

## Activity

El elemento de activity en Android Studio se entiende como una pantalla que renderiza componentes como 'button', 'Textview', 'Image'. En Java cuando creamos un proyecto nuevo sin actividades Predefinidas, se nos genera un archivo llamado 'Main activity' que contendrá las instrucciones para dicha actividad.

### Ciclo de vida de una app

Lo más importante de una actividad es su ciclo de vida los cuales se clasifican en:

`onCreate()`, `onStart()`, `onResume()`, `onPause()`, `onStop()`, `onRestart()` y `onDestroy()`.

`onCreate()` es el que se ejecuta siempre al inicio de una actividad, en este definimos la interfaz de la actividad, este método solo es invocado una vez así que sería útil para definir variables de inicio.

`onStart()` aquí es donde la actividad se comienza a mostrar al usuario, pero es breve este método antes de continuar con `onResume()`; en esta actividad

se ejecuta entrando en primer plano e interactúa con el usuario con la actividad, o sea la aplicación está corriendo. También están los métodos para cuando la aplicación pasa a segundo plano o se pausa.

`onStop()` la aplicación se congela pero mantiene las variables en pantalla, `onPause` la aplicación pasa a estar parcialmente oscurecida como cuando presionamos la multi tarea antes de pasar a otra app abierta.

`onRestart()` cuando la actividad reanuda operaciones pasando rápidamente por `onStart()` y luego `onResume()`.

onDestroy() se ejecuta cuando la aplicación se cierra, aquí un paso útil para ejecutar sería limpiar la memoria RAM del equipo de los posibles objetos que creamos durante la ejecución de este y cerrar conexiones a BDD en la nube que sigan abierta tras post-ejecución.

## Layouts

El diseño de aplicaciones android es parecida a la de un sitio web Pero en vez de utilizar HTML, utilizamos XML que es más estricto en su sintaxis lo cual es idóneo para android ya que no permite el paso a diversos errores que se podrían provocar. Para nuestras apps existen diferentes capas para la acomodación de nuestros componentes que funcionan como un <Views> en un sitio web <div> que usualmente es un contenedor. Pues tenemos los siguientes

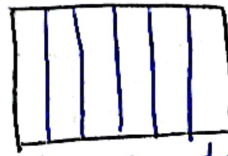
- Linear Layout
- Relative Layout
- Absolute Layout
- Table Layout
- Frame Layout

explicando los 3 principales son:

Comenzando con Linear Layout es el más sencillo de utilizar ya que acomoda los elementos de forma vertical o de forma horizontal. Si elegimos la forma vertical listará los elementos a modo de lista una sola columna y horizontal distintas columna pero una sola fila



vertical



horizontal

## Relative Layout

Este contenedor es relativo, la posición de los elementos tienen "movilidad", si queremos comparar con algo conocido podría ser un elemento HTML con CSS de position "relative".

## Frame Layout

Este elemento junto a Linear Layout son los más fáciles de comprender, este elemento es un frame que siempre renderiza los elementos en la esquina superior izquierda.