ActionScript® 3.0 for
ADOBE® FLASH® PROFESSIONAL CS5

CLASSROOM IN A BOOK®

The official training workbook from Adobe Systems

CD-ROM Included for Windows and Mac OS

ActionScript 3.0 for Adobe® Flash® Professional CS5 Classroom in a Book® Instructor Notes

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names in sample files are for demonstration purposes only and are not intended to refer to any actual organization.

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110-2704, USA

Adobe Press books are published by Peachpit, a division of Pearson Education located in Berkeley, California. For the latest on Adobe Press books, go to www.adobepress.com. To report errors, please send a note to errata@peachpit.com. For information on getting permission for reprints and excerpts, contact permissions@peachpit.com.

Writer: Chris Florio
Project Editor: Rebecca Freed
Development Editor: Judy Ziajka
Production Editor: Danielle Foster
Copyeditor: Scout Festa

Technical Editor: Angela Nimer
Compositor: Danielle Foster
Indexer: Rebecca Plunkett
Cover design: Eddie Yuen
Interior design: Mimi Heft

Printed and bound in the United States of America

ISBN-13: 978-0-132-48156-4
ISBN-10: 0-132-48156-1

9 8 7 6 5 4 3 2 1

INSTRUCTOR NOTES

Getting Started

The Adobe ActionScript 3.0 for Flash CS5 Classroom in a Book course presents students with tips, techniques, and solutions for using the Adobe ActionScript 3 for Flash CS5 software. The Instructor Notes are intended to complement the information in the Adobe ActionScript 3.0 for Flash CS5 Classroom in a Book.

The information is organized to follow the sequence of instruction in each lesson.

About the workbooks

It is recommended that each student in the class have an individual copy of the Adobe ActionScript 3.0 for Flash CS5 Classroom in a Book. Students will use this book as you lead them through projects. Additionally, the book can be used as a self-paced tutorial.

Course strategy

The book contains and an introductory chapter and 15 lessons, some of which may take a while to complete. If you're teaching a 15-session class, you can teach approximately one chapter of this book per class with the introduction and lesson one combined into a single session. Depending on the length of the class sessions students may need to complete some of the lessons as homework assignments. All of the lessons introduce new material progressively and should be approached sequentially.

If you are teaching a shorter session class, you may want to follow the chapter sequence for the earlier lessons and assign the later lessons as extra-credit, out-of-class assignments. We recommend that, at an absolute minimum, you teach Lessons 1–8, which introduce many essential concepts and techniques. Of course, completing all the lessons makes a far superior introduction. Lessons 9 and 10 work together to introduce techniques for controlling sound with ActionScript and working with XML in ActionScript. Lesson 11 covers some of the video features in ActionScript and expands on the XML techniques introduced in Lesson 10. Lesson 12 covers a number of recent Flash features, including the IK tools and Pixel Bender filters, as well as working with the user's camera and bitmap data in ActionScript. Lesson 13 covers printing and sending email with ActionScript; Lesson 14 is a brief overview of developing Adobe AIR 2 applications with Flash; and Lesson 15 introduces techniques for working with third-party ActionScript libraries such as PaperVision 3D.

Before beginning a lesson, encourage students to browse through it and read any sidebars.

Managing student projects

One way to simplify file storage and retrieval in classroom situations is to ask students to create a folder on their hard disks and name it [student's name] Lessons. They should then copy each project folder into that Lessons folder. Having students keep all their working files in their own Lessons folders makes it easy for you to clean up files when a class is over.

Additional resources

For complete and up-to-date information about using Flash CS5 panels, tools, and other application features, visit the Adobe website. You can also find Flash learning resources from within Flash CS5 by choosing Help > Flash Help. Here you can search Flash Help and support documents, as well as link to websites relevant to Flash users.

For additional resources, such as tips and techniques and the latest product information, check out the Adobe Community Help page at community.adobe.com/help/main.

Adobe Support

The U.S. Adobe website at www.adobe.com contains a variety of Flash training and support resources, including the Adobe Design Center and the Adobe Developer Center for Flash. Many step-by-step tutorials, galleries, and other resources are available at the Adobe website, and the User-to-User Forums are a great place for students to learn from more experienced users. Adobe Flash Support, the Adobe Design Center, and the Adobe Developer Centers provide hundreds of tutorials from experts and authors in the community, as well as thoughtful articles about Flash and ActionScript.

Go to:

www.adobe.com/support/flash/

www.adobe.com/designcenter/

www.adobe.com/devnet/flash/

Adobe Certified Expert (ACE) program

This program is designed to validate an expert skill level of Adobe Flash CS5 and ActionScript 3.0. Careful testing of candidates ensures that each ACE has demonstrated expert product knowledge of the current release of Adobe Flash, resulting in increased marketability and an added credential. Training for the ACE program is available through Adobe Authorized Training Center (AATC) and self-study.

For more information about this program, send an e-mail to certification@adobe.com or visit the Adobe website at: www.adobe.com.

Introduction to ActionScript 3.0

This introductory chapter provides a brief overview of the history of Flash and ActionScript, as well as some information on the changes between ActionScript 3 and earlier versions of the language. The amount of time that you spend on this material should vary depending on the experience of your students. If you are not already aware of whether they have experience with ActionScript 1 or ActionScript 2, this would be a good opportunity to discuss this with them.

The introduction also includes some valuable information about the various formats for viewing Flash projects online and offline. It also addresses questions your students may have about the differences between Flash CS5 Professional, Flash Builder, and Flex.

Lesson 1: Using Code Snippets and Navigating the Flash Timeline

This lesson will introduce the new Code Snippets feature in Flash CS5 as well as the techniques for adding code to the Flash Timeline to control playback. It will also introduce a few essential ActionScript 3.0 programming concepts. When covering the Code Snippets panel you may want to encourage students to take advantage of its convenience and its usefulness as a learning tool but discourage reliance on code snippets at the expense of understand how to write their own code.

