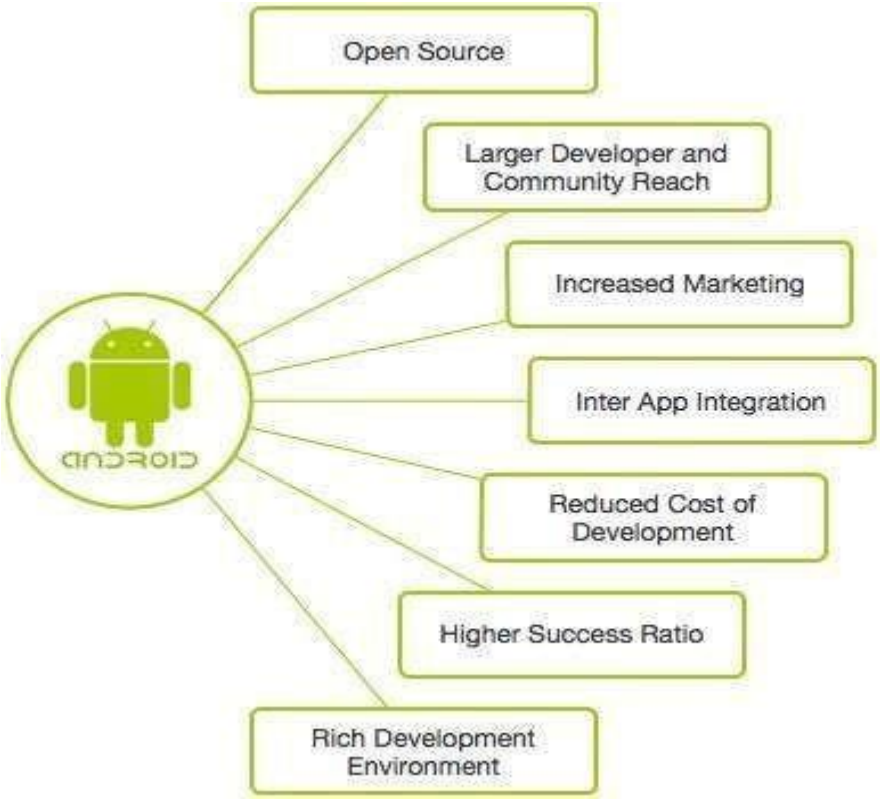
Introduction to Android


Who am I ?

- Android is an **open source and Linux-based** Operating System for mobile devices such as smartphones and tablet computers. Android was developed by the Open Handset Alliance, led by Google, and other companies.
- Android offers a **unified approach** to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.
- **The first beta version of the Android Software Development Kit (SDK) was released by Google in 2007 where as the first commercial version, Android 1.0, was released in September 2008.**
- On June 27, 2012, at the Google I/O conference, Google announced the next Android version, 4.1 Jelly Bean. Jelly Bean is an incremental update, with the primary aim of improving the user interface, both in terms of functionality and performance.
- The source code for Android is available under free and open source software licenses. Google publishes most of the code under the Apache License version 2.0 and the rest, Linux kernel changes, under the GNU General Public License version 2.

Why Android ?


Open Source
Larger Developer and Community Reach
Increased Marketing
Inter App Integration
Reduced Cost of Development
Higher Success Ratio
Rich Development Environment
android

Android is a mobile operating system (OS) based on the Linux kernel that is currently developed by Google.

Designed primarily for:

Touch screen mobile devices

With specialized user interfaces for televisions (Android TV)

Eg. OnePlus TV

Cars (Android Auto)

- **Android Auto** is your smart driving companion that helps you stay focused, connected and entertained with the Google Assistant.
- **Android Auto** allows you to connect your phone to your **car** to avoid distractions while driving.

Wrist watches (Android Wear os)

Wear OS is a **version of Google's Android operating system designed for smartwatches and other wearables**.

Game consoles, digital cameras, and other electronics

Features of Android

Beautiful UI : Android OS basic screen provides a beautiful and intuitive user interface.

Connectivity : GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX.

Storage : SQLite, a lightweight relational database, is used for data storage purposes.

Media support : H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, and BMP.

Messaging : SMS and MMS

Web browser : Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.

Multi-touch : Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero.

Features of Android

Multi-tasking : User can jump from one task to another and same time various application can run simultaneously.

