James Vann P. Lopez 08/28/2025

BSIT-3D ITELEC1

Assignment #1

**Research about the Activity lifecycle, get information, code on how to use it [onCreate]**

**[onStart], [onResume], [onPause], [onStop], [onDestroy]**

In Android, an Activity represents a single screen with a user interface (like a page). The Activity Lifecycle is the way Android manages different states of this screen — from when it is first created, becomes visible, interacts with the user, and is finally destroyed.

1. onCreate(Bundle savedInstanceState)

**Purpose:** Called once when the activity is first created.

**Best for:** UI initialization, setting up listeners, loading data.

**Analogy:** Like building the house — you set up the walls, furniture, and decoration.

**Example Code:**

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

Log.d(TAG, "onCreate: Activity is being created");

// Initialize UI elements, listeners

}

2. onStart()

* **Purpose**: Called every time the activity becomes visible to the user.
* **Best for**: Restarting things paused/stopped in onStop(), UI refresh.
* **Analogy**: You open the door — the house is visible now.

**Example Code**:

@Override

protected void onStart() {

super.onStart();

Log.d(TAG, "onStart: Activity is becoming visible");

// Start animations, refresh data if needed

}

3. onResume()

* **Purpose**: Activity is now in the foreground, and the user can interact with it.
* **Best for**: Start real-time things like camera, location updates, audio.
* **Analogy**: You're inside the house and actively using everything.

**Example Code:**

@Override

protected void onResume() {

super.onResume();

Log.d(TAG, "onResume: Activity is interactive now");

// Resume any paused tasks, like animations, media

}

4. onPause()

* **Purpose**: Called when another activity is taking focus, but your activity is still partially visible.
* **Best for**: Pause ongoing operations, save unsaved changes, pause video, etc.
* **Analogy**: Someone knocked on your door, you pause what you were doing to answer.

**Example Code**:

@Override

protected void onPause() {

super.onPause();

Log.d(TAG, "onPause: Activity is partially obscured");

// Pause media playback, stop sensors or cameras

}

5. onStop()

* **Purpose**: Activity is completely hidden.
* **Best for**: Heavy clean-up, release resources like broadcast receivers.
* **Analogy**: You leave the house, everything turns off.

**Example Code**:

@Override

protected void onStop() {

super.onStop();

Log.d(TAG, "onStop: Activity is no longer visible");

// Release resources, save state, unregister receivers

}

6. onDestroy()

* **Purpose**: Called before the activity is destroyed. May be due to:
  + Finishing the activity (finish())
  + System is temporarily destroying it (e.g. low memory)
* **Best for**: Final cleanup.
* **Analogy**: The house is demolished.

**Example Code:**

@Override

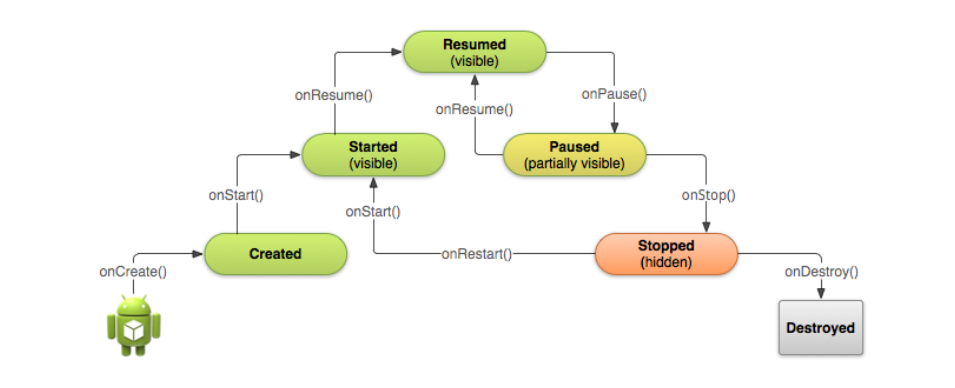
protected void onDestroy() {

super.onDestroy();

Log.d(TAG, "onDestroy: Activity is about to be destroyed");

// Final cleanup, free up memory

}

**Figure 1: Activity Life Cycle**