While much of the code in these lessons is written on the Flash Timeline, it is important to emphasize to the students the pros and cons of developing on the Timeline as opposed to working in external ActionScript files. Many students may do all of their Flash and ActionScript development on the Timeline while others may be learning ActionScript in order to move on to object-oriented application development with ActionScript that does not use the Timeline at all. Most Flash and ActionScript users are somewhere in between, and you can emphasize to students the value of knowing when to use the Timeline and when to use code exclusively. This is a discussion that can be returned to as your students get more comfortable with ActionScript.

Adding ActionScript using Code Snippets

Page 16

In addition to introducing the Code Snippets panel, this section will be the students' first use of the Actions Panel. Whatever time is needed to establish comfort with this panel would be well spent. Spend time with the code snippet example used in this lesson making sure that the students understand the comments in the code snippet.

Placing code on the Flash Timeline

Page 20

Depending on the users' experience, you may want to review the concept of keyframes and emphasize that no ActionScript can be stored on any frame in a Flash Timeline unless a keyframe has been added first.

Working with frame labels

Page 20

When demonstrating how to use frames in the Flash Timeline, you can emphasize the value of adding labels. The use of labels in the Flash Timeline both helps clarify the project but also makes it much easier to update the navigation in a project when changes are made to the Timeline.

Looping playback with ActionScript

Page 21

When Demonstrating the `gotoAndPlay()` and `gotoAndStop()` methods emphasize the usefulness of this action for navigating Timelines in Flash. You may want to mention the number of variations available for these methods , which allow for things like navigating between scenes, navigating other Timelines, and navigating to the next and previous frames.

Keeping track of the number of loops

Page 22

This section introduces the essential concept of storing data in a variable and modifying its value. It is worth taking the necessary time to emphasize the importance of variables in ActionScript and talk about the various datatypes that can be stored in a variable. How in-depth you go into this topic at this point may depend on the experience level of your students.

Using a conditional statement to control the Timeline

Page 25

Conditional statements are another essential concept that you may want to particularly emphasize. Consider providing examples of how conditional statements are used in websites, games, e-commerce applications, training programs, etc. Focus on uses that may be of particular interest to your students.

Updating the text on the home frame

Page 26

As you introduce the idea of manipulating text fields with ActionScript, you can talk about the wide range of possibilities for creating, formatting, and controlling text with code. Let students know that they will be covering many of these possibilities in coming lessons. If the students have not spent time with the new TLF text tools in Flash CS5, you may want to go over the new text features in the authoring environment at this time.

Questions

1. How is a code snippet added to a Flash CS5 project?
2. What is the keyboard shortcut to open the Actions panel on a Windows machine? On a Macintosh?
3. How do you create a label name for a frame in the Timeline?
4. What are the rules for naming a variable?

Answers

1. A code snippet is added to the Actions panel by opening the Code Snippets panel (Window > Code Snippets) and double-clicking a desired code snippet. If the selected snippet contains code that is triggered by interacting with an onstage graphic, then that graphic needs to be selected onstage before the code snippet is double-clicked.
2. Open the Actions panel by pressing F9 (Windows) or Option+F9 (Mac).
3. To add a label to a frame in the Timeline, select the frame and add a keyframe to the frame if one does not exist already. Then, with the frame still selected and the Properties panel visible, type a new label name into the label area of the Properties panel and press Enter.
4. You can choose any name for a variable, as long as you follow these three rules: Do not use spaces; use underscores instead. Except for underscores or dashes, do not use special characters; stick to letters and numbers. Do not start your names with a number.

Lesson 2: Working with Events and Functions

Understanding the event model in ActionScript 3.0 is probably the biggest step in mastering the basics of the language and being able to create rich interactive applications. This lesson introduces the event model using mouse click events. Emphasize to your students the wide range of events that are possible in Flash. Explain that by understanding the techniques covered in this lesson they will have learned the process used to work with all events in ActionScript.

This lesson also introduces functions. You can tell students that when they get comfortable with the material in the first two lessons—including variables, conditional statements, events, and functions—they will have an understanding of the basic building blocks of not only the ActionScript language but also of many other programming languages.

Creating event listeners and event-handling functions

Page 32

This is one of the core concepts in ActionScript. Many students grasp this quickly, while others struggle with it for a considerable time. Take as much time with this section as you feel is necessary to be sure that your students grasp the concepts. Clarity on the materials in this section will help in all the coming lessons.

This material will be reinforced many times in the coming lessons.

Using Code Snippets to create navigation

Page 34

This chapter first shows how to create an event listener using a code snippet and then goes through the process of writing an event listener by hand. You can determine how much emphasis to place on using code snippets based on students' experience level.

Creating button instances to control navigation

Page 35

One of the most common mistakes for ActionScript beginners is to forget to create instance names for interactive objects. This is a good time to emphasize the importance of instance names.

Creating Event Listeners

Page 40

When introducing the `CLICK` event you may want to let students know that this is just one of many possible mouse events that they can use in ActionScript.

The syntax of functions is critical for ActionScript students to master, but often it takes time to grasp the meaning of some of the syntax. How much time you take explaining the concepts of parameters and return types can depend on the level of your students. Sometimes it is best to just go over these concepts briefly and return to them when students have more experience.

Questions

1 What is a function?

2 How is a button, component or movie clip given an instance name in the Flash CS5 interface?

Answers

1 A function is a block of code that performs one or more, usually related, tasks. Functions are created using the keyword "function" followed by the function name and a pair of parenthesis that may contain parameters. The tasks to be performed are contained within curly braces.

2 To give an object an instance name in the Flash interface, select the object with the Properties panel visible. In the Properties panel, type a name in the Instance Name field and press Enter.

Lesson 3: Creating Animation with ActionScript

This lesson lays a foundation for creating motion with ActionScript. It introduces the `ENTER_FRAME` event and uses it to trigger functions that animate the properties of a movie clip. The lesson also introduces the Tween and Easing classes.

