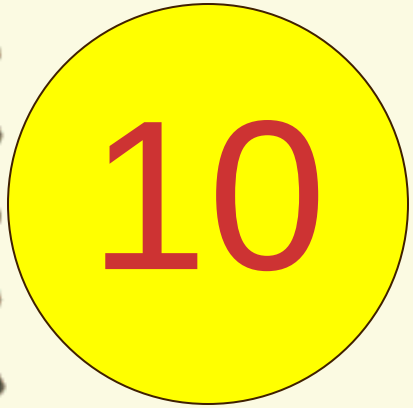
COMSC-032

Web Site Development-Animate

Part-Time Instructor: Joenil Mistal

Feb 13, 2014

Topic 10



Adding Interactivity with ActionScript

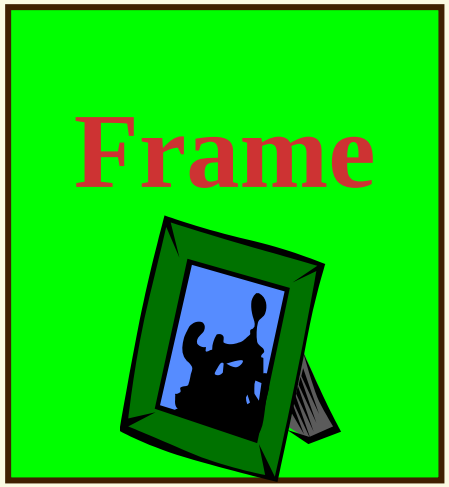
You can use ActionScript to ask your Animate movie for information, and to tell your Animate movie what to do.

Topics: Adding Interactivity with ActionScript

- Introduction to Animate Actions
- Using the Action Panel
- Assign Frame Actions
- Add Action to Movie Clips
- Jump to a Specific Frame or Scene
- Assign Stop and Play Actions
- Load a new Movie into the Current Movie
- Control Instances with Behaviors
- Link a Button to a Web Page
- Customize the Action Panels
- Add a Component

Introduction to Animate Actions

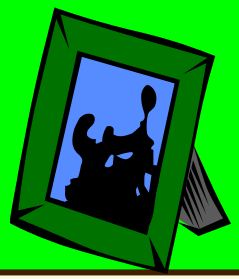
- You can add interactivity to your Animate movies by assigning an **action** or **behavior** to a frame, a button, or movie clip instance.
- Action are based on principles of **cause and effect**.



Introduction to Animate Actions

- The occurrence that triggers the action is called an **event**.
- An **event** might be a click of button or reaching a certain frame in your movie.

Frame



Button

Movie Clip

Introduction to Animate Actions

- The result of the action is the **target**, the object that is affected by the event.
- For example, a target might be linking to a Web page or playing another Animate movie

Introduction to Animate Actions

Action and ActionScripts

- Animate actions are built on a programming language called **ActionScript**.
- This scripting language allows you to write **scripting programming** code, you can certainly write your own actions in Animate.
- However, you do not need to know a scripting language to create actions.
- Animate includes hundreds of **pre-written scripts – Code Snippets**, or actions you can assign.

Introduction to Animate Actions

Using Actions in Animate

- You can use the Actions panel to add actions to **frames**, **buttons** or **movie clips** -mini movies within the main movie.
- You can also assign any of the built-in actions found in the **Behaviors** panel.

Introduction to Animate Actions

Using Actions in Animate

- When you assign an action, Animate adds it to a list of actions for that particular frame or button.
- This list is called an **action list** or script. Animate then executes the actions in the list based on the order in which they appear.

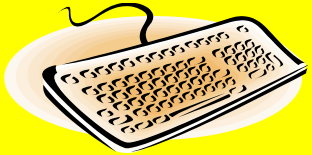
Introduction to Animate Actions

Events

- Anything that causes an action is called an **event**.
- In Animate terminology, an event **triggers** an action in your movie.

Mouse

Keyboard



Frame

Clip

Introduction to Animate Buttons

Events

Animate recognizes several types of events:

- **Mouse** event occurs when you a user interacts with a button.
- **Keyboard** events occur when a user presses a keyboard key.
- **Frame** events are placed in keyframes in your movie.
- **Clip** events control movie clips

Introduction to Animate Buttons

Targets

- A target is the **object affected** by the action.
- Targets are directed toward the current movie (called, the default target), other movies (called the Tell Target) or a browser application (called an external target)

Introduction to Animate Buttons

Targets

- For example, you might place a button in your movie that, when clicked, opens a Web page.
- You direct most of your frame action toward the current movie, which is the default target.

Introduction to Animate Buttons

Types of Actions

- Animate group actions into categories in the **Actions panel**.
- The most common actions include navigational actions such as: Go To, Play and Stop; browser actions, such as Get URL and Load Movie; and movie clip control actions.

Introduction to Animate Buttons

Action Script Versions.

- You can find more than one version of Action Scripts in Animate.

ActionScript 1.0	ActionScript 2.0	ActionScript 3.0

Introduction to Animate Buttons

Action Script Versions.

- You can find more than one version of Action Scripts in Animate.
- **ActionScript 3.0** works best for users familiar with object-oriented programming.
- **ActionScript 2.0** is simpler than 3.0, but is slower to execute in the Animate Player

Using the Actions Panel

- You can use the **Actions panel** to add actions and write your ActionScript for your Animate movies.
- Actions enable you to **add interactively** to your movies.

Using the Actions Panel

The Action panel offers the following:

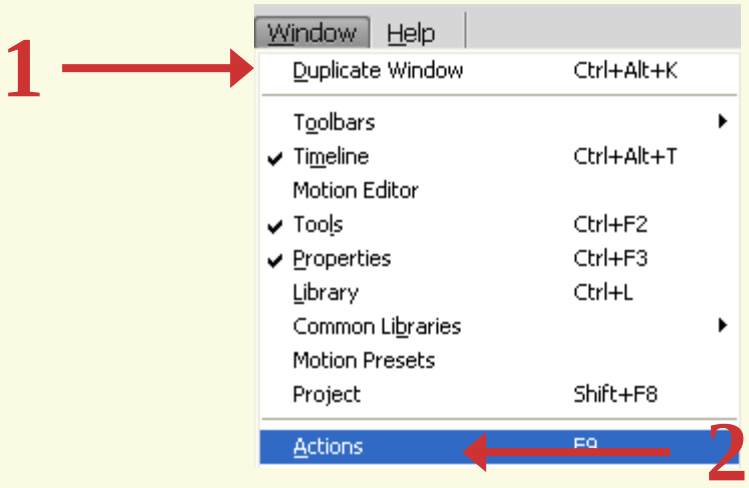
- **Scripting environment** in a single panel, with a script pane for assembling scripts.
- An **Action Toolbox** with pieces of code you can use to write scripts
- **Script Assist** mode to help you write scripts accurately.

Using the Actions Panel

Using the Action Panel

1. Click **Window**
2. Click **Actions**

You press F9 for the Actions Panel.



Using the Actions Panel

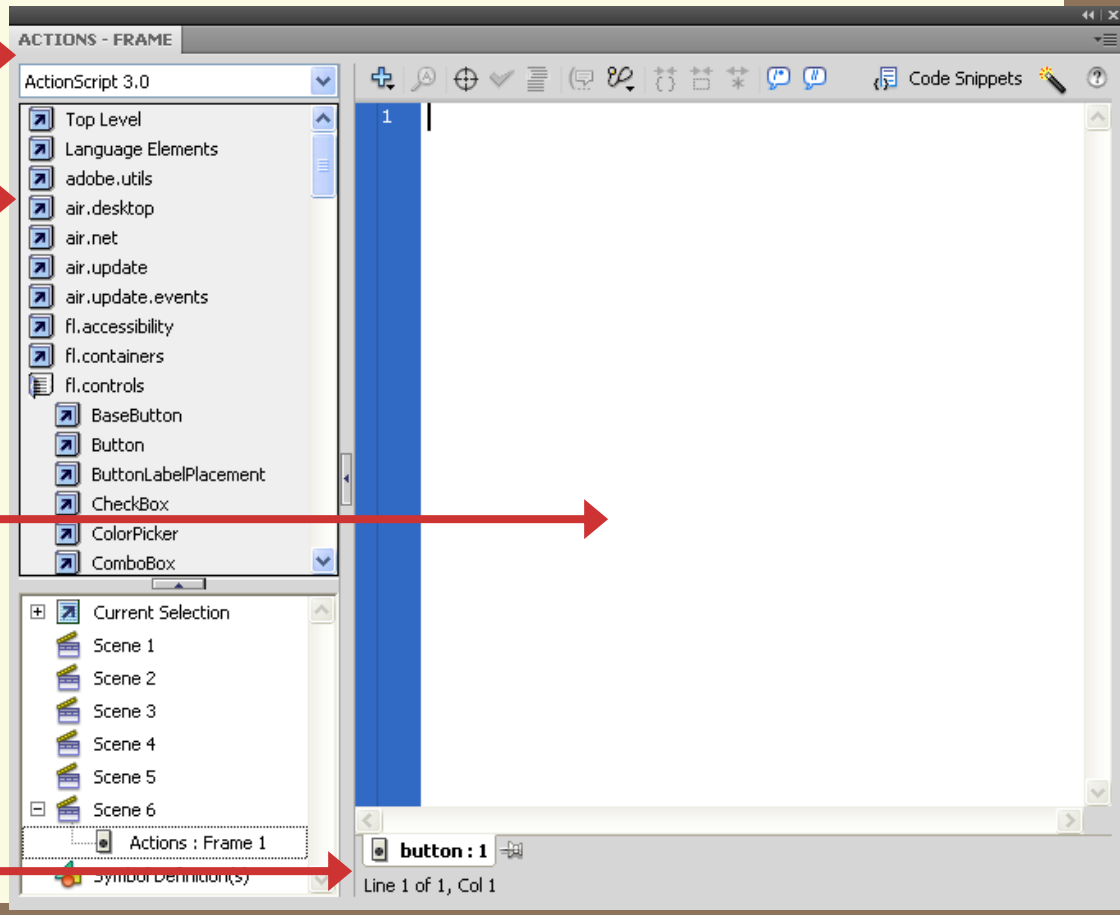
Using the Action Panel

The Action panel opens.

Toolbox - Allows you to select the ActionScript version and displays a menu of all the commands available

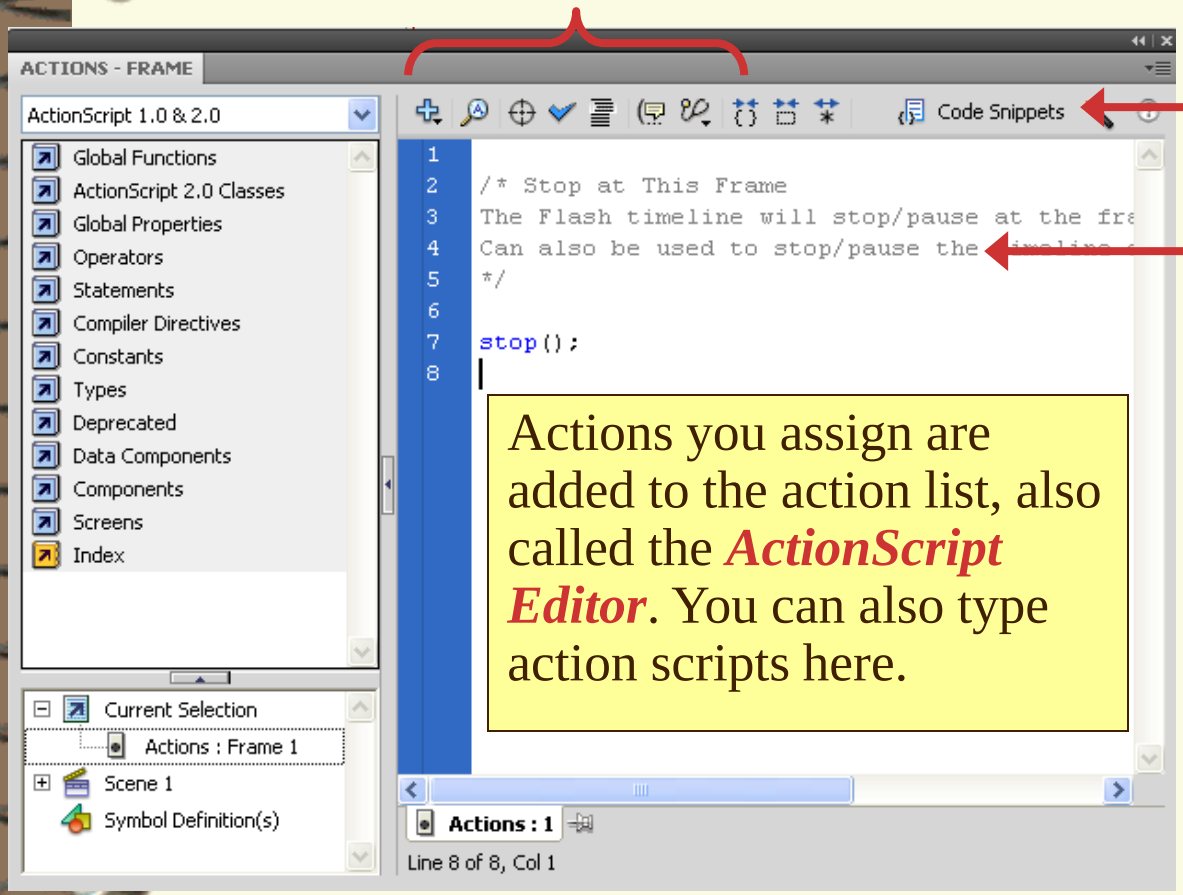
ActionScript Editor - displays a visual representation of your file structure.

Pin Active Script- the current frame or object is listed here.



Using the Actions Panel

You can use the buttons to help you edit and construct your ActionScript statements.



Code Snippets- assist tool to help you write script in Animate.

When you assign an action, a code hint tooltip appears with information about which parameters to assign.

Actions you assign are added to the action list, also called the ***ActionScript Editor***. You can also type action scripts here.

Using the Actions Panel

Can I move the panel out of the way?

You can **move** and **resize** the Actions panel just as you can with other panels available in Animate.

You can also dock the panel.

Click the panel's title bar to quickly hide or display the panel contents.

Assign Frame Actions

- You can use **frame actions** to control the main Timeline or to control a movie clip.
- Frame actions must be placed on a **keyframe**.
- For clarity, it is a best practice to reserve a layer in the Timeline only for actions and to place other elements on the other layers.

Assign Frame Actions

- When you assign an action, it appears in the actions list on the right side of the **ActionScript Editor**.
- As soon as you assign an action to a frame, the frame is marked with a **tiny icon**  of the letter **a**, for action.
- After assigning an action to a frame, you can return to the Actions panel and make changes to the action as needed.

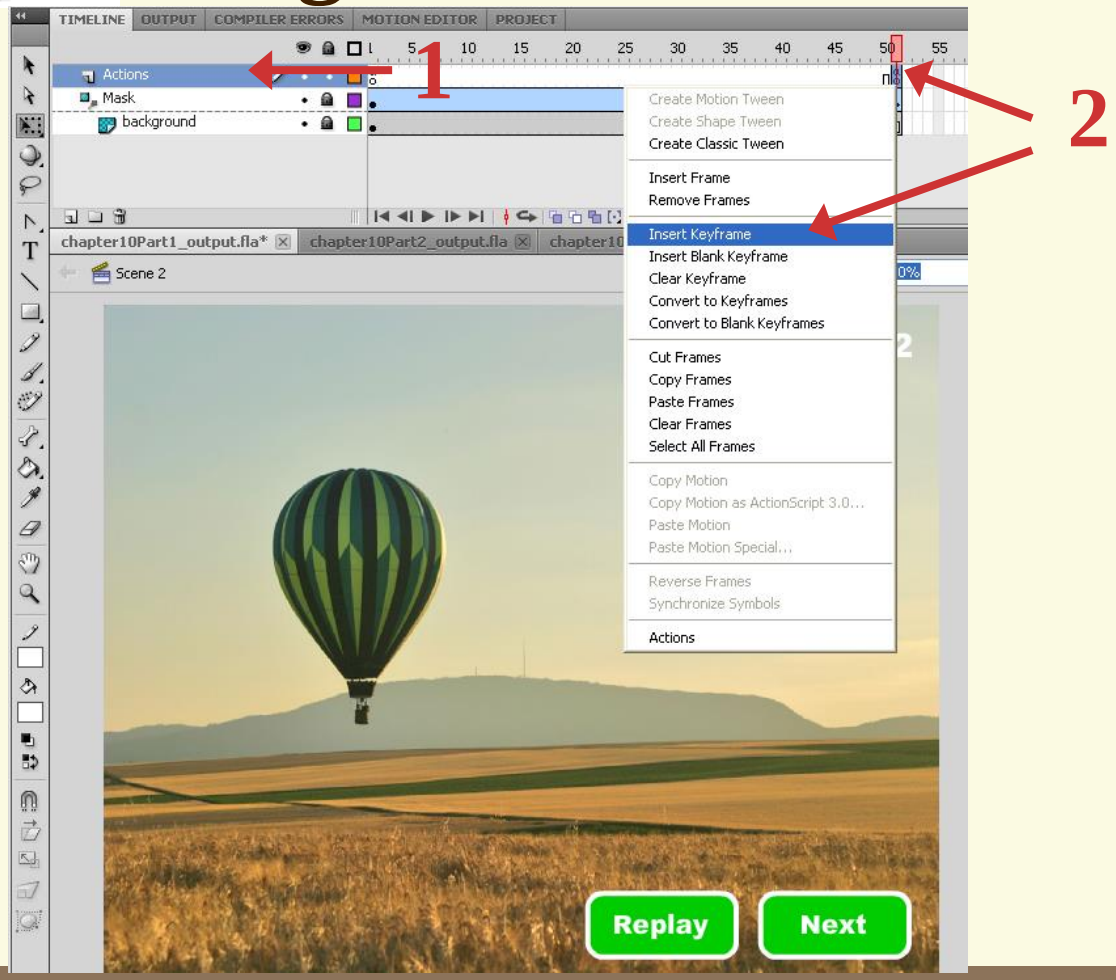
Assign Frame Actions

Assign Frame Actions

1. Insert a **New Layer** for the ActionScripts
2. Right-click on frame 1 and select **Insert Keyframe**.
3. Press **F9** for the ActionScript Panel
4. In the Action window, type **stop();**
5. Close the Action Panel.
   Note: This save the ActionScripts

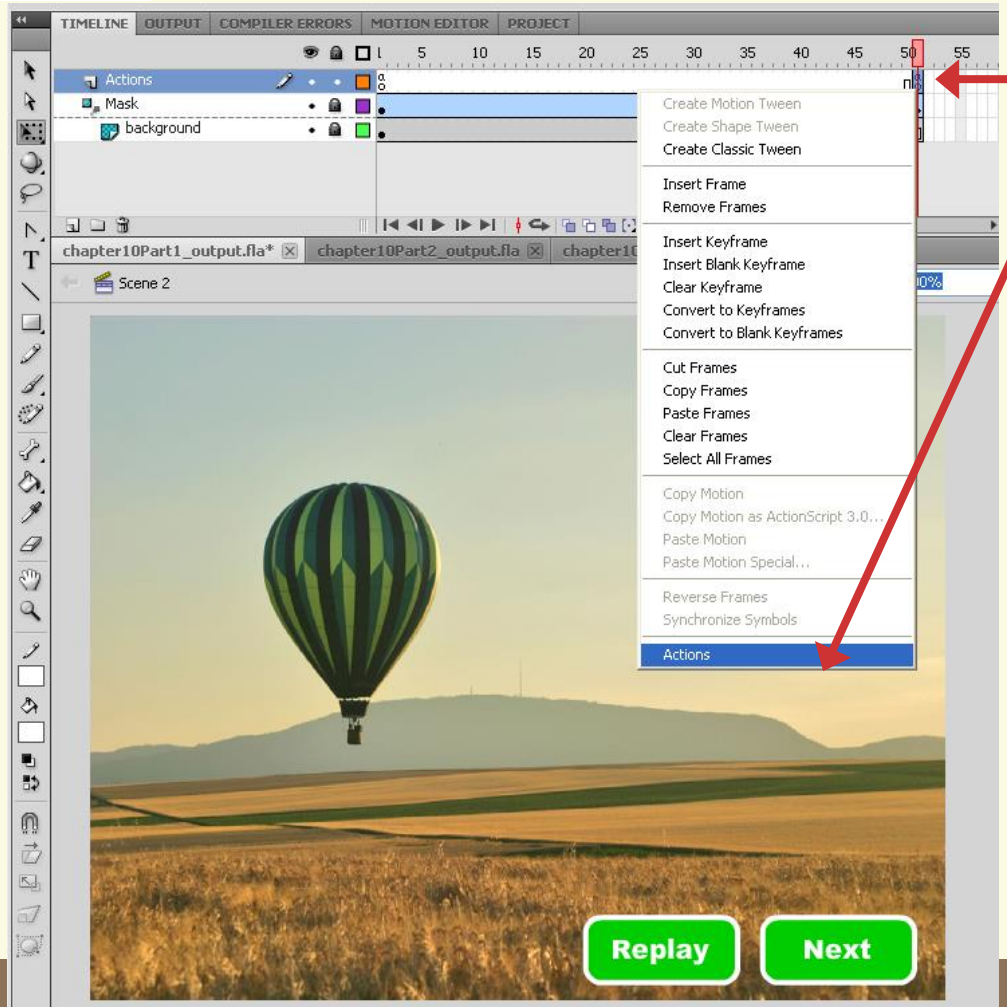
Assign Frame Actions

Assign Frame Actions



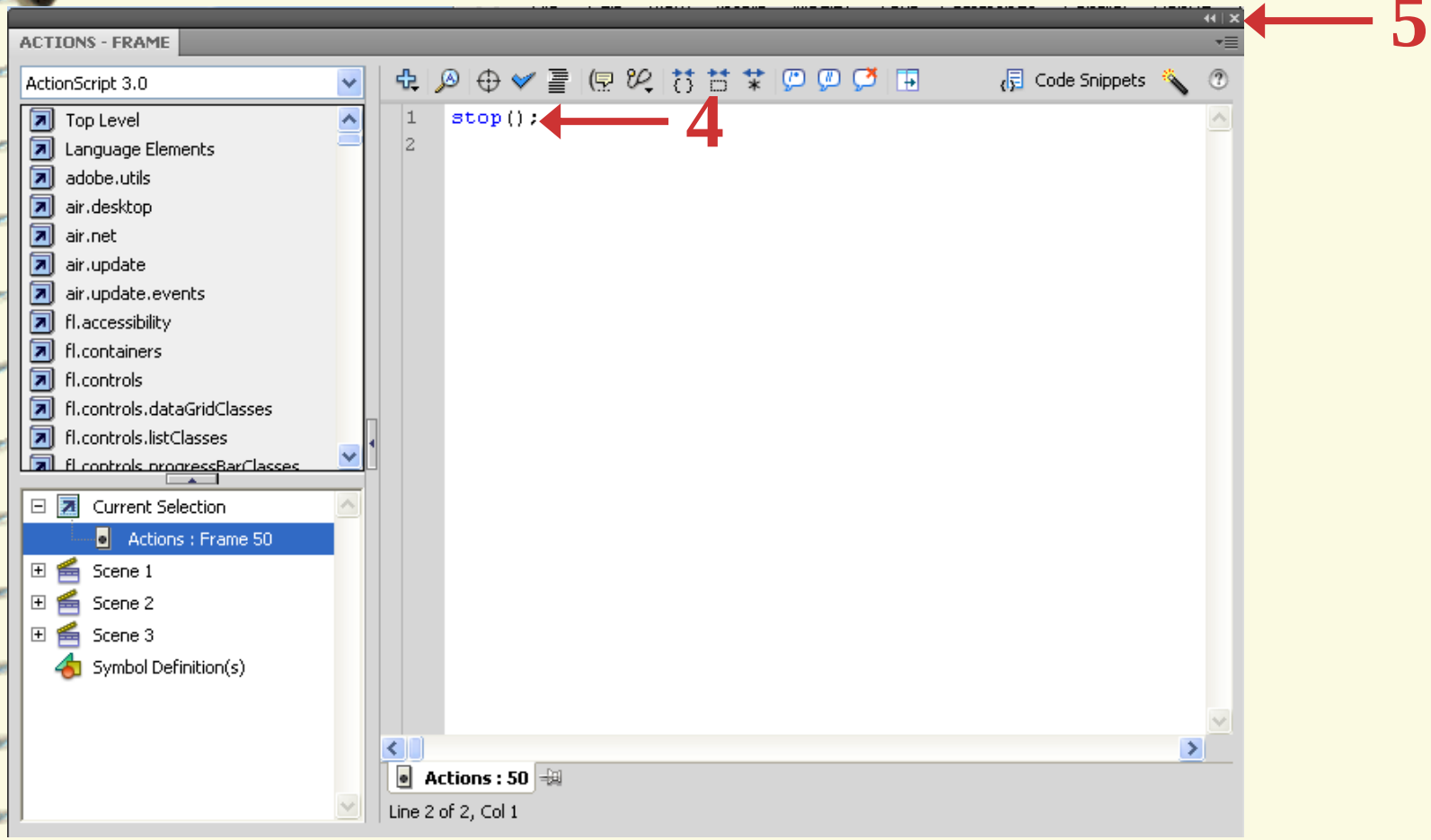
Assign Frame Actions

Assign Frame Actions



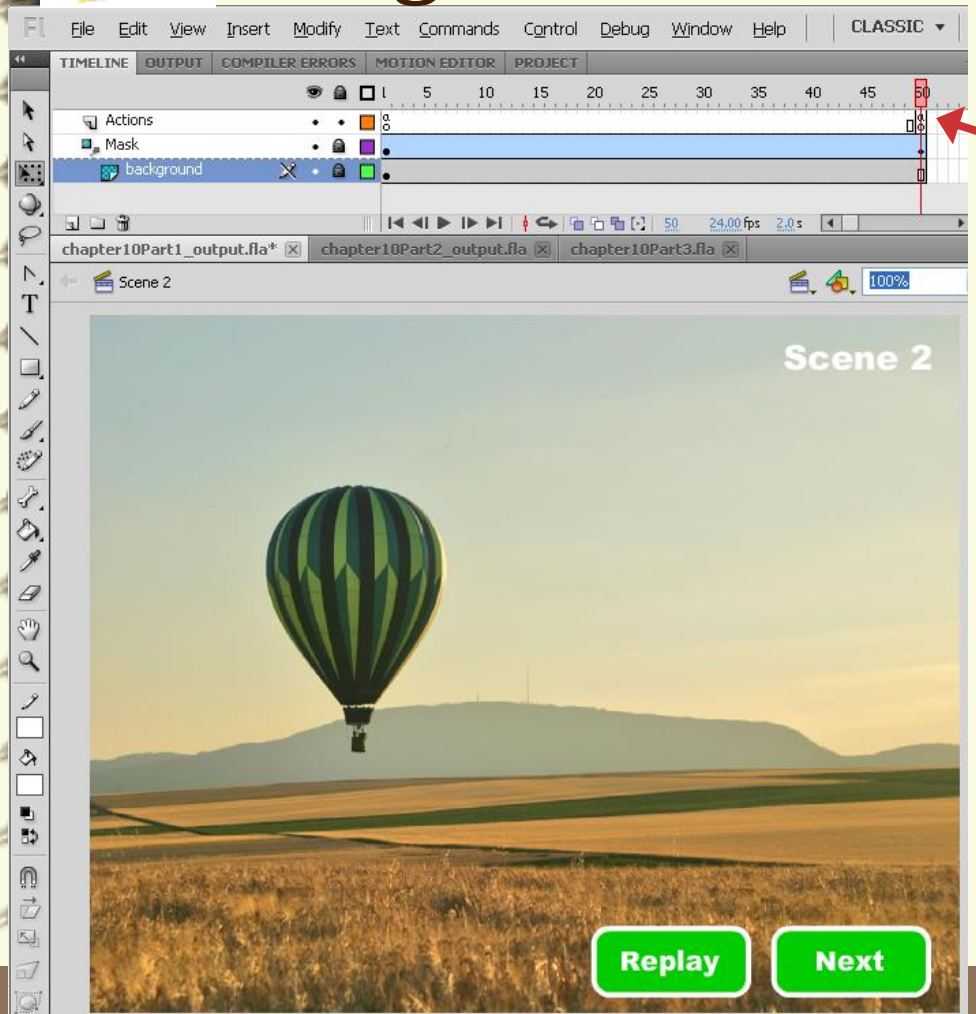
Assign Frame Actions

Assign Frame Actions



Assign Frame Actions

Assign Frame Actions



Animate also add a tiny le to indicate that an action is assigned to the frame.

