



MOBILE APPLICATION DEVELOPMENT

ANDROID

By: AbuBakar Ubaid



WHAT IS ANDROID?

- Mobile operating system based on **Linux Kernel**
- User Interface for touch screens
- Used on **over 80%** of all smartphones
- Powers devices such as watches, TVs, and cars
- Over **2 Million Android apps** in Google Play store
- Highly customizable for devices / by vendors
- Open source



IDE (Integrated Development Environment)

- Software suite that consolidates basic tools required to write and test software.
- Without an IDE, a developer must **select, deploy, integrate and manage** all of these tools separately.
- An IDE can be a **standalone application** or it can be part of a larger package.
- **For example:**
Net beans, Eclipse, Android Studio

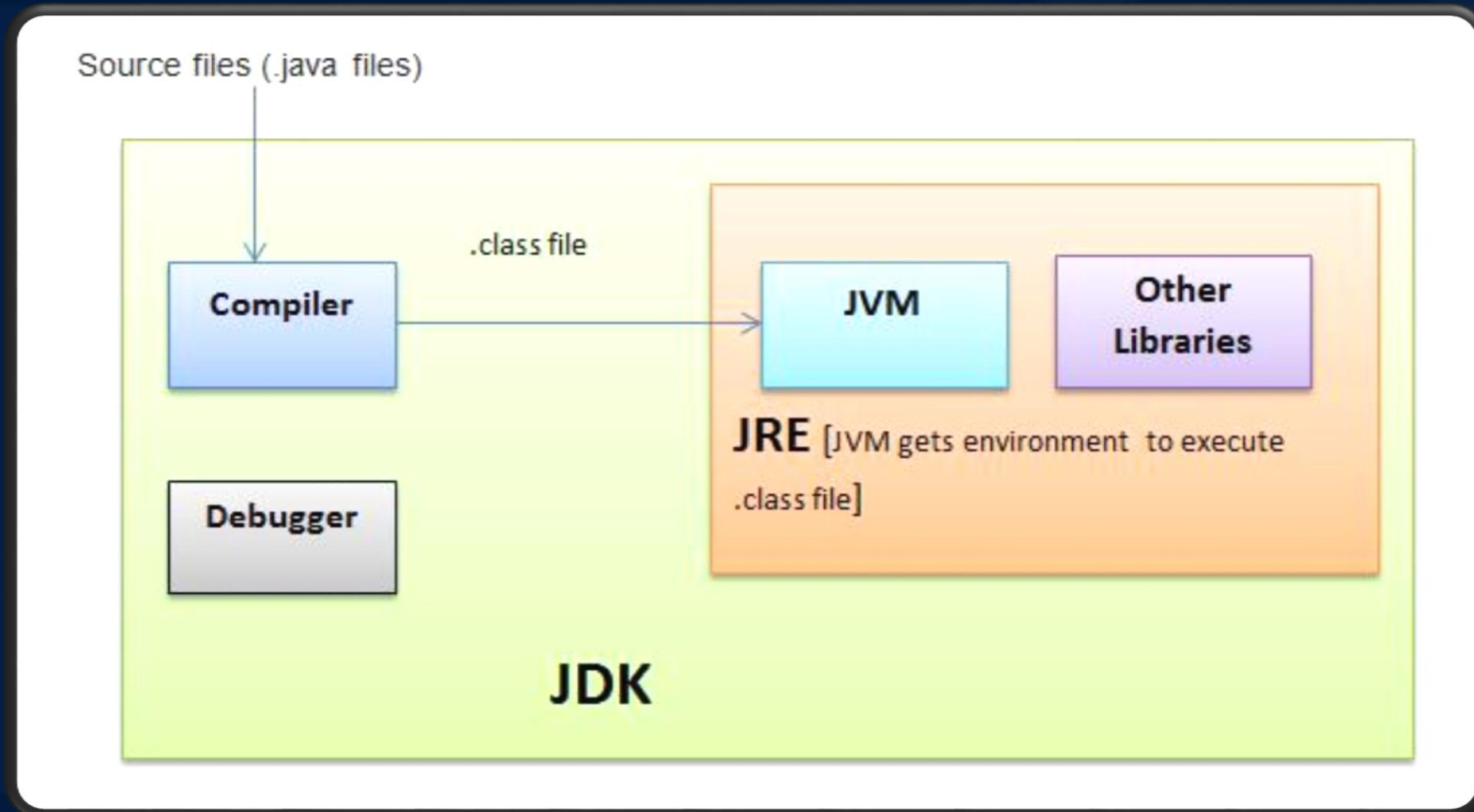


JDK (Java Development Kit)

- The **JDK** is one of three core technology packages used in Java programming, along with the **JVM** (Java Virtual Machine) and the **JRE** (Java Runtime Environment).
- The **JVM** is the Java platform component that executes programs.
- The **JRE** is the on-disk part of Java that creates the JVM.
- The **JDK** allows developers to create Java programs that can be executed and run by the JVM and JRE.