Controlling movie clip properties with ActionScript

Page 50

If your students are comfortable with the design tools in Flash, then the material in this lesson can be presented as techniques that let them do things in ActionScript that they are already doing in the interface. Reinforce that there are situations when animation in the interface makes sense and when using ActionScript to create animation is preferable.

Changing a property value

Page 51

The ability to change a movie clip property in code allows for many creative possibilities. Give students a chance to experiment with altering movie clip properties either in class or in homework assignments. You may want to show the students the full list of MovieClip properties in the Flash help files.

Increasing or decreasing a property's value

Page 52

This section introduces operators that increment and decrement a movie clip's properties. Depending on your students' experience, you may want to discuss operators in ActionScript in more detail and talk about some of the other operators available in ActionScript 3.

Using an ENTER _FRAME event to animate a MovieClip property

Page 53

This section introduces the very useful `ENTER_FRAME` event. Point out to students that this event repeats at the current frame rate independent of what is happening on any Timelines. The `ENTER_FRAME` event is extremely useful for situations where code needs to be repeated constantly. You can point out that there are also other possible techniques (such as the Timer class) for repeating code.

Creating a variable to store a movie clip reference

Page 54

This is the first time that students have used a variable to store a reference to a movie clip instance. Remind students that variable can store many types of data in ActionScript. For more experienced students you may want to give a more detailed discussion of data types at this point.

Changing MovieClip properties with buttons

Page 56

This section combines many of the concepts that have been introduced so far to change a movie clip's properties when a `CLICK` event takes place. For many ActionScript beginners, getting confident with these techniques creates an overall comfort with learning the language. If students have doubts about their ability to work with code, it may be worth taking a little extra time with these sections to make sure they are secure with this material.

Creating animation using ActionScript tweens

Page 57

The Tween class is fun to work with and is also a good opportunity to clarify how parameters work. Students often quickly understand the purpose of Tween instance parameters, which can lead to a grasp of working with parameters throughout the ActionScript 3.0 language.

Importing the Tween and easing classes

Page 59

For simplicity, the lessons in the book only explicitly import classes when necessary to make the lesson files work. Make sure that the students understand what is happening when Flash CS5 automatically adds import statements. If your students are learning ActionScript to eventually develop object-oriented applications, you may want to encourage them to import all the classes they use.

Adding ActionScript tweens

Page 59

Students often enjoy playing with ActionScript tweens, and you may want to show them some additional easing options with ActionScript.

Questions

1. Describe the ActionScript syntax to import an external class into a Flash project.
2. When is the syntax to describe the current width of the Flash Stage? The current height?
3. What is the range for the alpha property of a movie clip?

Answers

1. To import an ActionScript class you use the keyword import followed by the path to the class that you wish to import. For example to import the Tween class you would type:

   ```
   import fl.transitions.Tween;
   ```

2. The width of the Flash Stage can be described as stage.stageWidth, the current height can be described as stage.stageHeight.
3. The alpha property of a movie clip can be set to a range between 0 (completely transparent) and 1 (completely opaque).

Lesson 4: Creating ActionScript in External Files

Up until this point, all the lessons have created ActionScript in frames of the Flash Timeline. In this lesson ActionScript will be created in an external .as file and integrated into a Flash project. Some students may wind up pursing OOP development in Flash and wind up working extensively or even exclusively with ActionScript files and others may choose to always work in the Timeline.

Try to use this chapter as an opportunity to explain the options for writing ActionScript and make it clear that there is no universal wrong or right approach but instead they should try to choose the approach that works best for each individual project.

Creating an ActionScript file

Page 68

Emphasize to students that the tools in the Flash application are not the only option for creating ActionScript files. Discuss the pros and cons of using Flash Builder or other development environments. In a more advanced class where Flash Builder 4 (or higher) is available, you may wish to show the students how to work with class files from a Flash CS5 project in Flash Builder.

Basic structure of an ActionScript 3.0 class file

Page 70

Frequently when students begin to grasp the way that class files work in ActionScript, their entire understanding of Flash expands. Take as much time as you can to fully explain the structure and use of class files. Make sure the students understand the concept of packages, constructor functions, methods, and their parameters.

This section also introduces the drawing API. If your students are more advanced, you may want to show them more features of the drawing API.

Creating instances of a class file in Flash

Page 74

This section demonstrates the very important technique of creating variables to store instances of custom classes. It also covers adding display objects to the Stage with the `addChild()` method.

Overriding the parameters of each Ellipse instance

Page 77

New users often need a while to fully grasp this concept. Explain the difference between required and optional parameters and how the parameters of a constructor function can be set when a new instance of a class is created.

Turning the makeShapes() function on and off

Page 78

This section introduces the `MOUSE_MOVE` event as well as the technique for removing an event listener.

Randomizing the color of the ellipses

Page 79

This section introduces the `random` method of the `Math` class and uses it to select a random hexadecimal number. You may wish to challenge the students to try create a random number in a specific range or guess the color that a specific hexadecimal value represents.

Questions

1 What is the suffix for an ActionScript file?

2 What ActionScript keyword is used at the start of an ActionScript file and describes the path to that file?

3 How does ActionScript indicate that a number is a hexadecimal value?

Answers

1 An ActionScript file is a plain text file whose file name ends with the suffix .as.

2 The "package" keyword is used at the beginning of an .as file to describe the path to that file.

3 A hexadecimal number in ActionScript is preceded by the characters 0x.

Lesson 5: Using ActionScript and Components to Load Content

Most large Flash projects consist of a number of SWF files in addition to supporting content and data that is loaded at runtime.

The main goal of this lesson is to integrate multiple files into a single Flash project, This lesson is particularly important for students who will be using Flash to design websites with limited interactivity but a lot of content.

