**BROADCAST RECIEVER :**

**Android apps can send or receive broadcast messages from the Android system and other Android apps, similar to the publish-subscribe design pattern.**

**OR**

**Broadcast Receivers can send or receive messages from other applications or from the system itself. These messages can be events or intents.**

**These broadcasts are sent when an event of interest occurs**

**For example: the Android system sends broadcasts when various system events occur, such as when the system boots up or the device starts charging.**

**Apps can also send custom broadcasts, for example, to notify other apps of something that they might be interested in (for example, some new data has been downloaded).**

**Apps can register to receive specific broadcasts. When a broadcast is sent, the system automatically routes broadcasts to apps that have subscribed to receive that particular type of broadcast.**

**Generally speaking, broadcasts can be used as a messaging system across apps and outside of the normal user flow. However, you must be careful not to abuse the opportunity to respond to broadcasts and run jobs in the**

**background that can contribute to a slow system performance.**

**SYSTEM BROADCAST:**

The system automatically sends broadcasts when various system events occur, such as when the system switches in and out of airplane mode. System broadcasts are sent to all apps that are subscribed to receive the event.

## Receiving broadcasts

Apps can receive broadcasts in two ways: through :

1) manifest-declared receivers (Statically) by adding reciever in manifest and creating class that extend broadcast reciever and implement onRecieve() method

As part of the Android 8.0 (API level 26) Background Execution Limits, apps that target the API level 26 or higher can no longer register broadcast receivers for implicit broadcasts in their manifest

If your app targets Android 8.0 or higher, you cannot use the manifest to declare a receiver for most implicit broadcasts (broadcasts that don't target your app specifically). You can still use a context-registered receiver when the user is actively using your app.

---

2) context-registered receivers. (Dynamically)

Context-registered receivers receive broadcasts as long as their registering context is valid. For an example, if you register within an Activity context, you receive broadcasts as long as the activity is not destroyed

The registration is done
using **Context.registerReceiver()** method.