Assign Frame Actions

Assign Frame Actions

1. Insert a **New Layer** for the ActionScripts
2. Right-click on frame 75 and select **Insert Keyframe**.
3. Press **F9** or right-click your new **keyframe** and select **Actions**
4. In the Action window, type **stop();**
5. Close the Action Panel.
   Note: This save the ActionScripts

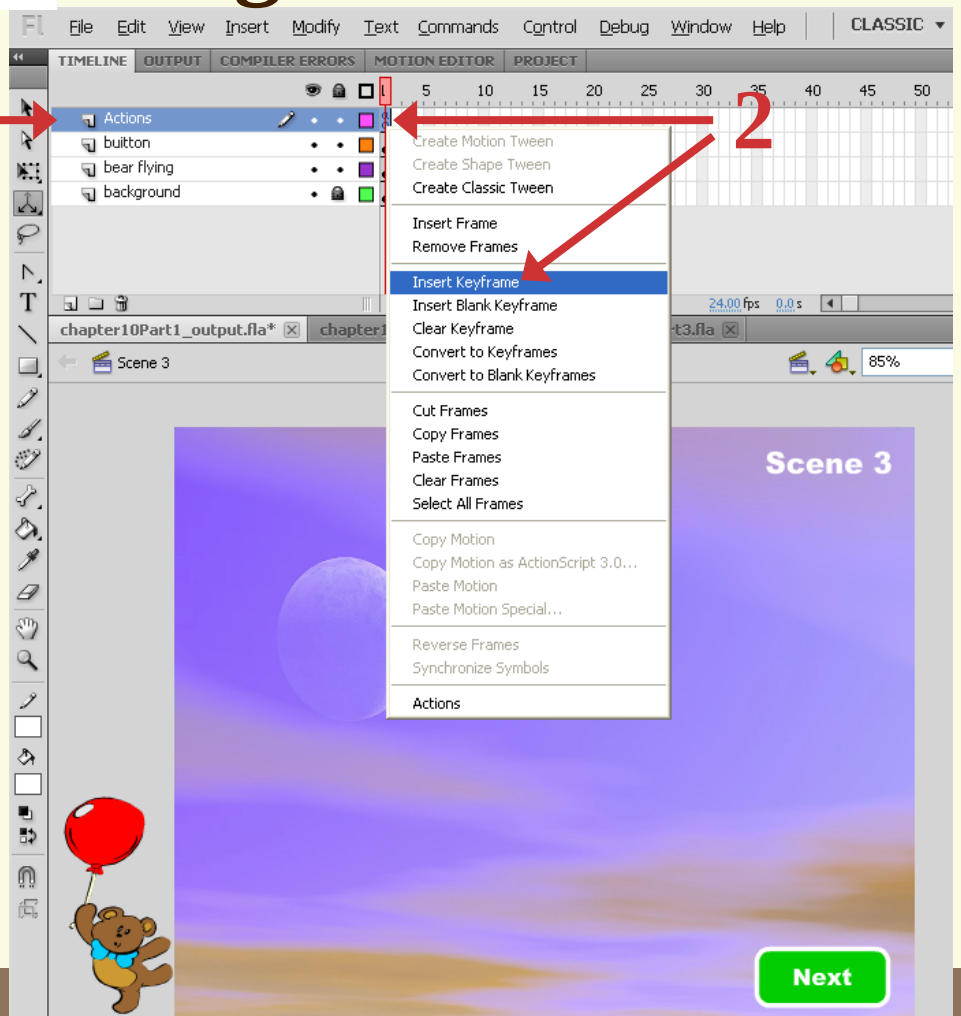
Assign Frame Actions

Assign Frame Actions

1. Insert a **New Layer** for the ActionScripts
2. Right-click on frame 50 and select **Insert Keyframe**.
3. Right-click your new **keyframe** and select **Actions**
4. In the Action window, type **stop();**
5. **Close** the Action Panel.
   Note: This save the ActionScripts

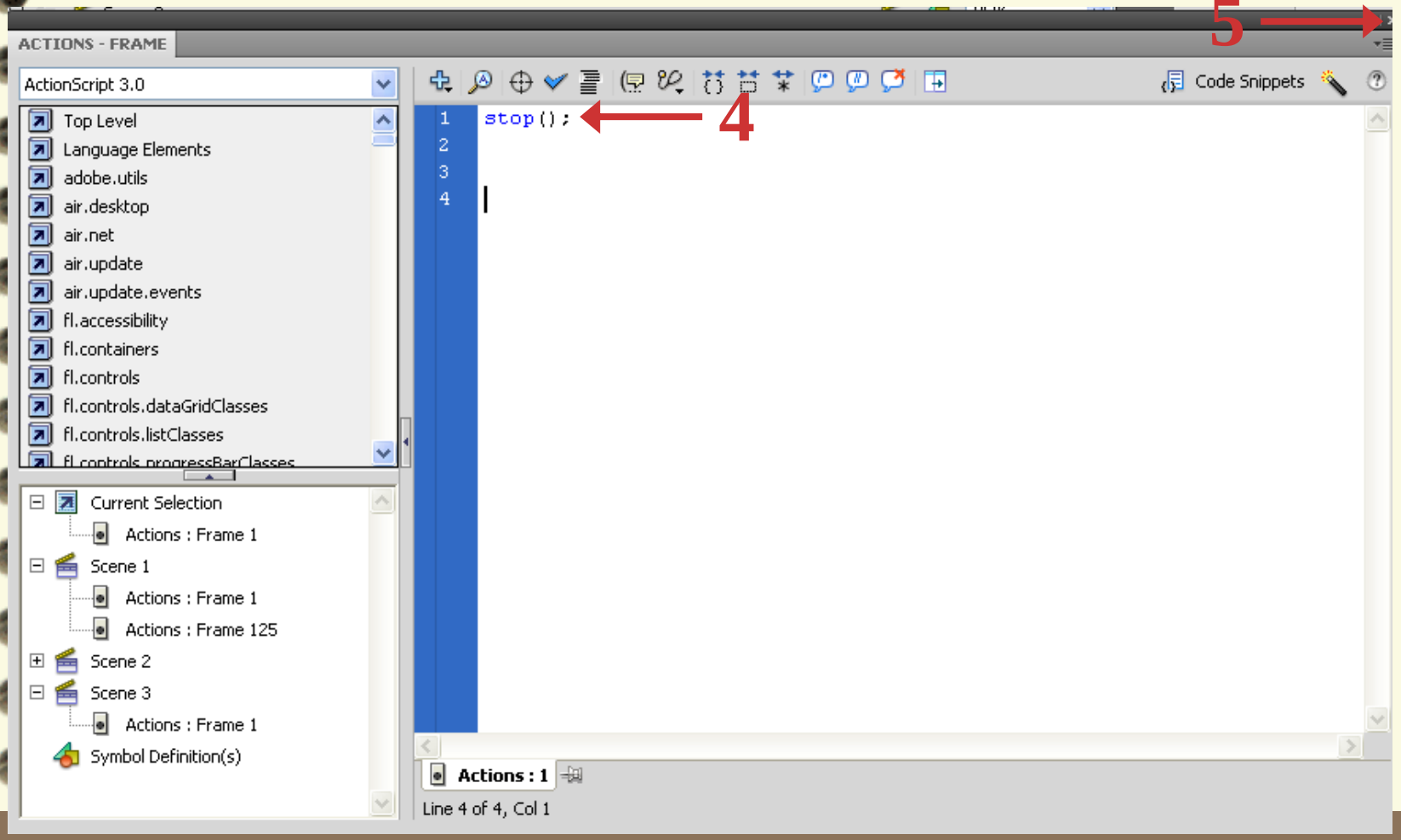
Assign Frame Actions

Assign Frame Actions



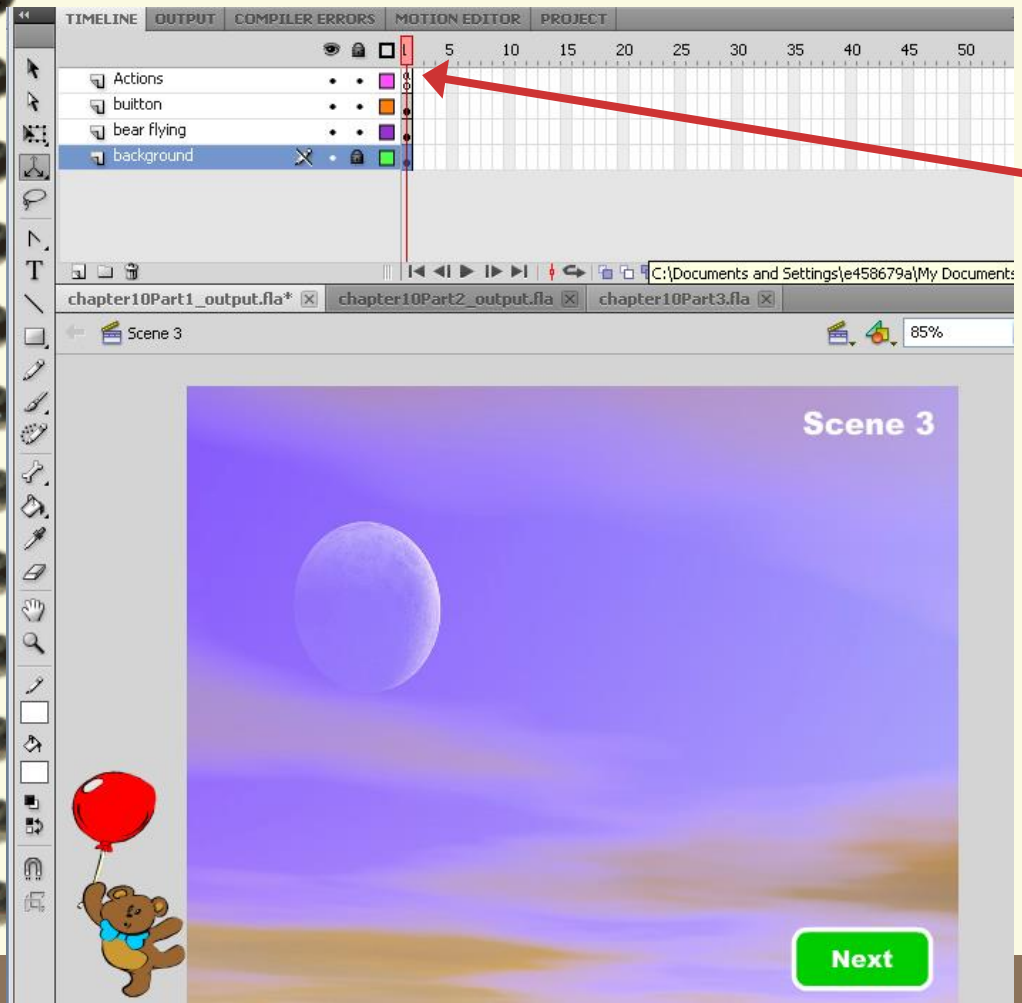
Assign Frame Actions

Assign Frame Actions



Assign Frame Actions

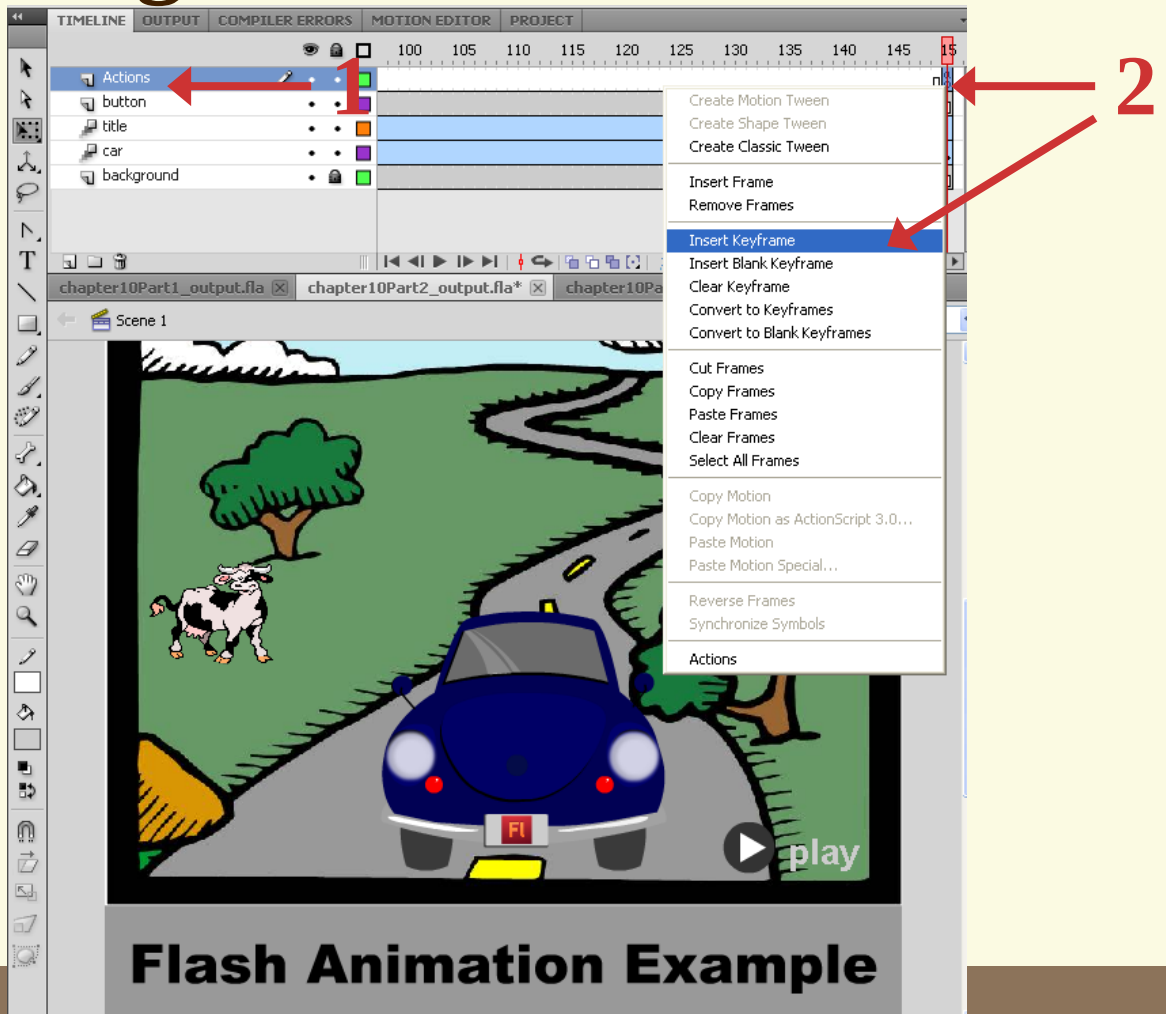
Assign Frame Actions



Animate also add a tiny le to indicate that an action is assigned to the frame.

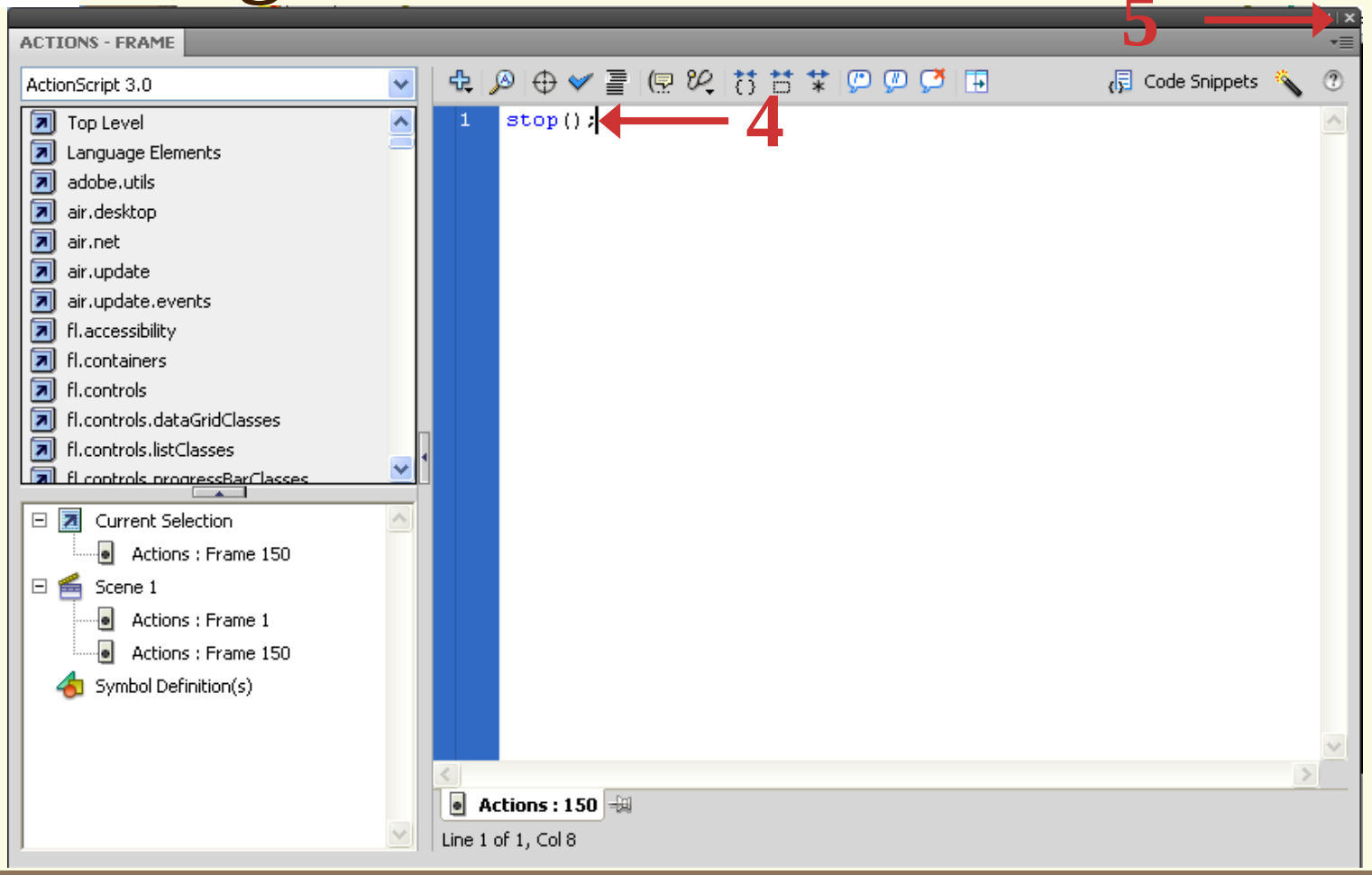
Assign Frame Actions

Assign Frame Actions



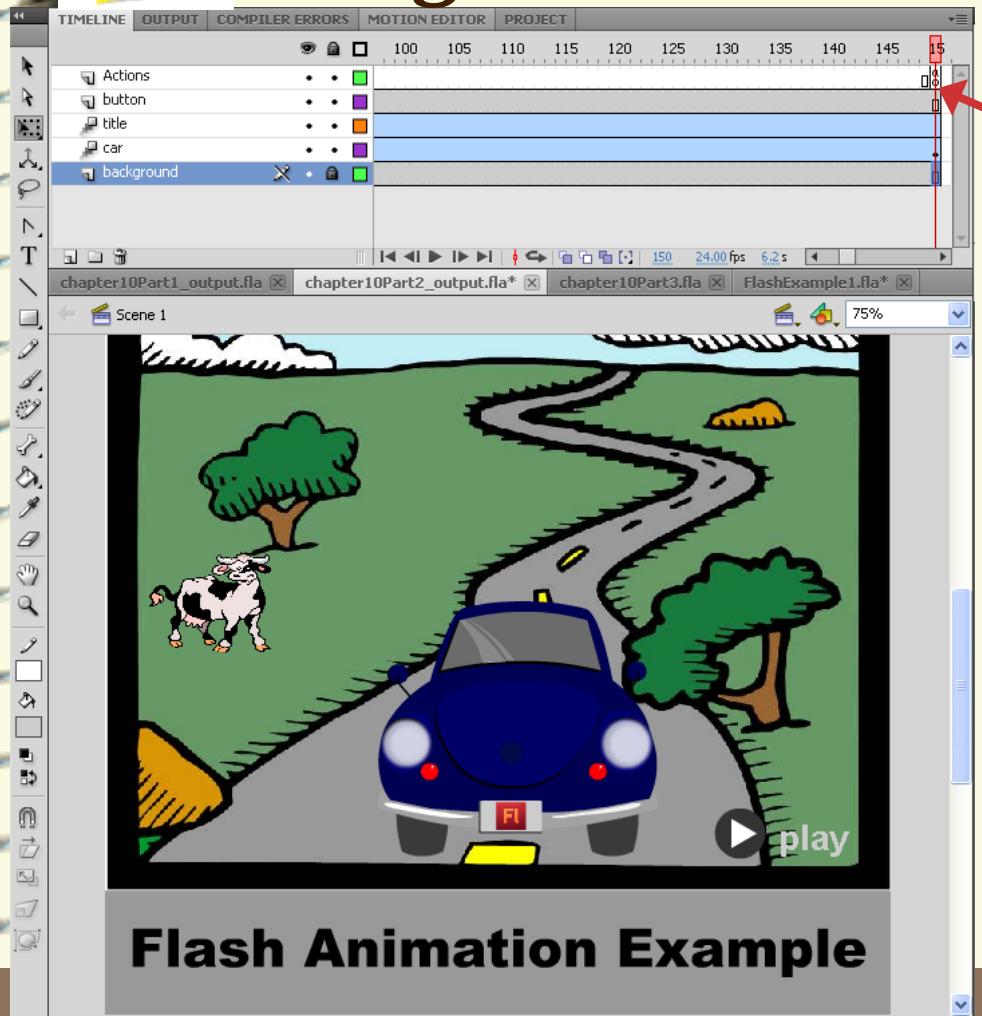
Assign Frame Actions

Assign Frame Actions



Assign Frame Actions

Assign Frame Actions



Animate also add a tiny le to indicate that an action is assigned to the frame.

