

Google Web Toolkit (GWT) allows Java developers to create Web 2.0-ready Ajax front ends. GWT compiles these Java artifacts into optimized JavaScript for all major browsers. This intensive, hands-on course explores advanced GWT features from user interface widgets to client-server integration. First, students will learn how to create and use GWT modules. From there, they will explore many advanced user interface widgets and panels. Then they will learn how to apply styles to GWT widgets with CSS. Next, students will learn to create their own composite widgets. JUnit testing is the focus of the next chapter, followed by a chapter on the GWT Java Emulator. Students will then learn about various connectivity and integration features of GWT as well as how to invoke native JSNI methods. Finally, the course ends with a chapter on GWT best practices.

Course Objectives:

- Create and use modules in GWT.
- Design user interfaces with more advanced GWT widgets including the ToggleButton, Tree, SuggestBox, and RichTextArea.
- Style GWT widgets with CSS.
- Create your own composite widgets.
- Develop and execute GWT unit tests with JUnit
- Discover the GWT's Java Emulator, its capabilities and limitations.
- Learn to use GWT's DOM management.
- Describe advanced connectivity mechanisms, polling and related concepts.
- Use the various integration options to work with servers, including XML, JSON and GWT-RPC.
- Declare and invoke native JSNI methods.
- Describe some best practices on performance and avoiding pit falls.

Audience: Java developers who wish to quickly create Ajax application with the Google Web Toolkit.

Prerequisites: GWT and Java programming experience is required. Basic knowledge of HTML, CSS, and JavaScript is recommended. Previous experience doing widget-based Form development using events as well as HTTP exposure is beneficial.

Number of Days: 2 days

1	Using Modules in GWT GWT Modules Using External Modules Modularizing an Application Deferred Binding Multiple-Module Applications Code Splitting	HTMLPanel FocusPanel StackPanel TabPanel The DeckPanel The DisclosurePanel Using Menus via a MenuBar Using a Tree Widget Using SuggestBox GWT RichTextArea Editing FormPanel
2	More on GWT User Interfaces ToggleButton and PushButton DecoratorPanel AbsolutePanel	

3	<p>Using DeferredCommand</p> <p>GWT UiBinder and SafeHtml</p> <p>What is UiBinder</p> <p>UiBinder Types</p> <p>UiBinder Benefits</p> <p>Binder XML Files and Java Files</p> <p>Creating Widget-Based UiBinder Templates</p> <p>Creating a UiBinder in Eclipse</p> <p>Passing Values to Binders via Tag Attributes</p> <p>Handling Events in HTML-Based UiBinder Templates</p> <p>Providing Instances for UiFields</p> <p>Using SafeHtml</p>	7	<p>Model-View-Presenter and Dependency Injection</p> <p>Introduction to Model-View-Presenter</p> <p>The Event Bus</p> <p>Managing Asynchronous Data in MVP Applications</p> <p>Unit Testing MVP with Mock UI Elements</p> <p>Dependency Injection</p> <p>Unit Testing with Dependency Injection</p>
4	<p>Styling with CSS in GWT</p> <p>Styling with CSS</p> <p>GWT Visual Themes</p> <p>Associating CSS Files with a Module</p> <p>Styling Widgets in GWT</p> <p>Complex GWT Styles</p> <p>Dependent Styles in Widgets</p> <p>More ClientBundle Elements: CSS Resources, Data Resources, and External Resources</p>	8	<p>More Server Integration Options in GWT</p> <p>Using HTTP Requests</p> <p>Posting Data Parameters</p> <p>Manipulating in XML DOM</p> <p>XML Support in GWT</p> <p>Generating XML Documents in GWT</p> <p>Integrating Using JSON</p> <p>GWT Support for JSON</p> <p>Generating JSON on the Server</p>
5	<p>Creating Composite GWT Widgets</p> <p>Custom GWT Widgets</p> <p>Composite GWT Widgets</p> <p>Developing Composite Widgets</p> <p>Manipulating Contained Widgets</p> <p>The initWidget() Method</p> <p>Event Handlers</p> <p>Styling Composites</p>	9	<p>GWT JavaScript Native Interface</p> <p>Defining JSNI Methods</p> <p>Sharing Data Between JavaScript and Java</p> <p>Talking to the Browser via JSNI</p> <p>Using External JavaScript</p> <p>Using a JavaScriptObject in GWT</p> <p>Using JSNI with JSON</p> <p>Java Callbacks from External JavaScript</p> <p>Using External Libraries</p>
6	<p>Testing and Debugging GWT Modules</p> <p>Unit Testing Concepts</p> <p>The JUnit Test Framework</p> <p>Writing a JUnit Test Case</p> <p>GWT Unit Testing with GWTTestCase</p> <p>Creating and Running Tests</p> <p>Testing Asynchronous Code</p> <p>GWT System Testing</p> <p>GWT User Interface Testing</p>	10	<p>Advanced GWT Connectivity</p> <p>Stateful and Stateless Servers</p> <p>The Model-View Presenter Approach</p> <p>Using Stateful Servers</p> <p>The Request Factory</p> <p>Using Editors</p>
		11	<p>Client-Side RPC Architecture in GWT</p> <p>Client-Side Polling</p> <p>Server Push</p> <p>Implementing Server Push</p> <p>Server Push</p> <p>Polling Protocols</p>

- 12 **GWT Best Practices**
 - General Coding Practices
 - Avoid Native JavaScript Code
 - Security Issues
 - Handling Long Processing
 - The GWT IncrementalCommand Class
 - Internationalization in GWT
 - GWT Pitfalls and Issues
 - Large Projects
 - Merging with HTML
 - GWT Styling
 - Testing Practices for GWT
 - Unit Testing Principles
 - Bandwidth-Sensitive Applications
 - Reduced Theme Downloads
 - Minimize Server State Requirements
 - Dependency Injection