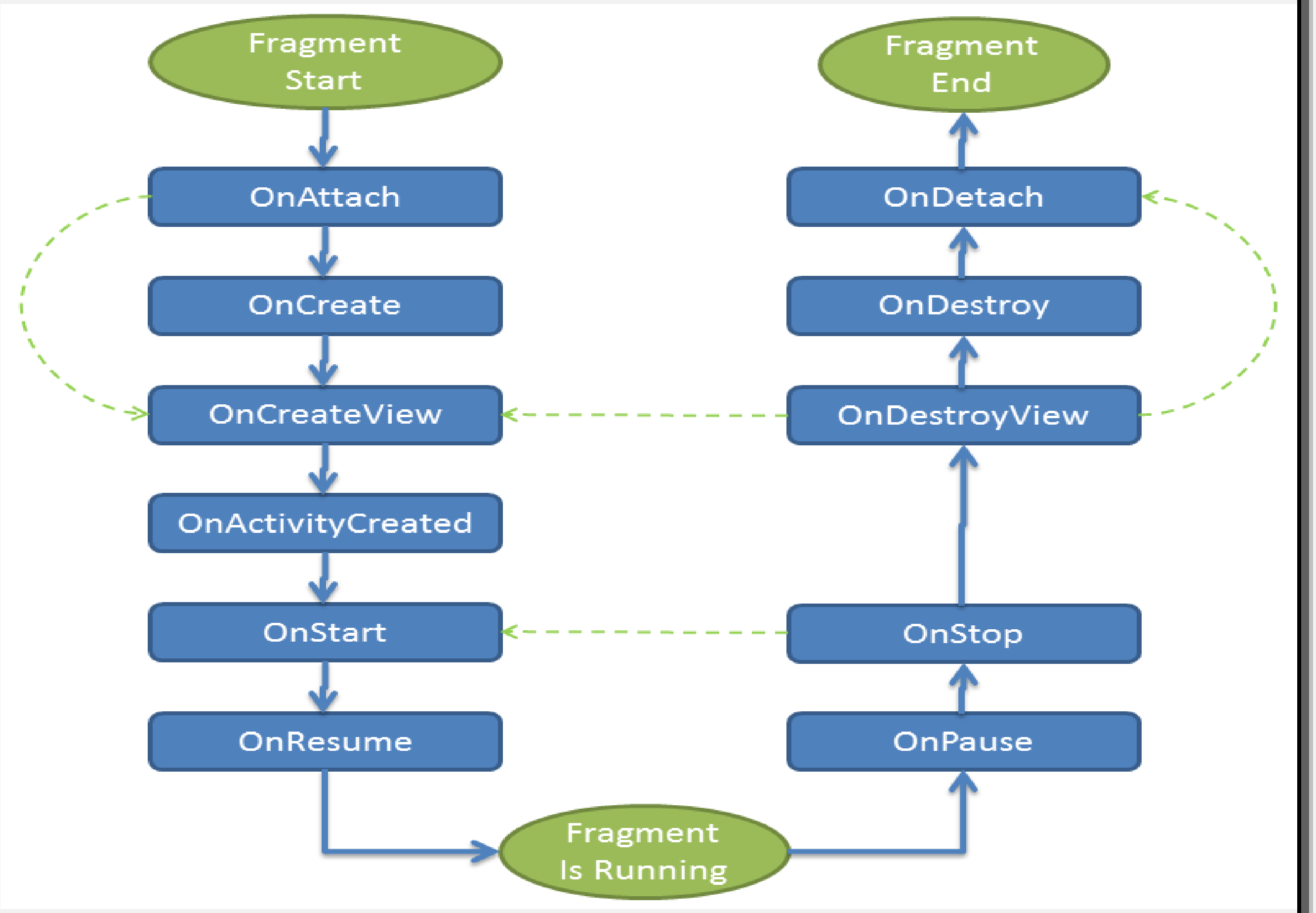
Fragments in Android

A fragment represents a conduct or a part of user interface in an activity. You can integrate more than one fragment in a single activity to build a multi-pane UI and reuse a fraction in multiple activities.

you may consider a fragment as a modular phase of an activity, which has its personal lifecycle, receives its very own enter events, and which you may upload or dispose of even as the interest is walking (kind of like a "sub interest" that you may reuse in exclusive sports).



