

# Introduction to Programming for Android Using Android Studio

Download Whitepaper: Accelerate Your Modernization Efforts with a Cloud-Native Strategy  
Get Your Free Copy Now

**Course Number: WA1965**

**Duration: 2 days**

## Overview

### Course Description

This Android Programming course teaches Java developers to build user-friendly interfaces for Android applications. Whether new to mobile development or looking to expand your skillset, this course provides a comprehensive foundation for creating engaging Android apps using Android Studio.

### Skills Gained

- The architecture of Android OS.
- Using the Eclipse based development environment.
- GUI development.
- Supporting multiple languages.

### Prerequisites

Experience programming Java applications.

## Course Outline

- Introduction to Android
  - What is Android?
  - A Short History
  - Open Source
  - Advantages of Android
  - Disadvantages of Android
  - Android Version Distribution
  - Android Market Share
  - Phones by manufacturer
  - References
- The Development Environment
  - What Do We Need?
  - Downloading the Tools
  - Setting Up the SDK
  - Android Studio
  - Starting Android Studio
  - Android Studio Projects
  - Android Studio Interface
  - Visual Designer
  - Gradle Build System
  - Modifying Android Project Settings
  - The Android Emulator
  - Hardware Virtualization and the AndroidEmulator
  - Create a Virtual Device
  - Device Definition
  - System Image
  - Creating a Project
  - Minimum SDK Version
  - Running Your Application in the Emulator
  - Running Your Application on a Device
  - Run Configuration
  - Debugging an Application

- Developing for Multiple Android Versions
- Migrating to Android Studio from Eclipse
- The Architecture of Android
  - Key Components Stack
  - The Kernel
  - Libraries
  - The Android Runtime JVM
  - Application Framework
  - Applications
- Support LibrariesApplication Fundamentals
  - Recapping The Basics
  - Packaging an Application
  - Application Reusability Architecture
  - The Manifest File
  - Activity
  - Creating an Activity Class
  - Activity Lifecycle
  - Launching an Activity
  - Pausing and Stopping an Activity
  - Destroying an Activity
  - Activity Destruction Under Resource Shortage
  - Saving State
  - Creating Views
  - Registering an Activity
  - The Default Activity
  - Process Lifecycle
  - The Application Class
  - The Application Context
- Basic GUI Programming
  - The Fundamentals
  - View Hierarchy
  - Creating Views Programmatically
  - Creating View in XML Layout
  - More on XML Layout

- Common Programming Tasks with Views
  - TextView and EditText
  - Multi-Line Text
  - Ellipsize Long Text
  - Button
  - CheckBox
  - RadioButton and RadioGroup
  - ToggleButton
  - ImageView
  - RatingBar
- Layout Management
  - Background
  - LinearLayout
  - LinearLayout Orientation
  - match\_parent (fill\_parent) Example
  - Layout Gravity
  - Weight
  - TableLayout
  - Managing the Width
  - RelativeLayout
  - RelativeLayout Example
  - ScrollView
  - HorizontalScrollView
- Resource Management and Localization
  - Introduction
  - Main Types of Resources
  - Defining Value Resources
  - Defining String Arrays
  - Creating Image Resource
  - Alternate Resource Folders
  - Alternate Folder Name Rules
  - Android XML Files in Android Studio
  - How Android Finds Resources
  - Dealing with Screen Orientation

- Orientation Change and Activity Lifecycle
- Fix Orientation
- Developing for Tablets
- Basics of Localization
- Testing for Localization
- Loading Localized String from Code
- Intents and Intent Filters
  - What is an Intent?
  - The android.content.Intent Class
  - Explicitly Specifying Component's Class
  - Implicit Target Component Specification
  - Intent Filters
  - Data Type Filter
  - Action Name
  - Category Name
  - Hint Matching Logic
  - Example Hint Matching
  - Default Component of an Application
  - Starting an Activity
  - Getting Output from Activity
  - Pending Intent
- Fragments and Supporting Different Screens
  - The Problem
  - Screen Density
  - Screen Size
  - Size Qualifiers in Android 3.2
  - Supporting Multiple Screens
  - Declaring Screen Support in Manifest
  - Screen Compatibility Mode
  - Providing Alternate Resources
  - Scaling Images
  - Providing Alternate Layout for Large Devices
  - Fragments
  - Fragment Example

- Creating a Fragment
- Adding a Fragment to an Activity
- Managing Fragments
- Detecting Fragments From Activities
- Communicating With an Activity
- Using Fragment Callback Interface
- Fragment State
- Supporting Fragments on Older Devices
- Putting It All Together
- Testing on Multiple Screens
- Menus With Action Bar
  - Android Navigation
  - Action Bar Overview
  - Anatomy of the Action Bar
  - Action Bars in the Wild
  - Using an Existing Menu Definition
  - Adding Action Items
  - Handling Clicks on Action Items
  - Split Action Bar
  - Up Navigation
  - Using Action Views
  - Drop-down Navigation
- Lab Exercises
  - Lab 1. Setting up the Development Environment
  - Lab 2. Developing a Simple Application
  - Lab 3. Basic UI Development
  - Lab 4. Basics of Activity and Process Lifecycle
  - Lab 5. Using Simple Views
  - Lab 6. Layout Management
  - Lab 7. Resource Management
  - Lab 8. Localization
  - Lab 9. Using Intent to Launch Activities
  - Lab 10. Using Fragments
  - Lab 11. Implementing the Action Bar