**Android Components**

Android components are fundamental pieces of an Android app, that work together to provide various functionalities such as Activities, Services, Broadcast Receivers and Content providers.

1. **Activities-** These represent the UI of the app, where the user can interact with the app. Each activity typically represents one screen of the app.
2. **Services-** These are the background tasks that run independently of the Ui, and perform long running operations such as downloading files, playing music, or processing data in the background.
3. **Broadcast Receivers-** Even if your app is not running, broadcast receivers will send you the info about your app. For instance, Weather app always doesn’t need to be opened, a broadcast receiver can listen for updates from the system about changes in the weather. When new weather data becomes available, the receiver can notify the app, allowing it to fetch the latest weather information and update the Ui with the updated data.
4. **Content Providers-** They are responsible for the security of your app. If a third-party app, wants to access your app info, such as location, call logs etc. Content providers make sure that if a third-party app is allowed to access your app information or not.

**Services and their Life Cycle**

A service runs in the background, to perform long- running operations without a user interface.

**Lifecycle of Services**

1. **Created-** The service is created when another component such as Activity or another service calls it by calling startService() or bindService().

1. **Started-** Once the service is Created, it goes to Started state, it will run indefinitely, even if the activity or component who calls it is destroyed. We need to explicitly stop it or it should be explicitly stopped by the system, if system is needed to reclaim the resources.
2. **Bound**- Services can also be "bound" to components, such as activities, allowing them to interact by sharing data and method calls.
3. **Destroyed- The** service is destroyed when it's no longer needed or when the system needs to reclaim resources. This can happen when the service is explicitly stopped using **stopService()** or **unbindService()**, or when the component that started or bound to it is destroyed.

**Types of Services**

1. **Foreground-** These are services that have a visible notification to the user, indicating that the service is running. For instance, when you accidentally go to the home screen of your phone, in that case the music player will be visible to the user, in the notification bar.
2. **Background-** These are the services which do not have a visible notification or a user interface to the user. For instance, if you are using an app, it will create some temporary files. We do not need to create a user interface or notification bar, instead, we can create a cleanupService, that will extend the Android Service class, and we can set it to run at particular intervals, such as twice per day, or once per day.
3. **Bound Services-** It means an activity, or the service can communicate with each other. In other words, an activity can set up the service, when to start or when to stop. So, bound service needs to follow up with the activity life cycle, because it is bound to the activity.