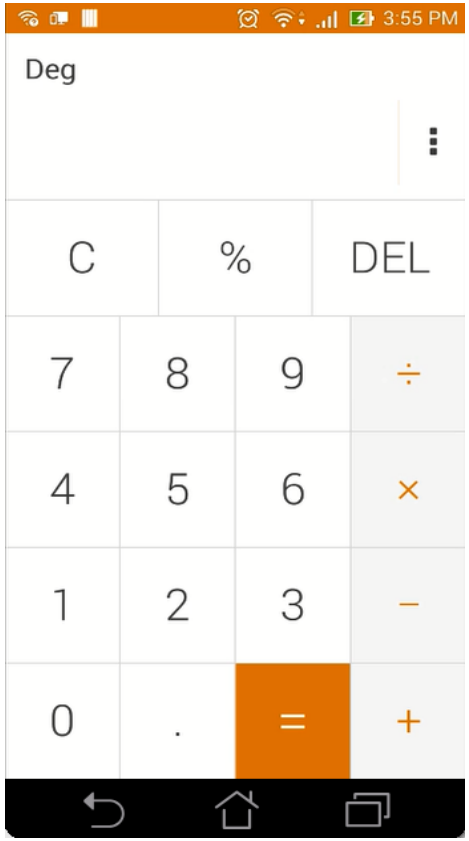
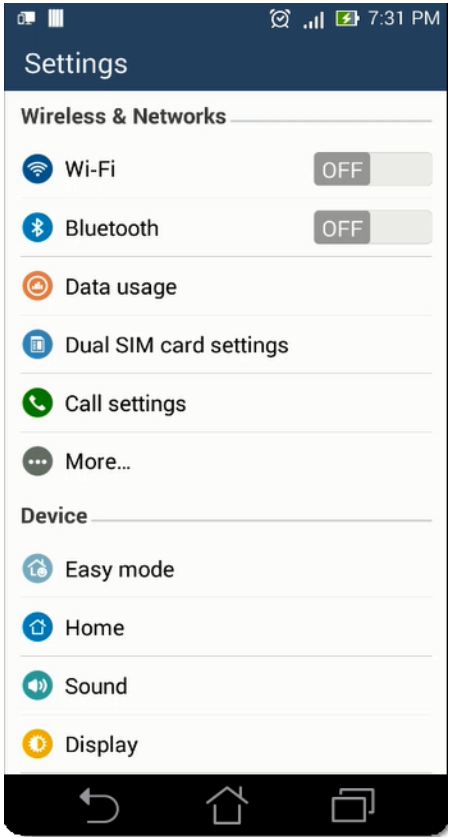
Activity and its Lifecycle

Application Fundamentals

What is an Activity?

