Introduction to Android

MNNIT Computer Coding Club

Objective

To make you answer the following questions yourselves:

1. Why you learn Android?
2. What actually is Android?
3. How to develop your first Android app?
4. What is an Activity?
5. What is a layout?

Reference

1. Head First Android Development: A Brain-Friendly Guide.
2. https://developer.android.com
3. https://www.androidhive.info/
4. Lots of online tutorial ;)

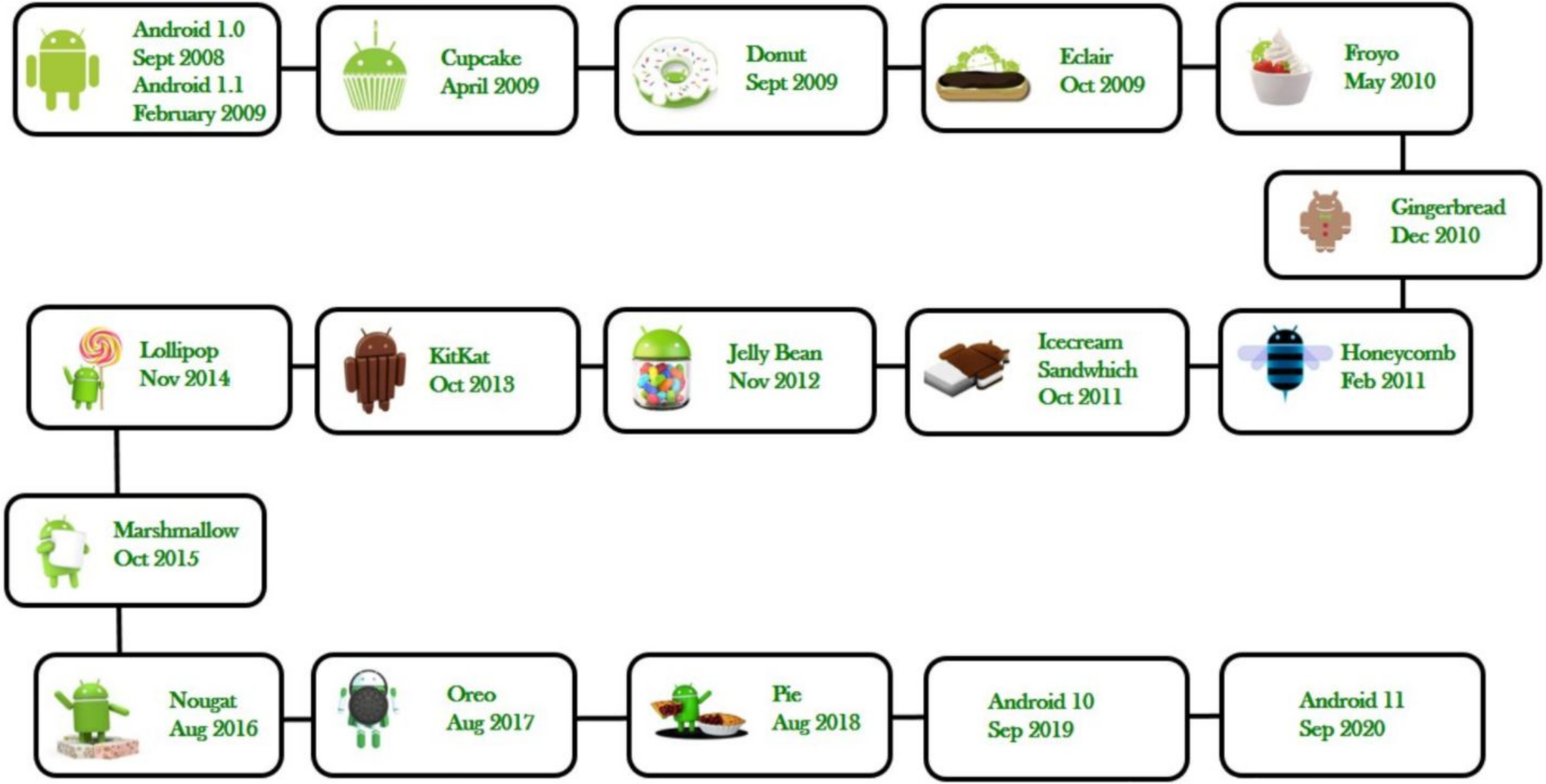
Introduction to Android

According to Wikipedia, Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets.

