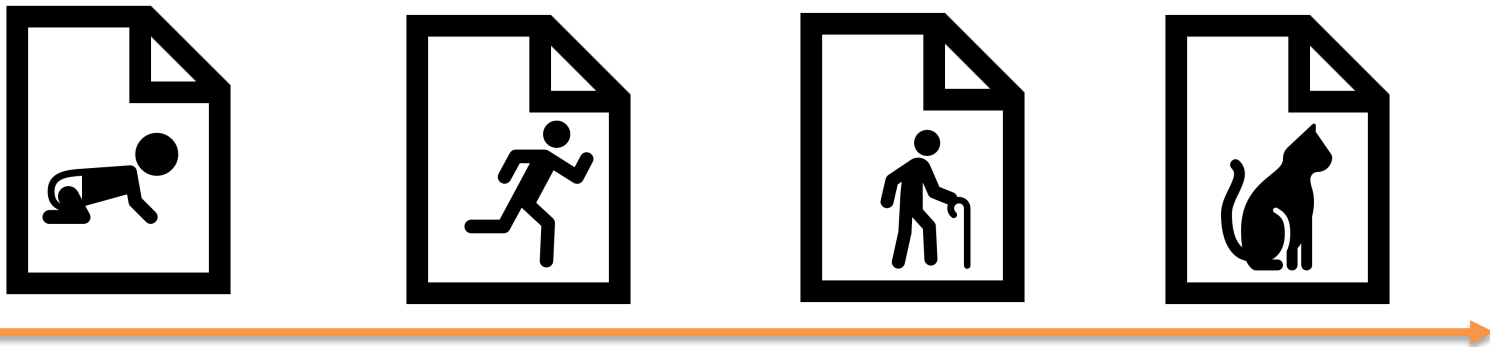


Ciclo de Vida de una Actividad



Desarrollo para plataformas móviles



Borja Martin Herrera
Borja.herrera@u-tad.com

Ciclo de vida de una actividad



- Las actividades tienen un ciclo de Vida que se inicia cuando se crea y muestra la actividad y que termina cuando se abandona la actividad.

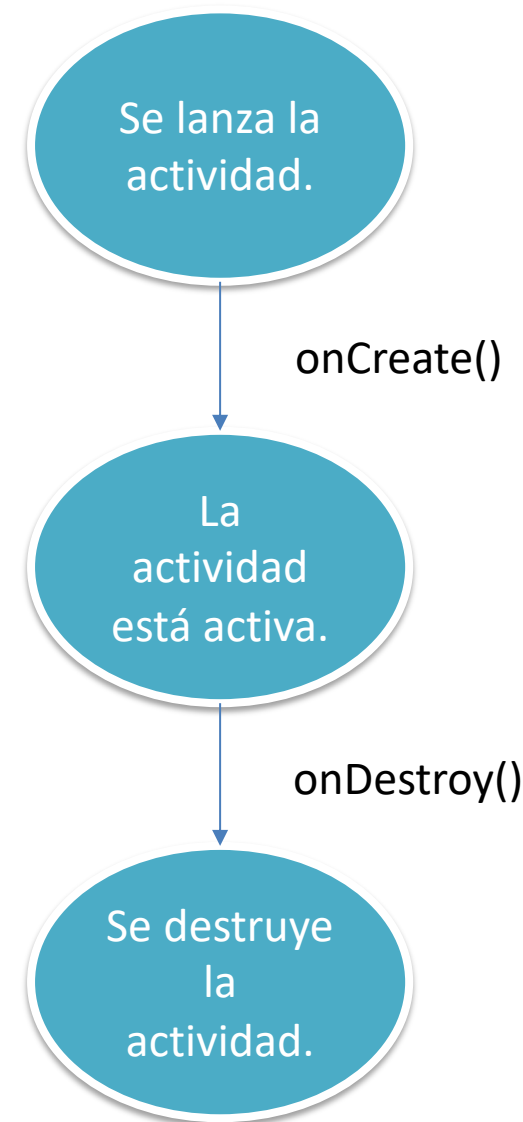
Ciclo de vida de una actividad

- Activa (Running): La actividad es visible y tiene el foco.
- Visible (Paused): La actividad es visible pero no tiene el foco (Hay otra actividad superpuesta pero que deja parte visible porque no ocupa toda la pantalla o tiene alguna parte transparente).
- Parada (Stopped): Cuando la actividad no es visible para el usuario.
- Destruída (Destroyed): Cuando la actividad termina al invocarse el método `finish()`, o es matada por el sistema.