Following are crucial points about fragment –

- A fragment has its very own layout and its own behaviour with its personal life cycle callbacks.
- You may add or do away with fragments in an interest whilst the interest is strolling.

- You may integrate multiple fragments in an unmarried hobby to construct a multi-plane UI.

A fraction can be utilized in multiple activities.

- Fragment life cycle is closely associated with the life cycle of its host interest which means when the pastime is paused, all the fragments available in the hobby will also be stopped.
- A fraction can put into effect a behaviour that has no person interface aspect.
- Fragments were added to the Android API in Honeycomb model of Android which API version eleven.
- You create fragments via extending Fragment elegance and you can insert a fraction into your hobby layout by way of affirming the fragment inside the hobby's format document, as a <fragment> detail.
- Prior to fragment introduction, we had a difficulty because we will display only a single hobby on the display screen at one given factor in time. So we had been not capable of divide device display and control extraordinary components one by one. However with the introduction of fragment we got greater flexibility and removed the difficulty of having a single activity on the display screen at a time. Now we will have a single pastime however every interest can incorporate of more than one fragments in order to have their own format, occasions and entire life cycle.

Right here is the listing of techniques which you can to override for your fragment magnificence –

1) **onAttach():**The fragment example is related to an interest example. The fragment and the hobby aren't always fully initialized. Generally you get on this approach a connection with the interest which makes use of the fragment for similarly initialization paintings.

2) **onCreate():** The device calls this technique when developing the fragment. You have to initialize critical components of the fragment which you need to keep when the fragment is paused or stopped, then resumed.

3) **onCreateView():** The system calls this callback while it is time for the fragment to attract its person interface for the primary time. to draw a UI on your fragment, you need to go back a View issue from this technique that is the foundation of your fragment's format. you may return null if the fragment does no longer provide a UI.

4) **onActivityCreated():** The onActivityCreated() is referred to as after the onCreateView() method whilst the host hobby is created. Activity and fragment example have been created in addition to the view hierarchy of the activity. At this

point, view may be accessed with the findViewById() method. Example. on this approach you may instantiate objects which require a Context object

5) **onStart():** The onStart() technique is known as once the fragment gets visible.

6) **onResume():** Fragment turns into lively.

7) **onPause():** The device calls this technique because the first indication that the person is leaving the fragment. this is generally where you need to dedicate any modifications that must be endured beyond the modern-day person session.

8) **onStop():** Fragment going to be stopped by using calling onStop()

9) **onDestroyView():**Fragment view will smash after call this technique

10) **onDestroy():** onDestroy() known as to do very last smooth up of the fragment's state however no longer assured to be referred to as by using the Android platform.

How to use Fragments?

- This involves wide variety of simple steps to create Fragments.
- To begin with determine what number of fragments you need to apply in an activity. as an instance permit's we need to apply fragments to address landscape and portrait modes of the device.
- Next based totally on number of fragments, create instructions to be able to amplify the Fragment class. The Fragment class has above noted callback capabilities. you could override any of the functions based totally to your requirements.
- Corresponding to each fragment, you'll need to create layout files in XML record. Those files may have format for the defined fragments.
- Eventually adjust activity document to outline the actual good judgment of changing fragments based totally to your requirement.

Forms of Fragments

- Basically fragments are divided as three ranges as proven underneath.

- **Single body fragments**− single frame fragments are using for hand keep devices like mobiles, right here we can show best one fragment as a view.

- **List fragments** − fragments having special listing view is called as list fragment

- **Fragments transaction**− The usage of with fragment transaction. We are able to flow one fragment to any other fragment.