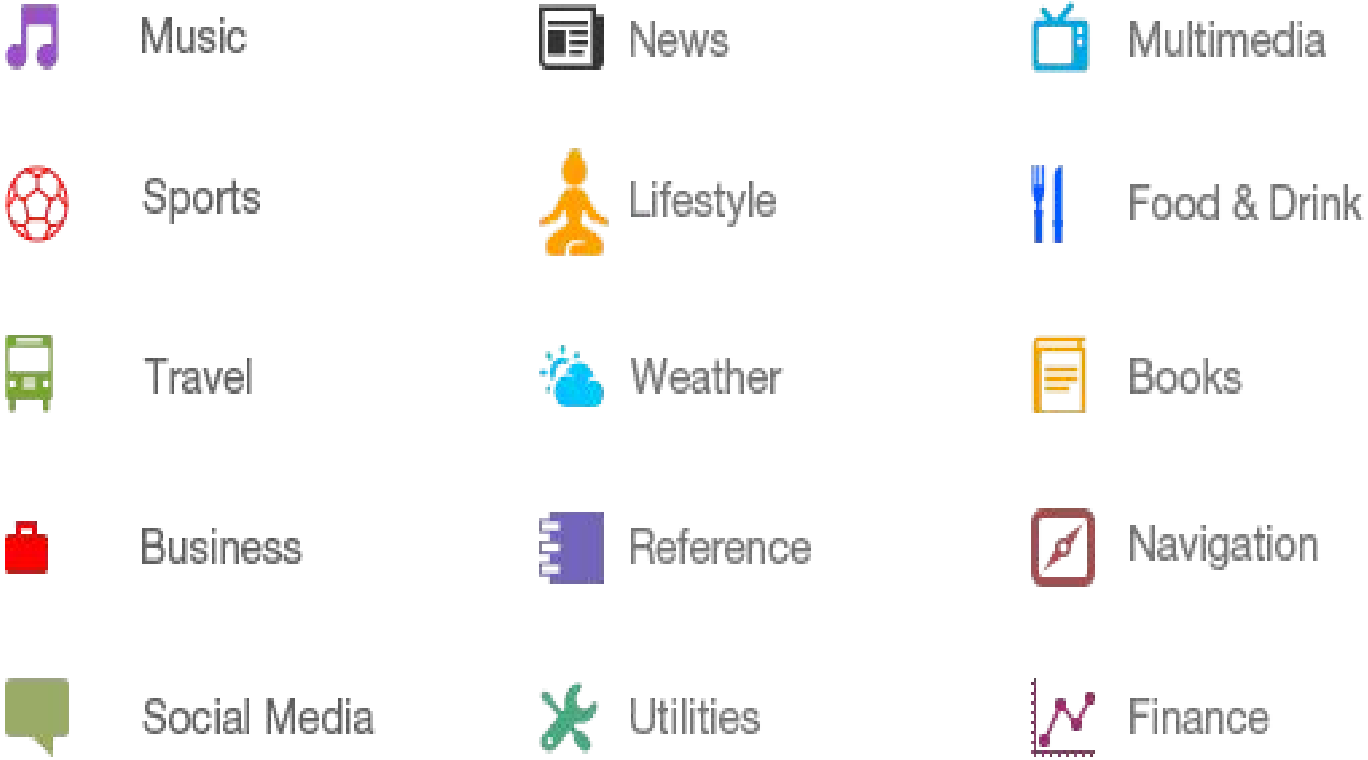
Resizable widgets : Widgets are resizable, so users can expand them to show more content or shrink them to save space.

Multi-Language : Supports single direction and bi-directional text.

GCM : Google Cloud Messaging (GCM) is a service that lets developers send short message data to their users on Android devices, without needing a proprietary sync solution.

Wi-Fi Direct : A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection.

Android Beam : A popular NFC-based technology that lets users instantly share, just by touching two NFC-enabled phones together.


Music
News
Multimedia
Sports
Lifestyle
Food & Drink
Travel
Weather
Books
Business
Reference
Navigation
Social Media
Utilities
Finance

History of Android

History of Android

- Initially, **Andy Rubin** founded Android Incorporation in Palo Alto, California, United States in October, 2003.
- In 17th August 2005, Google acquired android Incorporation. Since then, it is in the subsidiary of Google Incorporation.
- The key employees of Android Incorporation are **Andy Rubin**, **Rich Miner**, **Chris White** and **Nick Sears**.
- Originally intended for camera but shifted to smart phones later because of low market for camera only.
- 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.
- In 2008, HTC launched the first android mobile.

