In that Library, you will see two movie clips: **Boy Monk Animation** and **Padi Animation**. Drag each one to the surface of the *content_mc* movie clip. You should be in your new "clips" layer, and it should be unlocked.
(Did you know you could use symbols from other FLAs this way?)
6. Save and test the movie. **You will get an error:**
`TypeError: Error #1006: stopMove is not a function.`
7. These two movie clips are looking for a function named *stopMove()* on the Main Timeline. However, that function does not exist in this FLA. We have another function in this FLA, and you need to find its name and fix the two photo movie clips by replacing *stopMove()* with the new name.
HINT: <mark>There are two functions at the very bottom of your script on frame 1.</mark> The giant script you pasted into this FLA. Those functions are very important for the Flash 4 exercise.
8. When you go into the script *inside* the photo movie clips, you'll see *another* function call to the Main Timeline. That function, *startMove()*, also does not exist in this FLA. But there is a corresponding function in the script you copied.

```
function startClip(e:MouseEvent):void {
      play();
      MovieClip(root).pauseSliding(e); // function on Main Timeline
}
function closeClip(e:MouseEvent):void {
      play();
      MovieClip(root).resumeSliding(e); // function on Main Timeline
}
```

9. Above is the script to be used INSIDE embedded movie clips (inside *content_mc*) in the Flash 4 assignment.

Please don't panic. The key (in Flash 4) is to build a working movie clip inside the Content movie clip. <mark>All the script has been provided for you.</mark> It all works. Nothing needs to be rewritten or changed in any way. Just use the script as it is, and it will work.