JDK - JRE - JVM



AVD (**Android Virtual Device**)

- **AVD** is an emulator configuration that allows developers to test the application by simulating the real device capabilities.
- We can configure the AVD by specifying the hardware and software options.
- AVD manager enables an easy way of creating and managing the AVD with its graphical interface



LIBRARY

- It can include **everything needed to build an app**, including source code, resource files, and an **Android** manifest.
- **Complete package** of something
- **For example:**
 - Photo manipulation in android.
 - Write code of scaling and animating for every time ☹️
 “Picasso” photo manipulating library



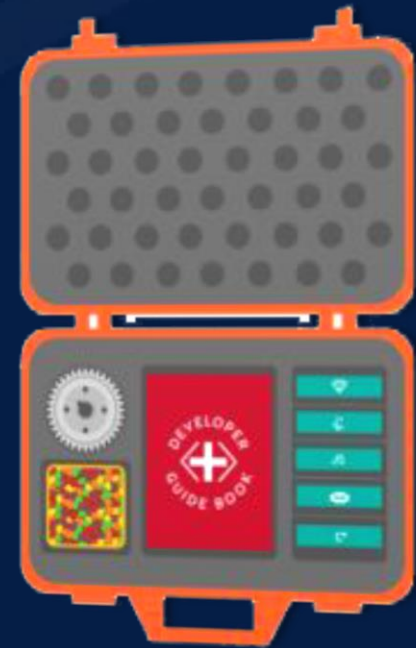
API (Application Programming Interface)

- **API** is a software intermediary that allows two applications to talk to each other
- **Connectivity** with server and hardware
- **For Example:**
 - **Waiter in a hotel**
 - Each time you use an app like **Facebook**, send an **instant message**, or **check the weather** on your phone, you're using an API.
 - **Use Google map**
 - **Book different flights** from different sites



SDK (Software Development Kit)

- **SDK** or **devkit**, providing a set of tools, libraries, relevant documentation, code samples, processes, and or guides that allow developers to create software applications on a specific platform.
- **For Example:**
 - Build a house without tools
 - For eat bread... seed...grow...harvest...eat ☹️
 - Write your own code from scratch ☹️



ANDROID OLDER VERSIONS

No	Name	Version	Released	API level
1	Alpha	1.0	Sep 23, 2008	1
2	Beta	1.1	Feb 09, 2009	2
3	Cupcake	1.5	Apr 27, 2009	3
4	Donut	1.6	Sep 15, 2009	4
5	Éclair	2.0 – 2.1	Oct 26, 2009	5 – 7
6	Froyo	2.2 – 2.2.3	May 20, 2010	8
7	Gingerbread	2.3 – 2.3.7	Dec 06, 2010	9 – 10
8	Honeycomb	3.0 – 3.2.6	Feb 22, 2011	11 – 13
9	Ice-cream Sandwich	4.0 – 4.0.4	Oct 18, 2011	14 - 15