Creating a List component instance and setting its parameters

Page 86

This lesson introduces the student to a number of useful user interface components in Flash. Use the `List` component to illustrate how the properties of a component can be set in either the Component Inspector panel or with ActionScript. For users of older versions of Flash, you may want to point out that the Component Inspector is no longer available and that the Property Inspector should be used instead.

Adding an instance of the UILoader component

Page 88

The `UILoader` instance is frequently used by web developers. This component will be seen again in Lesson 6, "Creating Preloaders in ActionScript 3.0."

Examining the gallery.fla file

Page 91

Be sure that students understand the setup of the gallery file, especially the way the thumbnails movie clip is designed with buttons inside it.

Adding event listeners to the thumbnails

Page 91

Be sure that students understand the process of targeting buttons that are nested inside movie clips.

Loading text from an external file

Page 93

This section introduces the `URLLoader` class and uses its methods to load text from external files.

It also shows how to set the properties of a text field with ActionScript. The techniques involved in loading external files in ActionScript can be confusing for beginners, so be sure to spend the needed time explaining each step of the process, especially the use of the `COMPLETE` event.

Adding a scroll bar to the text field

Page 97

This section shows how to link a `UIScrollBar` component to a text field to create scrolling text. If a student's scrollbar is not working, have them check that the multiline property is selected for their text field in the Properties panel.

Questions

1 What is the parameter of a List component that stores the list's labels and data?

2 What ActionScript class can be used to load text or data into a file?

Answers

1 The `dataProvider` parameter of a list component stores the components labels and data.

2 The `URLLoader` class can be used to load text or data into a file.

Lesson 6: Creating Preloaders in ActionScript 3.0

This chapter is focused on the tools for monitoring load progress and giving the user feedback while the content of a Flash project is loading.

The lesson goes over a number of tools in the Flash testing environment as well as the basics of using a preloader to track the loading of content into Flash Player, give the user feedback while loading content, and to respond appropriately when the content is fully loaded.

Tools in the testing environment

Page 102

By taking advantage of the tools in the testing environment, Flash developers can work with their projects offline and still test and optimize their files.

Bandwidth Profiler

Page 103

The Bandwidth Profiler is an extremely useful tool that gives the developer visual feedback on the size of the content in the frames of a Flash Timeline. This feedback helps identify places that may cause potential problems downloading Flash files.

Simulate Download

Page 105

This tool is used with the download settings (View > Download Setting) in the testing environment. Choosing Simulate Download lets you experience what it would be like to stream and view a Flash project at the connection speed currently selected in the download settings.

Creating a text field and progress bar to track loading

Page 106

Preloaders can be designed in many ways, but this lesson focuses on the most common type of preloader, which consists of a progress bar and a text field, which gives the user feedback on download progress.

Inserting ActionScript to track progress of the UILoader component

Page 108

The preloader that is created in this lesson uses the built-in methods and properties of the `UILoader` class. Let students know that preloaders can be created to track the loading of any content, not just content loaded into a `UILoader` component.

Adding event listeners for PROGRESS and COMPLETE

Page 109

The `PROGRESS` and `COMPLETE` events of the `UILoader` class make the process of creating preloaders much easier. Again, emphasize that preloading can be done without these events simply by tracking the `BytesLoaded` and `BytesTotal` properties of any Flash content.

Controlling the frames of a MovieClip to reflect loading progress

Page 113

This is a simple example of an alternative way to give the user feedback while content is loading. Encourage students to be creative in creating variations on the preloader concept. If there is time, have your students try to create their own preloaders with their own content.

Adding ActionScript for the loadAnimation clip

Page 114

Emphasize the advantage of having 100 frames in a movie clip that advances as the download progresses. If there is time, you may want to demonstrate how to do the math to use this technique with a movie clip of more or less than 100 frames.

Questions

1 What is the purpose of a preloader in Flash?

2 Name one way to enter the Flash testing environment.

3 Which user interface component gives a user visual feedback on loading progress?

Answers

1 A preloader is used to track the loading of content into Flash Player; to give the user feedback, when needed, on that loading; and to respond to the results of the loading.

2 The Flash testing environment can be entered by choosing Control > Test Movie from within Flash.

3 The Progress Bar component is used to give a user visual feedback on loading progress.

Lesson 7: Using Arrays and Loops in ActionScript 3.0

This lesson introduces two very important ActionScript tools: the array and the `for` loop by creating a simple game. Explain that arrays and `for` loops are important tools in many programming environments. The students should now think of themselves as going beyond the beginning steps in ActionScript 3.0. Assure your students that once they are comfortable working with arrays and `for` loops, many types of functionality will be possible in their projects. You may want to give them some other examples that are relevant to their areas of interest.

Adding MovieClip instances to the Stage from the Library

Page 123

Adding clips from the library to the Stage with ActionScript is a good use of the hybrid design/programming nature of Flash.

It is not covered in the chapter, but an alternative way to accomplish the technique covered in this step is to use the `embed` tag that is available in Flash CS5. (See the Flash Help files for details).

Setting the linkage properties of a MovieClip

Page 124

Many beginners forget this step, so be sure to emphasize to students the importance of setting the linkage properties of a movie clip if it needs to be added to the Stage at runtime with ActionScript.

Adding a new instance of the Apple class to the Stage

Page 126

Emphasize to students that the code for making instances of clips in the library is identical to the code for making instances of other ActionScript classes.

Introducing the Array

Page 127

Students who do not immediately grasp the concept of arrays often are helped by seeing multiple examples of how arrays are typically used. Discuss a few uses of arrays in projects that relate to the students' areas of interest.

Generating multiple instances with a for loop

Page 129

Many students who understand the basics of working with conditional statements take a while to grasp the use of `for` loops.

Often writing out a long "if" statement that does the same thing as a `for` loop helps clarify the syntax and illustrates the convenience of using a `for` loop instead of the longer technique.

