

Flex 4.5

INTRODUCING ADOBE FLEX 4.5

Introducing the Adobe Flash Platform

Understanding Flex development technologies

Understanding Flex application deployment

Exploring the Flash Builder interface

Creating a Flex project

UNDERSTANDING FLEX 4.5 FUNDAMENTALS

Understanding Adobe Flex namespaces

Understanding Flex components

Compiling and running an application

Creating custom components

Introducing skinning

Accessing and using structured data within a custom component

Introducing data binding

Updating visual controls using dynamic data

INTRODUCING OBJECT ORIENTED PROGRAMMING

Introducing OOP terminology

Understanding MXML classes

Creating a class property

Writing a class definition in ActionScript

Creating instances of an ActionScript class

Creating class methods

UNDERSTANDING COMPONENTS AND LAYOUTS

Introducing visual components

- Introducing Spark vs MX components
- Understanding components

Introducing Spark layout classes

- Defining the layout property
- Understanding the BasicLayout class
- Understanding the HorizontalLayout class
- Understanding the VerticalLayout class
- Understanding the TileLayout class
- Custom layout classes

Introducing the Spark container classes

- Understanding how the containers work
- Understanding the Spark containers
- Reviewing the Application container
- Introducing the Group container
- Introducing the Panel container
- Introducing the BorderContainer
- Sample application layout
- Defining layout properties

Using constraints to control component layout

- Understanding constraint-based layout basics
- Rules to control component layout
- Constraint-based vs. traditional container layout
- Visually assigning anchors

Adding scrollbars to containers

- Implementing the Scroller class

HANDLING EVENTS

Understanding events

- Handling framework events
- Handling user events

Implementing event handlers

- Using inline ActionScript in MXML

Creating ActionScript event handler functions

- Defining a function in an MXML document
- Creating functions in external files

Understanding the event object

- Introducing the event object
- Exploring event object properties
- Datatyping the event object

Understanding event bubbling

- Distinguishing currentTarget and target
- Introducing event bubbling

Adding event listeners with ActionScript

USING THE FORM CONTAINER

Using the Form container

- Introducing the Form components
- Creating a Form custom component

Customizing the FormItem container display

- Using sequence labels
- Using help content
- Using prompts with text controls
- Using a custom required icon

Displaying the Spark Form in a stacked layout

- Creating a Form with the stacked layout using CSS

VALIDATING AND FORMATTING DATA

Using formatters

- Introducing MX formatters
- Introducing Spark formatters
- Defining formatters
- Using formatters

Validating form data

- Introducing MX validators
- Introducing Spark validators
- Creating a validator with MXML
- Creating a validator with ActionScript
- Handling required values

Triggering validation with events

- When validation fails
- Using a custom error icon
- Using the Spark NumberValidator class

Triggering validation with ActionScript

- Suppressing automatic validation
- Using the validate() method
- Triggering multiple validators

CONTROLLING NAVIGATION

Understanding MX navigator containers

- Controlling container display
- Introducing the MX navigator containers

Using the ViewStack container

- Understanding ViewStack basics
- Navigating with the MX LinkBar control
- Navigating with the Spark TabBar and ButtonBar controls
- Displaying and resizing the ViewStack child containers
- Deferring instantiation of ViewStack child containers

Using a Spark container inside MX navigator containers

Creating custom navigation for the ViewStack container

- Setting the active ViewStack child container
- Enabling buttons dynamically

Using the TabNavigator container

Using the Accordion container

CONTROLLING APPLICATION STATE

Understanding view states

- Introducing view states
- Creating view states in Design mode
- Creating view states with code

Controlling view states

- Introducing the currentState property
- Switching between states

ANIMATING COMPONENTS AND STATES

Applying effects to components

- Interacting with users using animated effects
- Understanding effect basics
- Reviewing available effects
- Defining an effect
- Playing an effect

Creating composite effects

Applying transitions to application state changes

- Creating transitions

Adding and removing components during animations

CONTROLLING VISUAL DISPLAY WITH STYLING

Creating global application styles

- Generating the styles using the Appearance view in Design mode
- Flash Builder automatically creates CSS file
- Global vs Application selector

Defining styles and skins for components

- Creating Flex component selectors
- Create custom component selectors
- Applying a component skin via CSS

