Java Basics Primer for Non-Developers (QA, Testers, Selenium Users) (TT2190) Training

Duration: **2 Day**

BILGINC IT Academy
info@bilginc.com
bilginc.com

Overview

Learn “Just Enough Java” syntax to quickly use Java in Selenium automation, testing script enhancement and more

Java is the primary option for implementing automation using Selenium. Selenium users need to have basic syntax experience, but not in-depth Java application development or extensive coding skills by any means. This Java Basics Primer for Non-Developers / Selenium Users is a hands-on, quick-start style introduction to basic Java syntax geared for non-developers, Selenium testers and users who need to learn Just Enough Java to quickly get up and running with the skills needed to work with Selenium. This course provides you with the basic skills needed to customize and enhance Selenium’s testing scripts.

Prerequisites

This is an introductory- level course, designed for students who need to learn just enough Java to successfully use and support Selenium / WebDriver. Attendees should have a basic working knowledge of developing or testing software applications. Students include testers, QA and BA or similar non-coding roles.

Who Should Attend

Web Testers, QA, Selenium Users.

What You Will Learn

Working within in an engaging, hands-on learning environment, guided by our expert team, attendees will learn to:

- Understand not only the fundamentals of the Java language in terms of supporting Selenium functionality
- Understand the basics of the Java language
- Work with the basic Java syntax for flow control (looping and conditionals) as well as string handling and basic data structures
- Learn to use Java exception handling features
- Understand and use classes, inheritance and polymorphism

Outline

Session: Java: A First Look

Java Basics Primer for Non-Developers (QA, Testers, Selenium Users) (TT2190) Training

Duration: **2 Day**

BILGINC IT Academy
info@bilginc.com
bilginc.com

Lesson: The Java Platform

- Java Platforms
- Lifecycle of a Java Program
- Responsibilities of JVM
- Documentation and Code Reuse

Lesson: Using the JDK

- Setting Up Environment
- Locating Class Files
- Compiling Package Classes
- Source and Class Files
- Java Applications
- Lab: Exploring MemoryViewer
- Lab: Exploring ColorPicker

Lesson: The Eclipse Paradigm

- Workbench and Workspace
- Views
- Editors
- Perspectives
- Projects
- Tutorial: Working with Eclipse Oxygen

Session: Getting Started with Java

Lesson: Writing a Simple Class

- Classes in Java
- Class Modifiers and Types
- Class Instance Variables
- Primitives vs. Object References
- Creating Objects
- Lab: Create a Simple Class

Lesson: Adding Methods to the Class

- Passing Parameters Into Methods
- Returning a Value from a Method
- Overloaded Methods
- Constructors
- Optimizing Constructor Usage
- Lab: Create a Class with Methods

Lesson: Language Statements

- Operators
- Comparison and Logical Operators
- Looping
- Continue and Break Statements
- The switch Statement
- The for-each() Loop

BILGINC IT Academy

info@bilginc.com

bilginc.com

- Lab: Looping

Lesson: Using Strings

- Strings
- String Methods
- String Equality
- StringBuffer
- StringBuilder
- Lab: Fun with Strings
- Lab: Using StringBuffers and StringBuilders

Lesson: Specializing in a Subclass

- Extending a Class
- Casting
- The Object Class
- Default Constructor
- Implicit Constructor Chaining
- Lab: Creating Subclasses

Session: Essential Java Programming

Lesson: Fields and Variables

- Instance vs. Local Variables: Usage Differences
- Data Types
- Default Values
- Block Scoping Rules
- Final and Static Fields
- Static Methods
- Lab: Field Test

Lesson: Using Arrays

- Arrays
- Accessing the Array
- Multidimensional Arrays
- Copying Arrays
- Variable Arguments
- Lab: Creating an Array

Lesson: Java Packages and Visibility

- Class Location of Packages
- The Package Keyword
- Importing Classes
- Executing Programs
- Java Naming Conventions

Session: Next Level Java Programming

Lesson: Inheritance and Polymorphism

BILGINC IT Academy

info@bilginc.com

bilginc.com

- Polymorphism: The Subclasses
- Upcasting vs. Downcasting
- Calling Superclass Methods from Subclass
- The final Keyword
- Lab: Salaries - Polymorphism

Lesson: Interfaces and Abstract Classes

- Separating Capability from Implementation
- Abstract Classes
- Implementing an Interface
- Abstract Classes vs. Interfaces
- Lab: Mailable - Interfaces

Lesson: Exceptions

- Exception Architecture
- Handling Multiple Exceptions
- Automatic Closure of Resources
- Creating Your Own Exceptions
- Throwing Exceptions
- Checked vs. Unchecked Exceptions
- Lab: Exceptions

Session: Collections and Generics

Lesson: Introduction to Generics

- Generics and Subtyping
- Bounded Wildcards
- Generic Methods
- Legacy Calls to Generics
- When Generics Should Be Used
- Exercise: ShoppingCart

Lesson: Collections

- Characterizing Collections
- Collection Interface Hierarchy
- Iterators
- The Set Interface
- The List Interface
- Queue Interface
- Map Interfaces
- Using the Right Collection
- Collections and Multithreading
- Lab: Collections Poker