It is free and open source software; its source code is known as Android Open Source Project (AOSP), which is primarily licensed under the Apache License.

History of Android

The story of Android dates back to 2003 when **Andy Rubin, Rich Miner, Nick Sears, and Chris White** co-founded a start-up ***Android Inc.*** in **Palo Alto, California**. However, the company was later faced with the insufficiency of funds which brought Google into the picture. Google could sense the potential the product carried within and sealed a deal worth $50 Million to acquire Android in 2005. All the four Co-founders soon moved to the **Googleplex** to continue to develop the OS further under their new owners. The first public Android Beta Version 1.0 was finally published on **5th November 2007**.



Android 1.0
Sept 2008
Android 1.1
February 2009

Cupcake
April 2009

Donut
Sept 2009

Eclair
Oct 2009

Froyo
May 2010

Gingerbread
Dec 2010

Honeycomb
Feb 2011

Icecream
Sandwhich
Oct 2011

Jelly Bean
Nov 2012

KitKat
Oct 2013

Lollipop
Nov 2014

Marshmallow
Oct 2015

Nougat
Aug 2016

Oreo
Aug 2017

Pie
Aug 2018

Android 10
Sep 2019

Android 11
Sep 2020

Installation

Find the installation instructions here:

https://cc-mnnit.github.io/2021-22-Classes/Android/2021_05_07_AndroidClass-1/Installation-Instructions.html

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.

There are following four main components that can be used within an Android application −

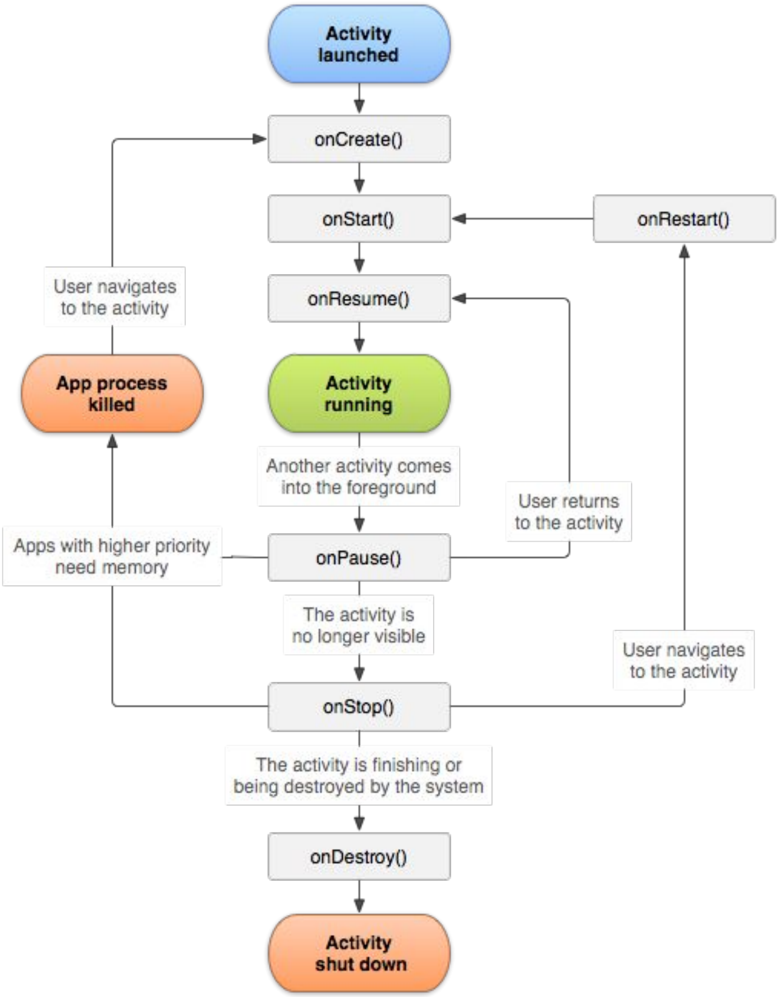
1. Activities.
2. Services.
3. Broadcast Receivers.
4. Content Providers.

