



# Android Programming

Trainer: Mr. Rohan Paramane



# Feature Phone vs Smart Phone

---

- Any phone that have its own OS is called as Smart Phone
- Feature phone is just a phone used to do limited task like call and sms.
- Smart phone is a phone which is having its own operating system.
- Smartphone OS - iOS, android, windows phone os, Symbian, oxygen os, tizen, palmOS....
- watch - watch os, android wear, tizen
- smart car - tesla
- smart tv - android tv, webos(LG), tvOS(apple)
- tablet - android os, ipados
- smart fridge - samsung os (based on tizen)



# Android History

---

- Android is Free and Open Source operating system for mobile devices
- Android is developed on top of Linux (with some changes)
- Developed by Andy Rubin in 2003 and then acquired by Google in 2005
- Android is the nick name of Andy Rubin given by coworkers because of his love to robots.
- In 2007, Google announces the development of android OS.
- Google started an OHA(Open handset alliance)
- In 2008, HTC launched the first android mobile.



# Android Versions

- Android made its official public debut in 2008 with Android 1.0 a release so ancient it didn't even have a cute codename.
- 2008 - 1.0 to 1.1
  - include a suite of early Google apps like Gmail, Maps, Calendar, and YouTube, all of which were integrated into the operating system
- Early 2009 - 1.5 – Cupcake
  - on-screen keyboard, framework for third-party app widgets
- Late 2009 - 1.6 - Donut
  - ability for the OS to operate on different screen sizes and resolutions
  - added support for CDMA networks
- 6 weeks after Donut - 2.0 to 2.1 – Éclair
  - addition of voice-guided turn-by-turn navigation and real-time traffic info
  - live wallpapers, platform's first speech-to-text function.
- 2010 - 2.2 – Froyo
  - addition of the now-standard dock at the bottom of the home screen
- 2010 - 2.3 – Gingerbread
  - Attractive UI enhancements



# Android Versions

- 2011 - 3.0 to 3.2 - Honeycomb (Tablet Release)
  - On-screen buttons and a card-like appearance for app-switching.
  - Organized UI for tablets to utilize the space
- 2011 - 4.0 - Icecream sandwich (Mobile Phones)
  - on-screen buttons and a card-like appearance for app-switching.
- 2012-2013 - 4.1 to 4.3 - Jelly Bean
  - Google Now was introduced, Placing widgets in lock screen
  - Multiuser support also came into play
- 2013 - 4.4 - KitKat
  - transparent status bar, first version of "OK, Google" support
- 2014 - 5.0 and 5.1 - Lollipop
  - Material Design standard
- 2015 - Marshmallow
  - screen-search feature called Now On Tap
  - app permissions, support for fingerprint readers, and support for USB-C.



# Android Versions

- 2016 - 7.0 to 7.1 Nougat
  - split-screen mode,
  - a new bundled-by-app system for organizing notifications, and a Data Saver feature.
  - Nougat's enhancements, however, was the launch of the Google Assistant
- 2017 - 8.0 to 8.1 - Oreo
  - native picture-in-picture mode, a notification snoozing option
- 2018 - 9.0 - Pie
  - hybrid gesture/button navigation system,
  - dashboard of Digital Wellbeing controls
- 2019 - Android 10
  - updated permissions system with more granular control over location data
  - a new system-wide dark theme, a new distraction-limiting Focus Mode
- 2020 - Android 11
  - The version's most significant changes revolve around privacy
  - Bubbles — a new kind of multitasking system to pop conversations out into floating windows



# Android Versions

---

- 2021 - Android 12
  - enhancements in the areas of performance, security, and privacy.
- 2022- Android 13
  - It lay the groundwork for a whole new type of multipurpose product, one that could function as a stationary Smart Display and then allow you to detach its screen and use it as a tablet.
- 2023 - Android 14 (Developers Preview)



# Installation of Android studio

- OS required – windows/Linux/macOS
- Language – Java / kotlin
- VM – JVM+JDK
- IDE – Android Studio
- SDK – android sdk
  - Software Development Kit similar to your JDK It contains
    1. Libraries - librt, libsqlite, etc..
    2. Packages - android.app, android.content, etc..
    3. Documentation - developer.android.com
    4. ToolChain – dx, javac, adb, etc..
    5. Execution Platform – android emulator(qemu)