Android Versions, Codename and API

Version	Code name	API Level
1.5	Cupcake	3
1.6	Donut	4
2.1	Eclair	7
2.2	Froyo	8
2.3	Gingerbread	9 and 10
3.1 and 3.3	Honeycomb	12 and 13
4.0	Ice Cream Sandwitch	15

Android Versions, Codename and API

4.1, 4.2 and 4.3	Jelly Bean	16, 17 and 18
4.4	KitKat	19
5.0	Lollipop	21
6.0	Marshmallow	23
7.0	Nougat	24-25
8.0	Oreo	26-27

List of Android Versions and Initial Stable Release Dates

Android 1.0
September 23, 2008

1.5 - Cupcake
April 27, 2009

1.6 - Donut
September 15, 2009

2.0/2.1 - Éclair
October 26, 2009

2.2 - Froyo
May 20, 2010

2.3 - Gingerbread
December 6, 2010

3.0 - Honeycomb
February 22, 2011

4.0 - Ice Cream Sandwich
October 18, 2011

4.1/4.3 - Jelly Bean
July 9, 2012

4.4 - KitKat
October 31, 2013

5.0 - Lollipop
November 12, 2014

6.0 - Marshmallow
October 5, 2015

7.0 - Nougat
August 22, 2016

8.0 - Oreo
August 21, 2017

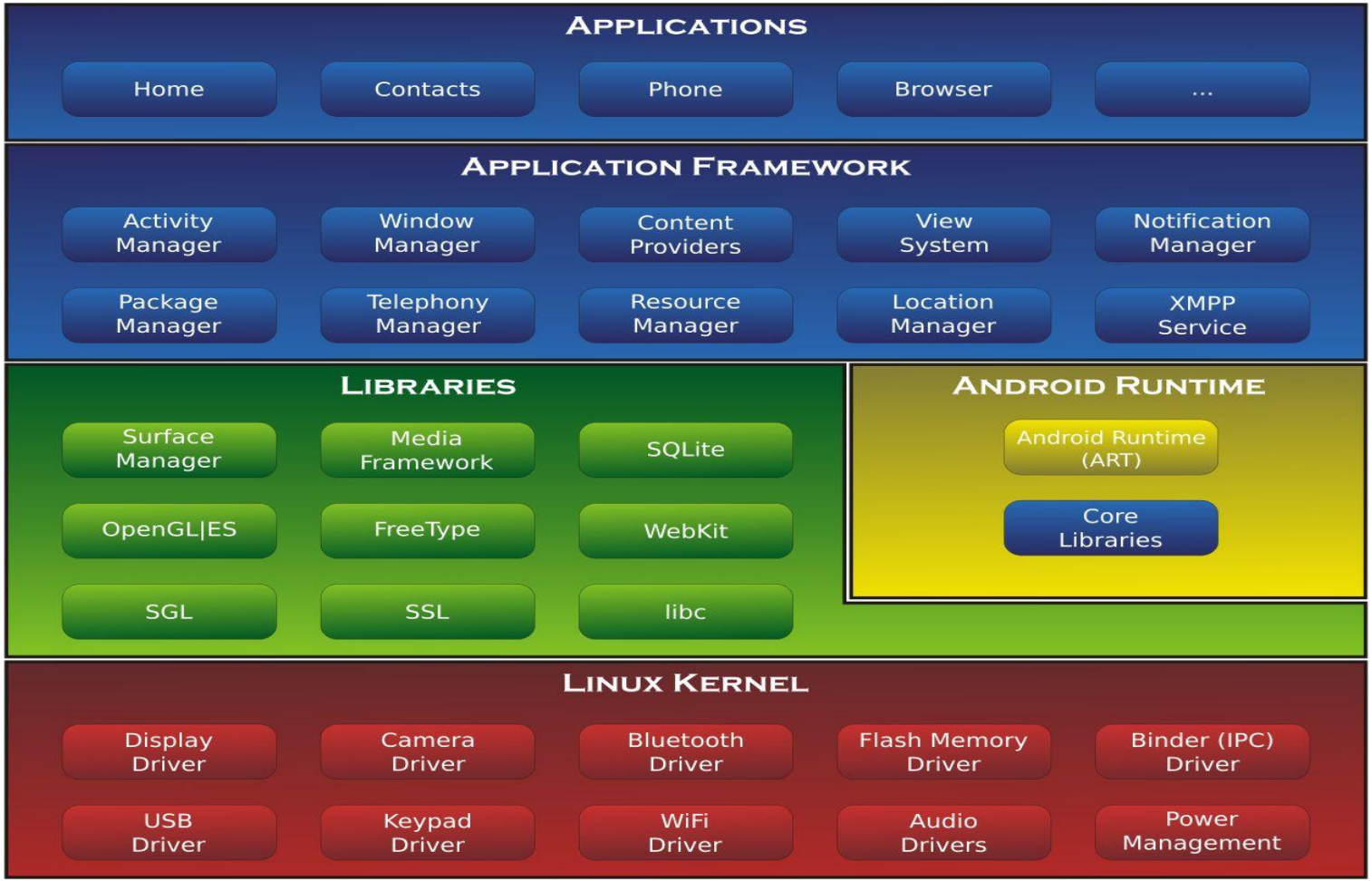
9.0 - Pie
August 6, 2018

Android 10
September 3, 2019

Android 11
September 8, 2020

Android 12
October 17, 2021

ANDROID ARCHITECTURE


Applications
Home
Contacts
Phone
Browser
...
Application Framework
Activity Manager
Window Manager
Content Providers
View System
Notification Manager
Package Manager
Telephony Manager
Resource Manager
Location Manager
XMPP Service
Libraries
Surface Manager
Media Framework
SQLite
OpenGL|ES
FreeType
WebKit
SGL
SSL
libc
Android Runtime
Android Runtime (ART)
Core Libraries
Linux Kernel
Display Driver
Camera Driver
Bluetooth Driver
Flash Memory Driver
Binder (IPC) Driver
USB Driver
Keypad Driver
WiFi Driver
Audio Drivers
Power Management

ANDROID ARCHITECTURE

- Android architecture contains different number of components to support any android device needs.
- Android software contains an open-source Linux Kernel having collection of number of C/C++ libraries which are exposed through an application framework services.
- Among all the components Linux Kernel provides main functionality of operating system functions to smartphones and Dalvik Virtual Machine (DVM) provide platform for running an android application.

ANDROID ARCHITECTURE

The main components of android architecture are following:-

- Applications
- Application Framework
- Android Runtime
- Platform Libraries
- Linux Kernel

Component : APPLICATIONS

- Applications is the **top layer of android architecture**. The pre-installed applications like *home, contacts, camera, gallery etc and third party applications* downloaded from the play store like chat applications, games etc. will be installed on this layer only.
- It runs within the Android run time with the **help of the classes and services** provided by the application framework.