Android Activities

An Activity is an application component that provides a screen with which users can interact in order to do something.

If you have worked with C, C++ or Java programming language then you must have seen that your program starts from main() function. Very similar way, Android system initiates its program with in an Activity starting with a call on *onCreate()* callback method.

There is a sequence of callback methods that start up an activity and a sequence of callback methods that tear down an activity as shown in the Activity life cycle diagram on next slide.



Activity launched

onCreate()

onStart()

onRestart()

User navigates to the activity

onResume()

App process killed

Activity running

Another activity comes into the foreground

User returns to the activity

Apps with higher priority need memory

onPause()

The activity is no longer visible

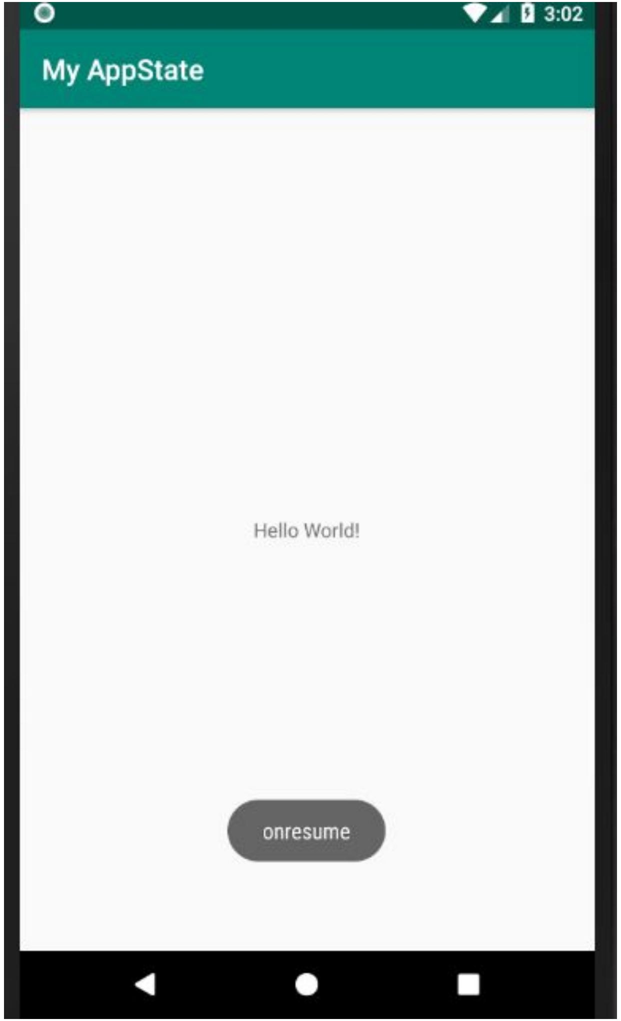
User navigates to the activity

onStop()

The activity is finishing or being destroyed by the system

onDestroy()

Activity shut down



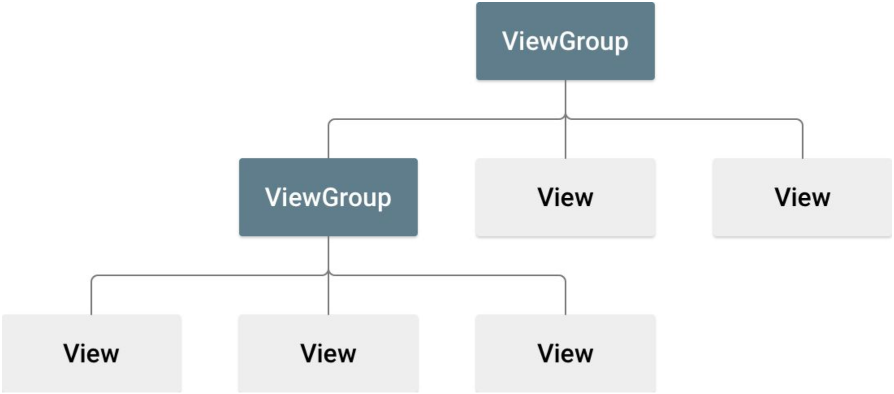
My AppState

Hello World!

