



Capstone Courseware, LLC

33 Boylston Street
Jamaica Plain, MA 02130

877-227-2477
capstonecourseware.com

172. Java Foundation Classes

Version 5.0

This 5-day course introduces the Java programmer to the Java Foundation Classes -- a.k.a. Swing -- the Java environment's comprehensive framework for GUI development. The student will study the fundamentals of the JFC architecture and quickly move to building simple JFC frame-based applications. By the end of the course the student will be comfortable building simple or complex interfaces with the most common Swing controls and classes -- including buttons, lists, combo boxes, checkboxes and radio buttons, text controls, trees, and tables -- controlling multiple windows and dialogs, using panes to manage related interfaces, implementing popup menus, and using data transfer packages for clipboard and drag-and-drop implementations.

The first module of the course provides an introduction to the JFC architecture and standard practices. AWT concepts such as the event model and basic layout management are reviewed as necessary. The standard controls are covered, including labels, text components, buttons, listboxes, and comboboxes. Architectural patterns are emphasized, especially JFC's strict use of the Model-View-Controller paradigm. Understanding the thorough use of this pattern in JFC is critical to using the framework effectively. Event handling is treated, both handling AWT-style events, such as action events from button clicks, and handling events fired by the model that lies under a particular control.

In the second module, more sophisticated and powerful controls are studied: the tree and table controls. Each is presented in a separate chapter, allowing students to dig into the underlying architecture and to develop a firm grasp of the many powerful features lying behind the direct use of the control classes themselves. Especially, customization of the controls using renderers and editors is considered. The final chapter of this module discusses the effective use of the separate model class and object, which pattern is built into all Swing controls, to implement trees and tables that present very large data sets. This allows the student to confront problems common to enterprise-class GUI building, and to find sound solutions using techniques such as lazy evaluation and LRU eviction.

In the third and final module, advanced GUI-management features are studied. Students learn to implement and/or customize scrolling, and to use splitter panes to combine related user interfaces. Popup elements such as dialog boxes, message boxes, and menus are also considered. The module concludes with a treatment of JFC's data transfer model, which empowers clipboard copy, cut, and paste features



as well as drag-and-drop. A simple application is developed over the course of this module that implements all the above features.

The course software also includes an optional overlay of workspace and project files to support use of the Eclipse IDE in the classroom. (This requires that the instructor be experienced in use of Eclipse and able to walk students through basic tasks in the IDE.)

Prerequisites

- Solid experience with Java programming is required. Course 103, "Java Programming," is excellent preparation.





Learning Objectives

- Understand the basics of the JFC architecture.
- Build simple GUI applications using JFC.
- Build more complex GUIs using various JFC controls.
- Use the many hooks into the JFC architecture to easily customize rendering and editing within JFC controls.
- Understand the significance of the MVC decomposition in using JFC controls.
- Build GUI classes that make effective use of events as fired from model, view and controller elements of the GUI itself.
- Implement JFC GUIs based on existing data structures, and use model implementations to adapt the JFC controls seamlessly to this data.
- Handle very large data sets, such as remote databases, without degradation of performance or user responsiveness.
- Implement scrolling, and customize scrolling for a particular scrollable element.
- Manage complex user interfaces by combining GUI areas with splitter panes and tab panes.
- Expand an application interface with popup dialogs, message boxes, and popup menus.
- Use standard dialogs such as file choosers and color choosers.
- Implement clipboard cut, copy and paste using the JFC data transfer model.
- Implement drag sources and drop targets for complete drag-and-drop capabilities.

Timeline: 5 days.

IDE Support: Eclipse 3.2

In addition to the primary lab files, an optional overlay is available that adds support for Eclipse 3.2. Students can code and build all exercises from within the IDE. Most exercises can be tested from within the IDE as well, though some must be tested from the command line. See also our orientation to Using Capstone's Eclipse





172. Java Foundation Classes

Outline

Overlays, and please be advised that this is an optional feature; it is not a separate version of the course, and the course itself does not contain explicit Eclipse-specific lab instructions.





Module 1. Introduction to JFC

Chapter 1. Introduction to JFC

- Abstract Windowing Toolkit Basics
- Simple Layout Management
- Simple Event Handling
- Lightweight Controls
- JFC Feature Set
- JFC Architecture and Relationship to AWT

Chapter 2. JFC Application Design

- Role of a JFrame
- Building a Frame-Based JFC Application
- Panes
- Using Dialogs

Chapter 3. JFC Components

- JFC Component Class Hierarchy
- JComponent Features
- Simple Control Types
- Text Components
- Menus
- Managing Look and Feel

Chapter 4. Architectural Patterns

- Observer Pattern
- Model-View-Controller Decomposition
- Strategy Pattern
- JList
- Factory Pattern
- JComboBox





Module 2. Trees and Tables

Chapter 1. Hierarchical Data and JTree

- Presenting Hierarchies
- JTree and Supporting Classes
- Using the Default Tree Model
- Customizing Look and Feel
- Implementing a Tree Model
- Custom Rendering
- Custom Editing

Chapter 2. Tabular Data and JTable

- Presenting Tabular Data
- JTable and Supporting Classes
- Implementing a Tree Model
- Customizing Look and Feel
- Custom Rendering
- Custom Editing

Chapter 3. Managing the Model

- Adapting Existing Data Structures
- Very Large Data Sets and GUIs
- Caching
- Lazy Evaluation Using Tree and Table Models
- Limiting the Cache with an Evictor
- Anticipating User Requests





Module 3. Advanced GUI Design

Chapter 1. Organizing Application Windows

- Viewport Abstraction
- JScrollPane
- Scrollable Elements
- Customizing Scrolling
- Tabbed Panes
- Splitter Panes

Chapter 2. Popup GUI Elements

- Dialog Boxes
- Message Boxes
- Using File Choosers
- Customizing File Choosers
- Using Color Choosers
- Custom Dialogs
- Tooltips
- Popup Menus

Chapter 3. Data Transfer

- The Data Transfer Model
- Transferable Objects
- Data Flavors and MIME Types
- The Clipboard API
- The Drag-and-Drop API

Appendix A. Learning Resources

System Requirements

Hardware Requirements (Minimum)

500 MHz, 256 meg RAM, 500 meg disk space.

Hardware Requirements (Recommended)

1.5 GHz, 512 meg RAM, 1 gig disk space.

Operating System

Tested on Windows 2000 Professional. Course software should be viable on all systems which support a J2SE 1.4 SDK.

Network and Security

Limited privileges required -- please see our standard security requirements at <http://capcourse.com/Guides/Security.html>.





172. Java Foundation Classes

Outline

Software Requirements

All free downloadable tools.

