* **Activities in Android**

Activities are part of the basic component of the Android application. Users cannot interact with an application on their mobile device without an activity. Activities form the bedrock upon which our mobile application is built. It consists of the layout, text fields, images, and different UI elements which the user can interact with.

Every mobile application you open, the first screen you see is an activity. In Android Studio activities are created using the Java or Kotlin programming language.

Activities are linked through intents, that are used for communication and navigation between different activities. Intents can be used to start a new activity, pass data between activities, or even trigger actions in other components of the app.

This article will provide a detailed explanation of activities, their importance, and activity life cycle along with its methods.

**An activity**

Activity is more like a container for one or more multiple screens in your app. It's not just seen as a screen but also as a unit that users interact with in your application.

Activities contain information about whether a user is currently interacting with the screen or if the activity is in the background.

It also serves as an entry point for your app. We can have multiple screens in an application that are bundled together into an activity.

For example, A user profile page activity can contain other screens like information details of the user.

However, Jetpack Compose has made everything easier. Using the jetpack compose UI we can have only one main activity with multiple screens while staying in the single main activity.

The main activity in Jetpack Compose serves as the entry point of our application. The main characteristic of activities in Android is the **LIFECYCLE** which means at some point our activity is created, destroyed, paused, etc.

**Basic components of an activity**

1. **Layout XML:** the layout XML file represents the user interface of an activity. These files contain view-groups (linear layout, constraint layout) and views such as buttons, images, app bars, etc.

The XML file defines the structure and positioning of these components which the user will interact with like clicking a button to open a new activity.

However, in Jetpack Compose you can define and structure the features of your app inside the main.activity.kt file.

1. **Activity class:** An activity class comprising a single screen with a user interface is represented by an instance(object) of the Activity class in Android.

To create a custom activity you must create a subclass of the Activity class and override its lifecycle methods and other crucial methods to the application's needs.

1. **Life cycle methods:** these methods perform various behaviors in activity and can be overridden in different stages. The methods are:`onCreate()`, onStart(), onResume(), onPause(), onStop(), onDestroy(), etc.