

Tutorial: **App Activity Life Cycle**

Activity

Single, focused thing that a user can do

Usually each screen has its own activity

An application may have multiple screens, hence multiple activities

An application runs in its own Linux process

Activity States

Active:

Running activity in foreground of screen

Paused:

Lost focus, but still visible

Retains all state information

In extreme memory situations may be killed

Stopped:

Not visible

Retains all state information

Often will be killed

Killed

To see this in action, run the following example.

Step 1: Create a new project and call it LifeCycle

Step 2: Paste the following code into MainActivity.java

```
package edu.itu.lifecycle;
```

```
import android.app.Activity;
```

```
import android.os.Bundle;
```

```
import android.widget.TextView;
```

```
public class MainActivity extends Activity {
```

```
    int paused = 0;
```

```
int killed = 0;
int stopped = 0;
TextView text;
```

```
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    if (savedInstanceState != null) {
        paused = savedInstanceState.getInt("paused");
        killed = savedInstanceState.getInt("killed");
        stopped = savedInstanceState.getInt("stopped");
    }
}
```

```
text = new TextView(this);
text.setText("Paused: " + paused + " stopped: " + stopped +
" killed " + killed);
setContentView(text);
}
```

```
protected void onResume() {
    super.onResume();
    text.setText("Paused: " + paused + " stopped: " +
stopped + " killed " + killed);
}
```

```
protected void onStart() {
    super.onStart();
    text.setText("Paused: " + paused + " stopped: " +
stopped + " killed " + killed);
}
```

```
protected void onStop() {
```

```
        stopped++;
        super.onStop();
    }

    protected void onPause() {
        paused++;
        super.onPause();
    }

    protected void onDestroy() {
        killed++;
        super.onDestroy();
    }

    protected void onSaveInstanceState(Bundle outState) {
        outState.putInt("paused", paused);
        outState.putInt("killed", killed);
        outState.putInt("stopped", stopped);
    }
}
```

Step 3: Compile and run the project. Observe what is going on.