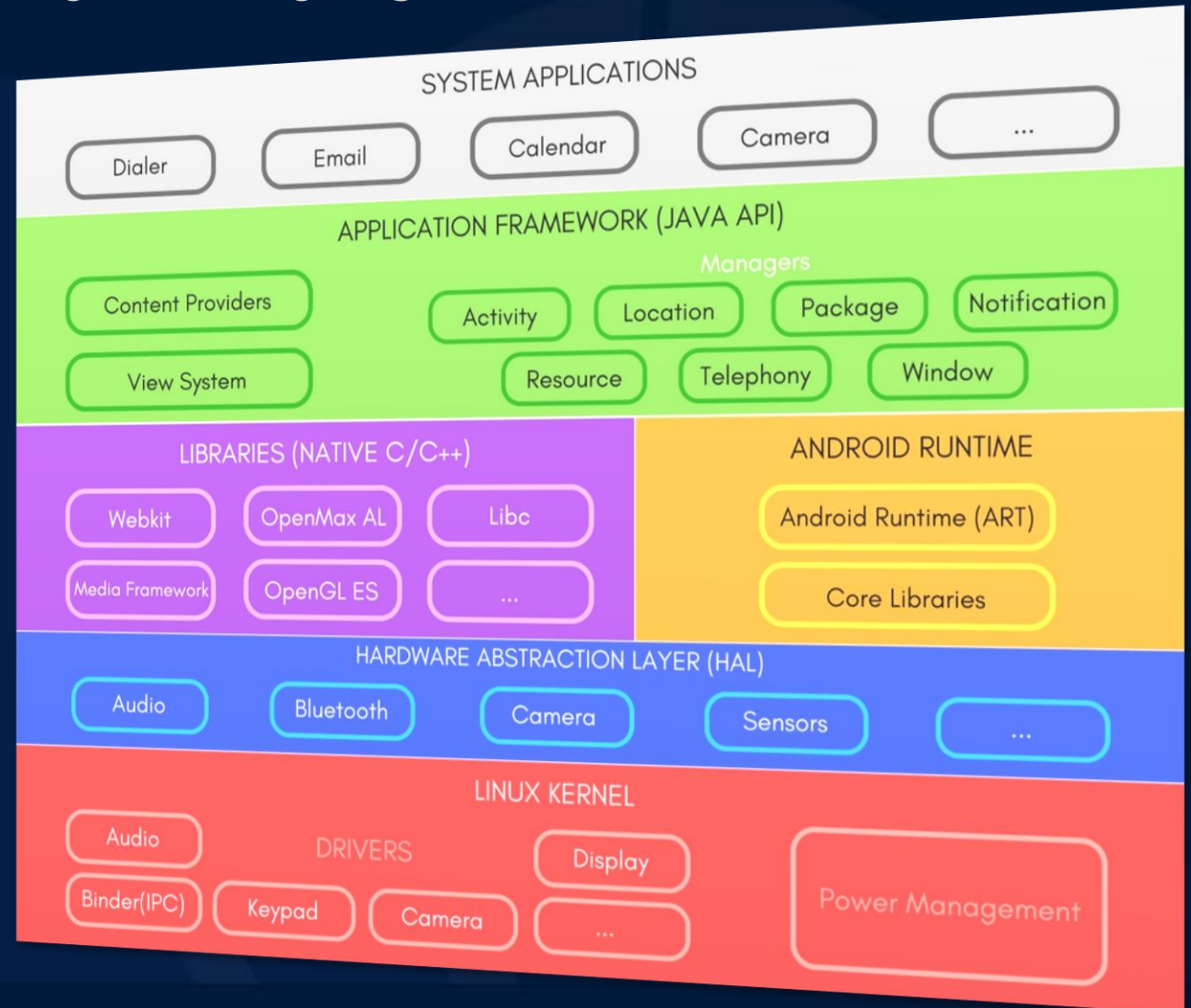
ANDROID OLDER VERSIONS

No	Name	Version	Released	API level
10	Jelly Bean	4.1 – 4.3.1	Jul 09, 2012	16 – 18
11	Kit Kat	4.4 – 4.4.4	Oct 31, 2013	19 – 20
12	Lollipop	5.0 – 5.1.1	Nov 12, 2014	21 – 22
13	Marshmallow	6.0 – 6.0.1	Oct 05, 2015	23
14	Nougat	7.0 – 7.1.2	Aug 22, 2016	24 – 25
15	Oreo	8.0 – 8.1	Aug 21, 2017	26 – 27
16	Pie	9.0	Aug 06, 2018	28
17	Android X	9.0-	Sep, 2019	28-



ANDROID PLATFORM ARCHITECTURE

- Linux Kernel
- HAL
- ART
- Libraries
- Application Framework
- System Applications



❖ LINUX KERNEL

- This layer is the **foundation of the Android Platform**.
- Contains all **low level drivers for various hardware components support**.
- **Android Runtime relies on Linux Kernel** for core system services like,
 - Memory, process management, threading etc.
 - Network stack
 - Driver model
 - Security and more.



❖ **HARDWARE ABSTRACTION LAYER**

- Provides Abstraction between hardware and rest of the software stack.



❖ ANDROID RUNTIME (ART)

- Designed to run apps in a **constrained environment** that has **limited muscle power** in terms of battery, processing and memory.
- Contains **set of core libraries** that enables developers to write **Android Apps using Java Programming**.
- It also has a very **efficient garbage collection**.



❖ APPLICATION FRAMEWORK

- It is a collection of APIs written in Java, which gives developers access to the complete feature set of Android OS.
- Enables and simplify the reuse of core components and services, like:

Activity Manager:

- Manages the Lifecycle of apps & provide common navigation back stack.

Window Manager:

- Manages windows and drawing surfaces, and is an abstraction of the surface manager library.

Content Providers:

- Enables application to access data from other applications or to share their own data



❖ APPLICATION FRAMEWORK

View System:

- Contains User Interface building blocks used to build an application's UI, including lists, grids, texts, boxes, buttons, etc. and also performs the event management of UI elements(explained in later tutorials).

Package Manager:

- Manages various kinds of information related to the application packages that are currently installed on the device.

Telephony Manager:

- Enables app to use phone capabilities of the device.



❖ APPLICATION FRAMEWORK

Resource Manager:

- Provides access to non-code resources (localized Strings, bitmaps, Graphics and Layouts).

Location Manager:

- Deals with location awareness capabilities.

Notification Manager:

- Enable apps to display custom alerts in the status bar.



❖ SYSTEM APPLICATION

- Top of the Android Application Stack, is occupied by the System apps and other Apps that users can download from Android's Official Play Store, also known as Google Play Store.
- A set of Core applications are pre-packed in the handset like Email Client, SMS Program, Calendar, Maps, Browser, Contacts and few more.
- This layer uses all the layers below it for proper functioning of these mobile apps.



Installation of ANDROID STUDIO



By: AbuBakar Ubaid

ANDROID

LECTURE - 01 “Android, Terminologies and Architecture”

THANK YOU 😊