When going through the syntax of a `for` loop, emphasize the semicolons between statements within the parentheses. Many new programmers make the understandable mistake of using commas between the statements.

Adding items dynamically to an array with push()

Page 132

If students are comfortable with the basics of using arrays, you may want to discuss other methods available in the `Array` class.

Generating drag-and-drop functionality

Page 132

If students are grasping the basics of drag and drop, you may want to go over the optional parameters for the `startDrag()` method.

Using hitTestObject() to check for collisions

Page 137

Depending on the level of the class, you may want to have a broader discussion of collision detection and discuss the pros and cons of `hitTestObject()`, as well as other alternatives for more advanced collision detection.

Adding conditional statements to determine the game's outcome

Page 142

Tracking the outcome of a game or quiz is a use of conditional statements that many students quickly grasp; they will need to do this frequently in their own projects. This can be an opportunity to solidify their knowledge of conditional statement syntax.

Questions

1. Once a movie clip in the library has its linkage properties set, how are instances of it created with ActionScript? How is that instance added to the Stage?
2. What method is used to make a movie clip draggable? What method is used to stop it from being dragged?
3. What method is used to remove an element from an array?

Answers

1. A new instance of a movie clip on the library is created with the keyword `new`, just like an instance of any other class. For example, to make an instance of an object in the library with a linkage identifier of `Ship`, you could write:

   ```
   var ship1:Ship = new Ship
   ```

   to add the object to the Stage you would write:

   ```
   addChild(ship1)
   ```

2. The `startDrag()` method makes a movie clip draggable, and the `stopDrag()` method stops it from being dragged.
3. The `splice()` method is used to remove specific elements from an array.

Lesson 8: Creating and Formatting Text with ActionScript

This lesson focuses on the basics of creating and formatting text with ActionScript. It also reviews the process of loading text from external files. For beginning students you may wish to review the text features in the Flash authoring environment before embarking on ActionScript control of text. For a more advanced group you can emphasize that the material covered in this lesson merely scratches the surface of the features available in ActionScript for working with text and note some of the lower-level text classes that are available.

Creating a TLF text field with ActionScript

Page 156

You may want to show students the range of methods and properties available in the TLFTextField class as well as compare that to the old TextField class. If the students have graphic design backgrounds, you can also point out how much better Flash is at working with text from InDesign projects.

Loading an external text file into a TLF text field

Page 158

This is a good opportunity to review the process of loading data from an external file. Many students need to do this a few times before getting comfortable with the techniques.

Using the TextFormat class

Page 159

The `TextFormat` class was chosen for this lesson so that new programmers could control text formatting without a huge learning curve, but you may wish to make advanced students aware of the other classes in ActionScript that provide more powerful control of text, such as `TextFlow` and its related classes.

Many students who have experience with web design understand the `TextFormat` class by comparing it to using CSS in an HTML web page.

Giving the user a custom panel to format text

Page 160

There are many features that could be added to this lesson, and you may wish to give students an out-of-class assignment to enhance this project.

After they have completed Lesson 14 on Adobe AIR, you may wish to revisit this lesson and convert the project to an AIR application. If there were time at this point you could show students how to create multiple windows for the formatting controls.

Creating a scroll bar using ActionScript

Page 168

This lesson shows a number of very useful ways to work with scroll bars. You can show students that in some cases the `TextArea` component can offer an alternate and very simple way to create text with scroll bars.

Questions

1 Name two keyboard events that are available in ActionScript 3.0.

2 Name a method that is available to the `UIScrollbar` component that allows you to pass both x and y coordinates to move the component.

3 What is one way that you could check whether the right arrow key on the user's keyboard was pressed?

Answers

1 The two keyboard events available in ActionScript are `KEY_DOWN` and `KEY_UP`.

2 The `move()` method allows you to pass both x and y coordinates to move a component.

3 To listen for whether the right arrow key was pressed, you would create an event listener for the `KEY_DOWN` event, and in the event handling function create a conditional statement to determine if the right key was pressed.

```
//within a keyboard event handling function with an event
¬parameter of “e”:
if(e.keyCode == Keyboard.RIGHT){
//do something
}
```

Lesson 9: Controlling Sound with ActionScript

This lesson focuses on three sound-related classes: the `Sound`, `SoundChannel`, and `SoundTransform` classes. They work together to give you control over individual sound files. This lesson uses these classes to build a simple music player application.

Adding sliders to the project

Page 180

The `Slider` components that are added in this step will be used later in the lesson to control the volume and panning of audio files controlled with ActionScript.

The Sound, SoundChannel, and SoundTransform classes

Page 183

This section introduces the relationship between the three sound-related classes that are used in the lesson. Be sure that students understand this relationship.

Creating Sound, SoundChannel, and SoundTransform instances

Page 183

This lesson uses one instance of each of these classes. This section creates the variables that will store those objects.

Adding more variables

Page 185

Three more variables are created, which will be used to keep track of the song that is currently playing in the music player, as well as the current volume and pan settings.

Creating the songList array

Page 185

In this lesson the list of songs to be used in the music player application will be stored in an array called `songList`. In the next lesson this project will be further developed and the song list will be moved to an external XML file.

Setting the song titles using a for loop

Page 185

A `for` loop is used to go through the elements of the `songList` array to place the titles of the songs into text fields. This will give students more practice working with arrays and `for` loops together.

Modifying text fields with the replace() method

Page 187

The `replace` method is used here to shorten the strings used in the title text fields. This illustrates a couple of techniques useful to students, including manipulating text strings and evaluating values to produce the name of an object.

Making the sliders invisible until needed

Page 188

Point out to students the value of using ActionScript to make interface elements visible and invisible as appropriate.

Adding event listeners to the song buttons

Page 189

Rather than writing a separate function for each button on the Stage that can choose a song, all the buttons call the same `chooseSong()` function.

Creating the chooseSong() function

Page 189

The `chooseSong` function uses a `switch` statement that takes the name of the button that was clicked to determine which song will be played. Be sure students understand the way the name property of the target is being used in this function.