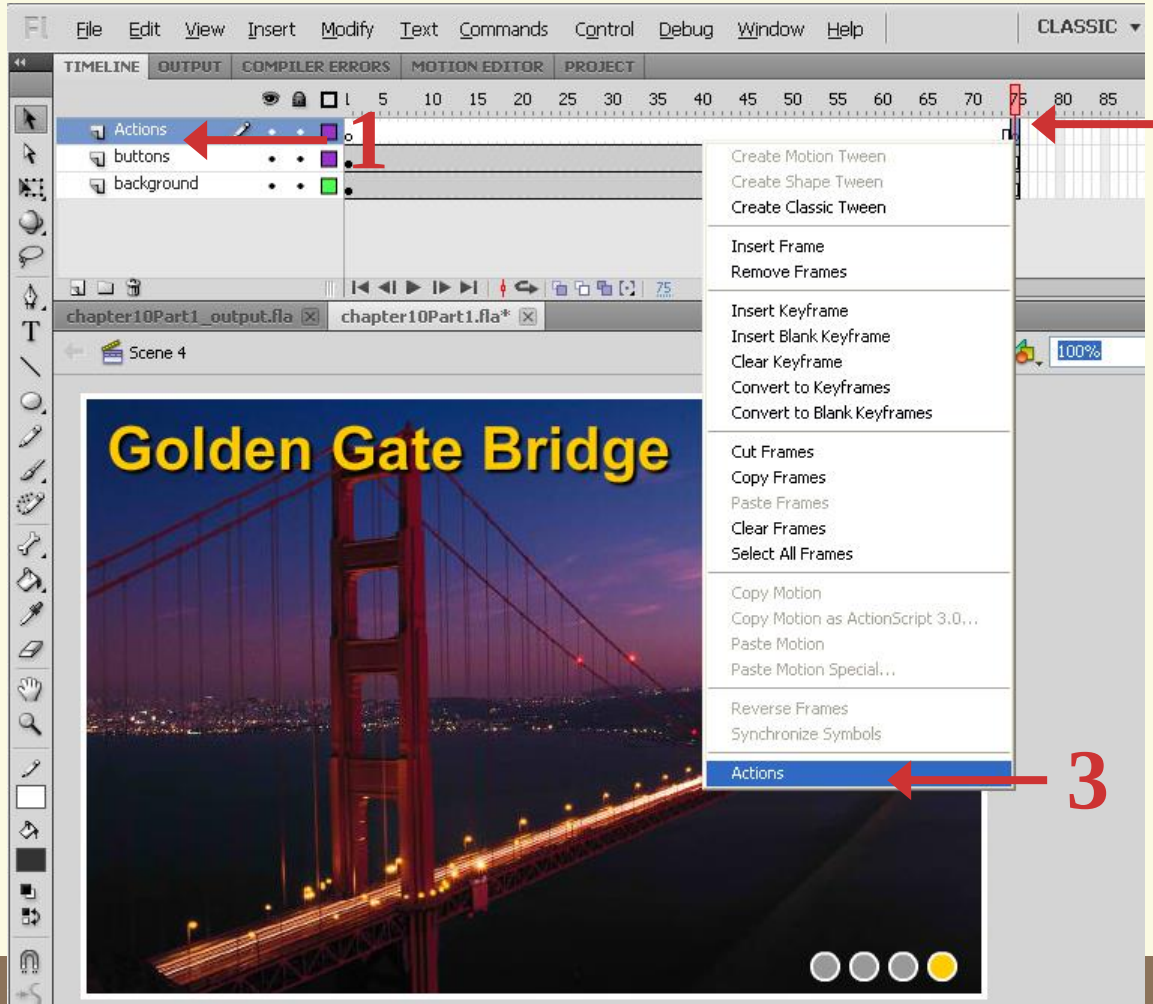
Assign Frame Actions

Assign Frame Actions

1. Insert a **New Layer** for the ActionScripts
2. Right-click on frame 75 and select **Insert Keyframe**.
3. Press **F9** or right-click your new **keyframe** and select **Actions**
4. In the Action window, type **stop();**
5. Close the Action Panel.
   Note: This save the ActionScripts

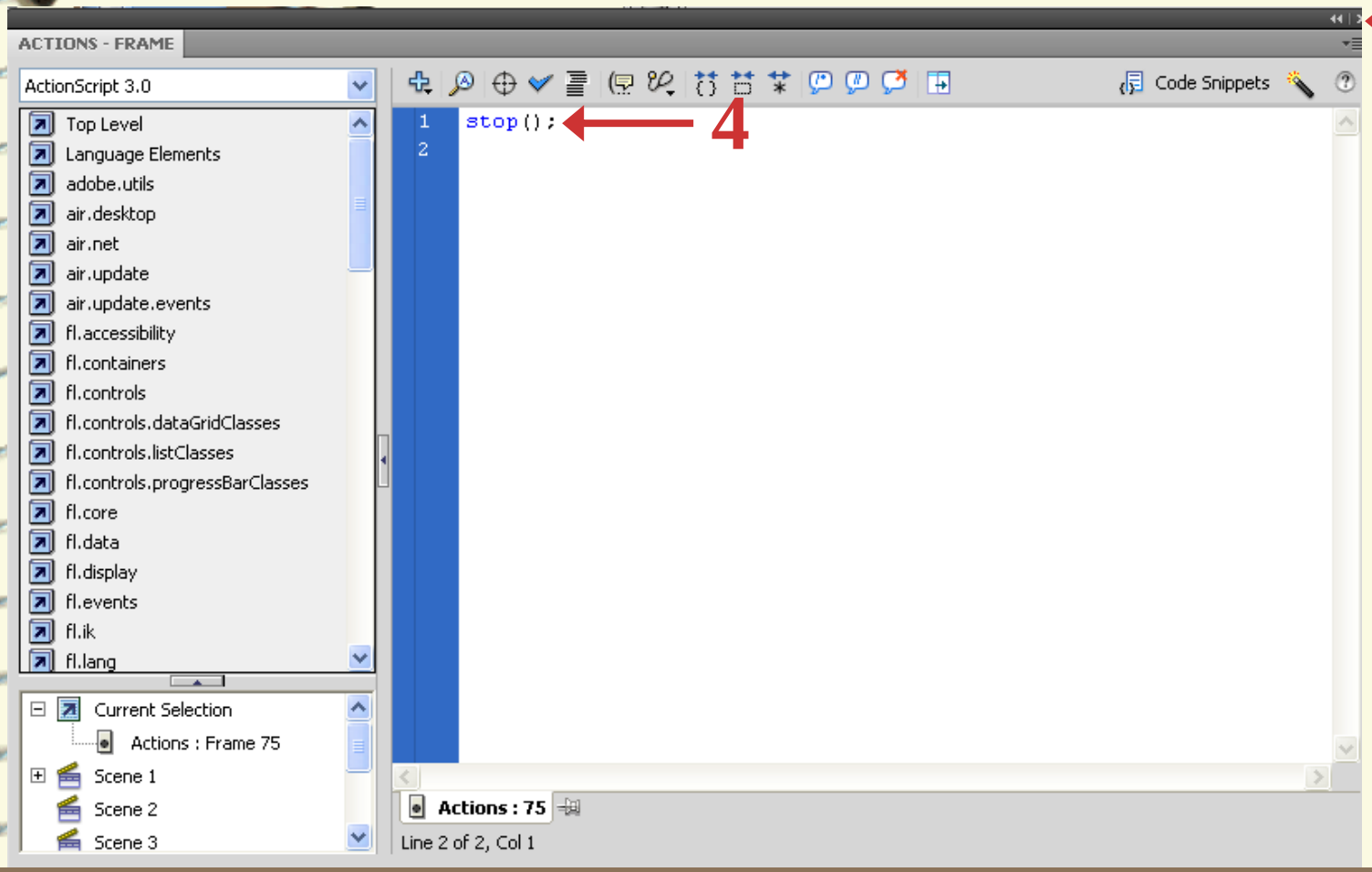
Assign Frame Actions

Assign Frame Actions



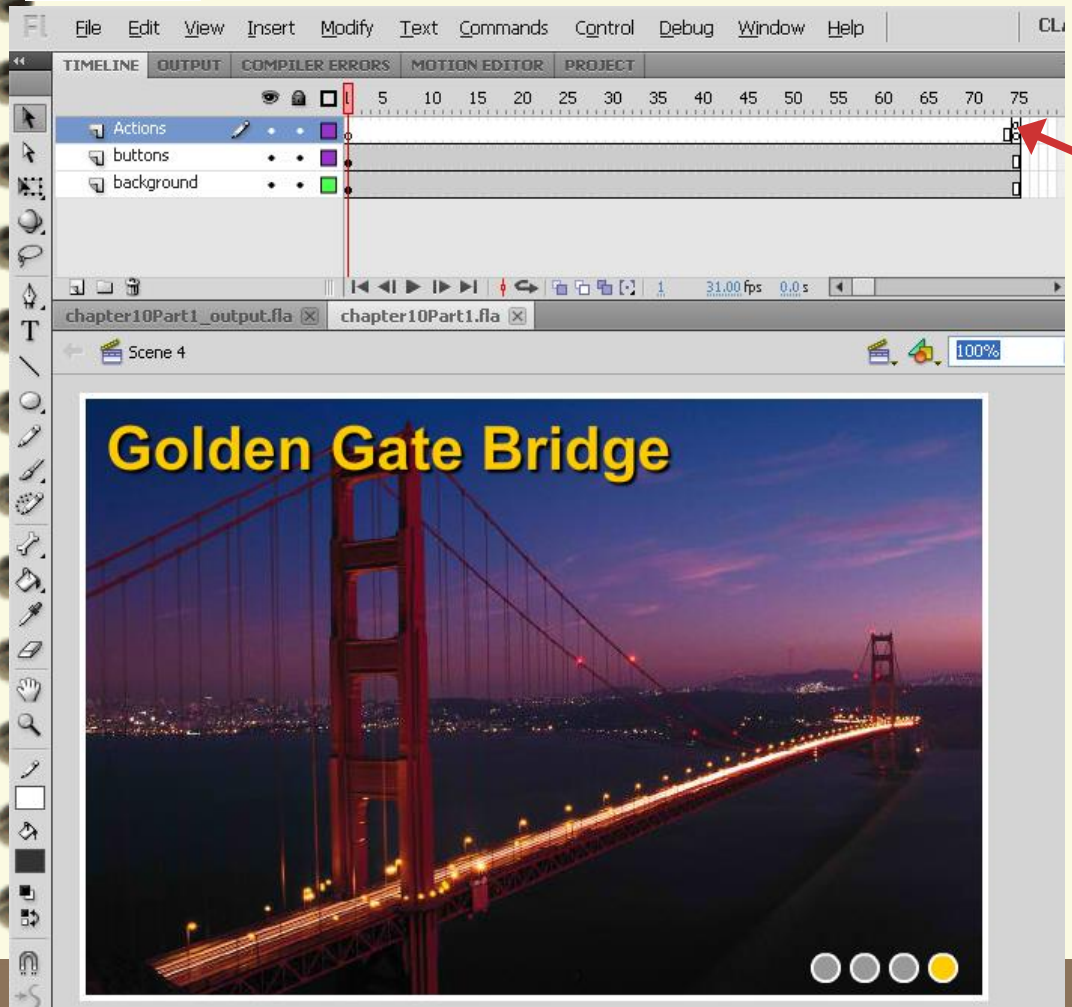
Assign Frame Actions

Assign Frame Actions



Assign Frame Actions

Assign Frame Actions



Animate also add a tiny le to indicate that an action is assigned to the frame.

Assign Frame Actions

How do organize actions in my movie?

To help you clearly identify actions you assign to frames, consider **creating a layer** specifically for actions in your movie.

This technique simplifies the process of finding the action you want to edit.

Assign Frame Actions

How can I edit my frame actions when the play head is on a different frame?

You can click the Pin Active Script button in the Actions panel. Doing so creates a tab in the Actions panel for the current script, which is accessible no matter what frame of the Timeline the play head is on.

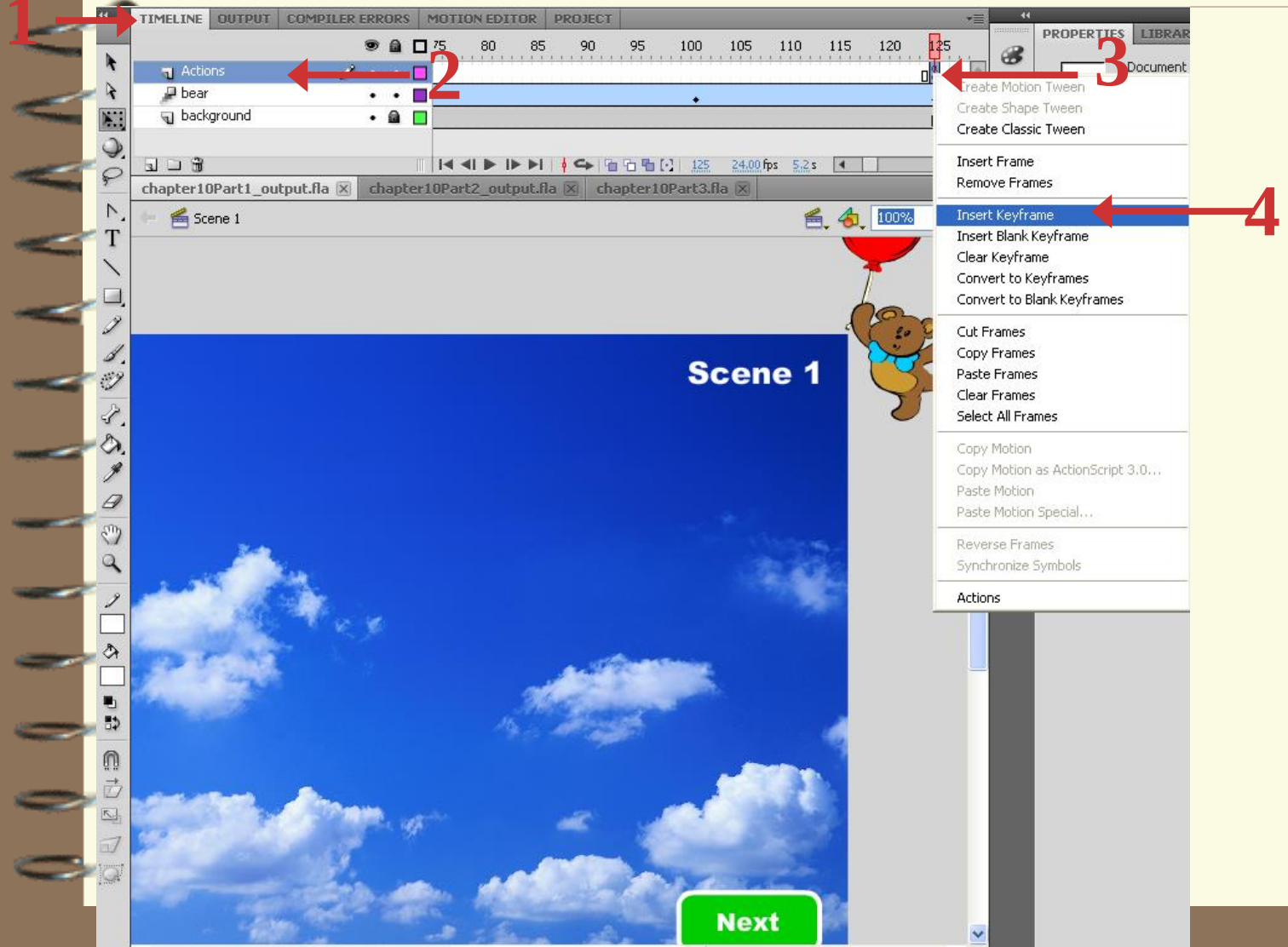
Add a gotoAndPlay() Action to the Timeline

- You can add actions to a movie clip's Timeline to control how it plays.
- You can make a movie clip animation play repeatedly using the gotoAndPlay() action, creating a looping effect..

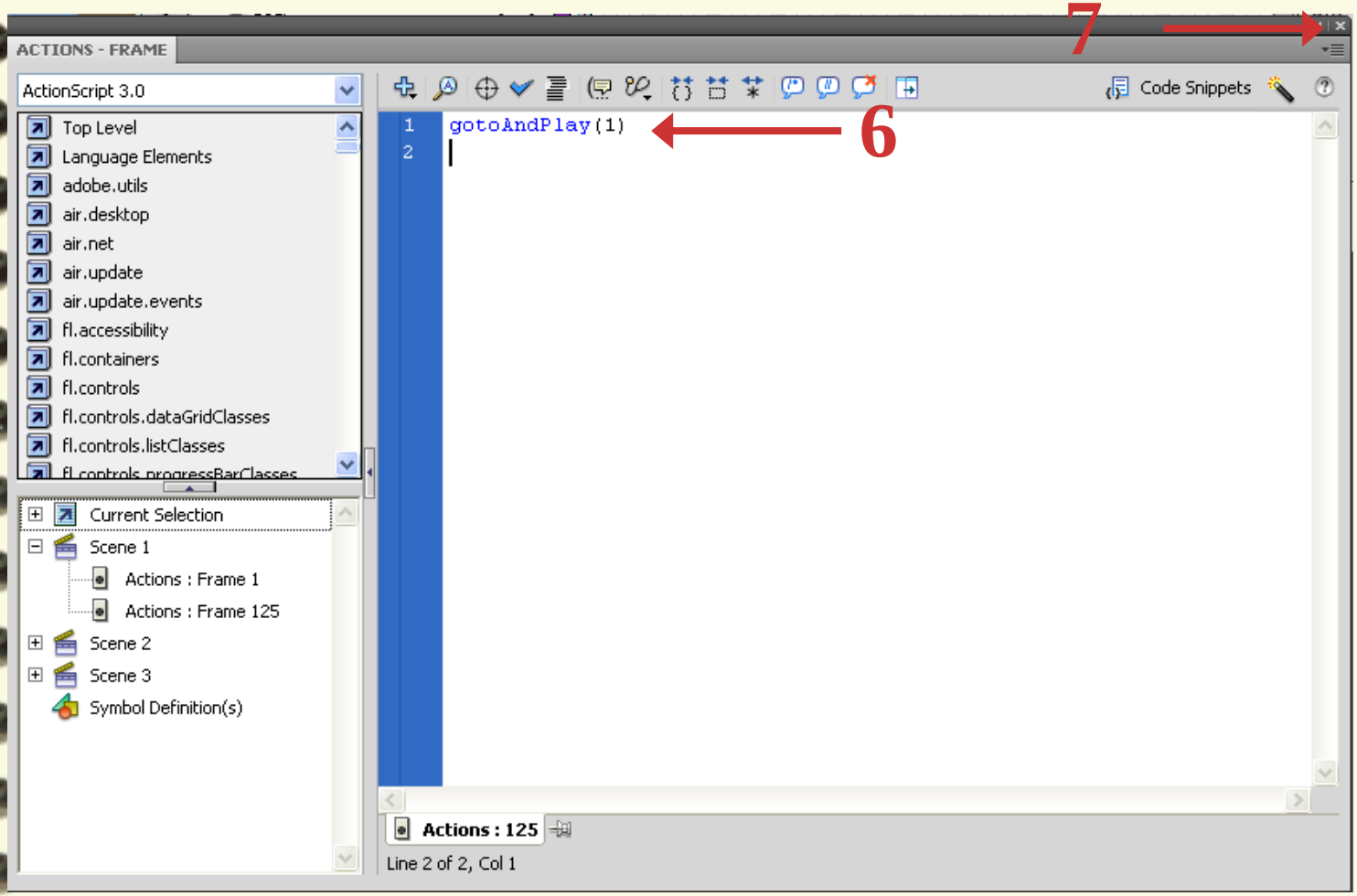
Add a gotoAndPlay() Action to the Timeline

1. Open the **Timeline.**
2. Insert a **New Layer** for the ActionScript.
3. Right-click on frame 125
4. Select **Insert Keyframe**.
5. After inserting the keyframe, press **F9** for the ActionPanel.
6. In the Actions panel, type ***gotoAndPlay(1);***
7. Close the Action panel.
   Note: This save the ActionScripts

Add a gotoAndPlay() Action to the Timeline



Add a gotoAndPlay() Action to the Timeline



Add a gotoAndPlay() Action to the Timeline

Can I preview my movie's frame actions without publishing or exporting?

Yes. Click Control and then Enable Simple Frame Actions. Animate allows certain actions like play(), stop(), and gotoAndPlay() to work when you play through the Timeline.

Working with Instances in ActionScript

- Any instances of a button, movie clip or other object with an instance name can have properties.
- These properties can be read and modified.
- Instances can also have functions that can be called that affect their behavior.

Working with Instances in ActionScript

Instance Names and Target Paths

- Any instances that you want to be able to talk to through ActionScript needs an instance name.
- This name allows you to target the instance you want.
- You need to use the full target path to talk to an instance that is nested inside other objects.

Working with Instances in ActionScript

Instance Names and Target Paths

- For example, from the main Timeline, to target a movie clip with the instance name “ball” on them main Timeline, you would use the following:

  this.ball

- To target a movie clip named ball that is inside an object named field that is inside an object named stadium, you would use the path:

  this.stadium.field.ball

Working with Instances in ActionScript

Relative Target Paths in Animate

- Animate reserves three keywords to make it easier to access your movie clips and objects in ActionScript 3:

  this, parent and root

- The keyword refers to the Timeline you are currently working on.
- The parent keyword refers to the object that your current objects is inside.

Working with Instances in ActionScript

Relative Target Paths in Animate

- For example, if you are writing a script on the Timeline of ball in the previous example, and you want to reference field, you target path would be the following:

  this.parent

- The root keyword always refers to the main Timeline of your Animate movie.
- So, to reference field another way from the Timeline of ball, you could use the following path:

  root.stadium.field

Working with Instances in ActionScript

Getting Instance Properties

- You can ask an object what the values of certain properties are in ActionScript.
- To find the x position of the ball movie clip from the previous example, you could use the following syntax:

```
var x_position = this.stadium.field.ball.x;
trace(x_postion);
```

Working with Instances in ActionScript

Setting Instances Properties

- You can also set instance properties through ActionScript
- To set the alpha of the ball movie clip to 50 percent, you can assign the value of the alpha property to 0.5 as showing the following:

this.stadium.filed.ball.alph= 0.5;

Working with Instances in ActionScript

Calling Instance Methods

- An instance method is a function that specifies a behavior that your object can have.
- Movie clip objects have instance methods like play (), stop(), gotoAndPlay(), gotoAndStop(), nextFrame(), and prevFrame().

Working with Instances in ActionScript

Calling Instance Methods

- To call an instance method, you first type the target path to your object, and a dot (.), and then the method name.
- For example, to tell a movie clip with the instance name ball to go to its next frame, you would use the following script.

```
this.ball.nextFrame();
```

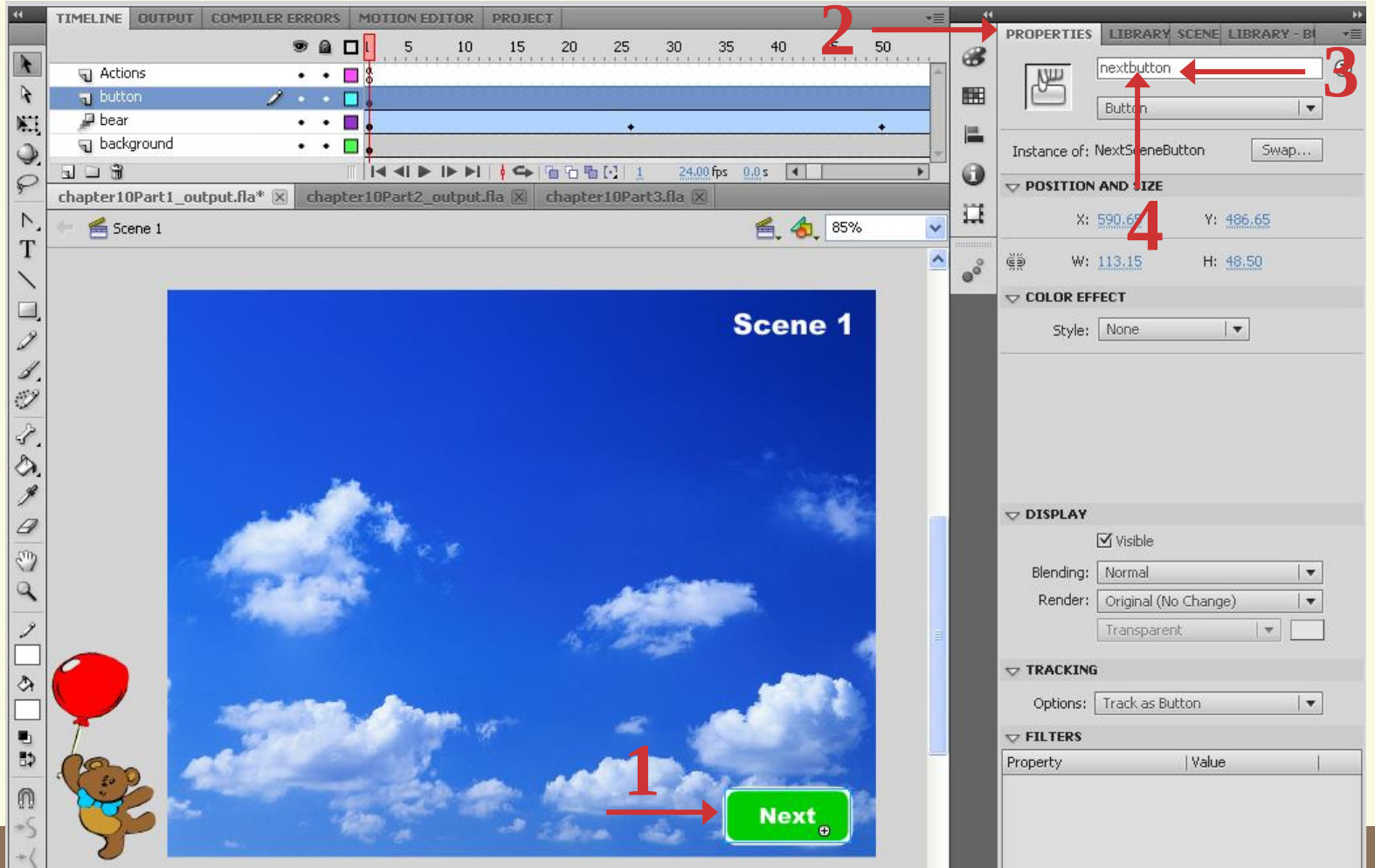
Working with Instances in ActionScript

Naming an instance

1. Click on the **symbol** on the stage
2. Select the **Properties** panel
3. Click on **instance name** box
4. Enter a **unique name** for the instance
5. Repeat steps 1 to 3 for all the symbols that will be used in ActionScript