onresume

Layouts

A layout defines the structure for a user interface in your app, such as in an activity. All elements in the layout are built using a hierarchy of View and ViewGroup objects. A View usually draws something the user can see and interact with. Whereas a ViewGroup is an invisible container that defines the layout structure for View and other ViewGroup objects.



Layouts continue...

The View objects are usually called "widgets" and can be one of many subclasses, such as Button or TextView. The ViewGroup objects are usually called "layouts" can be one of many types that provide a different layout structure, such as LinearLayout or ConstraintLayout.

You can declare a layout in two ways:

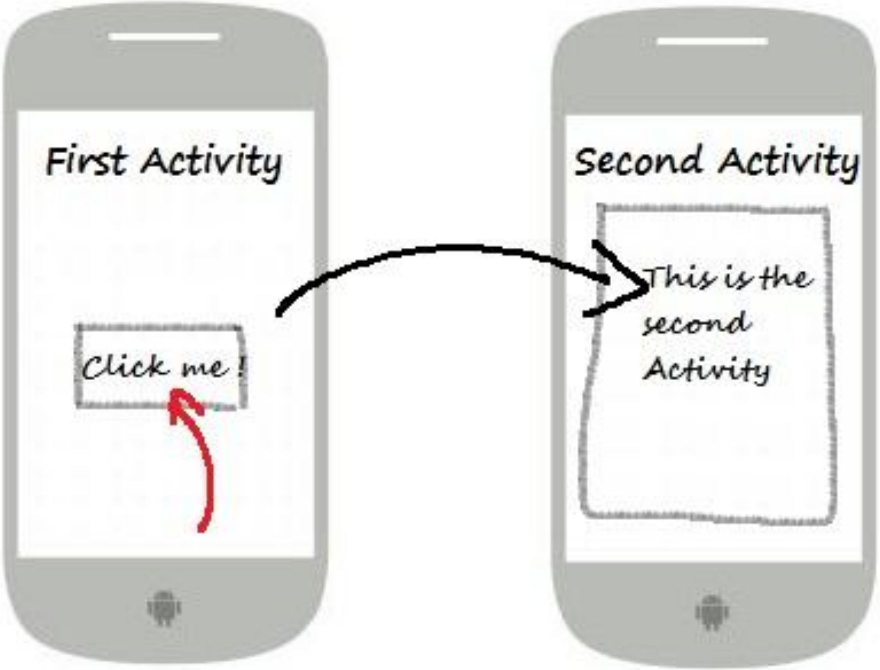
1. Declare layout elements in XML.
2. Instantiate layout elements at runtime.

First Android App

Let’s now together develop your first Android app.

Home Assignment

Study about R.java file.



Next we want to learn how to do that? For this we need to learn about Intent

Intent

An `Intent` is a messaging object you can use to request an action from another app component. Although intents facilitate communication between components in several ways, there are three fundamental use cases:

1. Starting an Activity.
2. Starting a Service.
3. Delivering a broadcast.

Intent Types

1. **Explicit** - It specify which application will satisfy the intent, by supplying either the target app's package name or a fully-qualified component class name. You'll typically use an explicit intent to start a component in your own app, because you know the class name of the activity or service you want to start.
2. **Implicit** - They do not name a specific component, but instead declare a general action to perform, which allows a component from another app to handle it. For example, if you want to show the user a location on a map, you can use an implicit intent to request that another capable app show a specified location on a map.