Creating a Sound class instance and checking for existing instances

Page 191

This section uses a conditional statement to check to see that a sound is not already playing before creating a new sound.

This step may need to be clarified for students, because the conditional refers to a `SoundChannel` instance (`channel`) that will not be created until a step that follows. This will be the `SoundChannel` that the sound (`snd`) will be played through. Emphasize the note on page 191 that points out that a web-based project would use events to track the loading of mp3 files. In order to focus on the sound-related classes being introduced and for the sake of simplicity, this lesson does not use progress or error events.

Creating the SoundChannel and SoundTransform instances

Page 191

The `SoundChannel` instance will control the playing and stopping of sounds and the SoundTransfrom instance will control the volume and panning controls.

It is a good idea to emphasize the warning on page 192 regarding the range for volume settings.

Controlling the visibility of the volume and pan controls

Page 193

Once a song has been chosen to play the volume and pan sliders that were hidden in an earlier step are made visible.

Adding a listener for the ID 3 tags of an MP3 file

Page 195

If there is time in class, it may be worthwhile to have students create their own ID3 tags using iTunes or another audio application.

Adding the Artist and Album information

Page 197

This section uses the ID3 tags from the currently loaded song to populate the text fields onstage.

Adding a text format object

Page 198

An opportunity to review the `TextFormat` class. Encourage your students to experiment with their own `TextFormat` settings.

Adding the slider controls

Page 200

This section adds ActionScript to set the volume and pan of the currently playing song based on the value of the two sliders onstage.

Adding the volumeChange() and panChange() functions

Page 200

When a slider is moved, then a `CHANGE` event listener is used to control volume and pan settings. Be sure students understand the simple math equations used to translate the volume and pan settings to user-friendly numbers.

Questions

1. Which ActionScript class must be imported in order to use the `CHANGE` event of a `Slider` component?
2. How are single-line comments added in ActionScript? Multiline comments?
3. What is the range for the volume property of the `SoundTransform` class? What is the range of the pan property?

Answers

1 The SliderEvent class must be imported in order to use the CHANGE event of a Slider component. You would import this class with the following code:

```
import fl.events.SliderEvent;
```

2 Single-line comments in ActionScript begin with two forward slashes. Multiline comments begin with a forward slash and an asterisk and end with an asterisk followed by a forward slash:

```
// this code is a single-line comment

/* this line and the
lines below this
are all
commented out
until the
asterisk-and forward space here */
```

3 The volume property of SoundTransform has a range from 0 (silent) to 1 (full volume). Values higher than 1 can be used but will overdrive the sound. The pan property has a range from -1 (left speaker) to 1 (right speaker). A pan setting of 0 is equally balanced between left and right.

Lesson 10: Working with an XML Playlist

This lesson introduces the very useful techniques for integrating XML data into a Flash project with ActionScript. Additional XML functionality is covered in Lesson 11.

Understanding the basic structure of an XML file

Page 206

The amount of emphasis you place on the XML file format should be determined by the experience level and priorities of your students. The brief overview of XML structure in this chapter is enough for students to be able to complete the lesson, but if they are new to XML it may be worth spending more time discussing the format and how it can be used both with Flash and with in other environments. You can point out that XML is used with everything from video applications such as Adobe After Effects and Apple Final Cut Pro to database software such as FileMaker.

The strong support for XML in ActionScript makes it a frequently used tool among ActionScript developers.

Replacing the songList array with an XML instance

Page 209

This lesson modifies the project from lesson 9 and replaces the array in that lesson, which stored a list of songs, with an XML file that contains a longer list of songs and information about them. Point out to students that a short, consistent list of media or other data is appropriate to store in an array, but if a list is long or will be changed regularly, it is much easier to store the information in an external XML file. By learning both techniques the students can choose the best option for any given project.

Loading an external playlist using the URLLoader class

Page 211

Students were already exposed to the `URLLoader` class in Lessons 5 and 8. This lesson uses this class to load XML data and work with it in Flash.

Responding to COMPLETE and IO _ERROR events

Page 211

Emphasize to students the importance of checking to see if loaded data and content is available before writing code that uses it. In this lesson, all the ActionScript that works with song information from the XML file is moved into the function that takes place when the XML data is completely loaded.

Accessing song title and artist from the XML data

Page 216

This section introduces very useful techniques for accessing data from loaded XML files. Be sure to point out the similarities between these techniques and the way that data is accessed from ActionScript arrays.

Navigating through the song list

page 223

The ActionScript that is triggered by clicking the more_btn will navigate through the song list and keep track of the songs from the XML data. Make sure that students understand the logic of the `songCount` variable and how it is used with the XML data in the `moreSongs()` function.

Questions

1 Which method of the URLLoader class is used to load the data in an XML file?

2 How can you be sure that data loaded with a URLRequest instance is available in Flash?

3 What is a method that creates a hyperlink in ActionScript?

Answers

1 The load method of the URLLoader class is used to load data into Flash. For example you could load data from a local file called songlist.xml with the code:

```
xmlLoader.load(new URLRequest("songlist.xml"));
```

2 To be sure that data loaded with a URLRequest instance is available, you could create an event listener for the COMPLETE event of the URLLoader and use the loaded data within that event-handling function.

3 The navigateToURL() method creates a hyperlink in ActionScript, such as:

```
navigateToURL(new URLRequest(link)"_blank");
```

Lesson 11: Using ActionScript and Components to Control Video

Flash video is a very popular technology, and many developers spend much of their time using the techniques covered in this chapter. An entire volume could be dedicated to ActionScript for Flash video. This lesson introduces the basics for creating a Flash video player that can work with cue points, play a series of videos from an XML playlist, and include captions from an XML caption file.