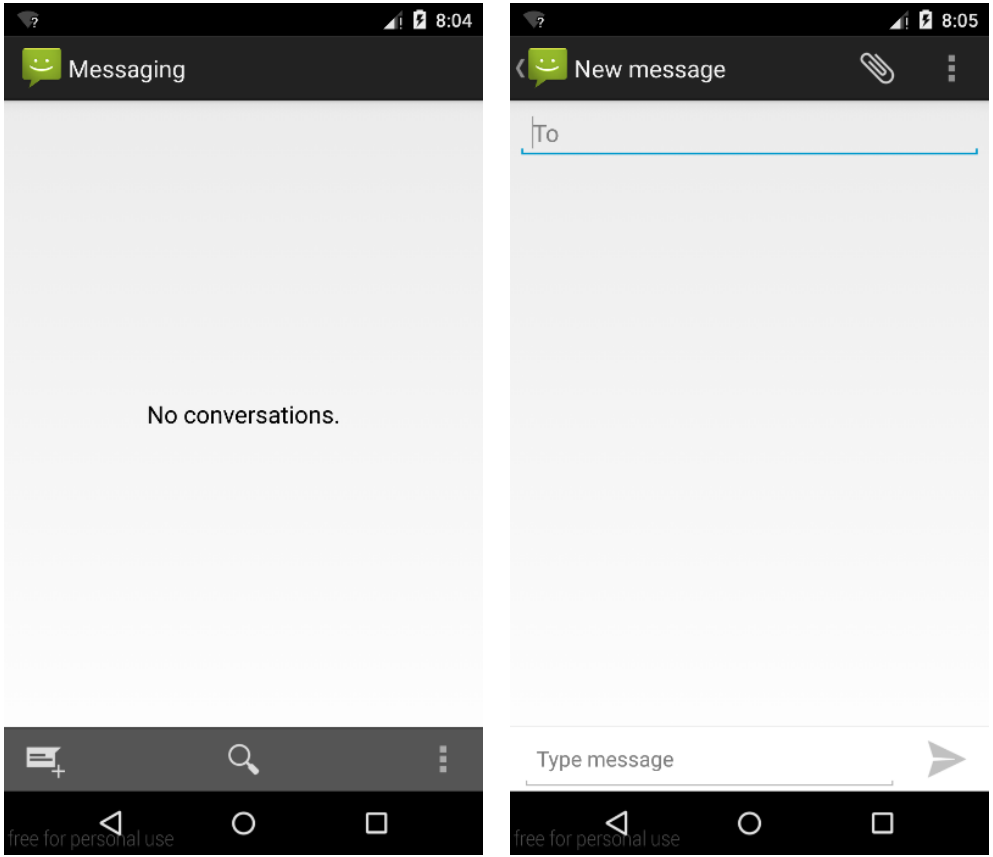




An application component that provides a screen

Draws its UI on its window

What is an Activity?



Every app has 1 main activity and other activities.

An app can start any Activity belonging to itself or other apps subject to certain conditions.

When a new Activity starts, the previous Activity is stopped and added to a stack known as BackStack.

How do I know when my Activity is visible to the user/paused/stopped?

Through special methods - Lifecycle Callback Methods

Why do I need these methods? What do I do with them?

- Simple - Define how your app behaves

Game must be paused when a call is on-going.

What is a callback method?

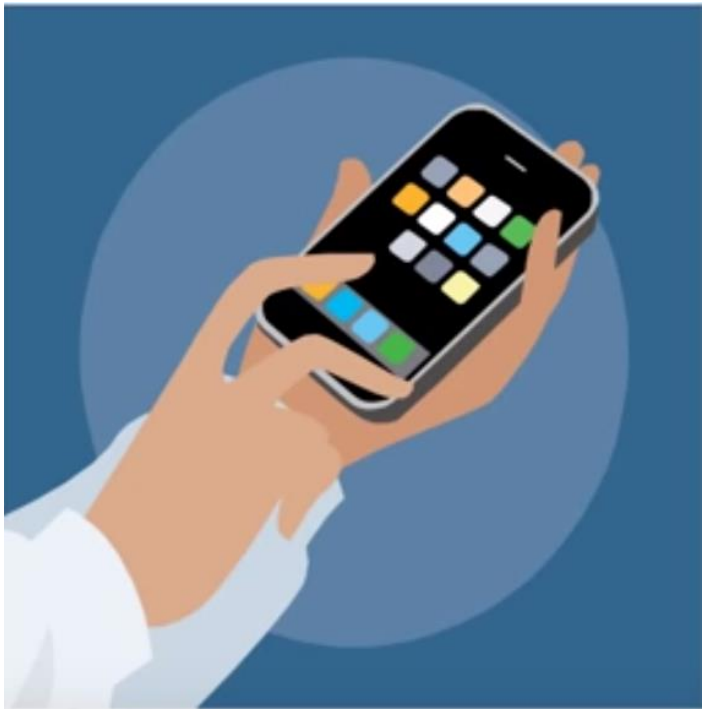
Android OS calls certain methods on your Activity class to notify whether your app is currently running or not.

Just like JVM calling public static void main

General Guidelines

1. Don’t do heavy processing or network consuming operations when user is currently away from your app.
2. App should not crash when another app is started.
3. Don’t lose the user’s progress or session data.

