**Activity Life Cycle**

Activity life cycle include six stages: **onCreate()**

This is the first callback and called when the activity is first created.

**onStart()**

This callback is called when the activity becomes visible to the user. **onResume()**

This is called when the user starts interacting with the application.

**onPause()**

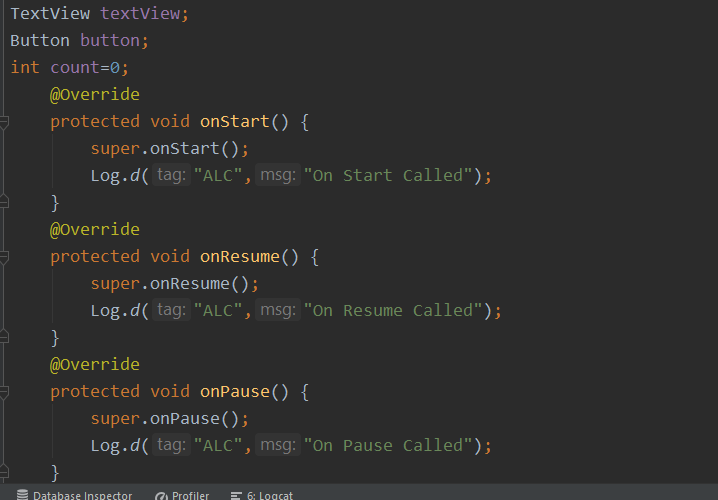
The paused activity does not receive user input and cannot execute any code and called when the current activity is being paused and the previous activity is being resumed.

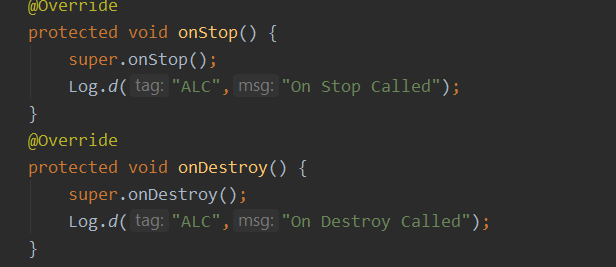
**onStop()**

This callback is called when the activity is no longer visible.

**onDestroy()**

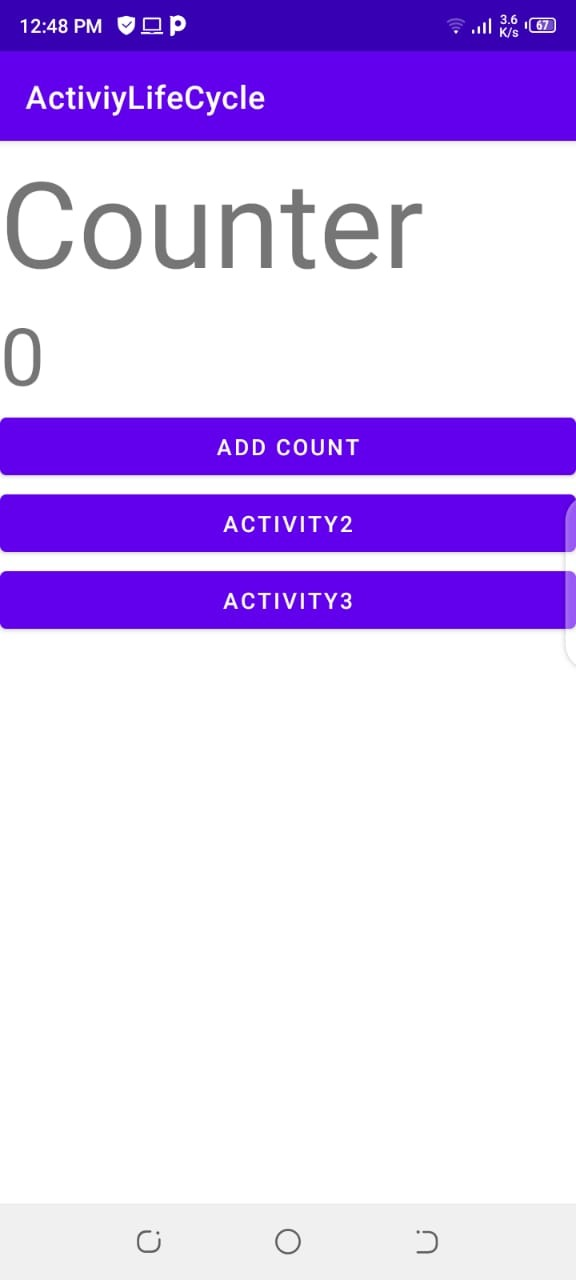
This callback is called before the activity is destroyed by the system.