Component : APPLICATION FRAMEWORK

- Application Framework provides **several important classes** which are used to create an Android application.
- It provides a **generic abstraction for hardware access and also helps in managing the user interface with application resources**. Generally, it provides the services with the help of which we can create a particular class and make that class helpful for the Applications creation.
- It includes different **types of services** *activity manager, notification manager, view system, package manager* etc. which are helpful for the development of our application according to the prerequisite.

Component : APPLICATION RUNTIME

- It contains components like **core libraries** and the **Dalvik virtual machine(DVM)**. Mainly, it provides the *base for the application framework and powers our application* with the help of the core libraries.

- Like Java Virtual Machine (JVM), Dalvik Virtual Machine (DVM) is a register-based virtual machine and specially designed and optimized for android to ensure that a device can run multiple instances efficiently. It depends on the layer Linux kernel for threading and low-level memory management. The core libraries enable us to implement android applications using the standard JAVA or Kotlin programming languages.

Components : PLATFORM LIBRARIES

The Platform Libraries includes various C/C++ core libraries and Java based libraries such as **Media, Graphics, Surface Manager, OpenGL** etc. to provide a support for android development.

- Media library provides support to play and record an audio and video formats.
- Surface manager responsible for managing access to the display subsystem.
- SGL and OpenGL both cross-language, cross-platform application program interface (API) are used for 2D and 3D computer graphics.
- SQLite provides database support and FreeType provides font support.
- Web-Kit This open source web browser engine provides all the functionality to display web content and to simplify page loading.
- SSL (Secure Sockets Layer) is security technology to establish an encrypted link between a web server and a web browser.

Component : LINUX KERNEL

- **Linux Kernel is heart of the android architecture.** It manages all the available drivers such as display drivers, camera drivers, Bluetooth drivers, audio drivers, memory drivers, etc. which are required during the runtime.
- The Linux Kernel will provide an *abstraction layer between the device hardware and the other components of android architecture.* It is responsible for management of memory, power, devices etc.