Activity Lifecycle Methods

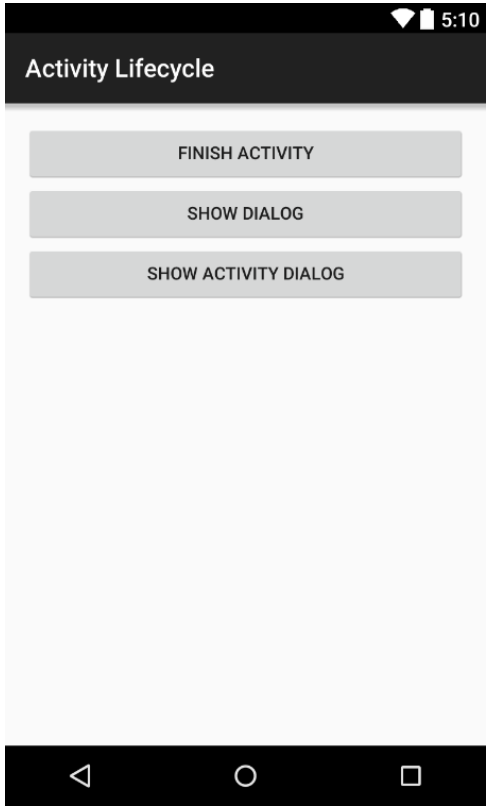


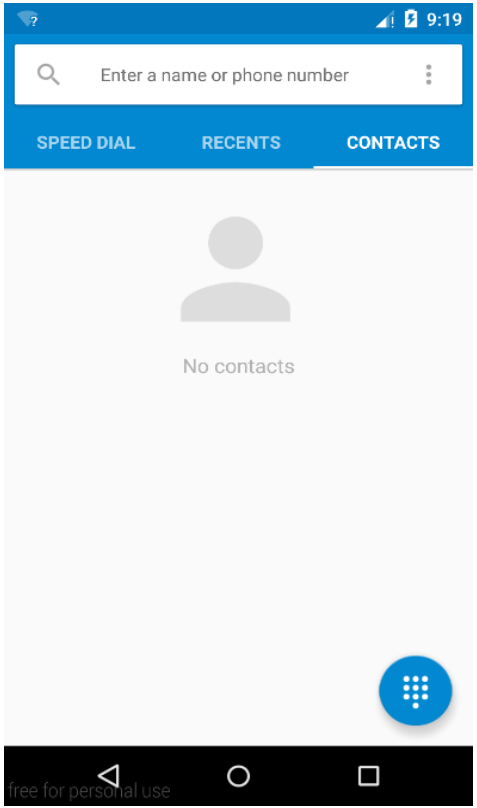
onCreate()

onStart()

onResume()

Activity Lifecycle Methods

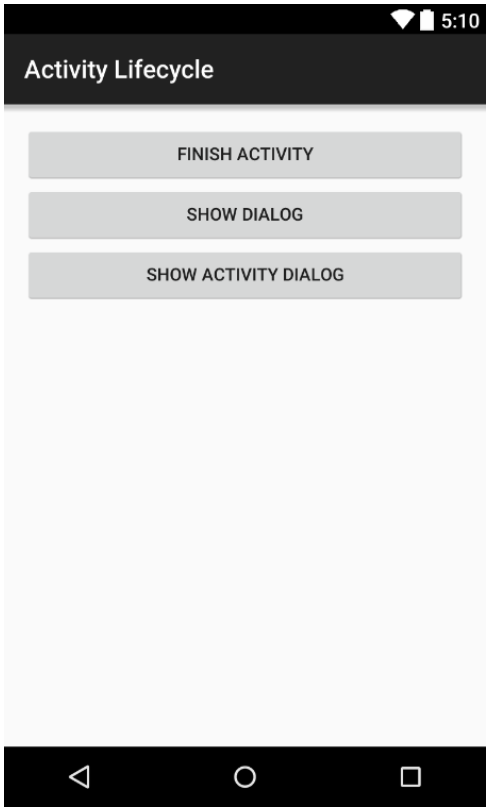




onPause()

onStop()

Activity Lifecycle Methods



onRestart()

onStart()

onResume()

Activity Lifecycle Methods

onDestroy()