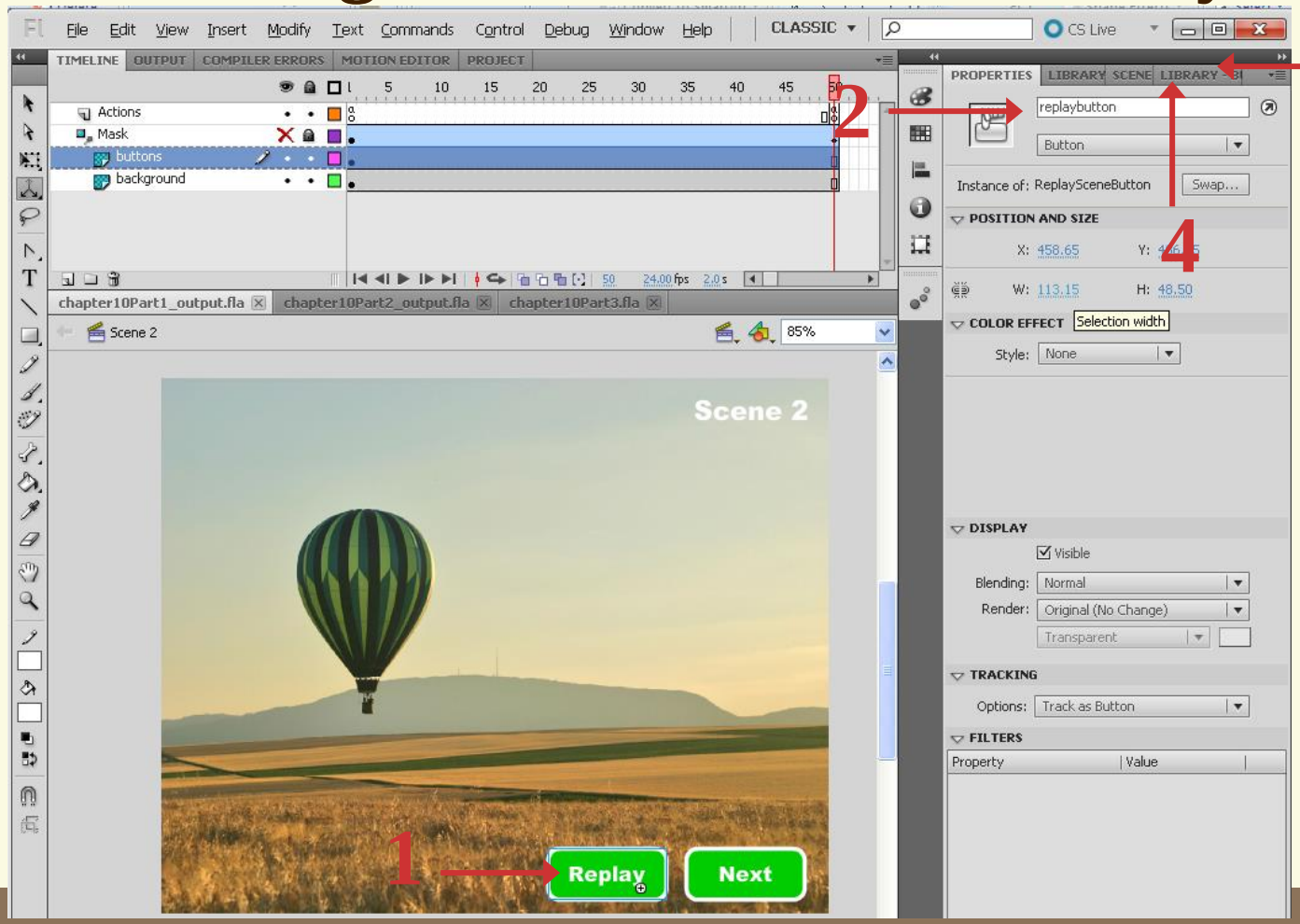
Working with Instances in ActionScript

Naming an instance for button symbol



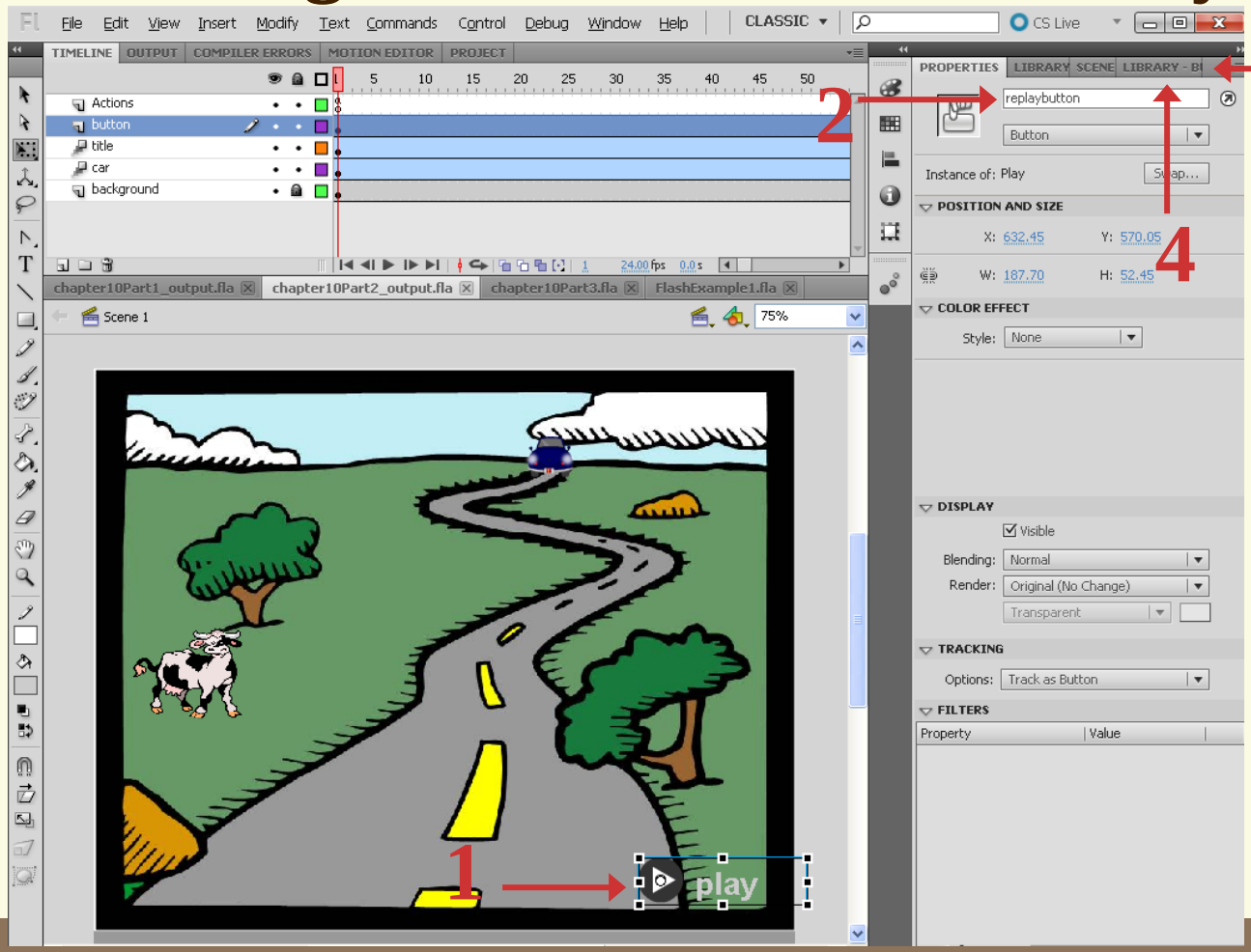
Working with Instances in ActionScript

Naming an instance for button symbol



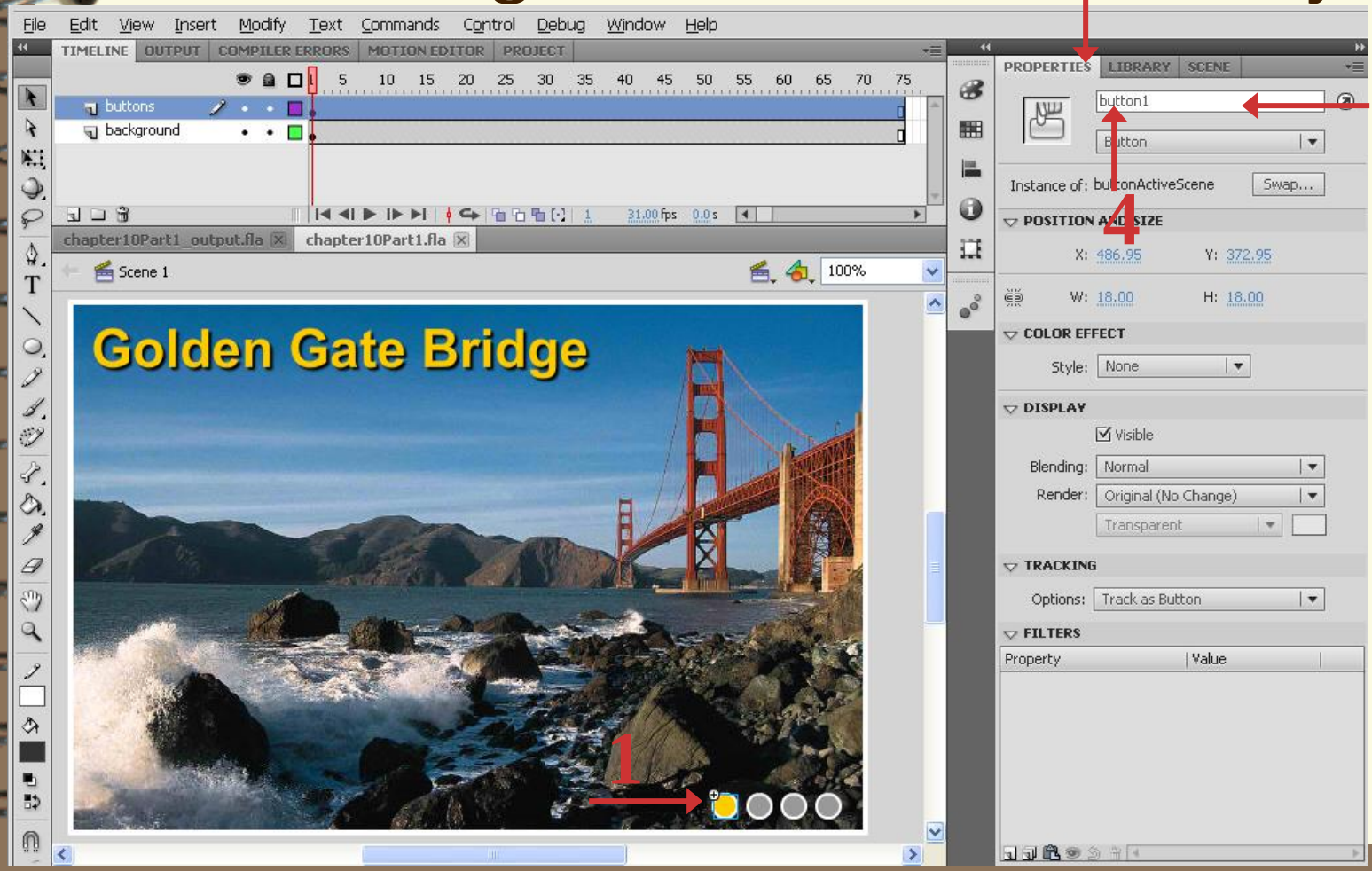
Working with Instances in ActionScript

Naming an instance for button symbol



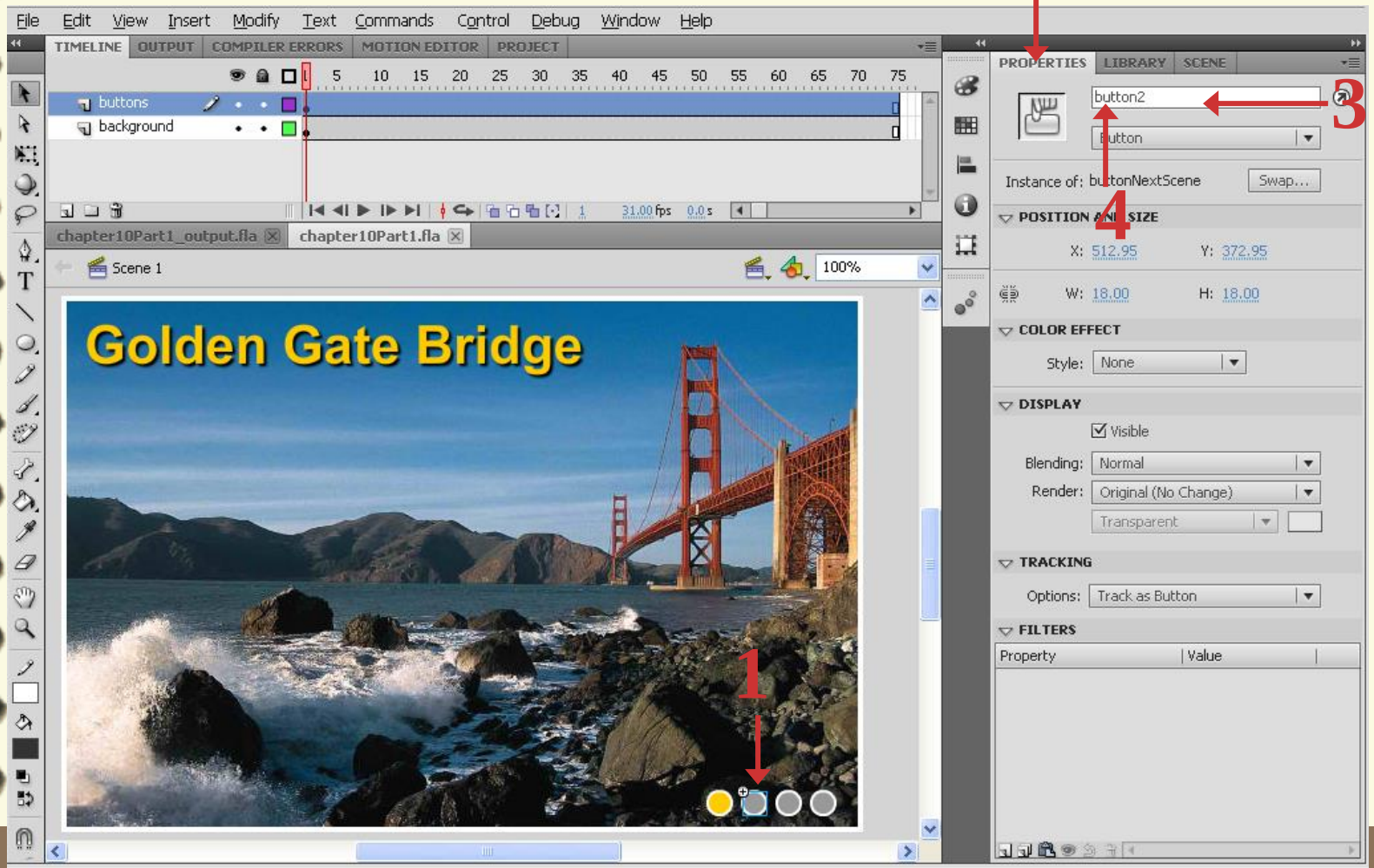
Working with Instances in ActionScript

Naming an instance for button symbol



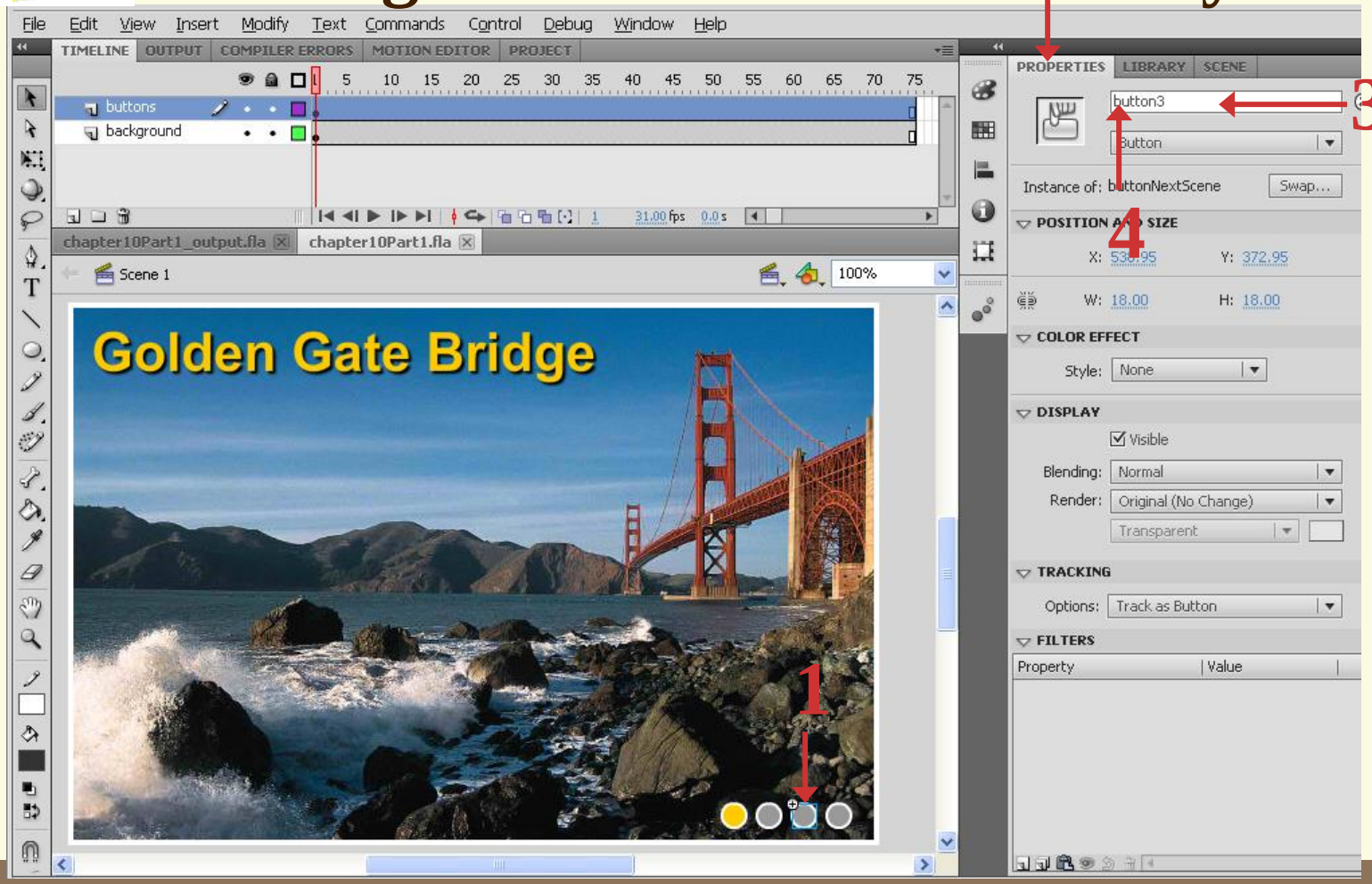
Working with Instances in ActionScript

Naming an instance for button symbol



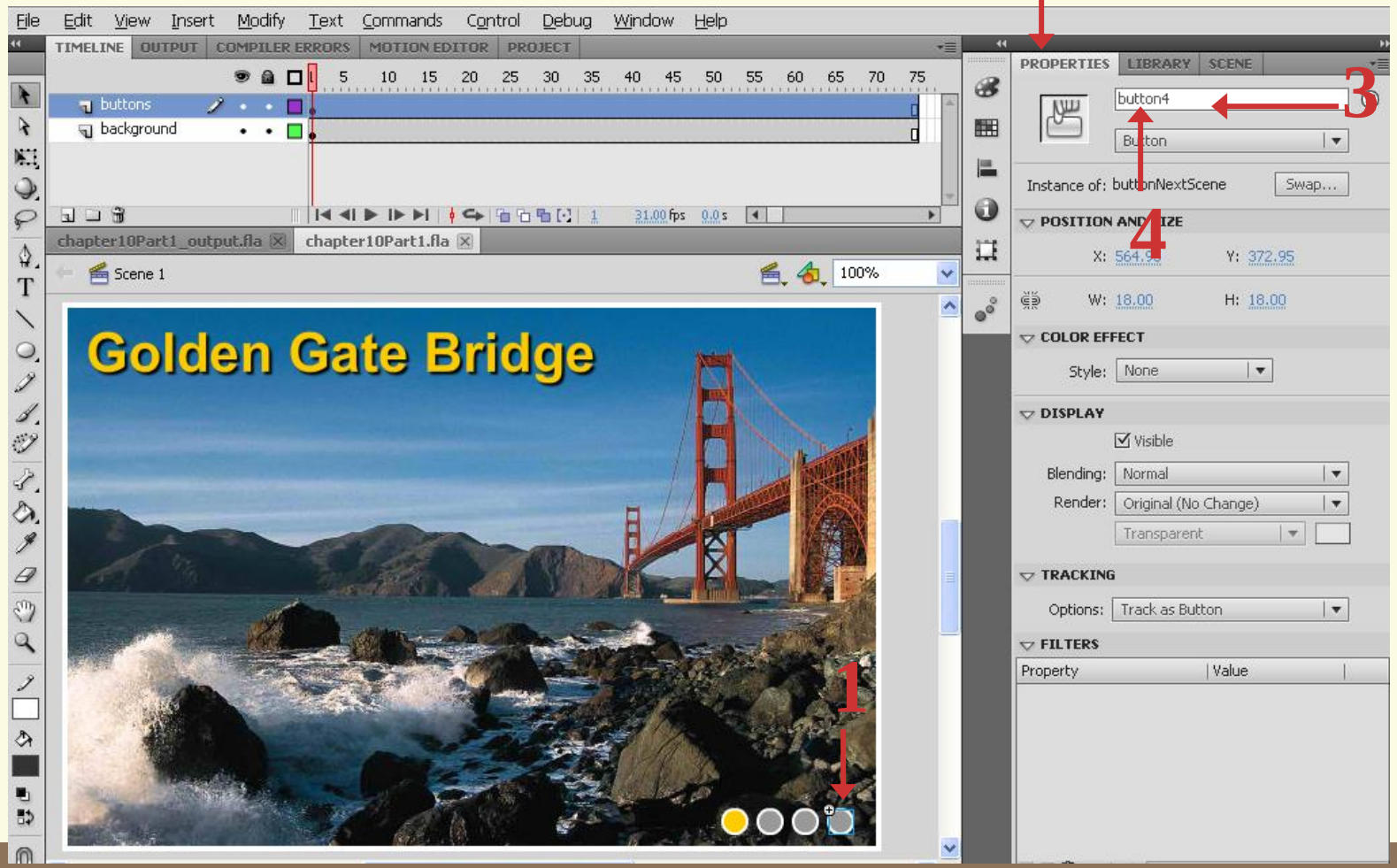
Working with Instances in ActionScript

Naming an instance for button symbol



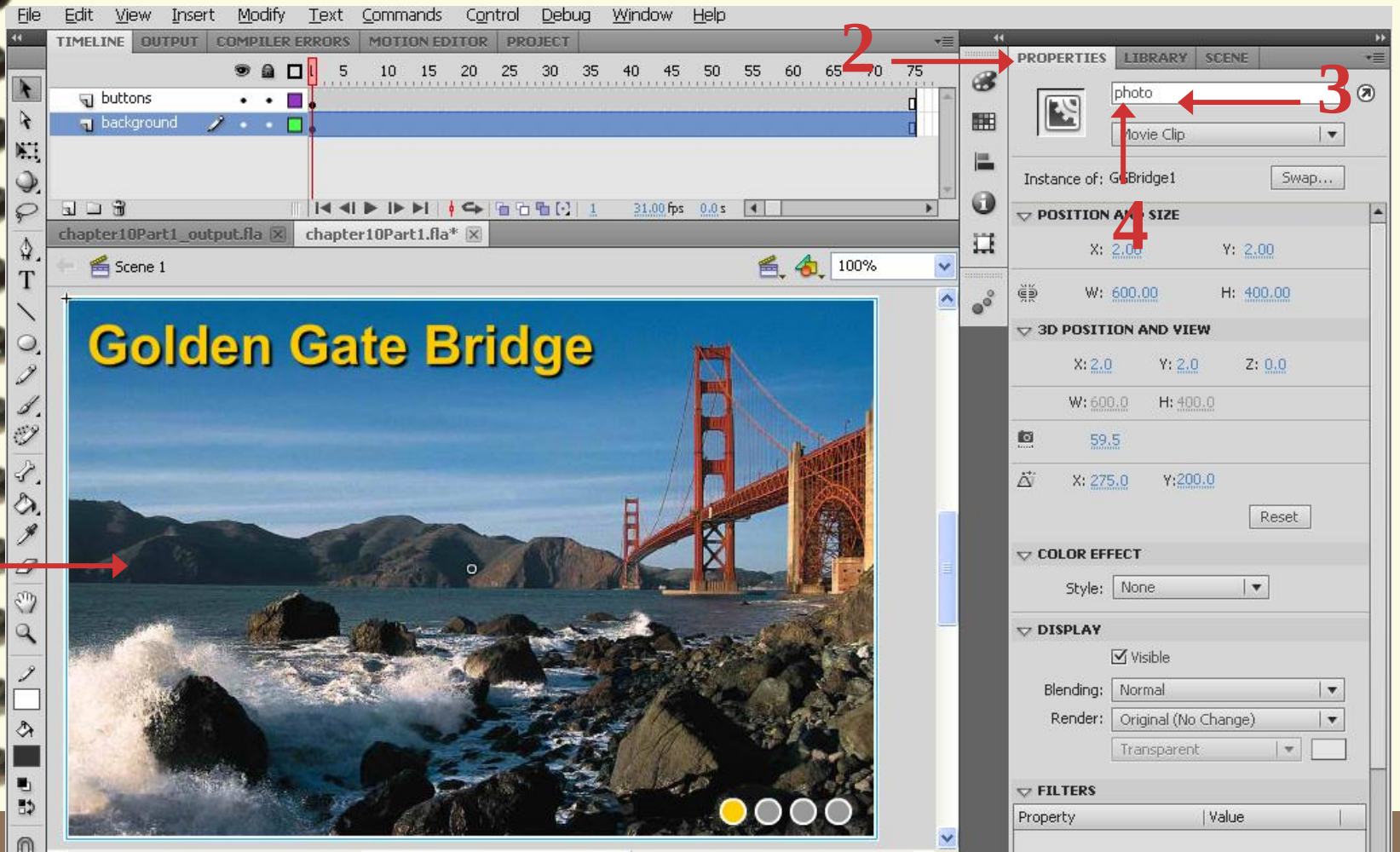
Working with Instances in ActionScript

Naming an instance for a button symbol



Working with Instances in ActionScript

Naming an instance for a movie clip



Move a MovieClip with ActionScript

- You can animate movice clip instances by listening for an ENTER_FRAME event.
- Every time Animate renders a frame, it broadcasts a message to every object that is listening.
- At 31 frames per second (fps), you can reposition a movie clip 31 times a second, create a smooth animation.

Move a MovieClip with ActionScript

1. Open the Library Panel
2. Drag a MovieClip symbol from the Library to the Stage
3. Click the movie clip you want to animate with the Selection tool
4. Open the Properties panel
5. Click here and type an instance name for your movie clip.

Move a MovieClip with ActionScript

6. Press F9 for the Action Panel.
7. Click the Target icon.
8. Click your movie clip to select it
9. Click OK
10. Click the Add new item to the script button.
11. Click Animate.events
12. Click EventDispatcher
13. Click Methods
14. Click addEventListener

Move a MovieClip with ActionScript

How can I set my font and color preferences for the ActionScript window?

Click Animate, Preferences, and then click on ActionScript in the left column.

The Preference panel has options to change your font and syntax colors.

There are also preferences that you can set for code hits, import and export file types, and language settings.

Add Actions to Movie Clips

- You can add actions to **movie clip instances** that appear in your main movie
- For example, you may have an animation sequence of moving car that includes movie clips for making each wheel rotate.

Move a MovieClip with ActionScript *(continued)*

- To move your movie clip, you can simply change its x and/ or y properties.
- These property values are in pixels, so make sure you do not set their values to a number larget than the Stage size (or to a negative number), unless you want your movie clip to disappear.

Move a MovieClip with ActionScript *(continued)*

What other properties can I animate using ActionScript?

You can animate any property of a movie clip or object using ActionScript.

To view a list of DisplayOjbect properties, click the Animate.display category in the Action Window.

Inside Animate.display, click DisplayObject. Inside DisplayObject, click Properties. All display properties are listed inside.

With this example, you can easily change the script to move your movie clip along the y-axis. Change evt.currentTarget.x to evt.currentTarget.y. Test the movice again, and watch your movie clip move vertically rather horizontally.

Move a MovieClip with ActionScript *(continued)*

How do I check my script for errors?

If you click the Check Syntax button , Animate displays the Compiler Errors panel with your errors.