This is how we add activity life cycle functions.

But Now we will understand its working.



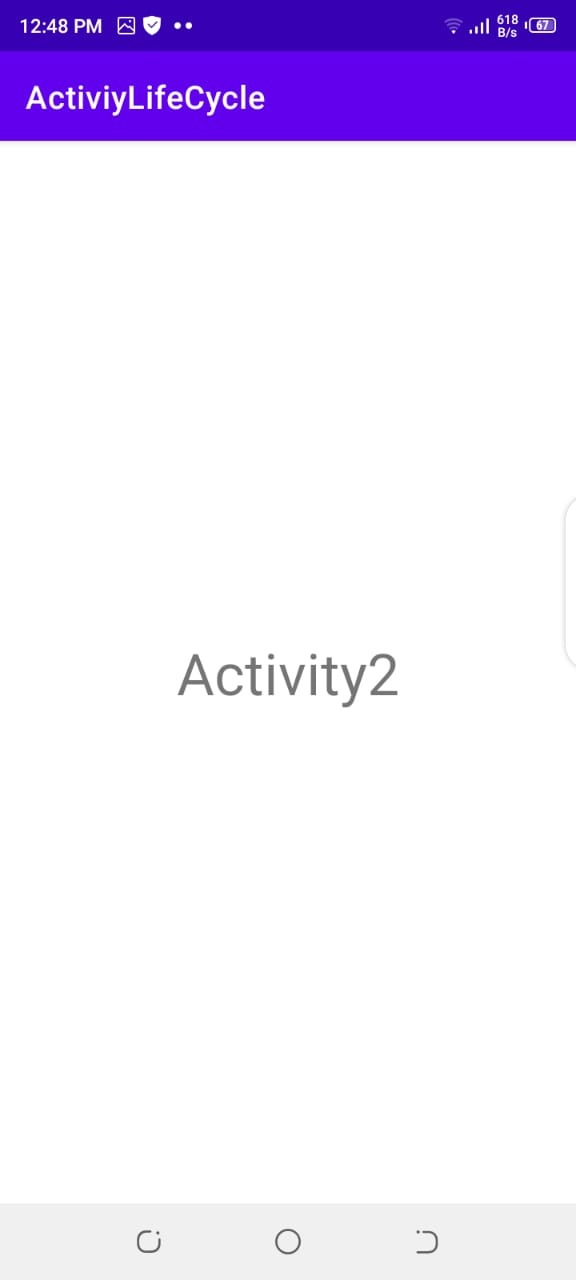
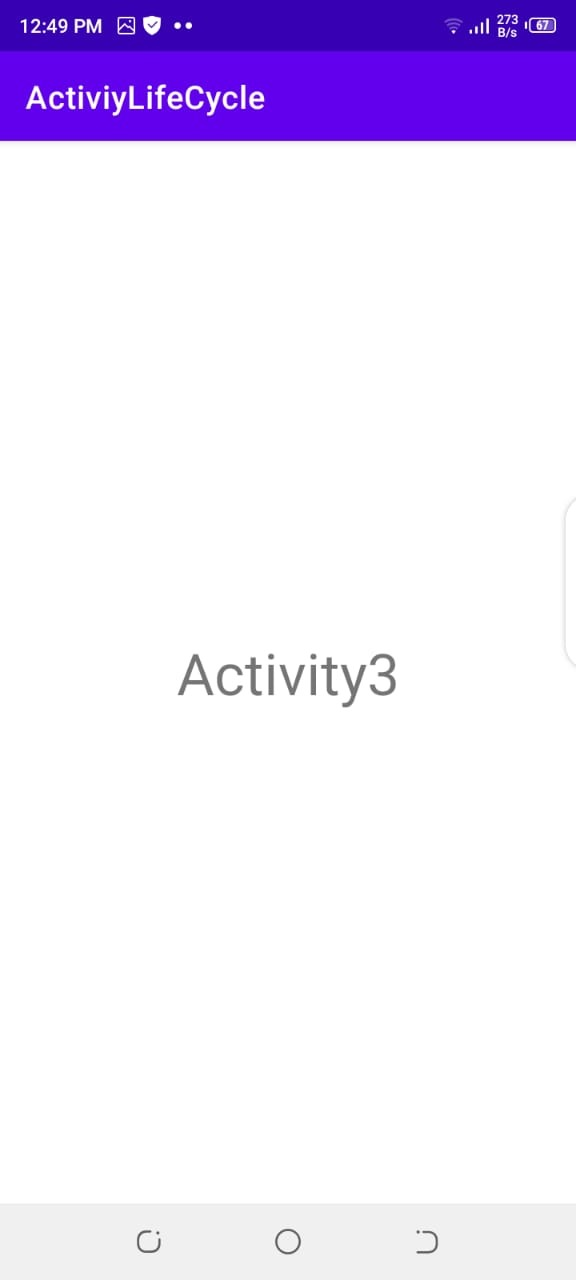
This is the main activity page .on that page in our log on

**create(), on start()** has been called. Now when we click on activity 2 its on create(), **on start() and on resume()** function will be called and the main activity **on stop()** function will be called.

