

Java for Android Training

Duration: 3 Days

Course Price: \$2,095

Course Description

Course Overview

Java for Android is a hands-on course designed to provide essential Java skills and experience in preparation for developing applications on Android mobile platform. This course is best suited for non-Java developers who are interested in getting into Android application development. Throughout the course, the students will perform hands-on exercises using Eclipse and will install Android SDK.

Course Objectives

Upon completion of the Java for Android course, you will be able to:

- Understand Java as a programming language, what it is composed of, and how it relates to other programming languages
- Install and configure the Eclipse Java development environment
- Create, run, and debug Java programs
- Understand object oriented programming (OOP) paradigm, explain the difference between OOP and structural programming, and understand the advantages of OOP
- Effectively use declarations, assignments, operators, flow-control structures, and program scope
- Understand the difference between Primitives, Strings, Classes, Objects, and Enumerations
- Control program flow with conditions and loops
- Practice OOP by reusing existing code through Inheritance, Polymorphism, and Composition
- Map a real-world system to Java classes and interfaces
- Organize Java code into classes, packages and archives (JARs)
- Read, write, and use Java code documentation (Javadoc)
- Understand Java error handling via try/catch blocks and exceptions
- Utilize built-in Java libraries in your applications
- Parse and process files using java.io and regular expressions
- Utilize design patterns and best practices developing Java applications

Audience

This course is for anyone who wants to get into Android application development but lacks the experience with Java programming language. General understanding of Object Oriented Paradigms (OOP) and previous experience with another programming languages is highly recommended.

Course Outline

- Java's Characteristics
- Java Virtual Machine
- Interpreted
- Architecture-Neutral and Portable
- Multithreaded
- Dynamic
- Object-Oriented
- Java Technology – Overview
- Java SE, EE and ME
- Java Virtual Machine (JVM)
- Software Based Technology
- Java Run-Time Environment (JRE)
- Garbage Collector
- Object-Oriented Paradigm
- Abstraction
- Encapsulation of Attributes and Behaviors
- A Software Object – (Using UML)
- An Application – Object Collaboration
- Exercise – Create Two UML Object Diagrams
- Messaging
- Method Definition and Calls
- Parameters / Arguments
- The return keyword
- Java Application Development
- Java Development Tools
- Create JAR files
- Write, Compile and Run a Java Application
- Java Language Basics
- Java syntax and key-words
- Identifiers and Naming Conventions
- Instantiation
- An Object's Blueprint
- Classes and Objects – Terminology
- Class Declaration
- An Instance
- Variables and Data Types
- Variable Declaration and Assignment
- Primitive and Reference/Object Data Types
- Type Casting
- Variable Scope

Operators and Expressions
Functionality
Precedence
Arrays
Array Overview
Declaration, Assignment & Instantiation
Cell Value Assignment and Retrieval
Exercise – Single Array
Control Flow Statements
Looping
Decision making and branching
String and StringBuffer
String
StringBuilder
Exercise – String & StringBuilder in the API
Inheritance
Inheritance hierarchy
SubClass and SuperClass
Overriding Methods
Dynamic Method Binding
Java's Framework
Packages of Classes
Application Programming Interface (API)
API documentation
Qualifiers and Access Specifiers
For: Class, Variable and Method
Object's Public Interface and Private Implementation
Polymorphism
Polymorphic variables, Parameters and Arrays
Exception Handling
Exception objects
try/catch/finally statement
Java's Catch or Specify Requirement
Collections
Vector, Enumeration, List etc.
Input – Output
Input and Output Streams
Read Write locally and remotely
Object persistence – Serialization

Prerequisites

Pre-requisites

Previous programming background is recommended. To get the most out of this training we also recommend that you install Sun/Oracle JDK (1.6.x) and recent version of eclipse ("Eclipse IDE for Java Development") prior to the class. Mac users can obtain JDK w/XCode.

Learn iT! San Francisco - 33 New Montgomery St. - Suite 300 - San Francisco, CA 94105 - 855.838.5028 or 415.693.0250

Learn iT! San Jose - 2025 Gateway Place - Suite 390 - San Jose, CA 95110 - 855.838.5028 or 408.200.0953

Learn iT! New York - 330 West 38th St. - Suite 301 - New York, NY 10018 - 855.838.5028

Live Online Training - Remote Training - Home, Office, Anywhere