



## Server-Side Java Programming with Servlets and JSPs

---

<b>Course #</b>	<b>JV-391</b>
<b>Type</b>	<b>Instructor-Led</b>
<b>Category</b>	<b>Technical</b>
<b>Length</b>	<b>5 Days</b>
<b>Summary</b>	<p>This course covers the Java technologies used to develop Java Web applications. The key topics discussed are Java Servlets, JavaServer Pages (JSP) and XML. Students will learn: the details of developing and deploying servlets and JSP pages, how to design Java Web applications using the Model-View-Controller (MVC) design, how JavaBeans fit into the MVC design, using JDBC and JavaBeans to create Data Access Objects. Students will learn how to deploy and administer Java Web applications on Apache Tomcat using the MySQL database.</p>
<b>Highlights</b>	<p><b>Servlets</b></p> <ul style="list-style-type: none"><li>• <b>Overview of Servlets: Programs that run in a Web server.</b></li><li>• <b>Server-side Development: Options for creating Web applications.</b></li><li>• <b>Advantages of Servlets: Powerful, portable and secure.</b></li><li>• <b>The Servlet API: The javax.servlet and javax.servlet.http packages.</b></li><li>• <b>HTTP: The Hypertext Transfer Protocol.</b></li><li>• <b>Requests: The HTTP Request header.</b></li><li>• <b>Responses: The HTTP Response header.</b></li><li>• <b>HTML Basics: A quick introduction to HTML and its syntax.</b></li><li>• <b>HTTP Servlets: The HttpServlet class.</b></li><li>• <b>The "Hello, World" Servlet: A simple HTTP servlet handling a GET request.</b></li><li>• <b>The Lifecycle of a Servlet: The servlet container.</b></li><li>• <b>A Servlet Counter: A demonstration of the lifecycle of a servlet.</b></li><li>• <b>HttpServletRequest: Encapsulating the HTTP request header of a client request.</b></li><li>• <b>HttpServletResponse: Encapsulating the HTTP response to a client.</b></li><li>• <b>Parameters: Obtaining data from the client request.</b></li><li>• <b>The Request Dispatcher: Including servlets and forwarding requests to other servlets.</b></li><li>• <b>Scope: Understanding the various scopes an object can have within a</b></li></ul>

servlet container.

- **Request Scope:** Objects associated with a client request.
- **Application Scope:** A container-wide object.
- **Session Scope:** Creating client sessions.
- **Cookies:** A sweet session tracker.
- **The HttpSession Interface:** Container session objects.
- **URL Rewriting:** Encoding a URL.
- **Servlet Threading Models:** The SingleThreadModel.
- **Connecting to a Database:** JDBC within a servlet.
- **Server-side Includes:** A servlet embedded in an HTML page.

## JavaBeans

- **Overview of JavaBeans:** Java software components.
- **The Bean Development Kit:** The bean specification and Sun's beanbox.
- **Properties:** Adding properties to a bean.
- **Methods:** A bean's behavior.
- **Events:** How beans communicate with each other.

## JDBC

- **Overview of JDBC:** An API for communicating with databases.
- **JDBC Drivers:** Understanding the various types of drivers available for JDBC.
- **Connecting to a Database:** Loading the appropriate driver and making the connection.
- **Statements:** Executing SQL statements.
- **Prepared Statements:** Executing precompiled SQL statements.
- **Results:** Working with result sets.

## JavaServer Pages

- **Server-side Development:** Comparing JSP to other technologies.
- **Overview of JavaServer Pages:** Java technology for creating dynamic Web content.
- **JSP Lifecycle:** Translation of JSP into servlets.
- **An Introduction to Servlets:** Understanding how servlets run on a Web server.
- **"Hello, JSPs":** A simple JavaServer Page.
- **JavaServer Page Tags:** The various JSP tags.
- **Templating:** The include directive.
- **Declarations:** Declaring variables and methods.
- **Expressions:** Java statements within a JSP.
- **The request Object:** The request header information.
- **Parameters:** Request data passed in to the JSP.
- **Scriptlets:** Embedding Java code within a JSP.
- **Directives:** Specifying options for the page.
- **Implicit Objects:** The implicit objects of a JSP.
- **JSP Actions:** The action tags of JSP.
- **Overview of JavaBeans:** Java software components.

- **JavaBeans in JSPs:** The useBean directive.
- **Error Pages:** Handling exceptions in JSP.
- **Scope:** The four scopes of JSP objects.
- **Sessions:** Creating sessions within JSP.
- **Request Scope:** Associating data with the request.
- **Application Scope:** The implicit application object.
- **Session Scope:** Creating client sessions.
- **The Implicit Session Object:** Session tracking made easy.
- **Beans with Session Scope**

### Custom Tags and JSTL

- **Creating your own JSP tags**
- **Writing Tag Handler classes**
- **The Tag Library Definition File**
- **The JSP Standard Template Library**

### Who should attend

Beginning Java programmers new to JDBC and JavaBeans who need to learn how to develop Web applications using the various Java technologies like Servlets, JavaServer Pages, and XML.

### Prerequisites

Students have completed JV 200, be familiar with the Java language and have experience with HTML. Familiarity with SQL is also helpful.

### At Course Completion

- Understand the various components in a Java Web application.
- Learn how Servlets and JavaServer Pages fit into the J2EE environment.
- Understand the Model-View-Controller design of a Java Web application.
- Learn how to install and setup the Tomcat Web server and MySQL database server.
- Understand the lifecycle of a Java servlet and JavaServer Page.
- Learn how to write and deploy a Java servlet and JavaServer Page.
- Understand the directory structure of Java Web applications.
- Understand the various scopes an object can have in a servlet container.
- Learn how to include and forward a request to another Web component.
- Learn how to create client session in JSP and servlets.
- Understand how JavaBeans are used to create Data Access Objects.
- Learn how to access data in a database using JDBC.
- Learn how to write effective and useful servlets and JSP pages.
- Understand the various elements of a JavaServer Page.
- Learn how to create a custom JSP tag library.
- Learn how to use the JSP Standard Tag Library.
- Understand XML, including its uses, syntax and advantages.
- Learn how to develop Java applications that create and use XML documents.