And on activity 2 if we pressed bac button it will destroy activity 2. And it will consume main activity state because its destroy function is not called.

So the conclusion is state remains maintain when we move between activities.

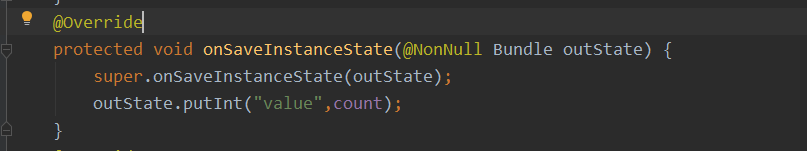
Same is the case with activity 3.

Now lets move forward and consider a scnerio in which state wil not be save.

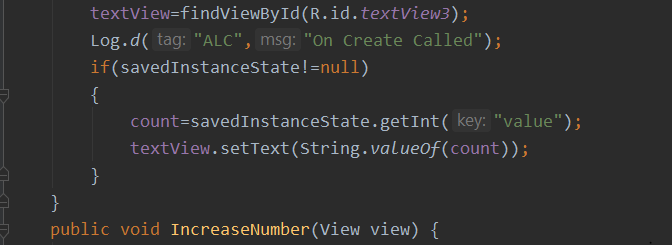
In main activity page when we do count and will roate the screen the count will be removed because on rotate first activity is destroyed and dthen moves to new avtivity .

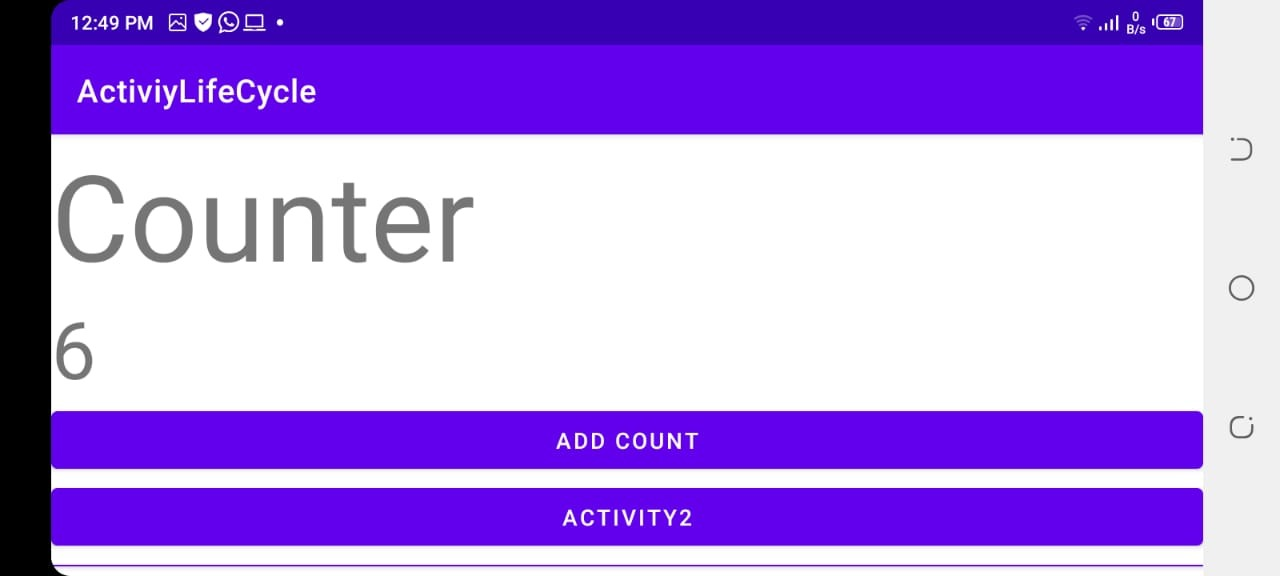
So in order to save our state we have to do somw work.



This is the function we need to save state of count .

So we will do this and in on create function we will check if state is saved or not.





This is how the things look like .