Android Programming

Trainer: Mr. Rohan Paramane

Fundamental components of Android

- Activity
- Service
- Content Provider
- Broadcast Receiver

- These fundamental components are also called as Pillars of Android

- AppCompat is a set of support libraries which can be used to make the apps developed with newer versions to work with older versions

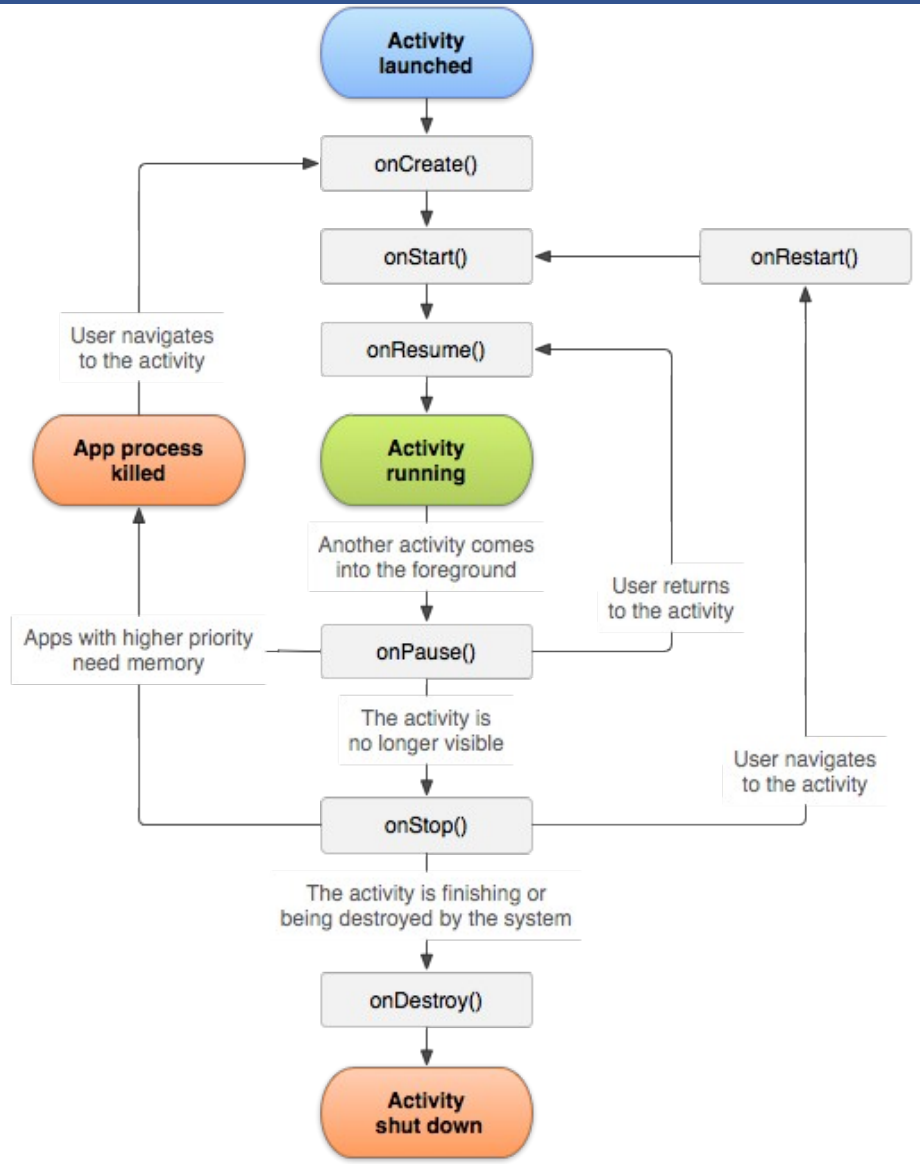
Activity

- The activity is a crucial part of Android app
- Like other programing languages where apps are launched with main() method.
- Here the android system initiates code in an Activity instance by invoking specific callback methods that specify to specific stages of life cycle
- The mobile-app experience differs from its desktop counterpart in that a user's interaction with the app doesn't always begin in the same place.
- For instance, if you open an email app from your home screen, you might see a list of emails. By contrast, if you are using a social media app that then launches your email app, you might go directly to the email app's screen for composing an email.
- The activity class facilitates this
- the activity serves as the entry point for an app's interaction with the user.
- You implement an activity as a subclass of the **Activity** class

Android Layouts

- Linear layout
  - A layout that organizes its children into a single horizontal or vertical row.
- Relative layout
  - Enables you to specify the location of child objects relative to each other (child A to the left of child B) or to the parent (aligned to the top of the parent).
- Frame layout
  - Frame Layout is designed to block out an area on the screen to display a single item.
- Constraint layout
  - Constraint Layout provides you the ability to completely design your UI with the drag and drop feature provided by the Android Studio design editor.

Activity Life Cycle



- To navigate transitions between stages of the activity lifecycle, the Activity class provides a core set of six callbacks
- onCreate() , onStart(), onResume(), onPause(), onStop(), and onDestroy().
- The system invokes each of these callbacks as an activity enters a new state.

- onCreate()
  - In the onCreate() method, you perform basic application startup logic that should happen only once for the entire life of the activity.
- onStart()
  - The onStart() call makes the activity visible to the user, as the app prepares for the activity to enter the foreground and become interactive.
  - For example, this method is where the app initializes the code that maintains the UI.
- onResume()
  - When the activity enters the Resumed state, it comes to the foreground, and then the system invokes the onResume() callback.
  - This is the state in which the app interacts with the user.
  - The app stays in this state until something happens to take focus away from the app.

- onPause()
  - The system calls this method as the first indication that the user is leaving your activity (though it does not always mean the activity is being destroyed).
  - It indicates that the activity is no longer in the foreground (though it may still be visible if the user is in multi-window mode)
- onStop()
  - When your activity is no longer visible to the user, it has entered the Stopped state, and the system invokes the onStop() callback.
  - This may occur, for example, when a newly launched activity covers the entire screen. The system may also call onStop() when the activity has finished running, and is about to be terminated.
- onDestroy().
  - onDestroy() is called before the activity is destroyed. The system invokes this callback either because:
    - the activity is finishing (due to the user completely dismissing the activity or due to finish() being called on the activity), or
    - the system is temporarily destroying the activity due to a configuration change (such as device rotation or multi-window mode)

Thank you!

Rohan Paramane<rohan.paramane@sunbeaminfo.com>