Ciclo de vida de una actividad

- Tenemos una serie de métodos que se ejecutan cuando la actividad “avanza” en su ciclo de vida.
- `onCreate` y `onDestroy` son dos de los métodos más importantes del ciclo de vida.
- **`onCreate`** se lanza según se crea la actividad.
- **`onDestroy`** se ejecuta justo antes de que la actividad sea destruida.



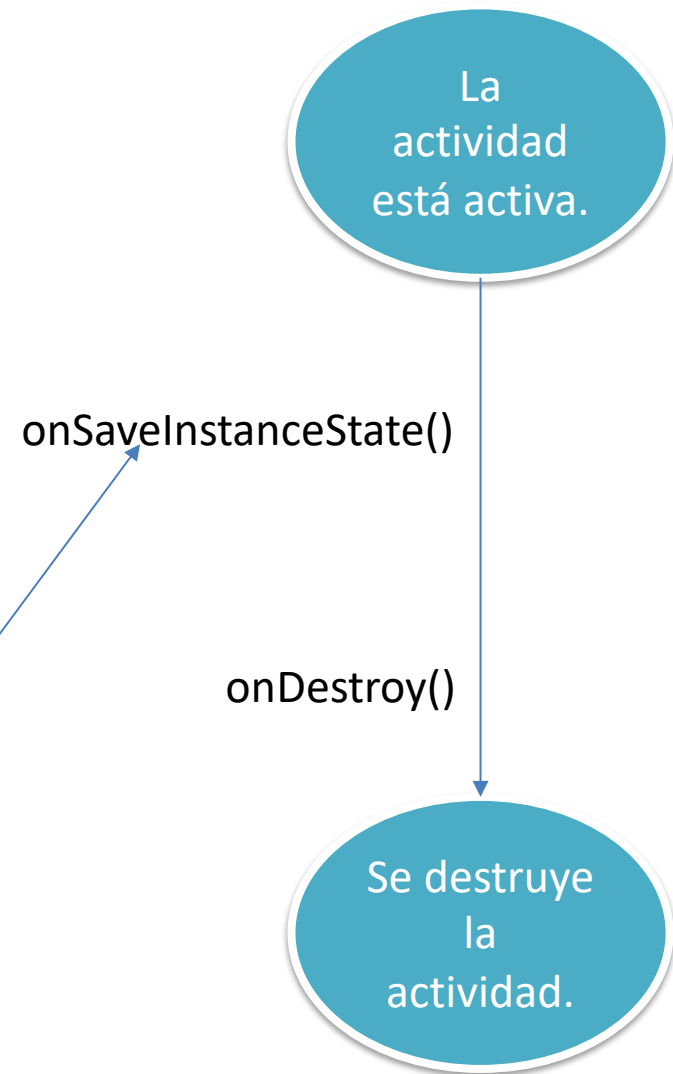
Ciclo de vida de una actividad

- onCreate:
 - Se ejecuta cuando la actividad es llamada por primera vez durante la ejecución de la app.
 - Todavía no es visible el Layout para el usuario. Es aquí donde elegimos el Layout que vamos a utilizar.
 - Una actividad puede ser destruida y puede volver a crearse durante una misma ejecución.
 - Cambiar la orientación de la app destruye y vuelve a crear la actividad.
- onDestroy:
 - Se llama cuando la actividad es destruida.
 - Por ejemplo al pulsar el botón de Back en una actividad o al usar multi ventana.
 - También si se llama al método finish() por alguna razón durante la actividad.



Ciclo de vida de una actividad

- Cuando cambiamos la actividad, la información es destruida.
- A veces podemos querer que la información se mantenga al volver a la actividad.
- Esto suele ocurrir cuando cambiamos la orientación del dispositivo por ejemplo.
- Hay un método que sirve para que podamos guardar los datos antes de que esto ocurra



Ciclo de vida de una actividad

- El método que tenemos que sobre escribir es onCreate con su parámetro savedInstanceState.

```
override fun onSaveInstanceState(outState: Bundle) {  
    super.onSaveInstanceState(outState)  
    outState.putInt("numero_guardado")  
    outState.putString("nombre_guardado")  
}
```

Ciclo de vida de una actividad

- Ahora queda leer esos datos cuando se vuelve a cargar la actividad.
- Esto lo hacemos en el método onCreate (). El objeto *Bundle* nos llega por parámetro.