The Compiler Error panel also appears when you test your movie if your script contains errors.

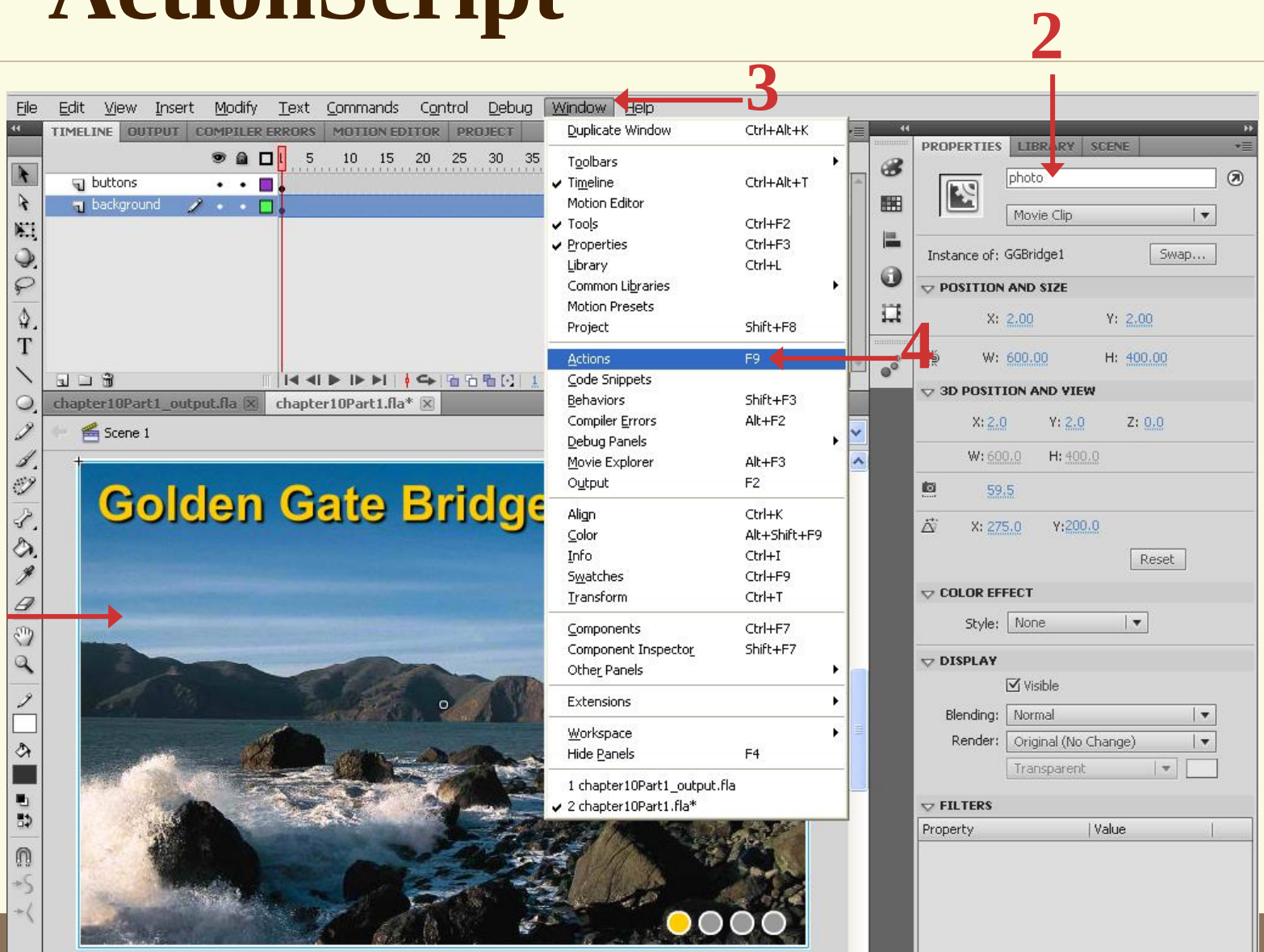
Fade In a Movie Clip with ActionScript

- You can use the ENTER_FRAME event to perform other tasks over time in Animate.
- The following demonstrates how to gradually change the alpha property of a movie clip, creating a fade effect.

Fade In a Movie Clip with ActionScript

1. Click on the instance of the **movie clip symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window**
4. Click on **Action Panel** or press F9

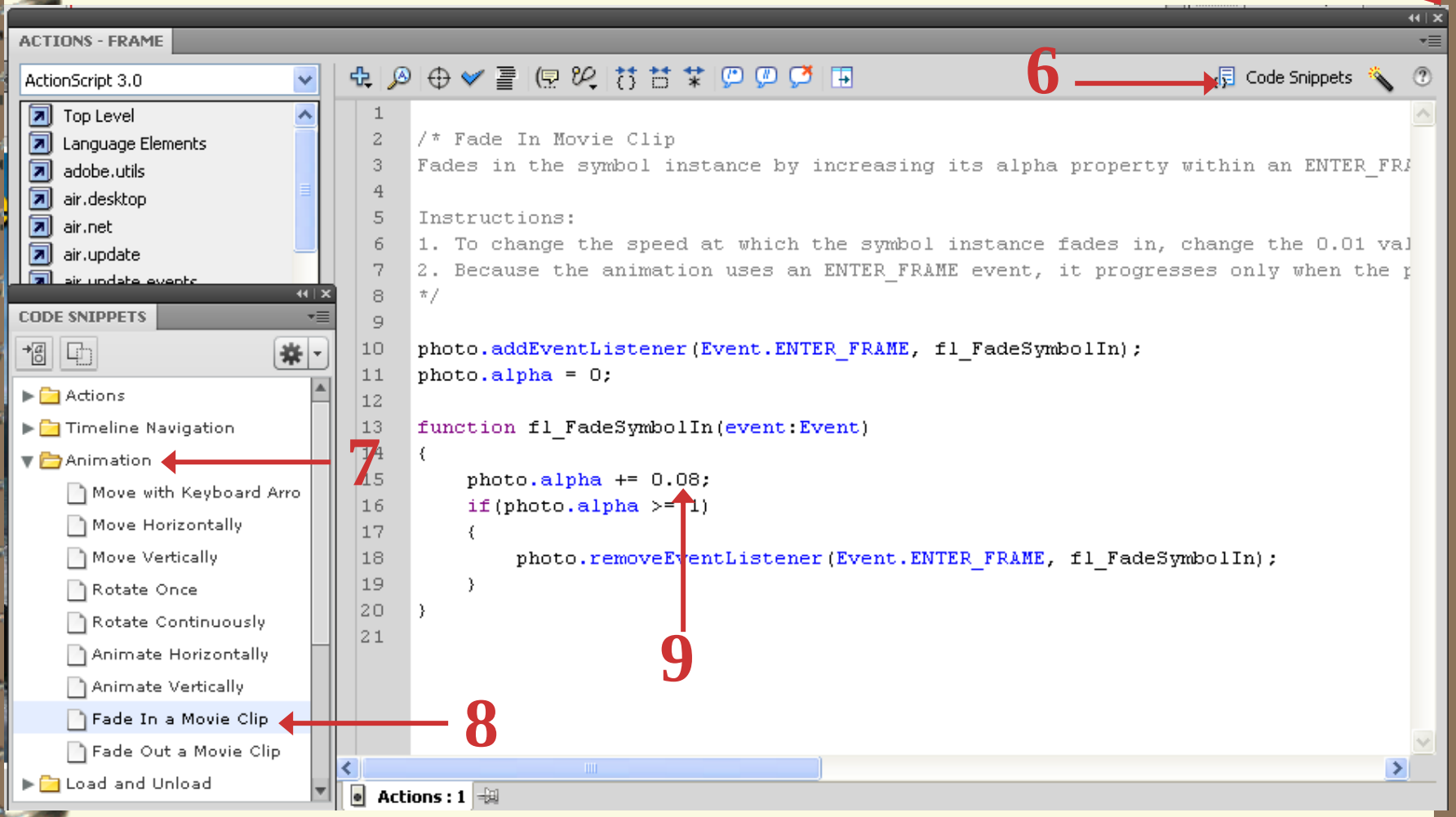
Fade In a Movie Clip with ActionScript



Fade In a Movie Clip with ActionScript

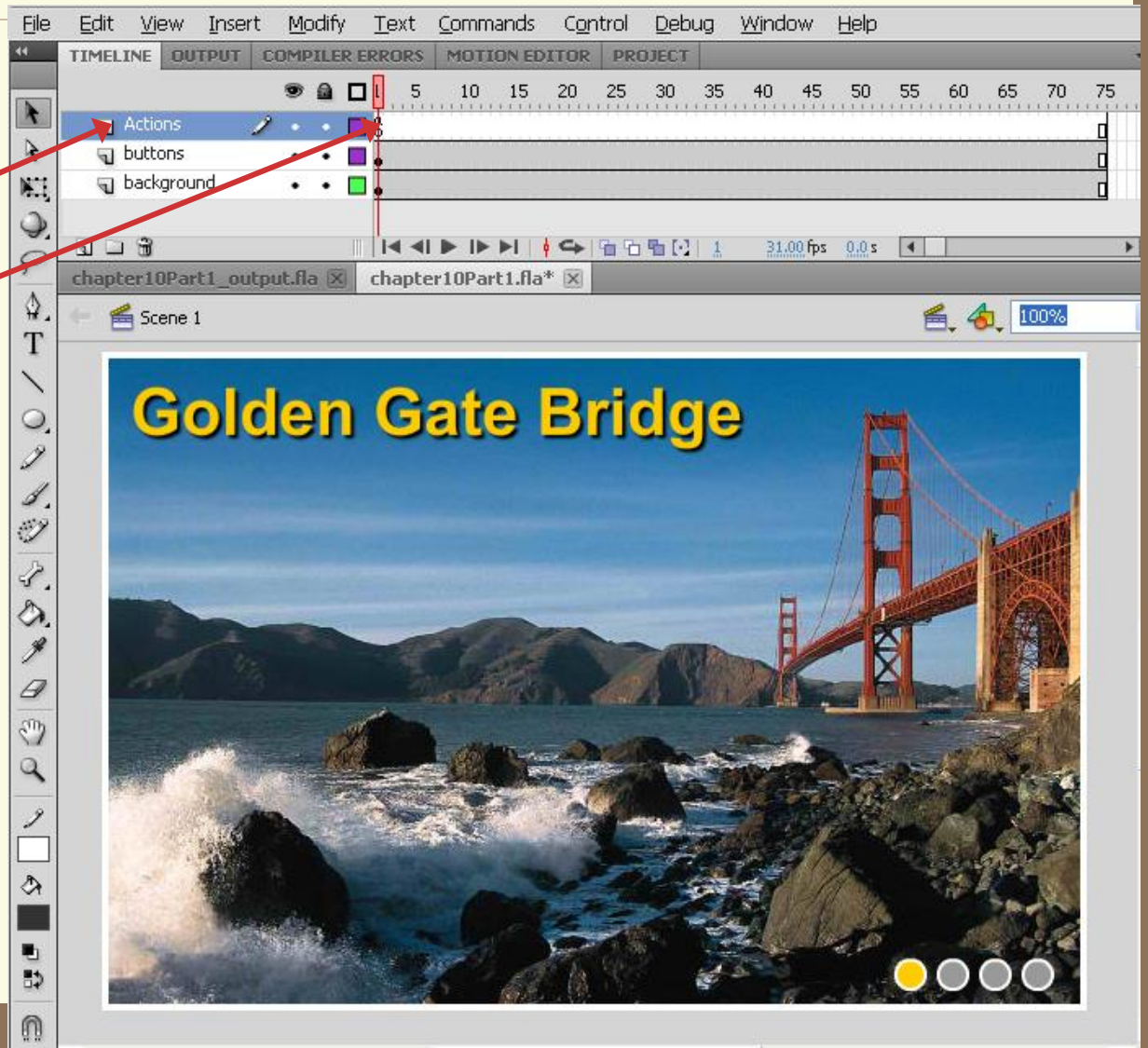
6. Click on the **Code Snippets**
7. Expand the **Animation folder**
8. Double click on "**Fade In a Movie Clip**"
9. Change the speed in which the symbol instance fade in to 0.08.
10. Close the Action Panel window.

Fade In a Movie Clip with ActionScript



Fade In a Movie Clip with ActionScript

The **Actions layer** and **action keyframe** are created automatically after the action script is entered into the Action Panel



Fade In a Movie Clip with ActionScript

How can I change the appearance of my scripts in the Actions panel?

You can click Animate, Preferences, and then click on the ActionScript category to customize the Actions panel.

Then, you can change your indentation preference, code hint timing, font, style, and syntax colors.

Fade In a Movie Clip with ActionScript *(continued)*

- By creating your animation in ActionScript, you can create scripts that can be used by multiple objects.
- This both saves you development time and gives your project a consistent animation style.

Fade In a Movie Clip with ActionScript *(continued)*

How can I test to see my if statement is working properly6?

You can place a trace() statement inside your if statement to see if the condition is met. In this example, you might add:

trace("face completed.");

Between the curly braces of your statement. If your condition is met, fade completed appears in the Output window when your test movie.

Understanding Events and Event Handlers

- In Animate, an Event is any type of interaction, such as a mouse click, mouse move, rollover, or key press.
- Events can also be triggered by object instances in Animate.
- For example, MoveClip object can broadcase events like ENTER_FRAME, and Loader objects can broadcase events like COMPLETE.

Understanding Events and Event Handlers

About Events and Event Handlers

- A good way to think about events and event listeners is to think about the police responding to a crime, like a robbery.
- In ActionScript, your event handler is the function that performs actions when an event occurs.

Understanding Events and Event Handlers

Adding Event Listeners

- In ActionScript 3.0 you can add events to objects by calling the addEventListener() function like

  myButton.addEventListener(MouseEvent.MOUSE_DOWN, onMouseClick);

- This statement tells Animate to listen for a mouse click on an instance called myButton
- When that event occurs, the function onMouseClick is executed.

Understanding Events and Event Handlers

Creating Event Handlers

- The event handler is the function that is executed when an event occurs.
- In ActionScript 3.0, event handlers take an object of type Event as a parameter.
- A handler for the mouse click example might look like this:

```
Function onMouseClick(evt:Event):void { trace
    (“my button was clicked! “);
}
```

Understanding Events and Event Handlers

Event Targets

- There are two places you use your event target in ActionScript.
- You provide the target when you create the listener, but you can also use the target in your event handler.
- For example, if you want the myButton object to disappear when clicked, your hander might look like this:

```
Function onMouseClick(evt:Event):void {
    myButton.visible= false;
}
```

Understanding Events and Event Handlers

Event Targets

- But, if you wanted any button to become invisible when clicked, you could add the same event listener to all of them, but change your handler to look like this:

```
function onMouseClick(evt:Event):void {
    evt.currentTarget.visible= false;
}
```

- In this case, evt.currentTarget is a reference to whichever listening button is clicked.

Understanding Events and Event Handlers

Removing Event Listeners

- Sometimes you want your objects to stop listening for evets.
- To do so, you remove the listener by calling removeEventListener.
- In the practical example, you would delete your mouse click listener like this:

```
myButton.removeEventListener(MouseEvent.
    MOUSE_DOUT, onMouseClick);
```

Timeline Navigation using a Button

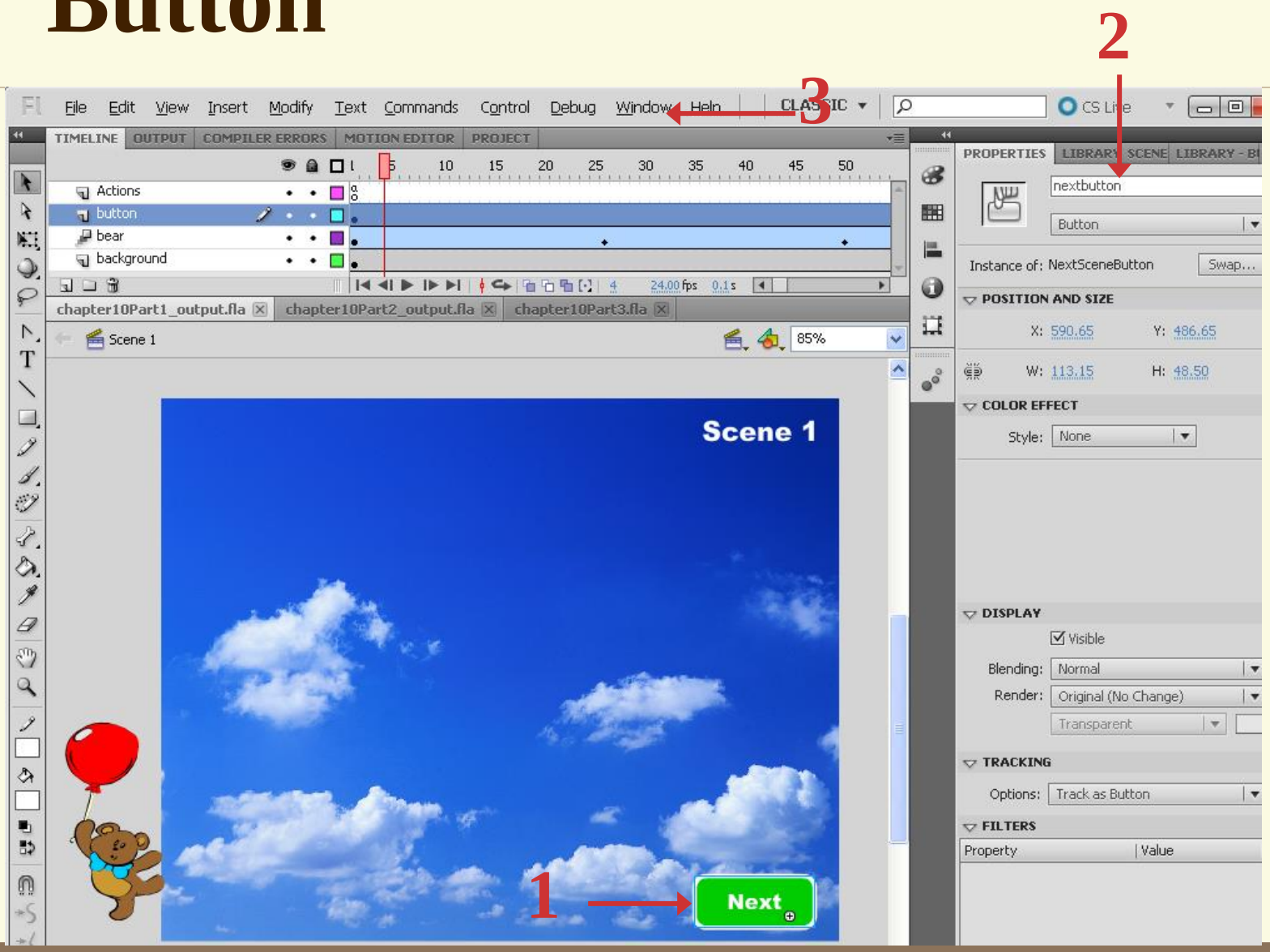
- You can create interactivity in your Animate movies by combining animation with controls like buttons.
- Your buttons can listen for mouse events, and perform actions when those events occur.

Timeline Navigation using a Button

Click to Go to Next Scene and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window** and click **Action Panel** or press **F9**

Timeline Navigation using a Button

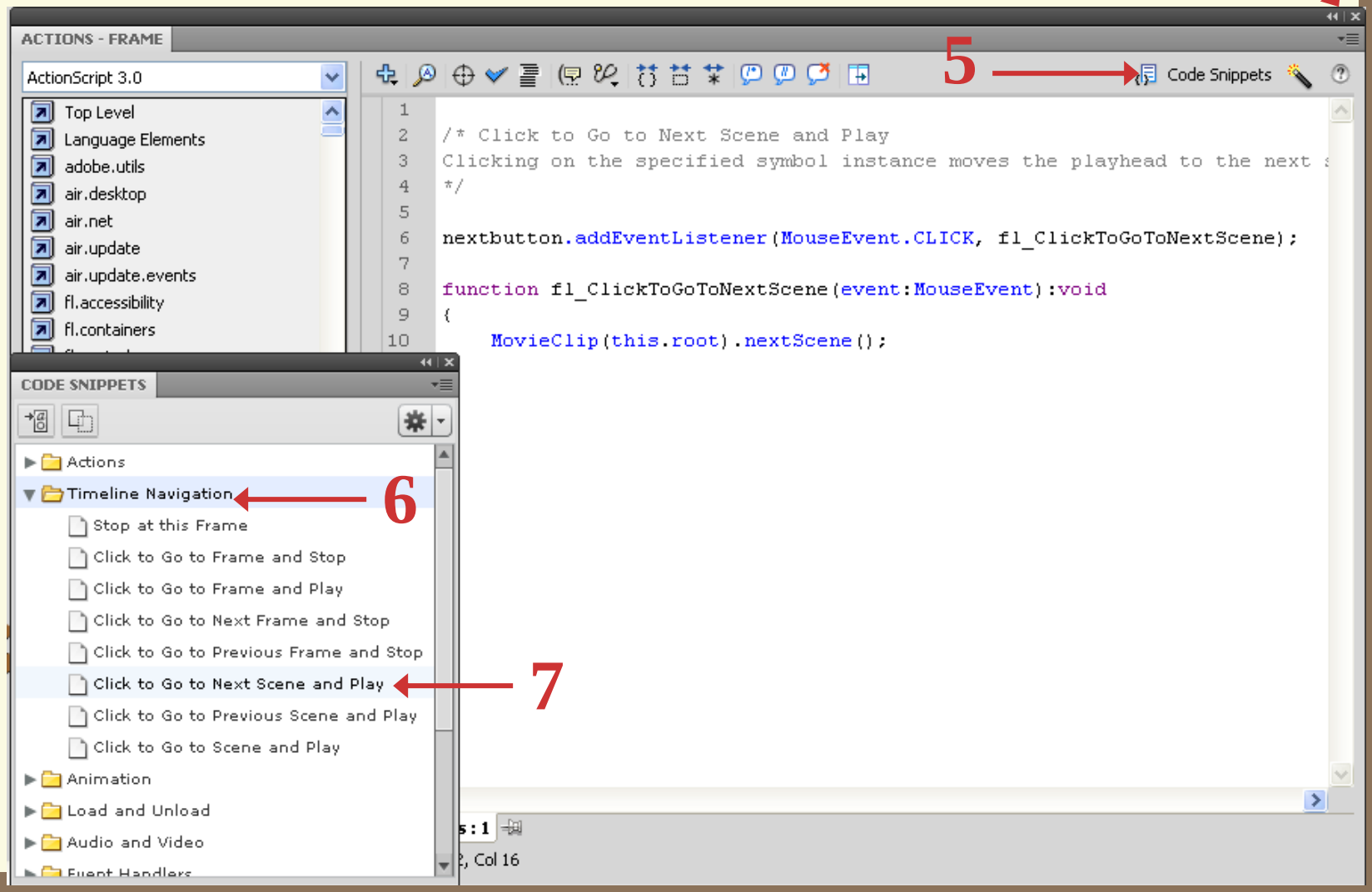


Timeline Navigation using a Button

Click to Go to Next Scene and Play

5. Click on the **Code Snippets**
6. Expand the **Timeline Navigation folder**
7. Double click on "**Click to Go to Next Scene and Play**"
8. Close the Action Panel window.