The features of Linux kernel

- **Security:** The Linux kernel handles the security between the application and the system.
- **Memory Management:** It efficiently handles the memory management thereby providing the freedom to develop our apps.
- **Process Management:** It manages the process well, allocates resources to processes whenever they need them.
- **Network Stack:** It effectively handles the network communication.
- **Driver Model:** It ensures that the application works properly on the device and hardware manufacturers responsible for building their drivers into the Linux build.

Application components

Application components are the essential building blocks of an Android application. These components are loosely coupled by the application manifest file AndroidManifest.xml that describes each component of the application and how they interact.

Four main components:

Activities : They dictate the UI and handle the user interaction to the smartphone screen

Services : They handle background processing associated with an application.

Broadcast Receivers :They handle communication between Android OS and applications.

Content Providers :They handle data and database management issues.

Activities

An activity represents visual representation of an Android application.
An activity represents a single screen with a user interface,in-short Activity performs actions on the screen.

An activity is implemented as a subclass of Activity class as follows –

```
public class MainActivity extends Activity {

}
```

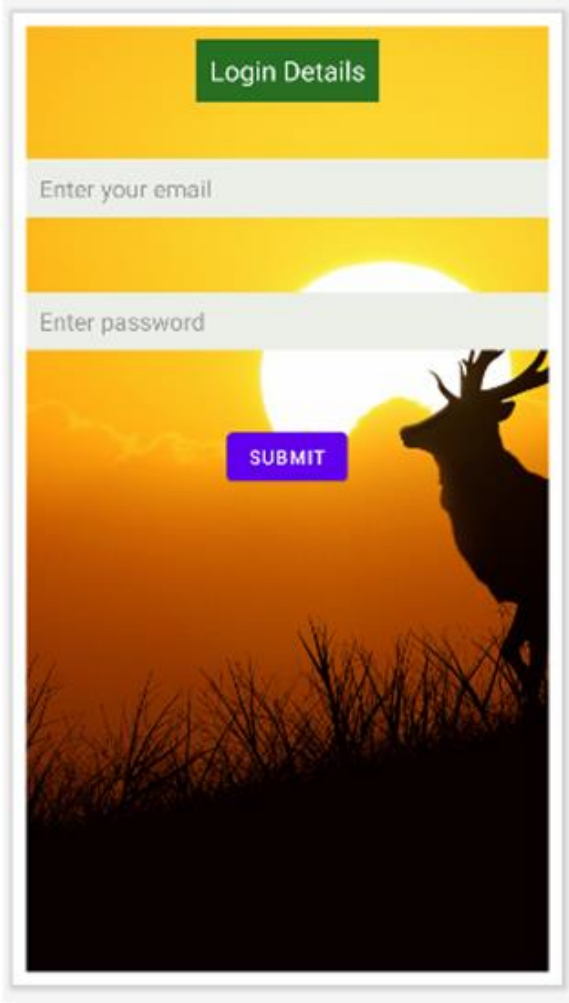
OR

```
public class MainActivity extends AppCompatActivity {

}
```

AppCompatActivity is a base class for activities that wish to use some of the newer platform features on older Android devices. Some of these backported features include:

- **Using the action bar,** including action items, navigation modes and more with the setSupportActionBar(Toolbar) API.
- **Built-in switching between light and dark themes** by using the Theme.AppCompat.DayNight theme and AppCompatDelegate.setDefaultNightMode(int) API.
- **Integration with DrawerLayout** by using the getDrawerToggleDelegate() API.


Login Details
Enter your email
Enter password
SUBMIT

Services

Services perform background tasks without providing UI.
A service is a component that runs in the background to perform long-running operations.

A service is implemented as a subclass of Service class as follows –

```
public class MyService extends Service {

}
```

Broadcast Receivers

Broadcast receivers perform an action in response to message from some other component.
Broadcast Receivers simply respond to broadcast messages from other applications or from the system.

A broadcast receiver is implemented as a subclass of BroadcastReceiver class and each message is broadcaster as an Intent object.