Explicit Intent

```
Intent intent = new Intent(ActivityOne.this, ActivityTwo.class);
startActivity(intent);
```

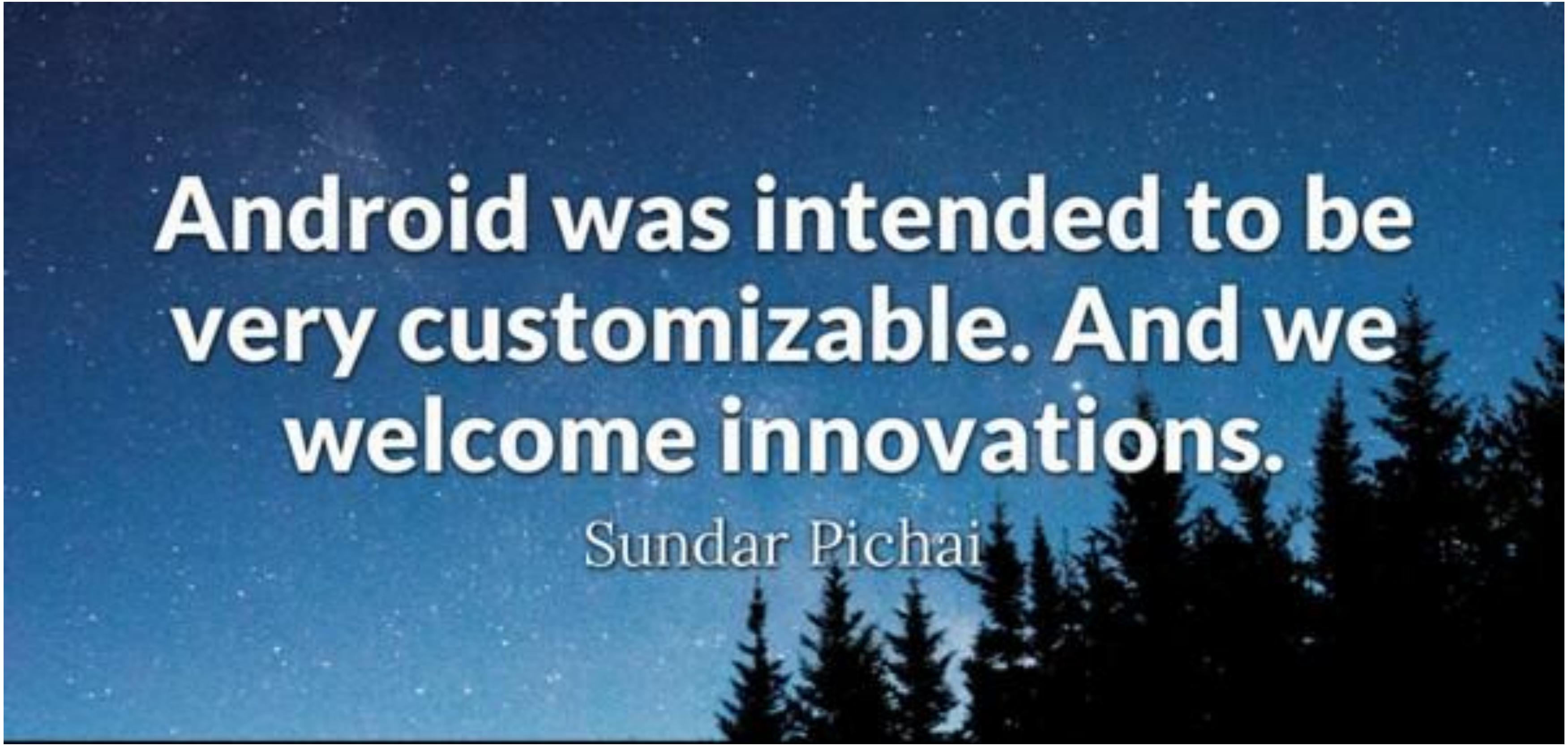
1. The Context in Android is actually the context of what we are talking about and where we are currently present. Here first parameter is context.
2. Second parameter is where we want to go next.

Home Assignment

Study about Metaprogramming. Identify where we have used this.

JAVA PROGRAMMERS
WEAR GLASSES

BECAUSE THEY
DON'T C#



Android was intended to be very customizable. And we welcome innovations.

Sundar Pichai

Thank You.