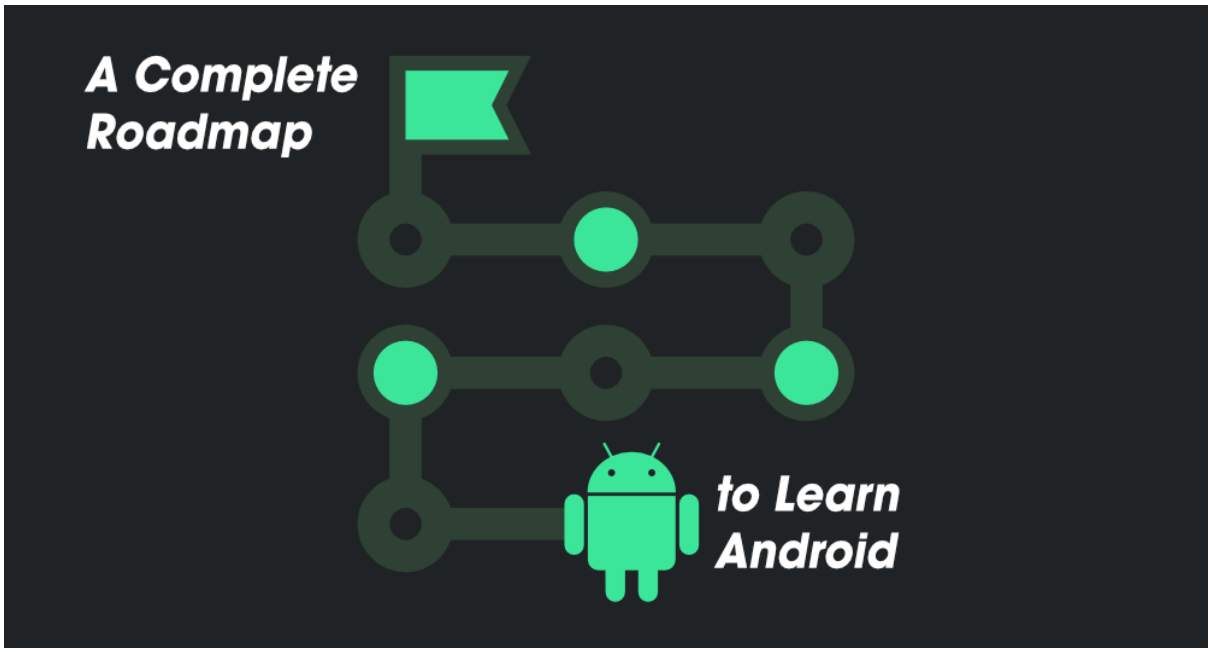
A complete roadmap for app developer

Android is an open-source operating system, based on the Linux kernel and used in mobile devices like smartphones, tablets, etc. Further, it was developed for smartwatches and Android TV. Each of them has a specialized interface. Android has been one of the best-selling OS for smartphones. Android OS was developed by Android Inc. which Google bought in 2005. Various applications like games, music player, camera, etc. are built for these smartphones for running on Android. Google Play Store features quite 3.3 million apps. Today, Android remains dominant on a global scale. Approximately 75% of the world population prefers using Android as against 15% of iOS. It is an operating system that has a huge market for apps.



It is known to everyone that how popular is Android nowadays. Now the questions arise are, Why Android (Decide the Goal First?), how to start? Where to start? What topics one should cover? etc, etc. Do you need to learn all the concepts from a book or you should go with some online tutorials or you should learn Android by doing some projects on it? So in this article, we are going to discuss all these things in detail.

Why Android? (Decide the Goal First?)

So before jumping into the complete Roadmap of Android one should have a clear goal in his/her mind that why he/she wants to learn Android? Is it for your college academic projects? or is it for your long-term career? or do you want to build your apps to start your business? So first make a clear goal. Why do you want to learn Android? For example, if you want to learn Android for your college Academic projects then it's enough to just learn the beginner things in Android. Similarly, if you want to build your long term career then you should learn the professional or

advanced things also. So it's on your hand and it's your decision why you want to learn Android.

How to Learn Android?

In Android, programming is done in two languages JAVA
And Kotlin and XML(Extension Markup Language). The XML file deals with the design, presentation, layouts, blueprint, etc (as a front-end) while the JAVA or KOTLIN deals with working of buttons, variables, storing, etc (as a back-end). And the biggest confusion for an Android beginner is which language to choose between Java and Kotlin? So let me try to overcome the confusion first.

Java or Kotlin?
Java is the official language for Android App Development and consequently, it is the most used language as well. Many of the apps in the Play Store are built with Java and it is also the most supported language by Google.

Kotlin is a cross-platform programming language that may be used as an alternative to Java for Android App Development. It has also been introduced as a secondary "official" Java language in 2017. The only sizable difference is that Kotlin removes the superfluous features of Java such as null pointer exceptions. It also removes the necessity of ending every line with a semicolon. In short, Kotlin is much simpler for beginners to try as compared to Java and it can also be used as an "entry point" for Android App Development.