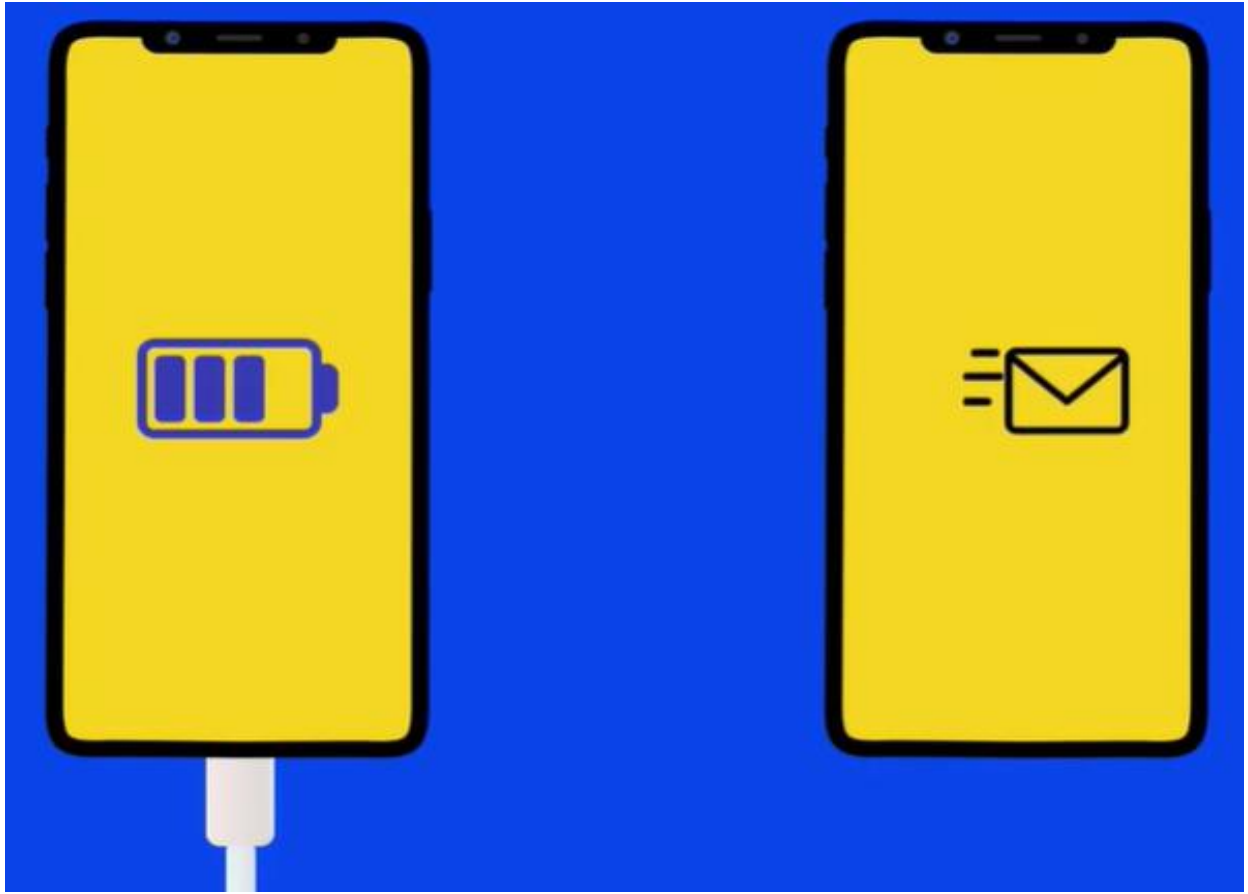
```
public class MyReceiver extends BroadcastReceiver

{

public void onReceive(context,intent){}

}
```

When device starts charging or when it receives messages our Android system sends broadcast on that particular event.



Content Providers

It is an object used to work with data.

A content provider component supplies data from one application to others on request. Such requests are handled by the methods of the ContentResolver class. The data may be stored in the file system, the database(SQlite),web server or somewhere else entirely.

A content provider is implemented as a subclass of ContentProvider class and must implement a standard set of APIs that enable other applications to perform transactions.

```
public class MyContentProvider extends ContentProvider {

public void onCreate(){}

}
```

Additional Components

Fragments : Represents a portion of user interface in an Activity.

Views : UI elements that are drawn on-screen including buttons, lists forms etc.

Layouts : View hierarchies that control screen format and appearance of the views.

Intents : Messages wiring components together.

Resources : External elements, such as strings, constants and drawable pictures.

Manifest : Configuration file for the application.

Additional Components

Intent :

It mainly used to perform following task :

- Start an activity
- Start a service
- Delivery a broadcast
- Display a Webpage etc.