Adding the FLVPlayback component

Page 231

For most users who work with Flash video, the FLVPlayback component is their primary tool. Let students know that there are lots of methods, properties, and events available for this component beyond what is covered in this chapter. Be sure that the students use the new FLVPlayback 2.5 component and not the older version in their projects.

Setting the FLVPlayback controls

Page 234

Again, stress to students the variety of ways there are to customize the look and controls for an instance of the FLVPlayback component. Show them some of the online resources with information on how to use this component and customize its controls.

Using cue points with Flash Video

Page 242

Be sure that students understand the differences between embedded cue points (navigation and event) and ActionScript cue points, and when to use each type. If you have time, give additional examples of the use of cue points.

Adding the FLVCaptioning component

Page 247

Remind students that in order to use the `FLVCaptioning` component, they should use an XML file that is specifically in the timed-text format.

Playing multiple video files from an XML playlist

Page 250

This is another opportunity to get comfortable with the XML techniques introduced in lesson 10. XML data is again loaded with a `URLLoader` instance and used in Flash as a playlist, this time for the video files that will be played in the FLVPlayback component.

Creating the changeVid() function

Page 253

Be sure that students are clear on the difference between the `COMPLETE` event of the `FLVPlayback` component and the `COMPLETE` event of the `URLLoader` class that they have used previously.

Questions

1 What are the two types of video compression available in the .flv file format?

2 What is the type of video compression used in the .f4v file format?

3 Which property of the `FLVPlayback` component is used to determine which video file is played in the component?

4 Which property of the `FLVPlayback` component determines which .swf file will be used as the controls for the source video?

Answers

1 The .flv file format can store video compressed with either Sorenson compression or On2 VP6 compression.

2 The .f4v file format stores video compressed using H.264 compression.

3. The `source` property of the `FLVPlayback` component is used to determine which video file is played in the component.

4 The `skin` property of the `FLVPlayback` component determines which .swf file will be used as the controls for the source video.

Lesson 12: Delving Deeper into Graphics and Animation with ActionScript

This chapter introduces a number of recently added ActionScript features. Using ActionScript to work with inverse kinematics and Pixel Bender filters offer very powerful (and fun) capabilities. In addition, the `Camera` class and `Bitmap` classes are introduced. The techniques in this lesson only scratch the surface of what is possible with all these classes, so encourage students to continue to explore them once they are comfortable with the basics.

Inverse kinematics in Flash CS5

Page 264

Since this book focuses on ActionScript, this lesson only briefly examines the IK tools in the Flash CS4 interface. If time allows, however, you may want to spend more time showing students how to use the bone tool in the Tool panel to create IK systems; how to animate IK armatures in the Timeline; and how to work with IK controls, particular the new spring features in the Properties Inspector.

Switching between authortime and runtime IK

Page 266

New users of Flash CS5 often forget to switch between authortime and runtime IK in the Properties panel: Emphasize this step with students.

Using the IKMover class

Page 270

This is the main class for creating IK animation. Most students grasp its use fairly quickly, but you may wish to have them experiment with this class in a separate file.

Adding the moveDown() function

Page 271

This is the main function in this lesson and it will be expanded on in the steps that follow this one.

Creating Sound and SoundChannel instances

Page 275

In Lessons 8 and 9 the concept of using multiple sound channels was introduced, but this is the first time in these lessons that two separate sounds have been used in a single project.

Accessing the user's webcam or video camera using ActionScript

Page 277

If students do not have access to video cameras on their computers, you may want to modify this lesson to use only the alternate video file that is included.

Using the Bitmap and BitmapData classes

Page 282

These two classes are very powerful and versatile, and you may want to let students know that the techniques in this lesson show only a fraction of the capabilities in these classes.

Examining the Pixel Bender Toolkit

Page 285

With more experienced students, you may want to go over the code in a few of the Pixel Bender sample filters that ship with Flash. Point students to the filter-writing tutorial and Pixel Bender reference file included with the Pixel Bender Toolkit install.

Examining the PBFilter.as file

Page 288

The book points out only a few of the significant points in this included file. If time permits, you may want to go through some of the code in this file in more detail, particularly the code that uses the FileReference class.

Manipulating a filter parameter with the Slider

Page 293

The PBFilter.as file refers to only a few filter parameters. More experienced users could create additional sliders to add controls for other parameters or even query the loaded filter to determine its parameters.

Questions

1 What does IK stand for?

2 Which ActionScript class is used to move joints in an IK armature?

3 Which method of the Video class is used to place the feed from a live camera into a video object?

Answers

1 IK is short for inverse kinematics.

2 The `IKMover` class is used to move joints in an IK armature.

3 The `attachCamera()` method of the Video class is used to place the feed from a live camera into a video object.

Lesson 13: Printing and Sending Email with ActionScript

This lesson introduces the ActionScript techniques for sending email from Flash in two different ways, as well as how to send content from a Flash project to the user's printer.

Adding a simple email link

Page 302

While this technique is initially attractive because of its simplicity, most users quickly decide that they prefer the technique for sending email from Flash using an external PHP file.

Sending email from Flash

Page 304

The PHP file for this section is included in the Lessons folder. If your students have some PHP experience, you may wish to show them how they can modify this file to offer additional functionality. The lesson file is based on a copy of the PHP file located on the author's server. If you alter the PHP file in the lesson, be sure to load it onto your own server and modify the URL that is used in the lesson.

Sending values using the URL Variables class

Page 305

Make sure that students understand the difference between normal ActionScript variables and variables that are created as properties of the `URLVariables` class.

For more advanced students, this is a good time to discuss the difference between dynamic and static classes.

Sending URL variables to the PHP file

Page 306

The `sendToURL()` method in Step 6 is the core of this section. Point out to your students that this method is useful anytime they wish to send data from a Flash project to any external URL.

Adding printing capabilities with the PrintJob class

Page 309

Many new users are initial confused by the use of the `PrintJob` class but once they understand the relationship between the `start()`, `addPage()`, and `send()` methods it usually becomes much clearer.