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    binding = ActivityMainBinding.inflate(layoutInflater)  
    setContentView(binding.root)  
  
    savedInstanceState?.getInt( key: "numero_guardado")  
    savedInstanceState?.getString( key: "nombre_guardado")  
  
}
```


Ejemplo de App

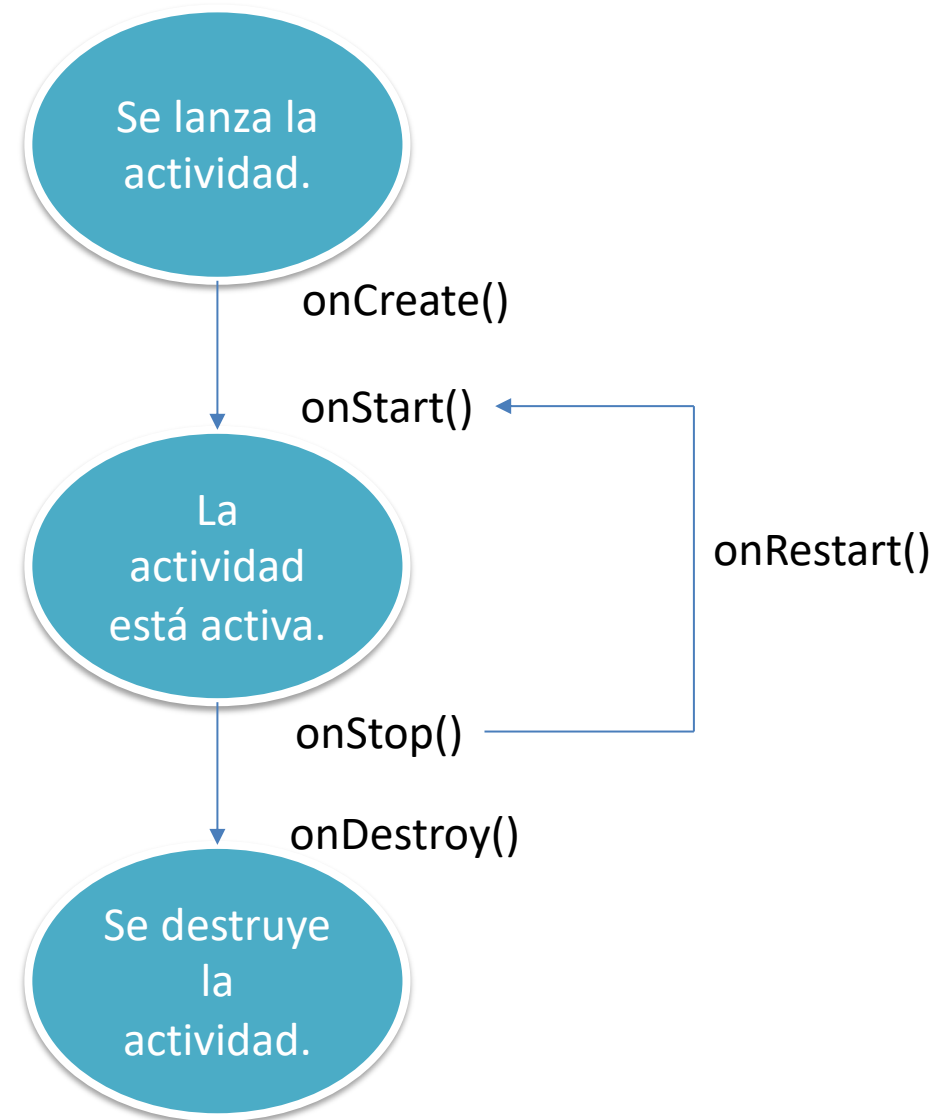
Ejercicio: Realizar un contador.

- Es cada vez que se pulse el botón de incremento o decremento se suma uno a un contador.
- Hay un textView que muestra el valor del contador todo el rato
- Existe un botón de reset que lleva el contador a 0
- El contador debe mantenerse siempre que se gire el móvil



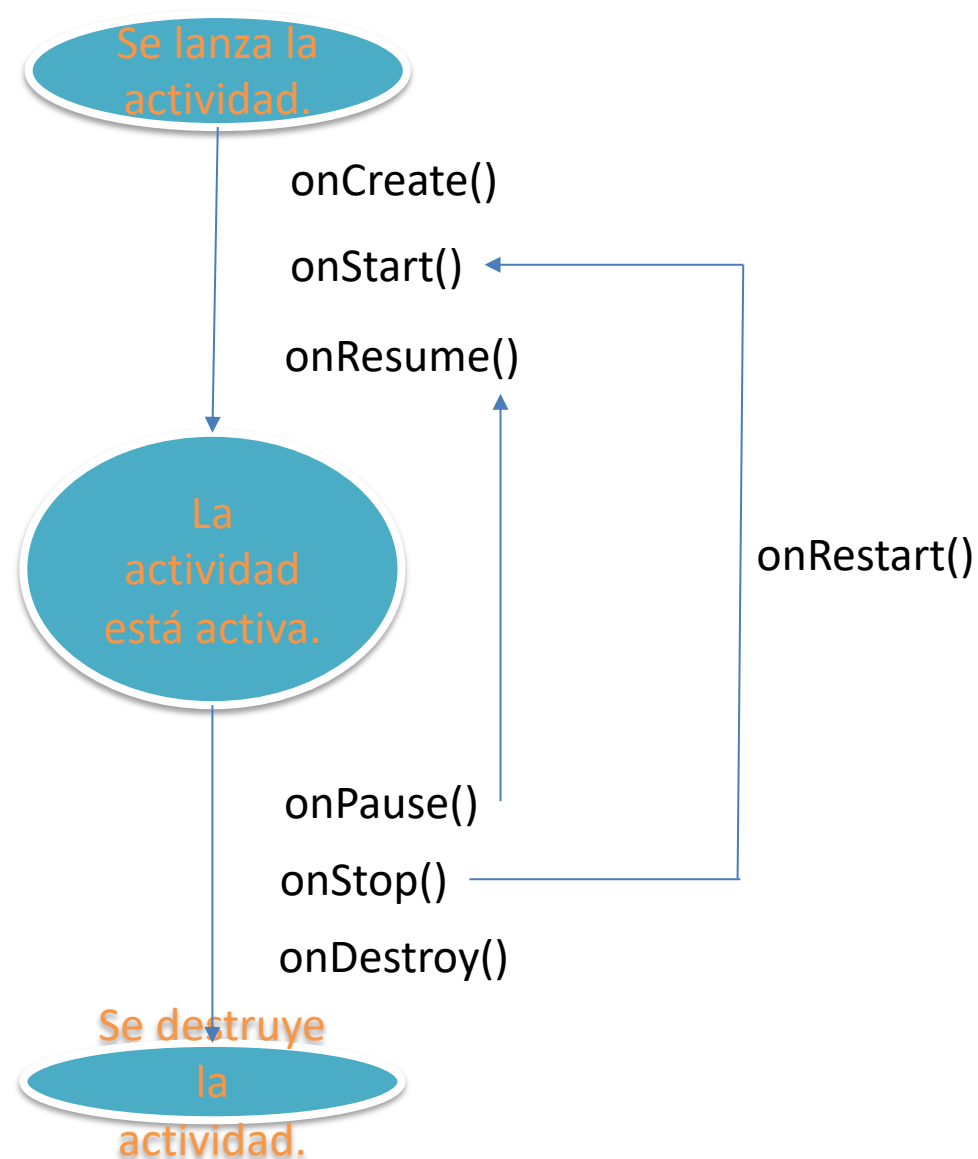
Ciclo de vida de una actividad

- Existen más métodos que se ejecutan durante el ciclo de vida de la actividad.