Timeline Navigation using a Button

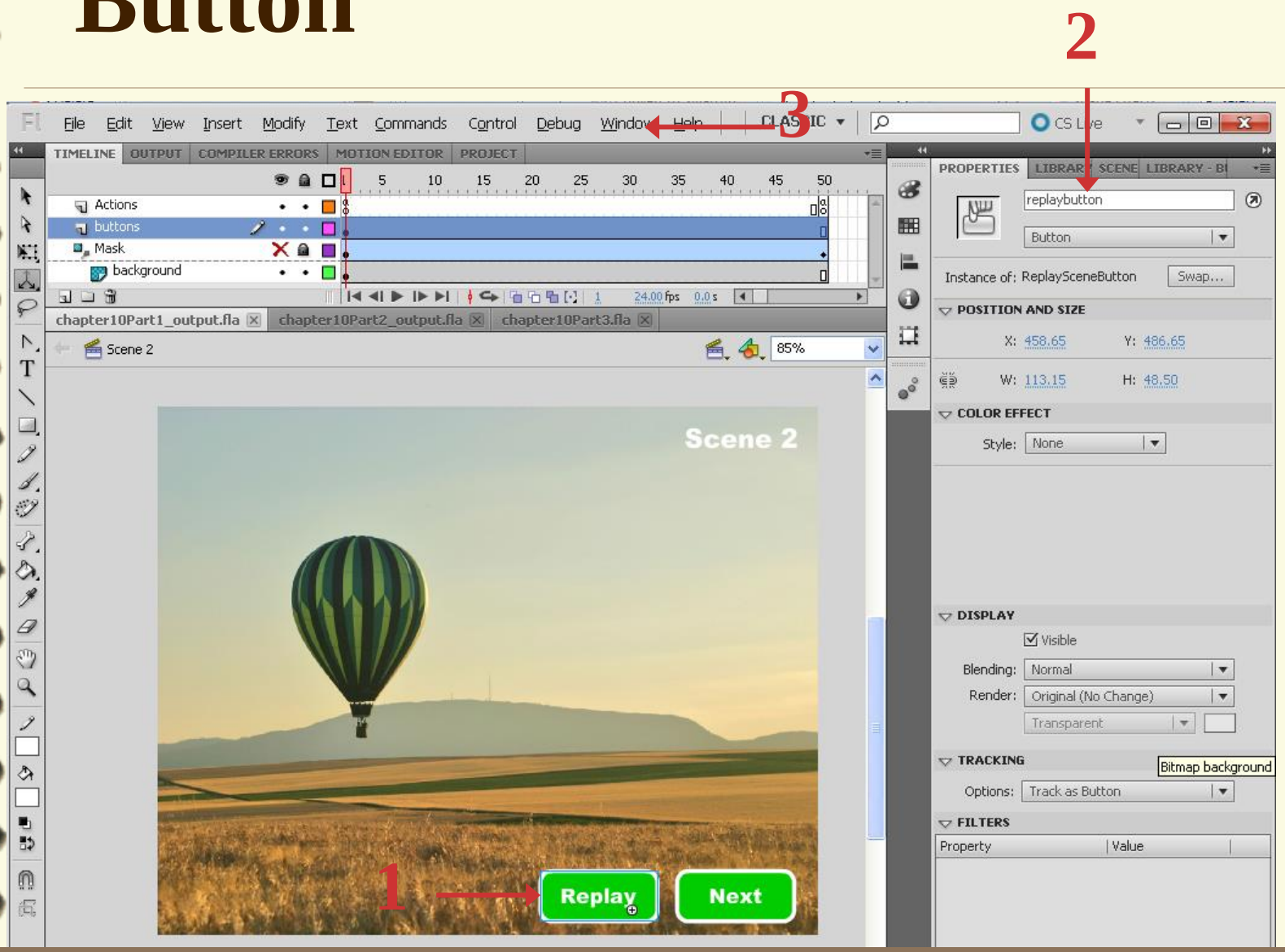


Timeline Navigation using a Button

Click to Go to Frame and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window** and click **Action Panel** or press **F9**

Timeline Navigation using a Button

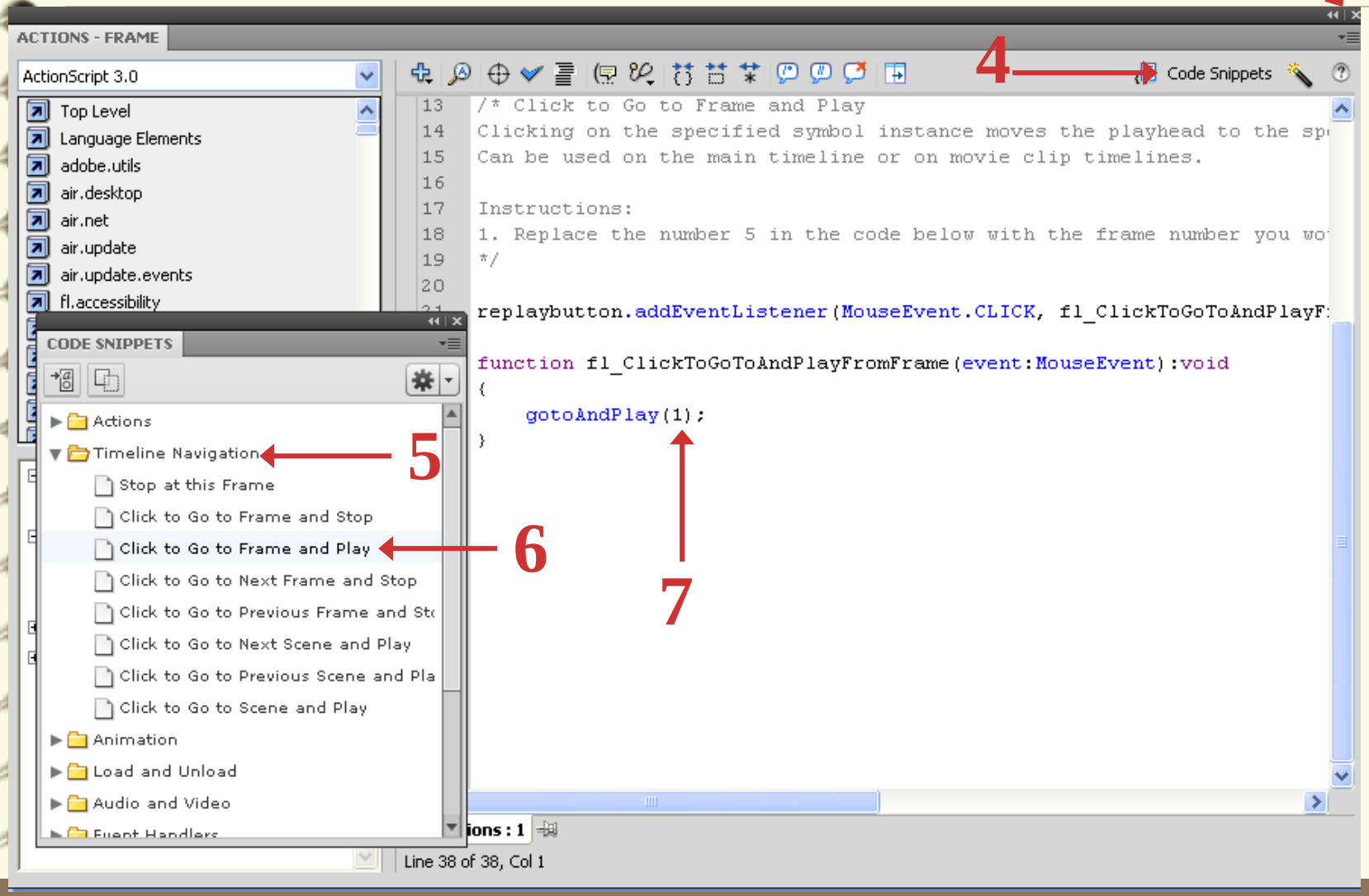


Timeline Navigation using a Button

Click to Go to Frame and Play

4. Click on the **Code Snippets**
5. Expand the **Timeline Navigation folder**
6. Double click on "**Click to Go to Frame and Play**"
7. Replace with the number with the frame you would like to go and play in the current scene.
8. Close the Action Panel window.

Timeline Navigation using a Button

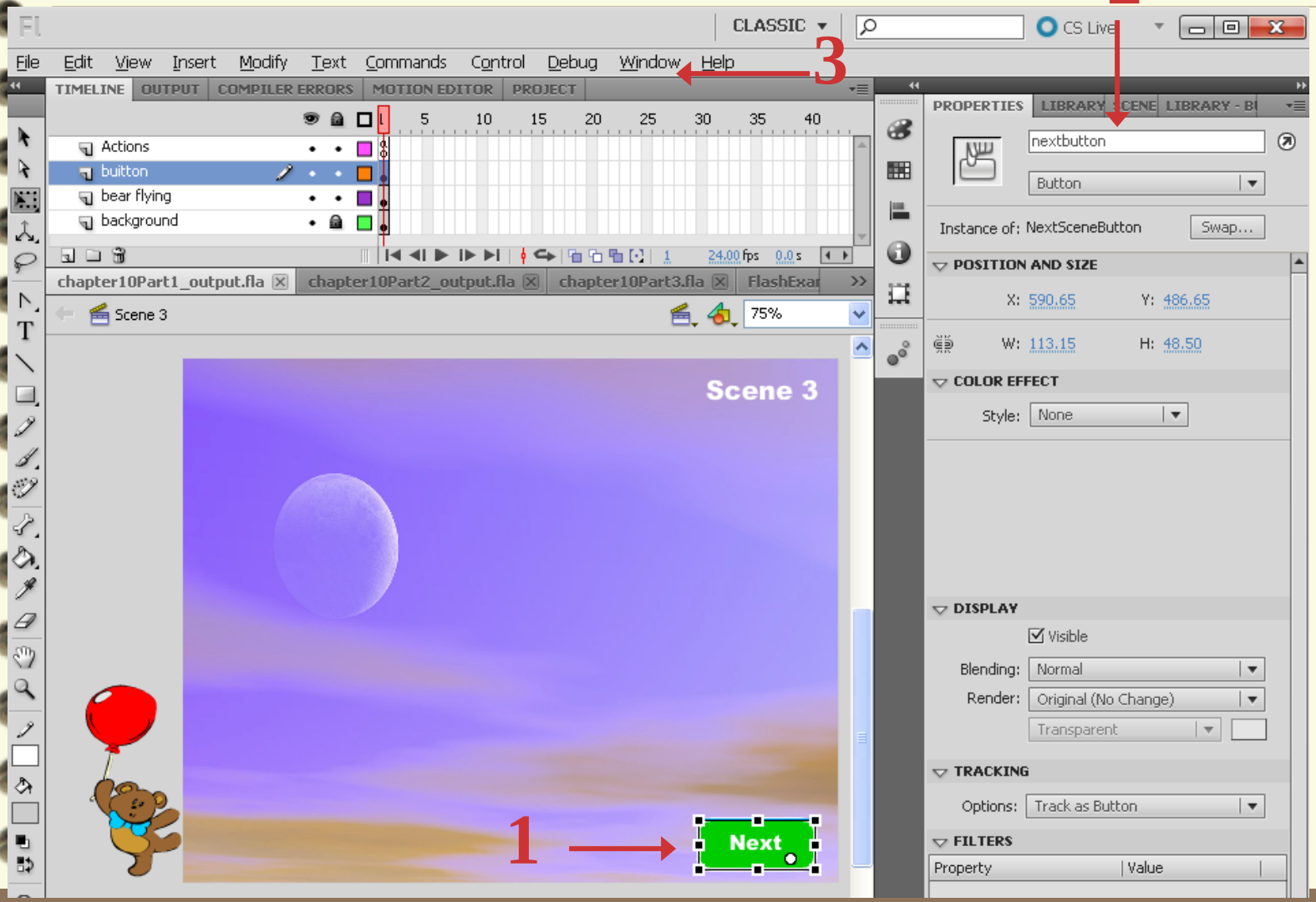


Timeline Navigation using a Button

Click to Go to Scene and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window**
4. Click on **Action Panel** or press **F9**

Timeline Navigation using a Button

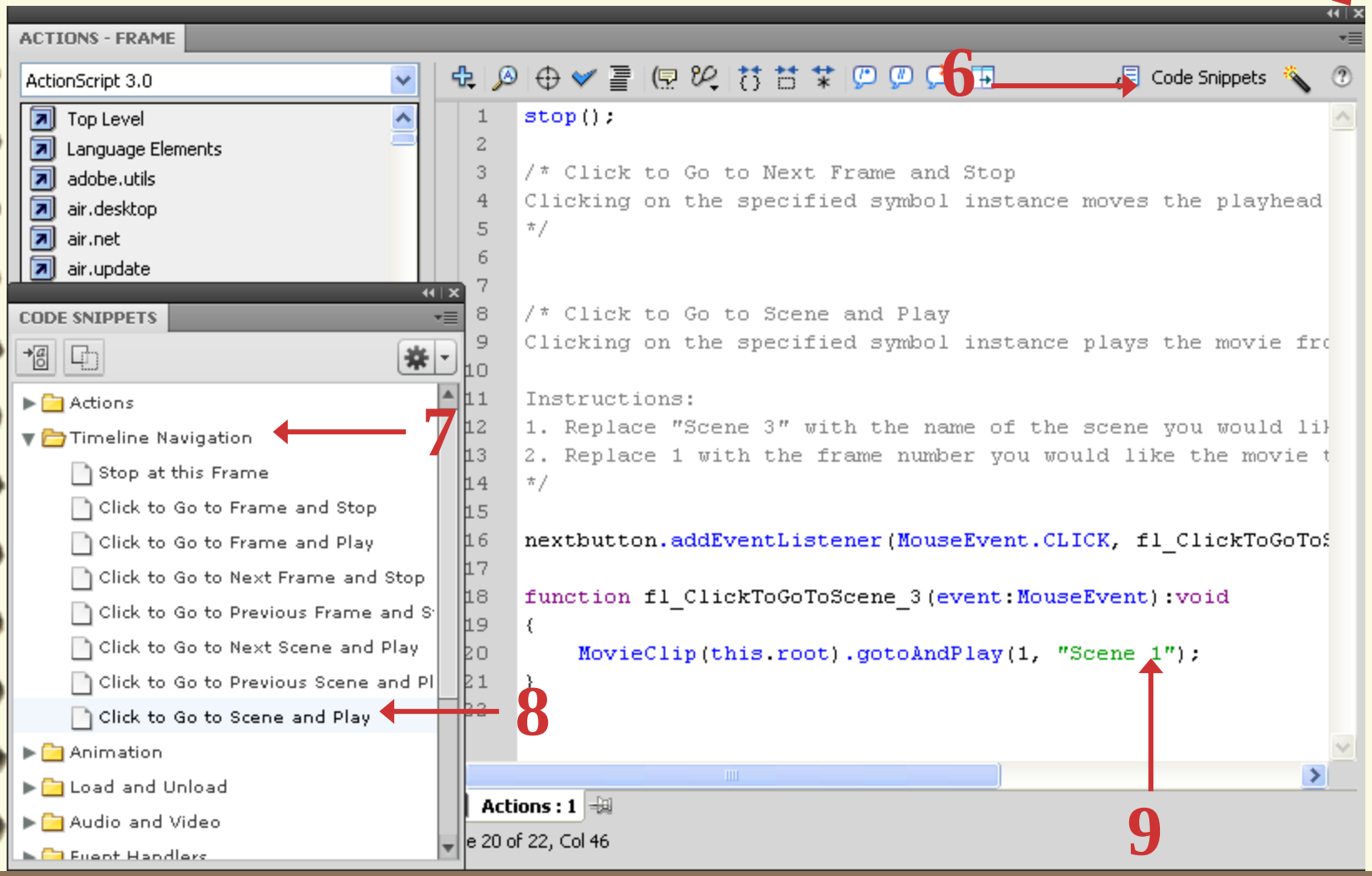


Timeline Navigation using a Button

Click to Go to Scene and Play

6. Click on the **Code Snippets**
7. Expand the **Timeline Navigation folder**
8. Double click on "**Click to Go to Scene and Play**"
9. Replace with the name of the scene you would like to play
10. Close the Action Panel window.

Timeline Navigation using a Button

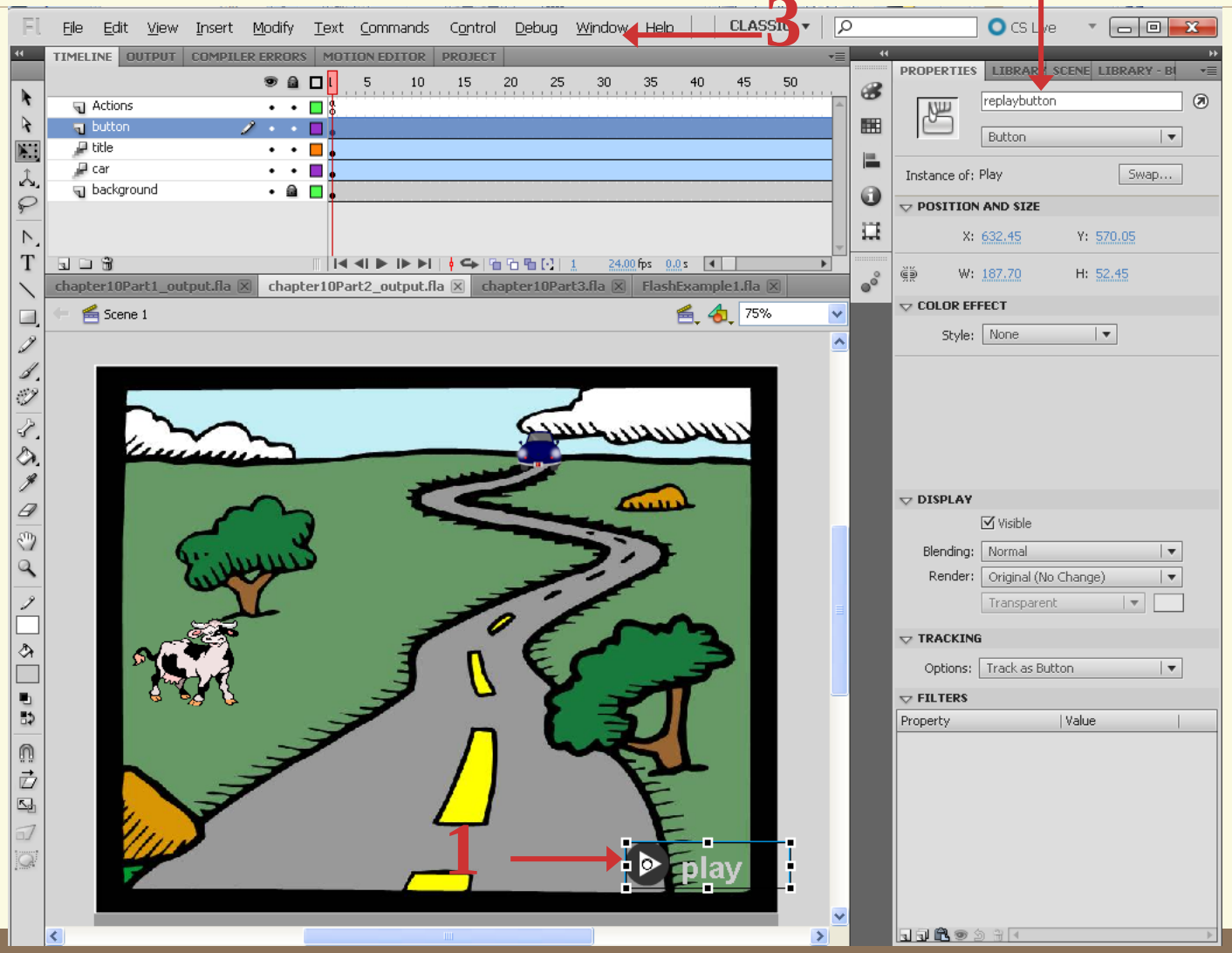


Timeline Navigation using a Button

Click to Go to Frame and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window** and click **Action Panel** or press **F9**

Timeline Navigation using a Button

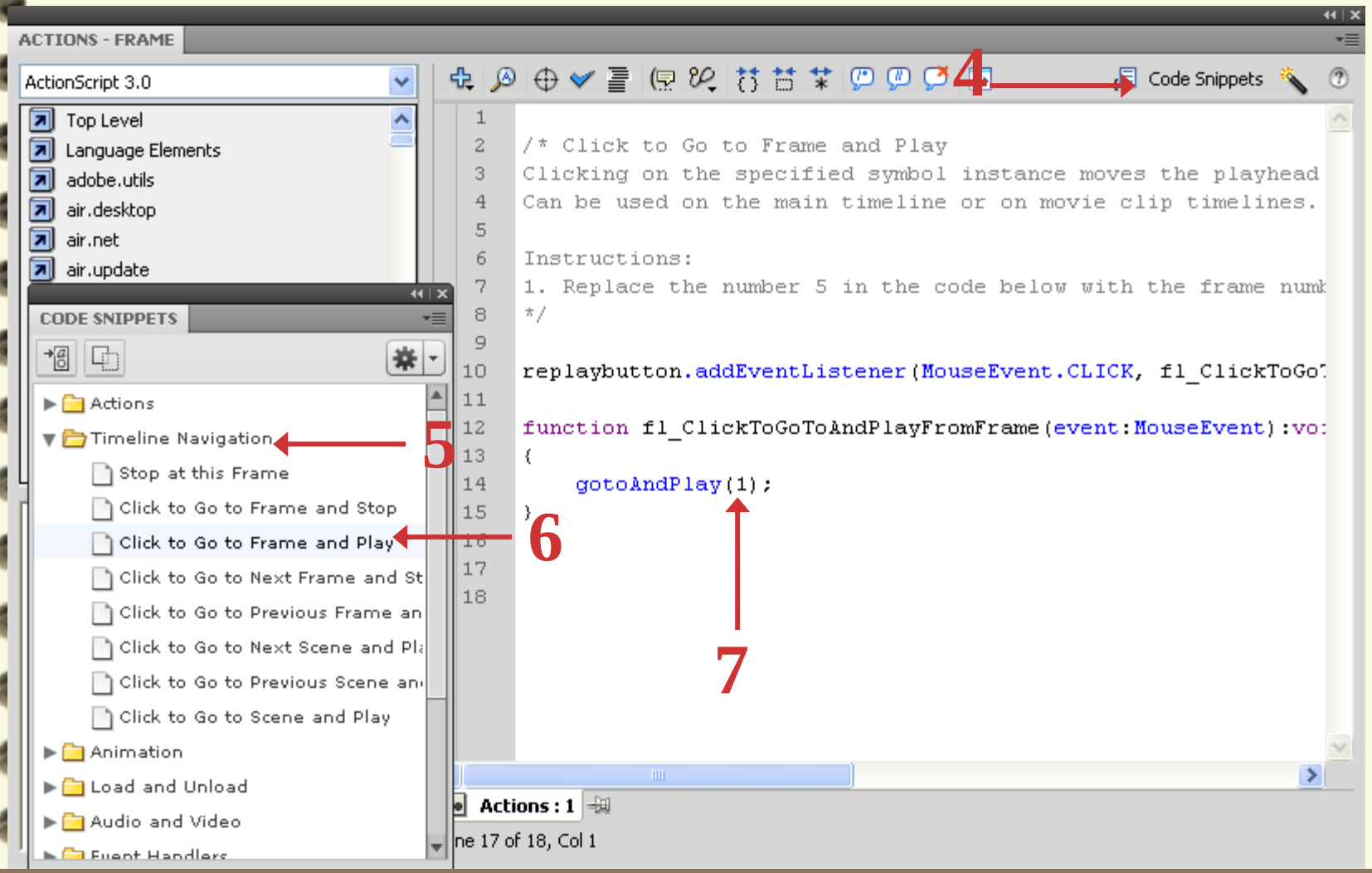


Timeline Navigation using a Button

Click to Go to Frame and Play

4. Click on the **Code Snippets**
5. Expand the **Timeline Navigation folder**
6. Double click on "**Click to Go to Frame and Play**"
7. Replace with the number with the frame you would like to go and play in the current scene.
8. Close the Action Panel window.

Timeline Navigation using a Button

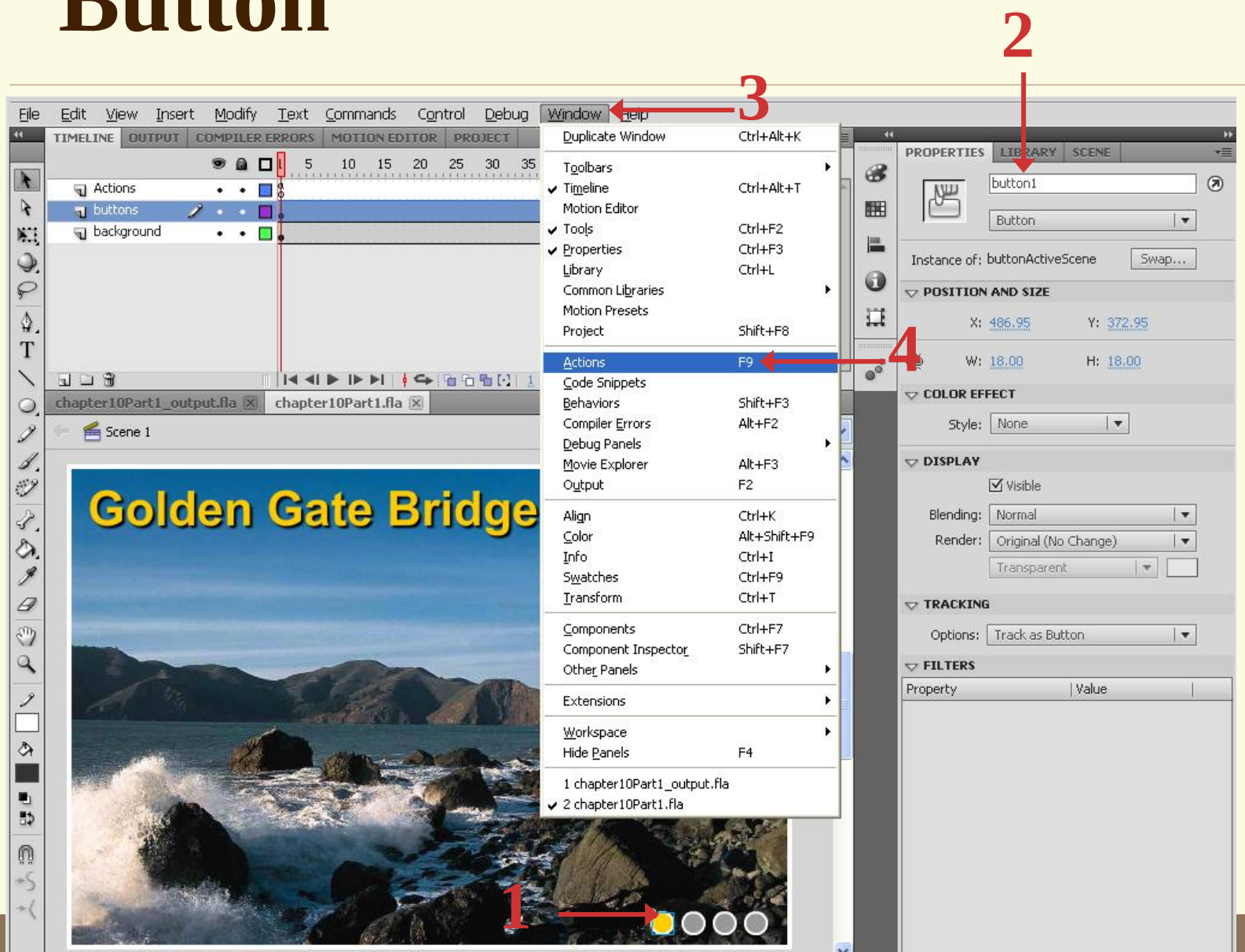


Timeline Navigation using a Button

Click to Go to Scene and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window**
4. Click on **Action Panel** or press **F9**

Timeline Navigation using a Button

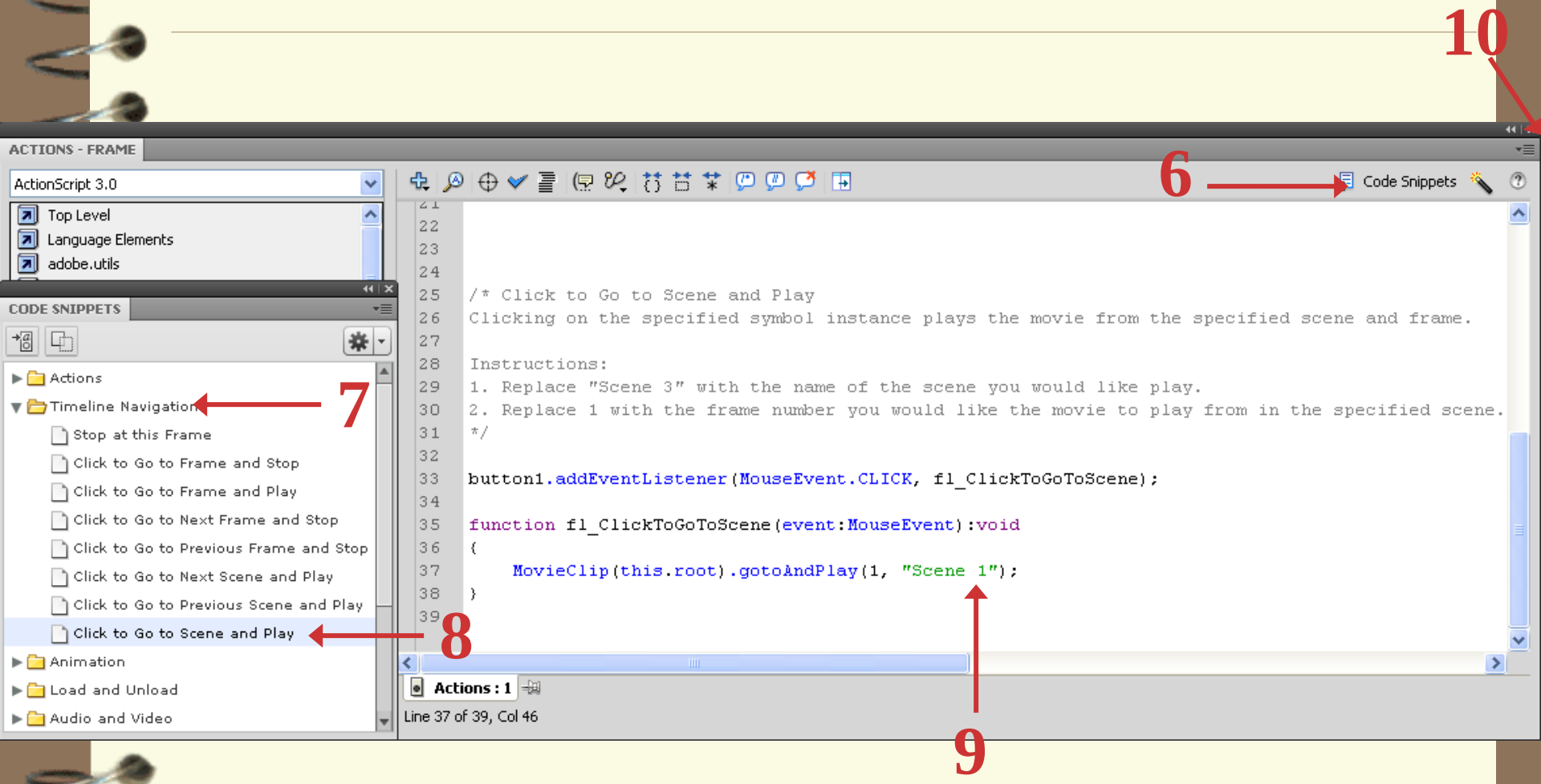


Timeline Navigation using a Button

Click to Go to Scene and Play

6. Click on the **Code Snippets**
7. Expand the **Timeline Navigation folder**
8. Double click on "**Click to Go to Scene and Play**"
9. Replace with the name of the scene you would like to play
10. Close the Action Panel window.