Capturing the current date and time with the Date class

Page 310

This lesson only uses the `Date` class in a very simple way. If time allows you may want to show your students ways that the `Date` class can be used to work with and format date and time information more creatively.

Questions

1. What is the syntax used to send email from a text field link?
2. What is an ActionScript method that can be used to send variables to a URL?
3. What does the `start()` method of the `PrintJob` class return when end users confirm the print job?
4. The constructor of what ActionScript class returns the current date and time?

Answers

1. To send email using a text field link in Flash you type `mailto:` followed by the recipient's email address in the link field of the Properties panel. Optional parameters can follow the address. Each parameter is preceded with a question mark, as in this example:

   ```
   mailto:yourName@yourISP.com?subject=From Lesson 13
   ```
2. `sendToURL()` is an ActionScript method that can be used to send variables to a URL.
3. When the end user confirms a print job, the `start()` method returns `true`. If the user cancels the print job, the `start()` method returns `false`.
4. The constructor of the `Date` class returns the current date and time.

Lesson 14: Creating Adobe AIR Applications with Flash and ActionScript

This lesson gives a very brief introduction to the world of Adobe Air development. If the number of weeks in your class does not allow for covering all of the lessons in the book, then this lesson is the easiest to skip without affecting the other lessons. However, this lesson may be many of your students' first exposure to Adobe AIR and may be valuable for them. If there is not class time for this lesson, encourage your students to go through the chapter on their own.

Using AIR-specific ActionScript

Page 322

When going through this section, it may be beneficial to refer to the ActionScript 3.0 and components reference guide and discuss the various AIR-only classes that are available. You will notice that many methods and properties that were formerly AIR-only are now supported by the latest version of the Flash player.

Specifying Flash publish settings for an AIR project

Page 324

After going through the publishing options for AIR files, you may want to open the resulting XML descriptor file and go through it with your students.

Creating a digital signature

Page 326

If it is relevant to your students, you may want to discuss in more detail the process of purchasing a commercial digital signature. More information can be found at: www.adobe.com/devnet/air/.

Controlling Printing with AIR

Page 332

Emphasize that the direct access to the user's printer that is used here with the `start2` method is not available for .swf files on the web.

Listening for drag events

Page 333

This section introduces some of the very useful ActionScript functionality that is Adobe AIR–specific, including the Drag/Drop capabilities and the `Clipboard`, `File`, and `FileStream` classes. Make sure the students understand the difference between the Native Drag features in AIR and the `startDrag` methods that were used in earlier lessons.

Questions

1. Name three ActionScript classes that are available for Adobe AIR only.
2. Which ActionScript class is used to determine whether a `PrintJob` prints vector or bitmap graphics?
3. What is a method of the `String` class that lets you search through the characters in a string?
4. What is a method of the `FileStream` class that can read data into Flash from an external file?

Answers

1. Many ActionScript classes are available for Adobe AIR only. These could include the `File`, `FileStream`, `NativeDrag`, and `NativeDragEvent` classes.
2. The `PrintJobOptions` class is used to determine whether a `PrintJob` prints vector or bitmap graphics.
3. The `indexOf()` method of the `String` class that lets you search through the characters in a string.
4. The `open()` method of the `FileStream` class can read data into Flash from an external file.

Lesson 15: Using Third-Party Libraries to Extend ActionScript

This lesson can open up a world of possibilities for your students. Some may wish to explore in much more depth the 3D possibilities in Papervision, while others may be inspired to look for other available third-party libraries that match their areas of interest. If you have time you can assist students in these explorations.

Downloading and installing third party ActionScript libraries

Page 344

This is a critical step that all students need to succeed at to proceed with this lesson. Take whatever time is needed to make sure the libraries are running successfully on all students' machines. If time is tight, you may want to preinstall the libraries in the classroom and just discuss the process with your students.

Setting the main class file in Flash CS5

Page 349

This introduces a technique that many developers use as their main work methodology. Make sure students understand the relationship of the main class file to the .fla file and to the flash Timeline. You may want to address the integration of Flash Builder if it is relevant for your students.

Using the Papervision 3D BasicView class

Page 351

The `BasicView` class is valuable for allowing new developers to easily work with a 3D scene. If time allows, you could create additional Papervision projects or assign them to students outside of class. For more advanced students, a more in-depth study of the parts of a Papervision 3D project used by the `BasicView` class would be helpful.

Overriding the onRenderTick() function

Page 359

Depending on the amount of time and the level of the class, you may want to go through the functionality found in the parent `onRenderTick()` function.

Using Caurina Tweener to animate the 3D Cube

Page 361

The brief use of Caurina Tweener in this lesson does not do justice to the usefulness of this class. Any additional time you take with students to make them comfortable with this tool would be well spent.

Creating NetConnection, NetStream and Video instances

Page 367

If time permits, it may be useful to create a simple example file that plays video in a Flash file but does not use Papervision 3D before working with these classes in the lesson file.

Questions

1. How can you indicate in Flash CS5 that an external ActionScript class file should be used as the main class?
2. What is a PaperVision3D class that creates instances of the `Scene3D`, `Camera3D`, `ViewPort3D`, and `BasicRenderEngine` classes?
3. What is the property of a Papervision Cube that stores the list of materials that will be used on the cube instance?
4. What keyword is used to call a function in a parent class from a child class?

Answers

1 In Flash CS5, an external ActionScript class file can be set as the main class by typing the path to the external file in the class field of the Properties Inspector.

2 The PaperVision3D `BasicView` class automatically creates instances of the `Scene3D`, `Camera3D`, `ViewPort3D`, and `BasicRenderEngine` classes when an instance is created.

3 The `MaterialsList` property of a Papervision cube contains a reference to an instance of the `MaterialsList` class, which is an array containing a list of the materials used on the cube.

4 The keyword `super` is used to call a function in a parent class from a child class.