



## Developing Applications for IBM WebSphere Portal 8.0 using IBM RAD 8.5 Information

<b>Length:</b>	5.0 Days
<b>Ref:</b>	WPL53G-X
<b>Delivery method:</b>	ClassroomInstructor Led OnlineSelf-paced Virtual Training
<b>Price:</b>	AUD

### Overview

This course is designed for users who are new to developing applications for IBM® WebSphere® Portal Server V8.0. It describes the fundamentals of designing and creating portlets and how to put the new features to use including managed pages. Specialized topics include migrating a version 6.1 theme to the new version 8.0 theme architecture by using client-side aggregation V2 (CSA2), developing custom themes by using CSA2, and customizing the user interface for tagging and rating. Developing multi-channel applications using RAD 8.5 is also covered in this course.

In addition, you have the opportunity to apply best coding practices throughout the course, developing robust portlet applications and portal themes by using Dojo, asynchronous JavaScript and XML (Ajax), and client-side aggregation. This course makes extensive use of Rational Application Developer V8.5 to develop and test JSR (Java Portlet Specification) 286-compliant portlets.

If you are enrolling in a Self Paced Virtual Classroom or Web Based Training course, before you enroll, please review the **Self-Paced Virtual Classes and Web-Based Training Classes** on our Terms and Conditions page, as well as the system requirements, to ensure that your system meets the minimum requirements for this course.

<http://www.ibm.com/training/terms>

### Public

The target audience for this intermediate course is experienced application developers who are familiar with Java, Web 2.0, and Web development and who are familiar with security concepts.

### Prerequisites

You should:

- Possess a solid understanding of J2EE application development, including the ability to read **and** understand servlets, **and** Java Server Pages (JSPs)
- Have a solid understanding of the World Wide Web functionality **and** be familiar with Asynchronous

JavaScript and XML (AJAX) **and** Dojo

- Have a basic understanding of Rational Application Developer **or** Eclipse development environment
- Have strong Windows user skills

## Objective

You should be able to:

- Compare Rational Application Developer and IBM Web Experience Factory
- Locate developer resources
- Review Portlet concepts such as the portlet lifecycle, WAR architecture, core, objects, and methods.
- Set up a development environment
- Perform basics of creating a project, deploying a portlet, starting and stopping a server and adding a portlet to a page
- Work with Core API (JSR 286)
- Leverage tag libraries
- Process portlet input
- Use Client side aggregation (CSA) framework
- Compare CSA and server-side aggregation (SSA)
- Customize portlet behavior
- Build interactive (collaborative) portlets
- Understand the basics of JavaServer Faces (JSF)
- Use JSF in the Portal environment
- Develop an iWidget
- Use Dojo and Ajax for portal development

## Topics

### Day 1

- Unit 1: Overview of IBM WebSphere Portal 8.0
- Unit 2: Working with portlet concepts
- Unit 3: Performing basic tasks
- Exercise 3: Creating, deploying, and testing a portlet
- Unit 4: Working with the core API
- Exercise 4: Working with core portlet objects

### Day 2

- Unit 5: Using tag libraries
- Exercise 5: Using tag libraries in portlet applications
- Unit 6: Processing portlet input

- Exercise 6: Processing client input requests
- Unit 7: Customizing portlet behavior
- Exercise 7: Customizing portlet behavior
- Unit 8: Working with client-side aggregation

### Day 3

- Unit 9: Interactive portlets
- Exercise 9: Developing and testing interactive portlets
- Unit 10: Introduction to JavaServer Faces (JSF)
- Exercise 10: Creating a JSF portlet application
- Unit 11: iWidget development
- Exercise 11: Developing an iWidget
- Unit 12: Serving resources
- Exercise 12: Creating a resource serving portlet

### Day 4

- Unit 13: Using Dojo and Ajax in portal development
- Exercise 13: Using Dojo in a portlet application
- Unit 14: Overview of portlet services
- Exercise 14: Creating a custom portlet service
- Unit 15: Working with portlet services
- Exercise 15: Using a standard portlet service

### Day 5

- Unit 16: Building portlet filters
- Exercise 16: Developing and testing a portlet filter
- Unit 17: Implementing personalization
- Exercise 17: Developing and testing components by using personalization
- Unit 18: Creating a portal brand
- Exercise 18: Creating a portal brand
- Unit 19: Creating multichannel Portal applications
- Exercise 19: Creating a multichannel Portal application