Learning Resources
There are plenty of resources and videos available online and it's confusing for developers where to start learning all the concepts. Initially, as a beginner, if you get overwhelmed with so many concepts then don't be afraid and stop learning. Have patience, explore, and stay committed to it.

The best way to learn Android is by developing some baby projects to some advanced projects.

A Roadmap to Learn

Start with the Overview of Android. Read some android related blogs and also research some Android-related things. For example read blogs on Introduction to Android Development, History of Android, Different Versions of Android, and also topics like Why Kotlin will replace Java for Android App Development, etc. etc. and make a complete mind makeup to start your journey on Android. Make yourself self-motivated to learn Android and build some awesome projects on Android. Do it regularly and also start learning one by one new concept on Android. It will be very better to join some workshops or conferences on Android

before you start your journey. Make your goal clear and move on toward your goal.

1) Programming
One may consider this step as a prerequisite. Learn these programming languages before you start learning Android.

- Java Programming Language
- Kotlin Programming Language
- Sound Knowledge of XML (Extensible Markup Language)

2) Android Studio
It's better to know your tools before you are going to use it. Android Studio is the official Integrated Development Environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development.

- **File Structure:**
    - AndroidManifest.xml file
    - Java file
    - Drawable file
    - Layout file
    - mipmap file
    - colors.xml file
    - strings.xml file
    - styles.xml file
    - build.gradle(Module: app) file
- **Android Studio Overview:**
    - Create a new project
    - Reopen, close, save the project
    - Create a new activity, classes, drawable resource files
    - Run the app on AVD of Emulator or in a real device etc.

3) Android Components
There are some necessary building blocks that an Android application consists of. These loosely coupled components are bound by the application manifest file which contains a description of each component and how they interact.

- **Activity:**
    - Activity life cycle
    - Handle Activity State Changes
    - Understand Tasks and Back Stack
    - Processes and Application Lifecycle
- **Services:**
    - Types of Android Services
    - The Life Cycle of Android Services

- **Content Provider:**
    - Content URI
    - Operations in Content Provider
    - Working of the Content Provider
    - Creating a Content Provider
- **Broadcast Receiver:**
    - Implicit Broadcast Exceptions

4) Simple UI Design
After you got an idea about the different components of Android then start exploring some simple UI design which is given below.

- **Explore different layouts:**
    - Frame
    - Linear
    - Relative
    - Constraint
- **View Elements:**
    - TextView
    - EditText
    - Buttons
    - ImageView
- **Intent:**
    - Implicit
    - Explicit
    - Intent Filter

5) Complex UI Design
Once you have a command on simple UI design then solely move to the complex UI design part such as:

- ListView
- RecycleVIew
- Fragments
- Dialogs
- Toast
- Bottom Sheets
- Navigation Drawer
- Tabs
- Material Design
- Some inserting Animations

6) Storage
In Android, there are three types of the storage system:

- Shared Preferences
- File System

- Database

7) Build

- Gradle
- Debug/ Release Configuration

8) Threading

- Threads
- Looper

9) Debugging

One of the most important skills of a developer is debugging skills. So the developer must learn these things:

- Exceptions
- Error Handling
- Logging
- Memory Profiling

10) Memory Leaks

- Cause of memory leaks
- Detecting and fixing memory leaks
- Context

11) Third-Party Libraries

- Image Loading Libraries
    - Glide
    - Picasso
    - Fresco
    - COIL
- Dependency Injection
    - Dragger
- Networking
    - Retrofit
- Multithreading
    - Coroutines
    - Rxjava

12) Android Jetpack

On its official site, it says Android Jetpack is a set of libraries, tools, and architectural guidance to help make it quick and easy to build great Android apps. It provides common infrastructure code so you can focus on what makes your app unique.

- AppCompat library
- Architecture components,
- Animation and transitions

- Android Ktx
- Navigation
- Paging
- Slices
- WorkManager

13) Android Architecture

The three famous architecture in the Android world are:

- MVVM (Model–View–ViewModel)
- MVI (Model-View-Intent)
- MVP (Model View Presenter)

14) Firebase

- FCM (Firebase Cloud Messaging)
- Analytics
- Remote Config
- App Indexing

15) Unit Testing

- Local Unit Testing
- Instrumentation Testing

16) Security

- Encrypt / Decrypt
- Proguard

17) App Release

- Signed APK
- Play Store

18) Keep Practicing and Read Some Android Tips

"Practice makes a man perfect" which tells the importance of continuous practice in any subject to learn anything. So keep practicing and read some Android tips such as Tips to Improve Your Android Development Skills, Tips to Get Your Android App Featured on Google Play Store, etc. Below is a complete diagrammatical representation of the Android Roadmap.

Android Developer Roadmap