Timeline Navigation using a Button

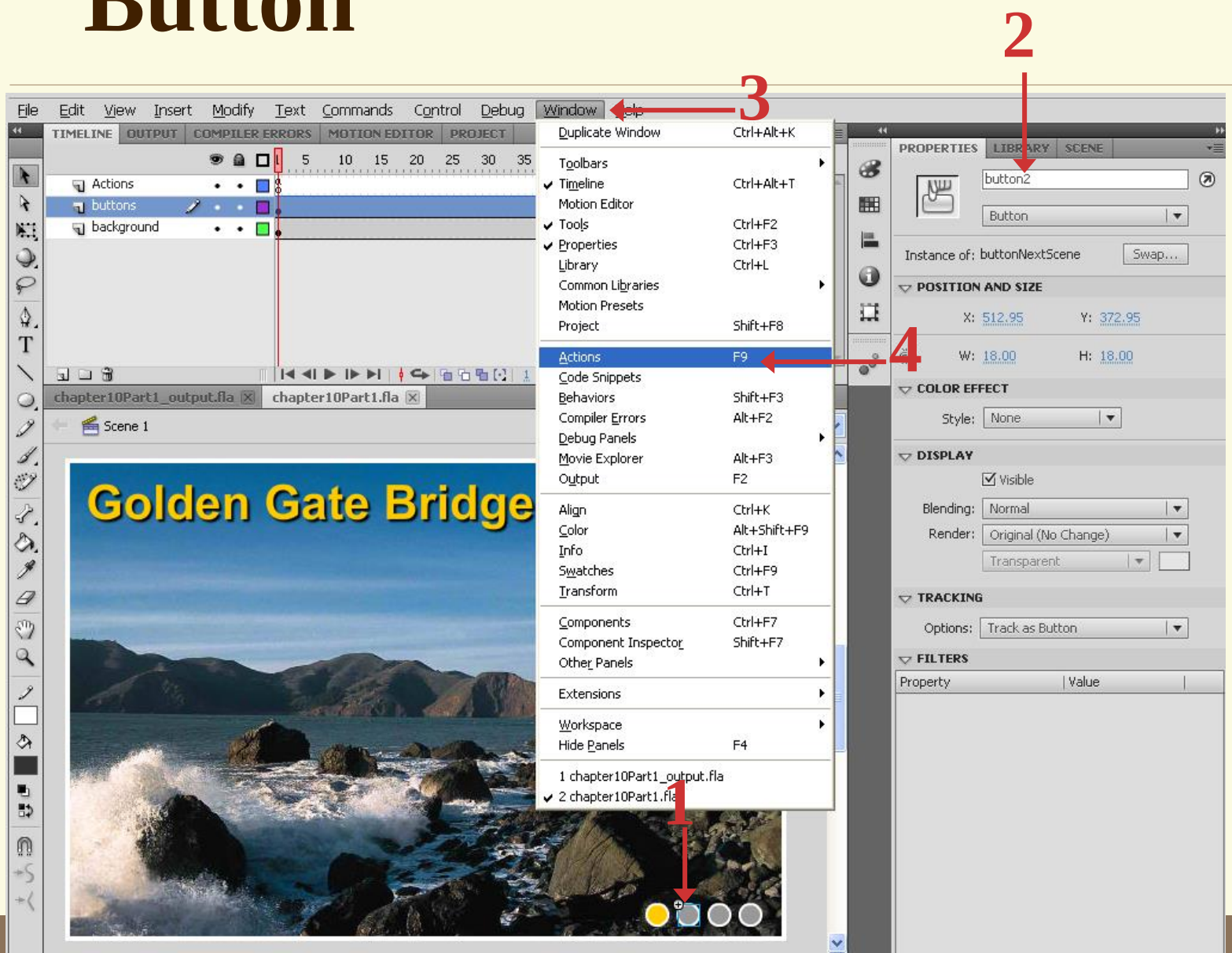


Timeline Navigation using a Button

Click to Go to Scene and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window**
4. Click on **Action Panel** or press **F9**

Timeline Navigation using a Button

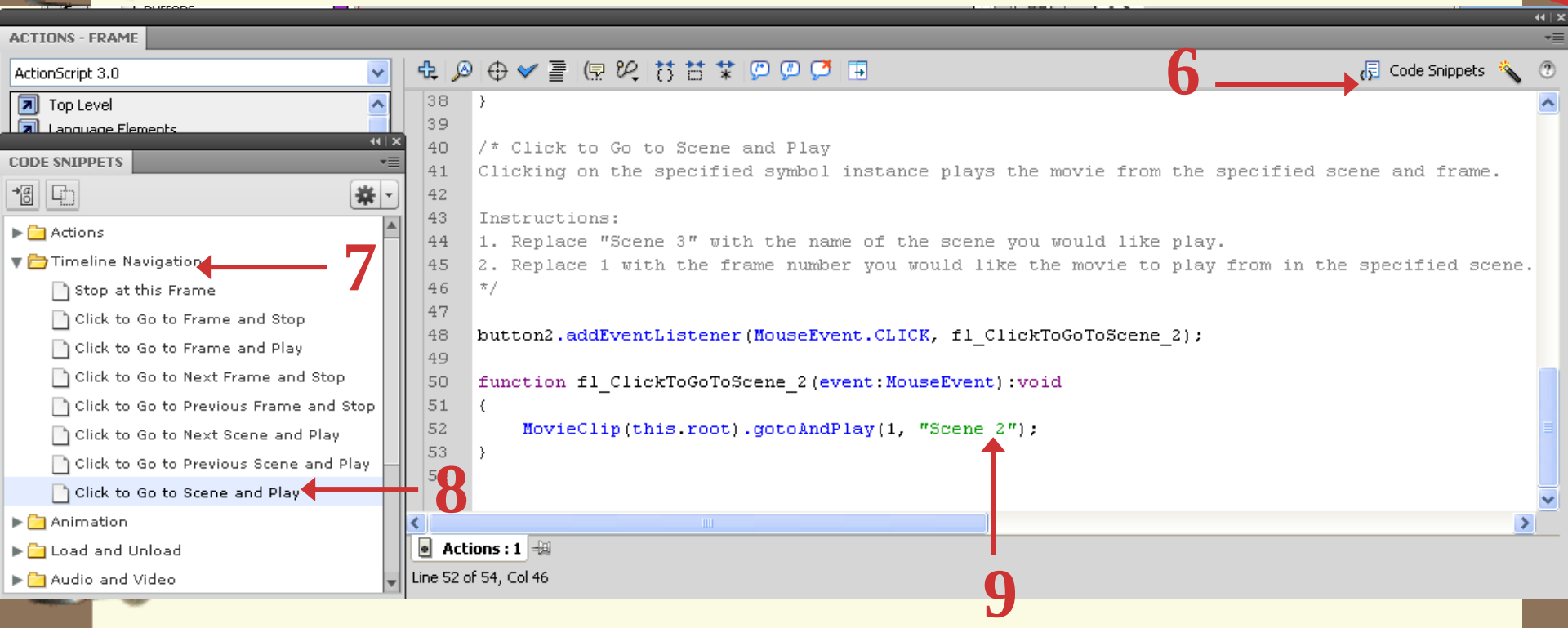


Timeline Navigation using a Button

Click to Go to Scene and Play

6. Click on the **Code Snippets**
7. Expand the **Timeline Navigation folder**
8. Double click on "**Click to Go Scene and Play**"
9. Replace with the name of the scene you would like to play
10. Close the Action Panel window.

Timeline Navigation using a Button

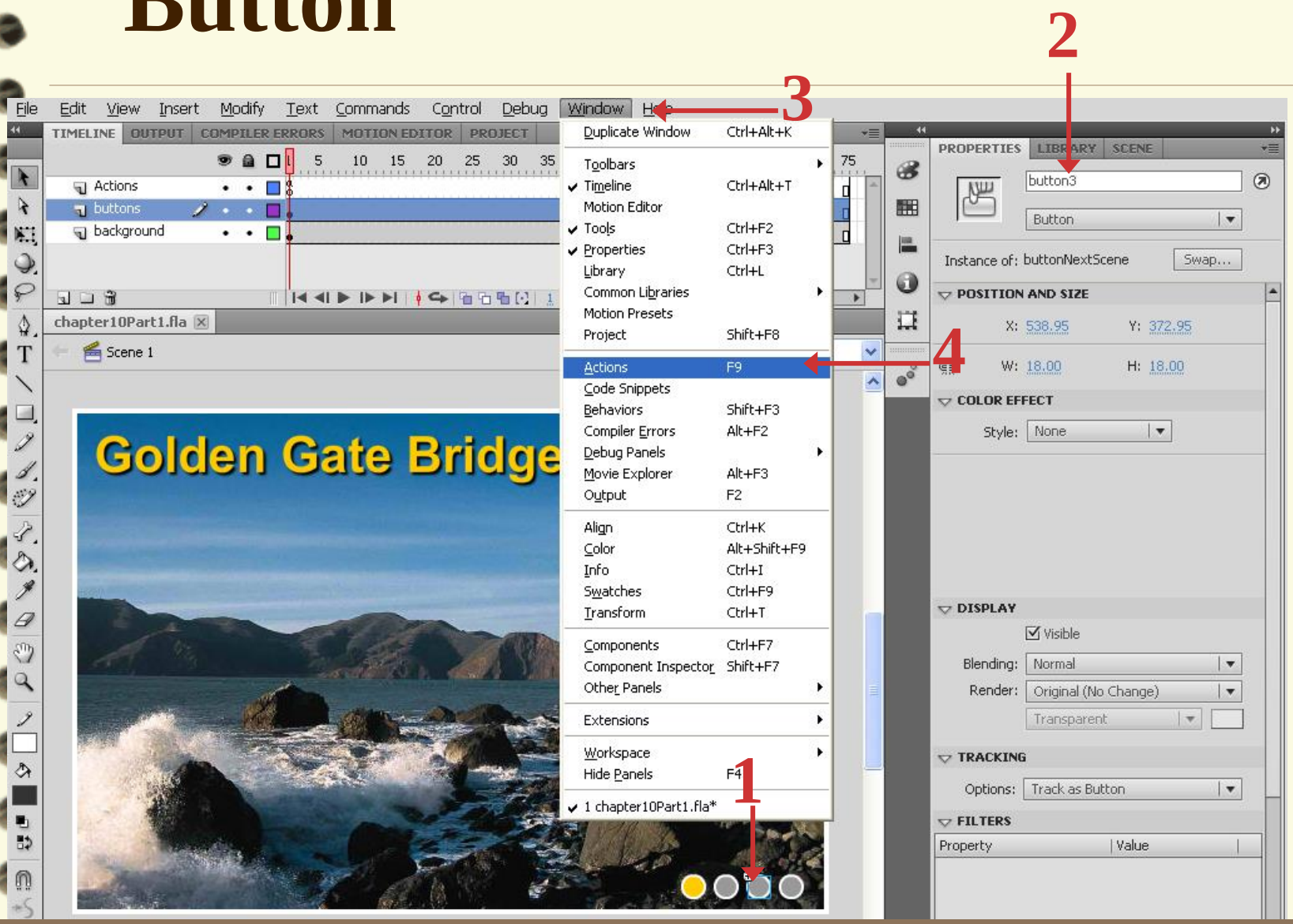


Timeline Navigation using a Button

Click to Go to Scene and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window**
4. Press **F9** or select on **Action Panel.**

Timeline Navigation using a Button

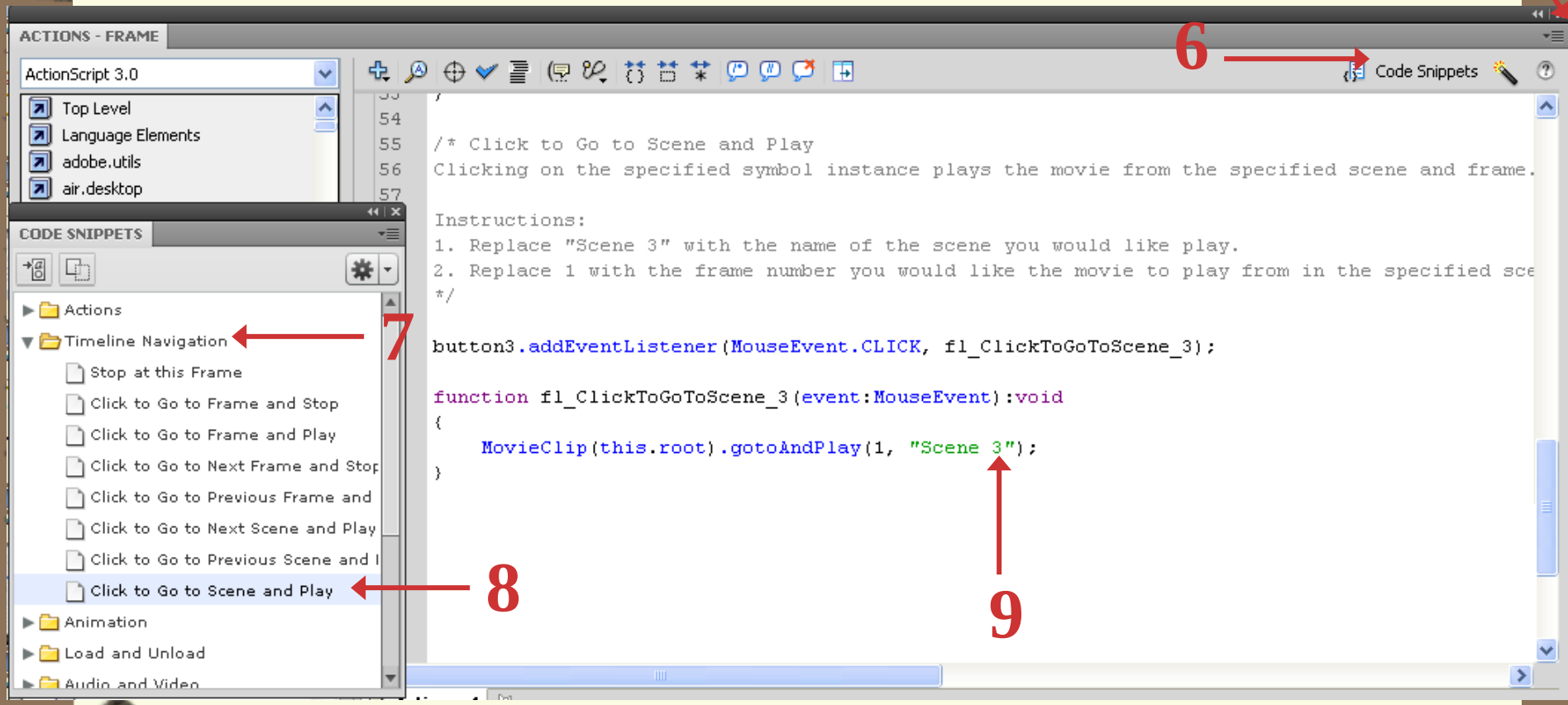


Timeline Navigation using a Button

Click to Go to Scene and Play

6. Click on the **Code Snippets**
7. Expand the **Timeline Navigation folder**
8. Double click on "**Click to Go Scene and Play**"
9. Replace with the name of the scene you would like to play
10. Close the Action Panel window.

Timeline Navigation using a Button

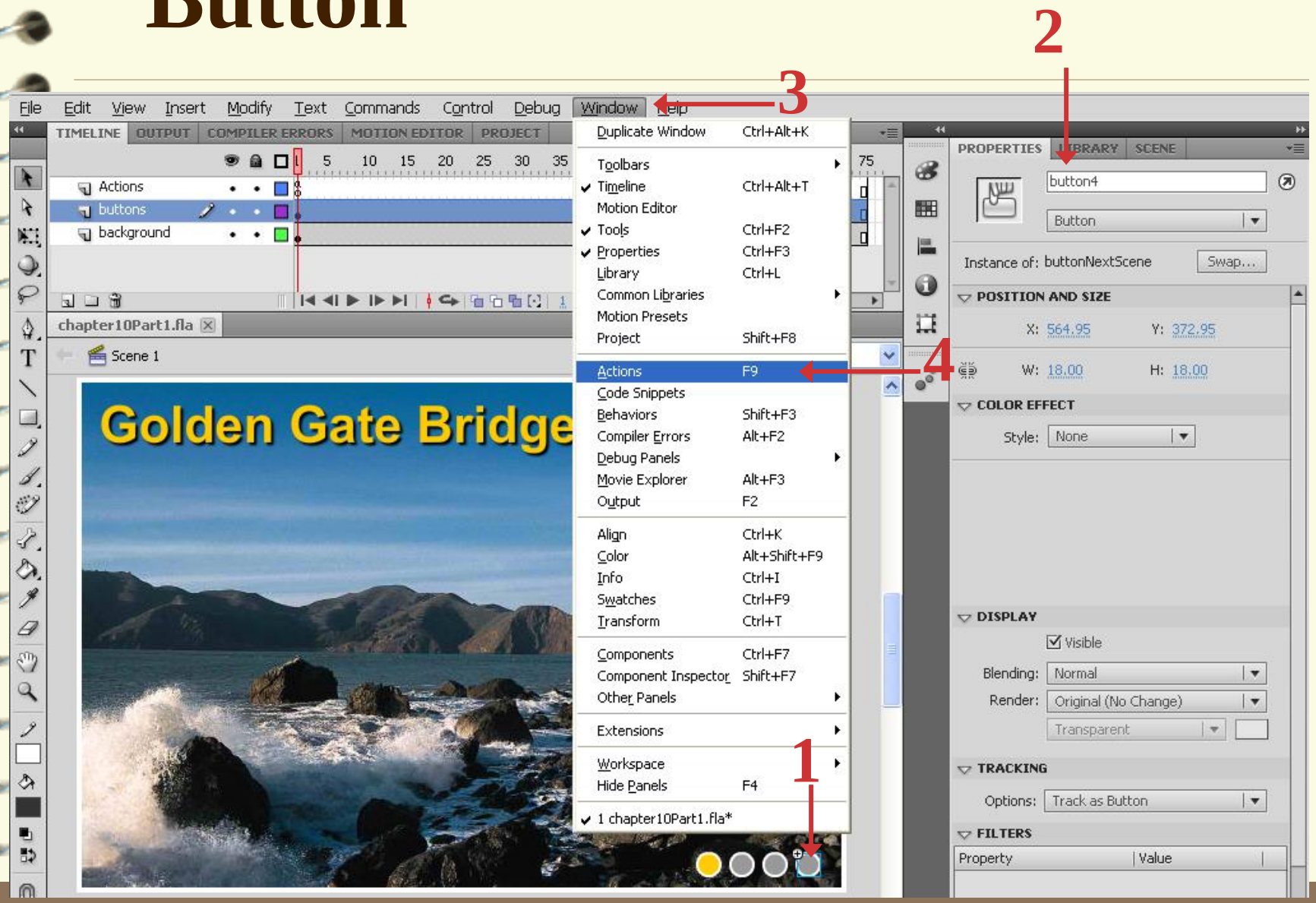


Timeline Navigation using a Button

Click to Go to Scene and Play

1. Click on the instance of the **button symbol** on the stage**.**
2. Verify the symbol has an **instance name** in the Properties panel
3. Select **Window**
4. Press **F9** or select on **Action Panel.**

Timeline Navigation using a Button

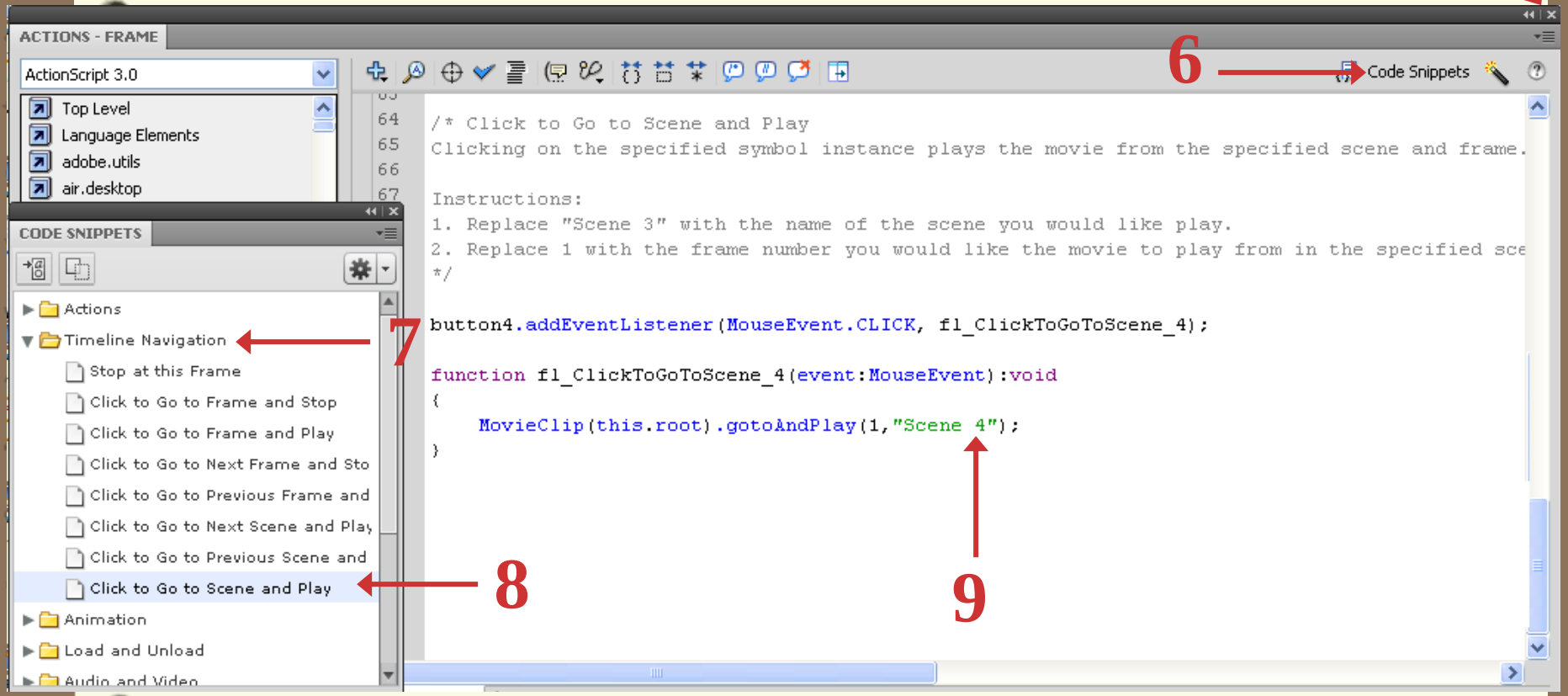


Timeline Navigation using a Button

Click to Go to Scene and Play

6. Click on the **Code Snippets**
7. Expand the **Timeline Navigation folder**
8. Double click on "**Click to Go Scene and Play**"
9. Replace with the name of the scene you would like to play
10. Close the Action Panel window.

Timeline Navigation using a Button



Start and Stop an Animation with a Button

How can I create comments to make my script more readable?

You can write comments in ActionScript to give yourself or someone else reading your code information that you want Animate to ignore.

A single line comment is written with two backslashes like this:

// this is a single-line comment

/ Multiline comments start with a slash followed by an asterisk. To end a multiline comment type an asterisk followed by a slash */*

Start and Stop an Animation with a Button

My ActionScript has odd indentation is there an easy way to format it properly?

You can click the Auto Format script button to have Animate automatically reset the indentation of each line of your script.

However, if you script contains errors, the formatting will fail.