# Understanding an Android Studio Project.

- The android project contains different types of app modules, source code files, and resource files. We will explore all the folders and files in the android app.
- Manifests Folder
  - Manifests folder contains AndroidManifest.xml for our creating the android application.
  - This file contains information about our application such as the Android version, metadata and other application components.
  - It acts as an intermediary between android OS and our application.
- Java Folder
  - The Java folder contains all the java source code (.java) files that we create during the app development, including other Test files.
- res (Resources) Folder
  - The resource folder is the most important folder because it contains all the non-code sources like images, XML layouts, and UI strings for our android application.
- Drawable Folder
  - It contains the different types of images used for the development of the application. We need to add all the images in a drawable folder for the application development.



# Understanding an Android Studio Project.

- Layout Folder
  - The layout folder contains all XML layout files which we used to define the user interface of our application. It contains the activity\_main.xml file.
- Mipmap Folder
  - This folder contains launcher.xml files to define icons that are used to show on the home screen. It contains different density types of icons depending upon the size of the device such as hdpi, mdpi, xhdpi.
- Values Folder
  - Values folder contains a number of XML files like strings, dimensions, colors, and style definitions. One of the most important files is the strings.xml file which contains the resources.
- Gradle Scripts
  - a. In build.gradle (Project) there are buildscripts
  - b. In build.gradle (Module) plugins and implementations are used to build configurations that can be applied to all our application modules.



# Gradle

---

- Gradle is an open-source build automation tool that is designed to be flexible enough to build almost any type of software
- Gradle in android studio is an automated build system and it contains a number of files that are used to define a build configuration that can be applied to all modules in our application.



# Android Architecture



- **Application**  
Default apps / downloaded from playstore
- **Application Framework**  
Used for application development
  1. Activity manager
  2. Service Manager
- **Libraries**  
It consists of system libraries
- **Android runtime/Dalvik VM**  
JVM for android, faster than desktop jvm.  
It uses 30% native instructions and 70% byte codes  
It is possible because of double compilation that android uses.  
The byte code is given to dex compiler which compiles and produces,  
70% byte code and 30% native code in .dex(dalvik executables) file.
- **Kernal**  
memory management ,process management ,HAL -> Device Drivers



# Android Architecture

- For android operating system the linux kernel is customized.
- Changes added to android from main line kernel is called as **Androdism**.
- Android has its own C runtime library.
- It is more light weight than Glibc which is used in desktop Linux
- The name for the Android's runtime library is **Bionic**.
- Android is meant to run on so many types of different devices - tablet, phones, STBs, treadmills, and what not - that the underlying hardware may greatly differ in its scope and support
- In an effort to combat this, Android defines a **Hardware Abstraction Layer (HAL)** which aims to promote standardization by defining an adapter
- The Hardware Abstraction Layer defines what an abstract camera, GPS, sensor, and other components look like to Android



# Androidism

- Anonymous Shared Memory (ASHMem)
  - A mechanism to allow shared memory. Applications can open a character device (/dev/ashmem) and create a memory region which can then be mapped into memory
- Binder
  - It is used for Inter Process Communication (IPC Mechanism)
  - Android services register with Binder, and clients can connect to them, with the help of service manager
- Logger
  - providing kernel-based ring buffers for fast, file-less logging.
  - Android logs are maintained by character devices in /dev/log
- ION memory Allocator
  - Introduced in ICS, and offers efficient memory allocation to kernel drivers
  - ION replaces an older Androidism, PMEM, and aims to standardize memory management in the various SoC(System on chip) architectures
- Low memory Killer
  - A layer on top of Linux's own Out-Of-Memory (OOM) killer, which terminates processes in case of memory exhaustion
- Sync Driver
  - The latest Androidism, introduced to allow fast synchronization primitives, used primarily by Android's Graphics stack
- WakeLocks
  - Originally a separate Androidism to control power management and prohibit the kernel's sleep functionality, wakelocks have gradually been merged with the kernel's own wakeup source mechanisms





Thank you!

Rohan Paramane<[rohan.paramane@sunbeaminfo.com](mailto:rohan.paramane@sunbeaminfo.com)>