Logcat

Used for debugging purposes

Print different messages to Logcat using android.util.Log class

Log.d(String tag, String message)

Log.d(“LIFECYCLE”, “onCreate was called”);

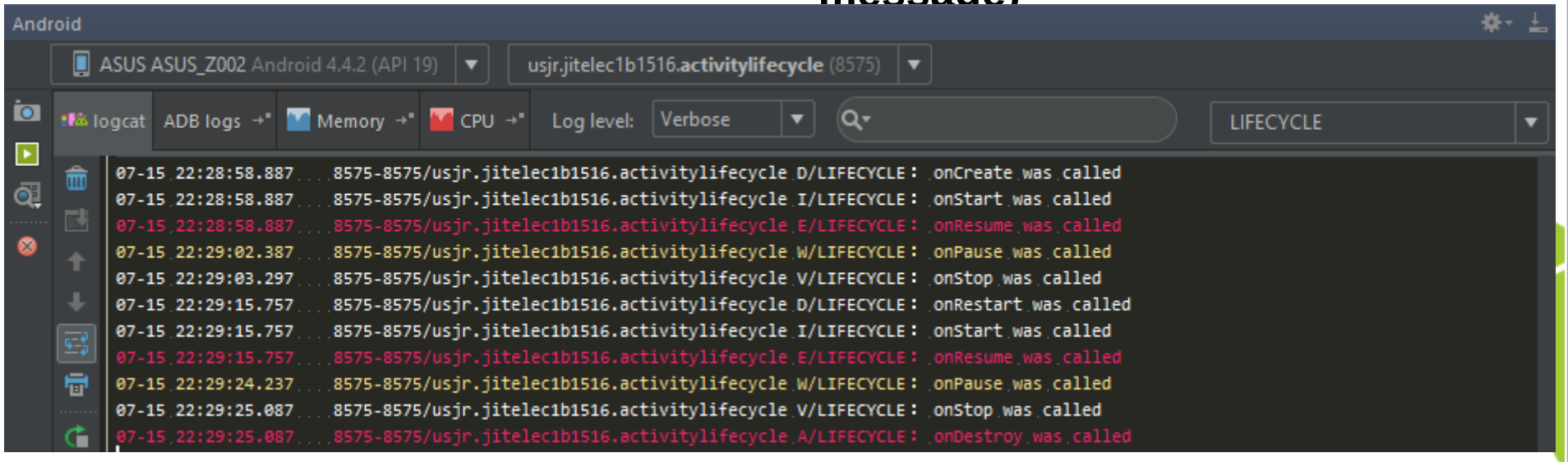
Logcat

Logging uses memory and CPU
You can log messages from real devices too.

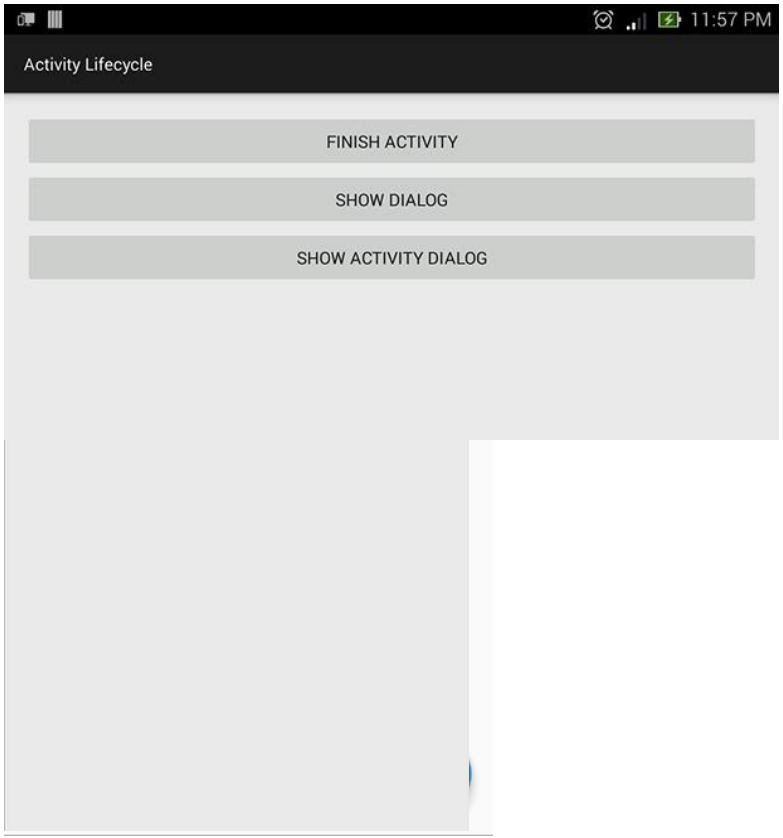
Log.i(String tag, String message)