Create and Include an External ActionScript File

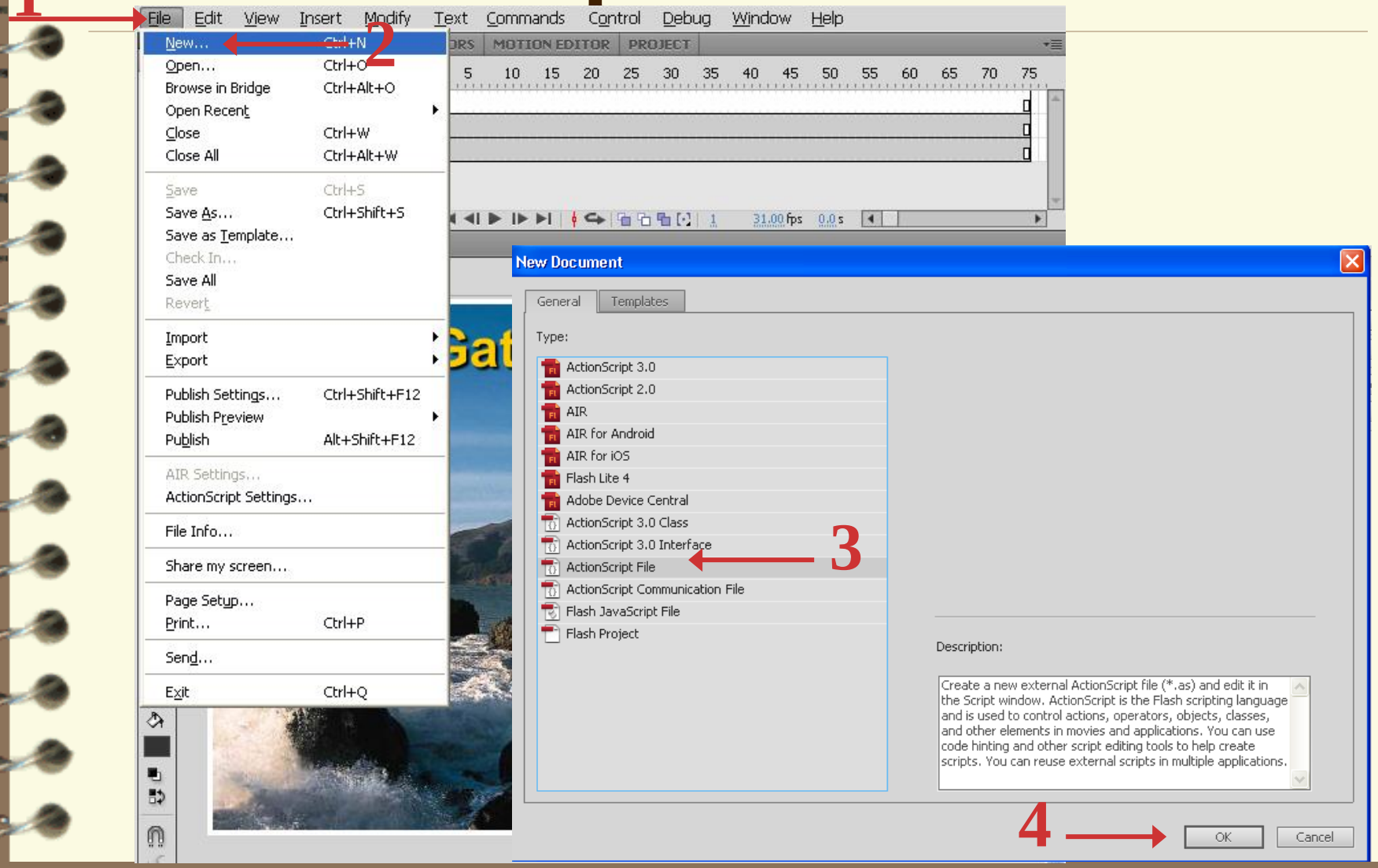
- You can compose your scripts in a separate **ActionScript (.as)** file and include them in your Animate movie.
- This is great way to keep your Scripts organized, and it also makes them easy to reuse in other projects.

Create and Include an External ActionScript File

Create the Script File

1. Click **File.**
2. Click **New**
3. Click **ActionScript** File
4. Click **OK**

Create and Include an External ActionScript File

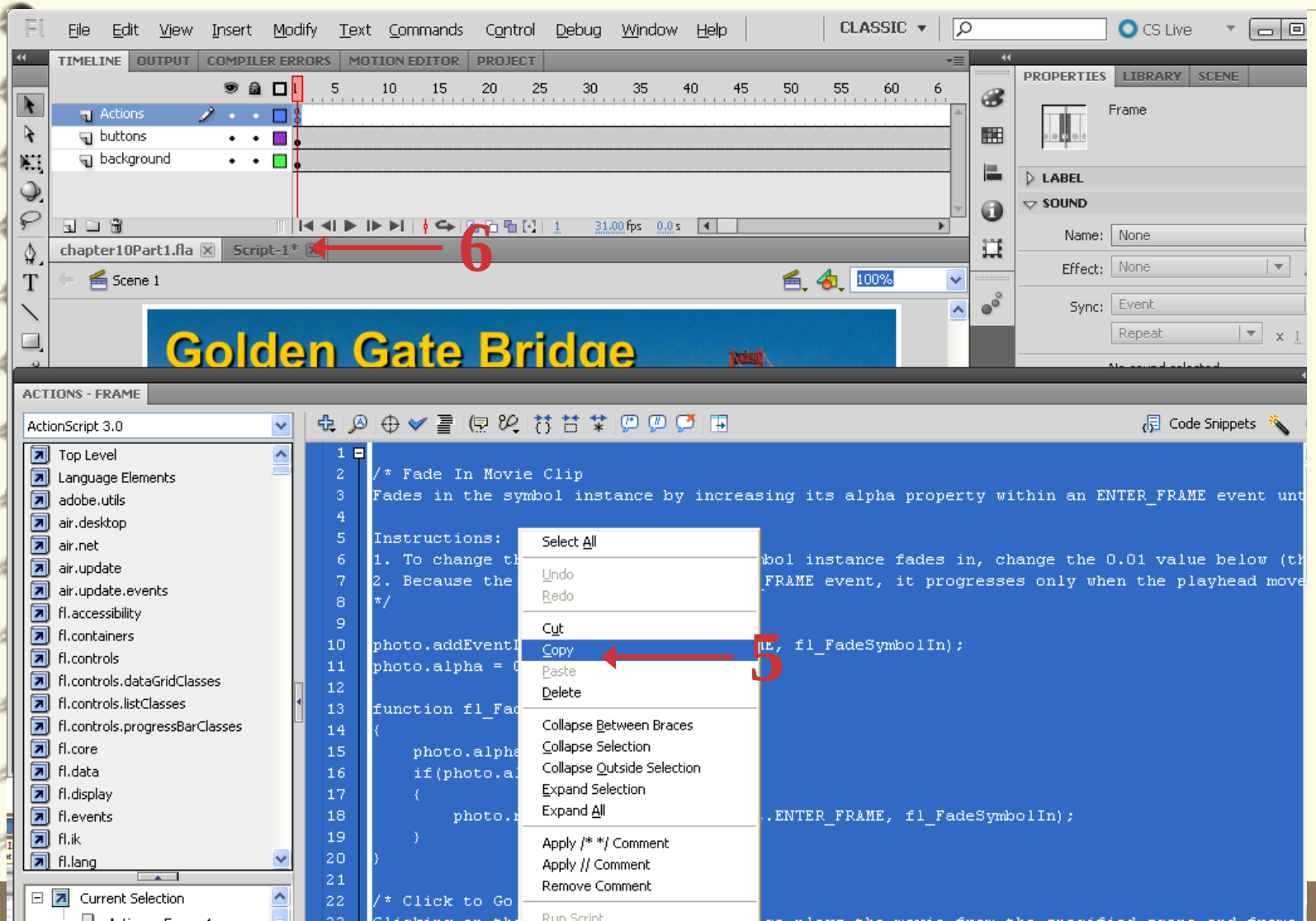


Create and Include an External ActionScript File

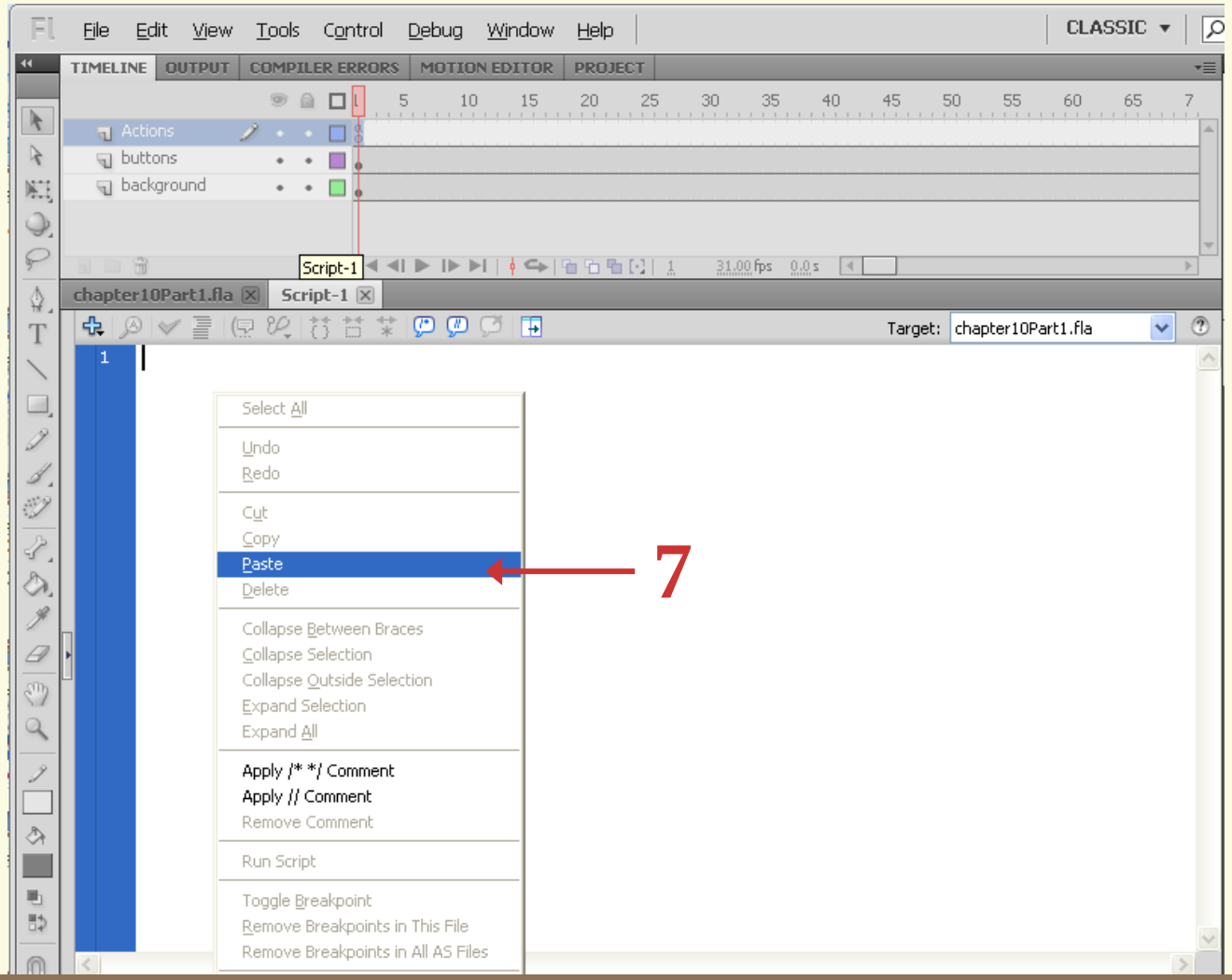
Create the Script File

5. **Copy** the ActionScripts you created in Scene 1 for the movie clip and buttons.
6. Toggle back the ActionScript
7. **Paste** the ActionScripts in the ActionPanel.
8. Remove all lines except the **calling functions**.

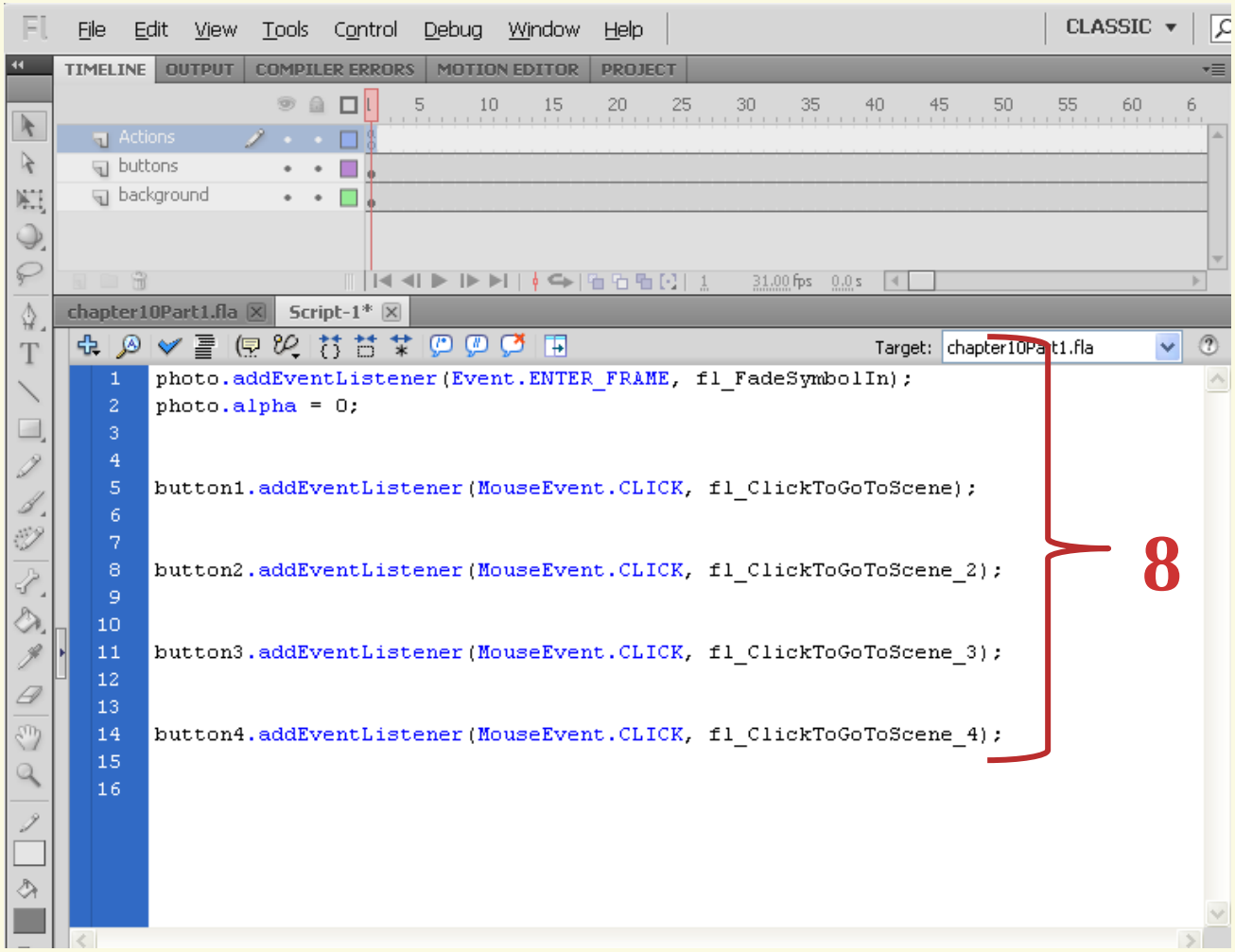
Create and Include an External ActionScript File



Create and Include an External ActionScript File



Create and Include an External ActionScript File

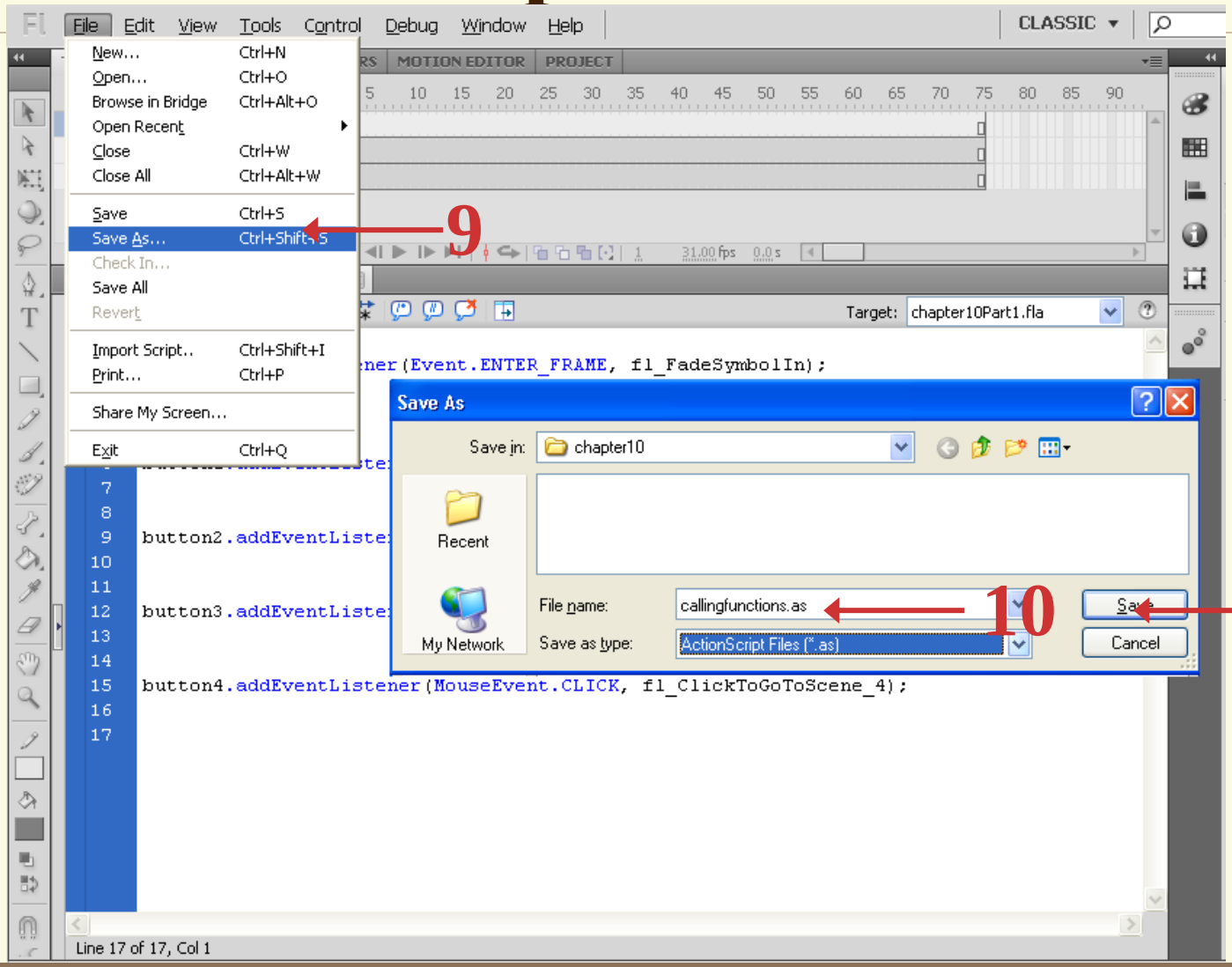


Create and Include an External ActionScript File

Create the Script File

9. Select **File** and then **Save As**
10. Enter the **name** of the ActionScript - *"callingfunctions.as"*
11. Click on the **Save** button

Create and Include an External ActionScript File



Create and Include an External ActionScript File *(continue)*

- Your external scripts are executed on the frame where you include them, unless they only contain function and variable declarations.

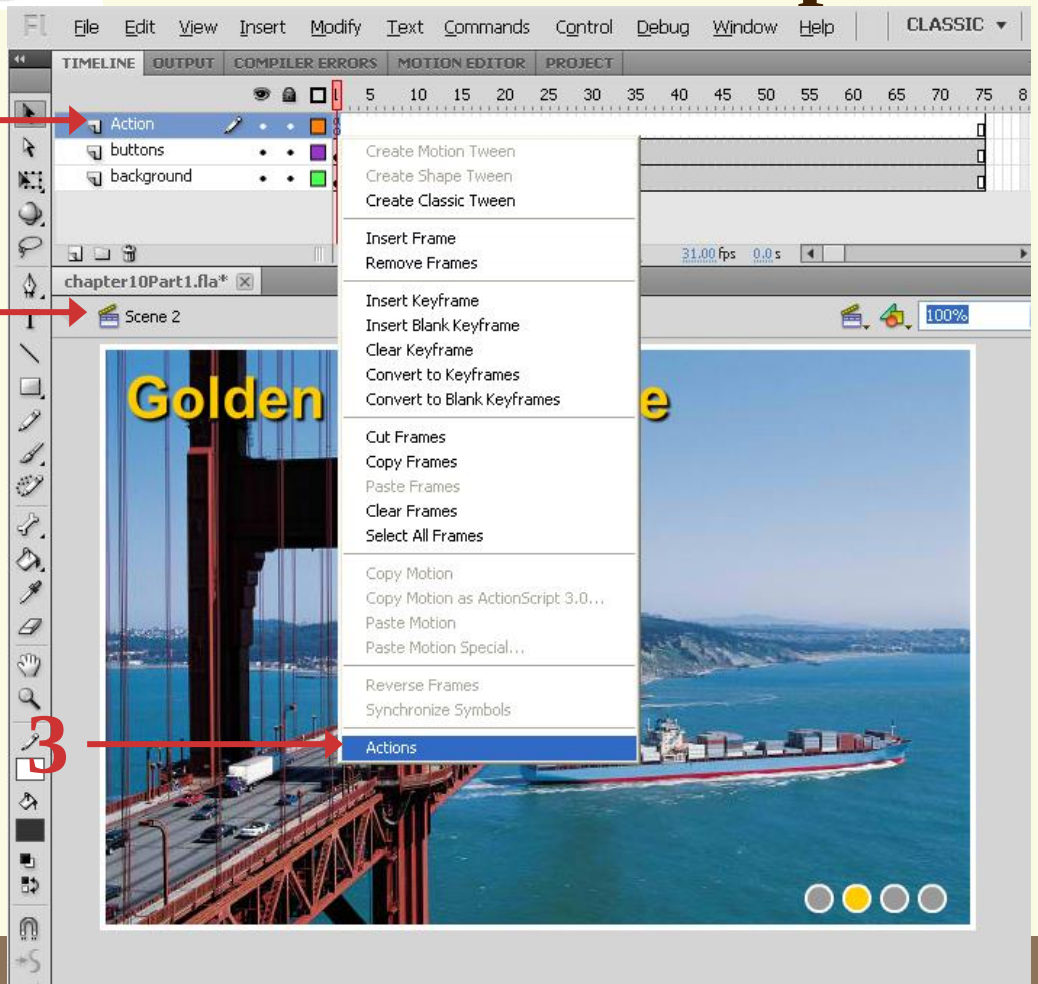
Create and Include an External ActionScript File

Include the Script in Your Movie

1. Go to the **Movie scene**
2. Insert a **New Layer**
3. Right click on the blank keyframe and select **Action**

Create and Include an External ActionScript File

Include the Script in Your Movie



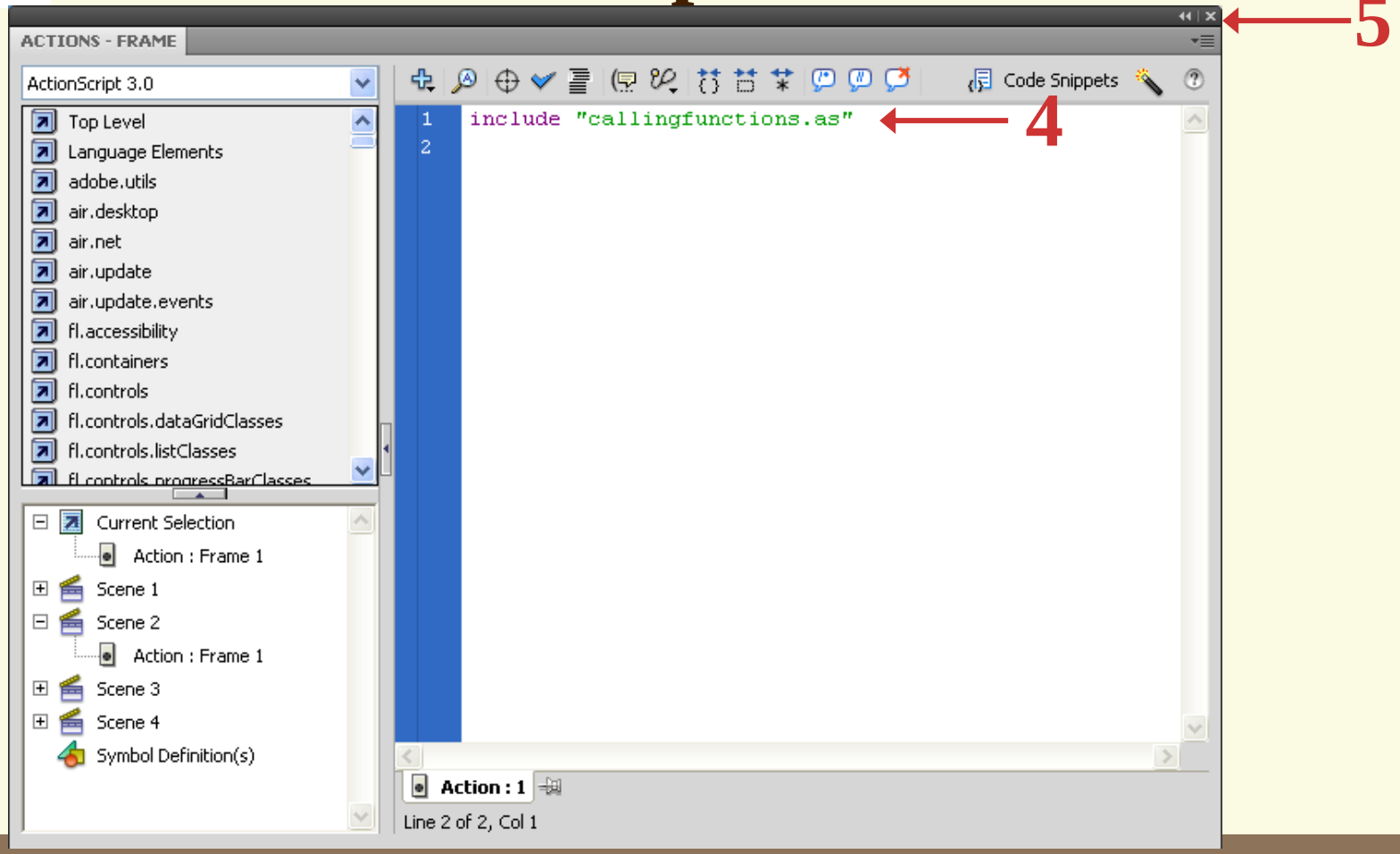
Create and Include an External ActionScript File

Include the Script in Your Movie

4. In the ActionPanel, type *include "callingfunctions.as"*
5. Close the Action Panel window
6. Repeat steps 1 to 5 for the rest of the scenes.

Create and Include an External ActionScript File

Include the Script in Your Movie



Create and Include an External ActionScript File

Can I create external ActionScript files with other programs?

Yes. Many developers prefer to use external editors to write their script, like TextMate or Eclipse.

You must make sure that whatever editor you use, that you save your file as plain text, to ensure that Animatees can read it properly.