- onStart, onStop, onRestart.
- **onStart** se llama cuando el Layout se muestra al usuario. Esto puede producirse por ejemplo porque la app pasa de estar en segundo plano para estar en foco.
- **onStop** se llama cuando la actividad deja de mostrarse al usuario (no tiene porque destruirse, puede ser por un cambio de actividad). Si va a ser destruida, es el método que se llama justo antes de onDestroy.
- **onRestart** se llama cada vez que la app es visible al usuario después de onStart. Tiene que haberse ejecutado el onStop() antes, por lo que a diferencia de onStart, no se llama la primera vez que se ejecuta la app.

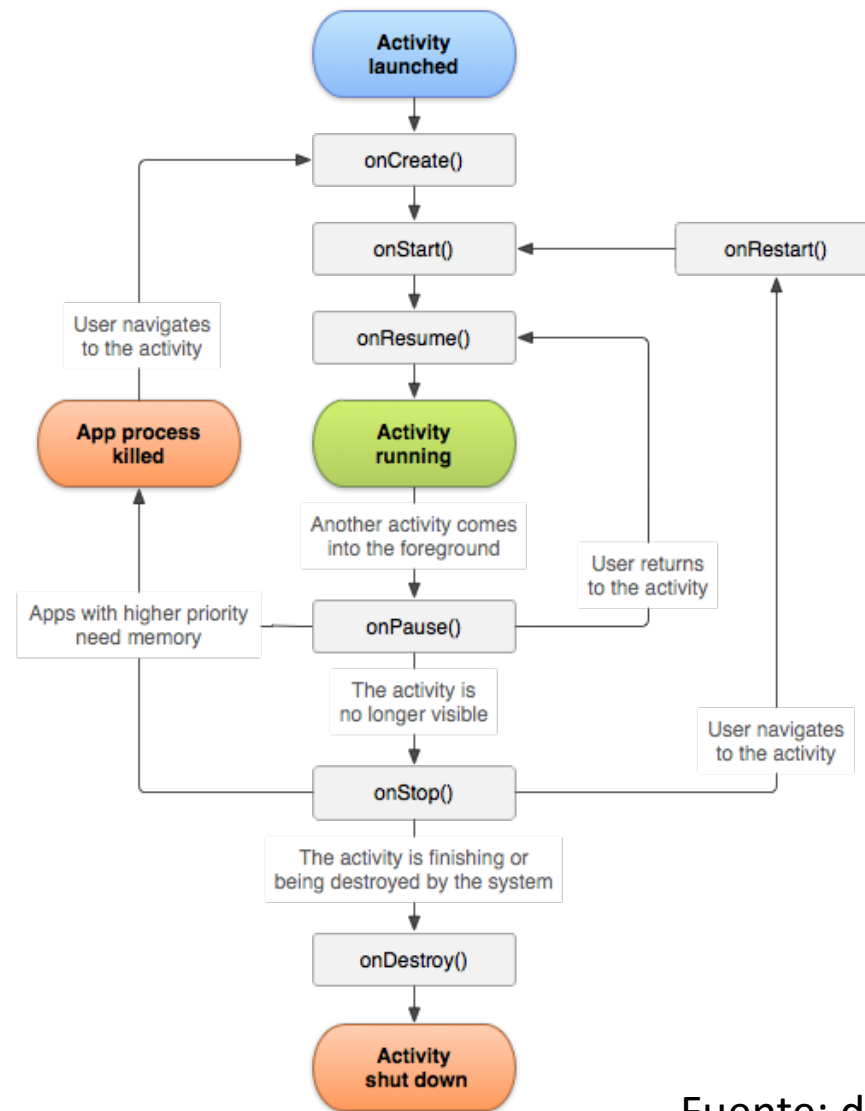


Ciclo de vida de una actividad

- Por último, nos queda ver dos métodos que se ejecutan cuando la actividad es tapada parcialmente, es decir, se ve pero no tiene el foco porque el foco lo tiene un menú emergente u otra actividad de otra app.
- onPause es llamado cuando la actividad pierde el foco.
- onResume es llamado cuando la actividad vuelve a tener el foco.



Resumen: Ciclo de vida de una actividad



Fuente: developer.Android.com

Ejemplo de App

Ejercicio: Realizad a app que sea capaz de recibir cualquier tipo de callback de su ciclo de vida.

Usad el método Toast cuando esto ocurra para mostrar un mensaje por pantalla.



Ejemplo de App

Ejercicio: Realizad un cronometro que mantenga la cuenta aunque la actividad pierda el foco, deje de ser visible o sea destruida por un cambio de orientación.

Podéis usar la clase *Chronometer*