Introducing Advanced CSS selectors

- Descendent selector
- id selector

- Class selector
- Pseudo selector

SKINNING SPARK COMPONENTS

Introducing skinning

- Understanding Spark skins
- Setting styles inline in the MXML tags
- Introducing the skinnable Spark container classes
- Creating a skin for Spark components

Incorporating visual elements drawn in other programs

- Introducing FXG

Changing the default display of skin parts

- Reviewing the contentGroup skin part for containers
- Understanding a Button control's skin part
- Understanding the Panel container's skin parts
- Finding skin parts for any component
- Choosing a component to display the skin part

IMPLEMENTING ADVANCED SKINNING TECHNIQUES

Implementing different visual states for a skin

- Understanding skin states in Button component
- Assigning properties for each state

Adding transitions in skin states

- Animating the button states

Accessing custom component properties from a skin

- Defining the custom property in the custom component
- Creating the skin and component contract
- Referencing the properties in the skin

Adding scrollbars to skins

- Using the Scroller with nonskinnable containers
- Using the Scroller with skinnable containers

ACCESSING REMOTE DATA

Using HTTPService to load remote data

- Using the HTTPService class
- Process for making HTTP requests
- Retrieving data with lastResult
- Understanding the ArrayCollection results
- Using results in data bindings

Handling returned data and faults

- Handling results
- Handling faults
- Displaying messages with an Alert pop-up
- Using result and fault handlers example

Making HTTP requests with parameters

- Using explicit parameter passing
- Using parameter binding

Using the Flash Builder wizards

- Understanding the CallResponder class
- Creating a master/detail form

CREATING A TYPED DATA MODEL

Understanding the need for a typed data model

- Working with a model of generic data
- Creating a model of typed data
- Understanding value objects
- Creating value objects

Creating an ActionScript class for typed data

- Reviewing classpaths and packages
- Creating class files
- Defining class access modifiers
- Creating class constructors
- Defining class properties
- Instantiating an ActionScript class

Populating an ArrayCollection with value objects

- Comparing ArrayCollection with ArrayList

Understanding data bindings with value objects

- Making ActionScript class properties bindable
- Binding the view to the model
- Binding two-way

Refactoring value objects

- Downcasting object references

EXTENDING EVENTS

Understanding the problem with bindings

- The problem of using bindings
- Creating loosely coupled components

Defining the event type

- Defining the event in the dispatching component
- Triggering the event from a user or system event
- Instantiating the event
- Dispatching the event to the main application

Handling the event in the main application

Extending the Event class

- Creating the extended class
- Extending the Event class
- Adding properties
- Creating the constructor

Using the extended event class

- Using the Metadata compiler directive
- Dispatch the extended event
- Handling the event in the main application

Dispatching a value object with the extended event

- Accommodating a value object in the extended event

Overriding the clone method

- Passing a parameter to toggle event bubbling

RENDERING CONTENT WITH THE DATAGROUP CONTAINER

Displaying string data in an item renderer

- Understanding DataGroup container basics
- Implementing the DataGroup container
- Using the default item renderer

Passing UI components in the data provider

- Rendering visual controls
- Mixing data item types

Creating a custom item renderer

- Understanding the limits of the default item renderer
- Creating a custom item renderer
- Creating a nested item renderer
- Referencing an external item renderer class
- Using data in an item renderer

Specifying how to visually render items

- Determining a renderer at runtime
- Using the Scroller with a DataGroup

Using the SkinnableDataContainer

- Implementing the SkinnableDataContainer
- Creating the skin component

DISPLAYING DATA USING THE DATAGRID

Using the DataGrid control

- Supplying data to a DataGrid control

Specifying DataGrid control columns

Formatting DataGrid control columns

- Setting column widths with a typical item
- Formatting data in a label function
- Creating reusable label functions

Using item renderers and item editors

- Understanding Item renderers versus item editors
- Introducing the types of item renderers and item editors

Creating custom item renderer and item editor components

Using events and selected items with a DataGrid component

Customizing the DataGrid skin

DEPLOYING FLEX AND AIR APPLICATIONS

Compiling the Flex application

- Compiling automatically
- Modifying compiler settings
- Embedding the SWF in the HTML wrapper
- Understanding the runtime shared libraries

Creating a Flex application production build

- Creating a release build

Creating a desktop application

- Creating an AIR project
- Understanding some differences between Flex and AIR applications
- Exporting a release build