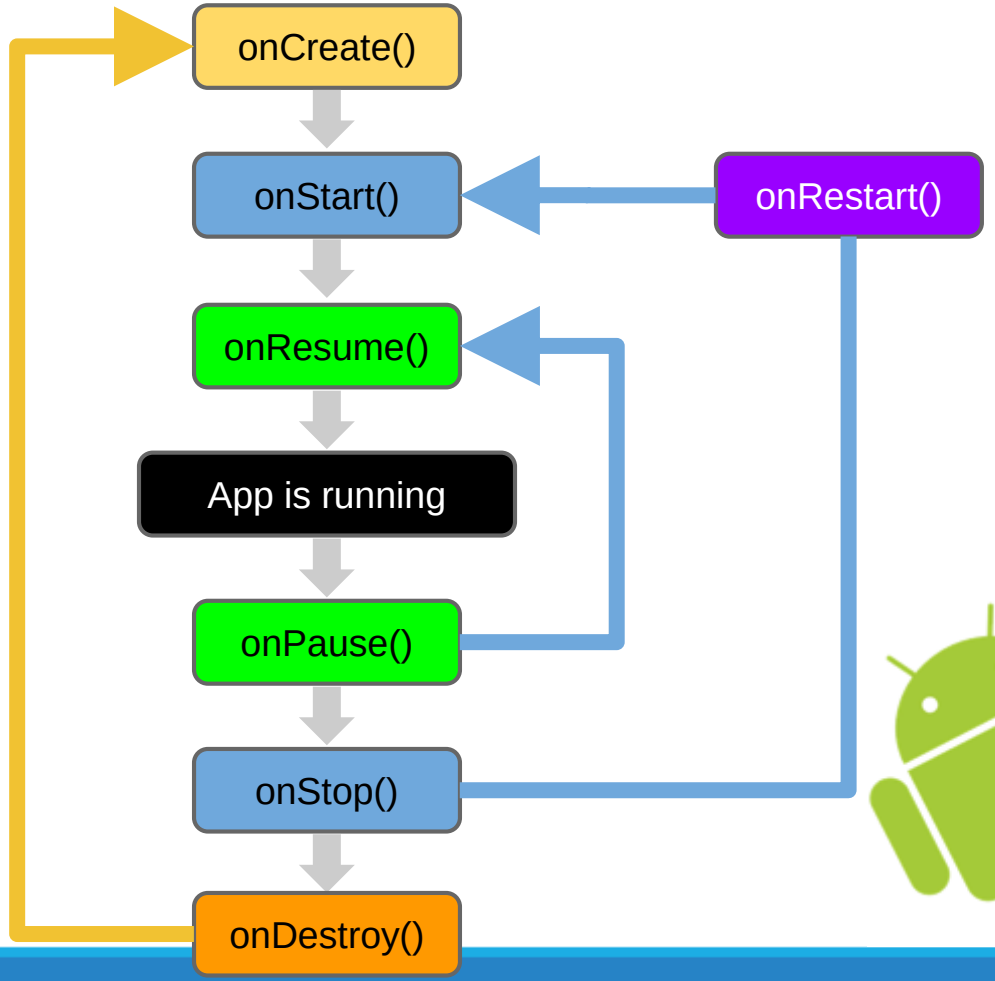
Log.e(String tag, String message)

Log.w(String tag, String message)



Activity Lifecycle





End of Slide