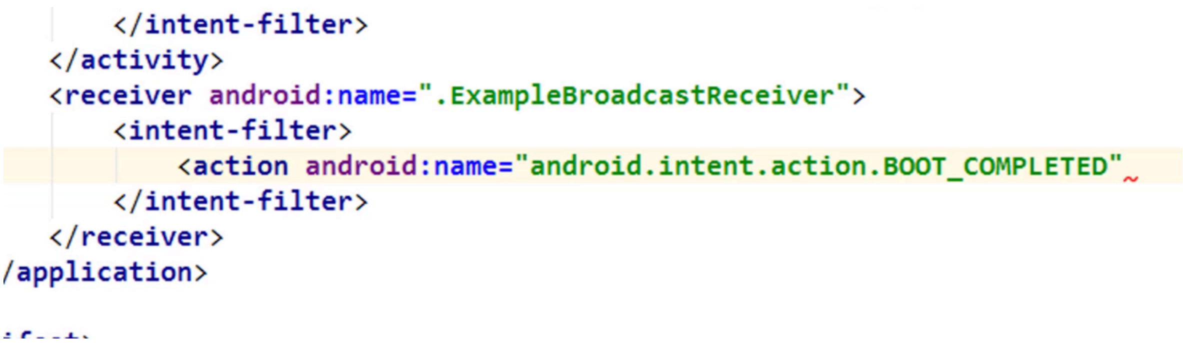
SESSION 08

export PATH=$PATH:/Users/sajib/Library/Android/sdk/platform-tools/

#BROADCAST RECEIVER

- listens to events
- system events
- app events
- class extends android.content.BroadcastReceiver [https://developer.android.com/reference/android/content/BroadcastReceiver]
- override
    - public void onReceive(Context context, Intent intent)
        - executed when a broadcast is triggered
        - the intent is for the action received / triggered
- add to Manifest
    - register broadcast
        - can have multiple actions against one broadcast

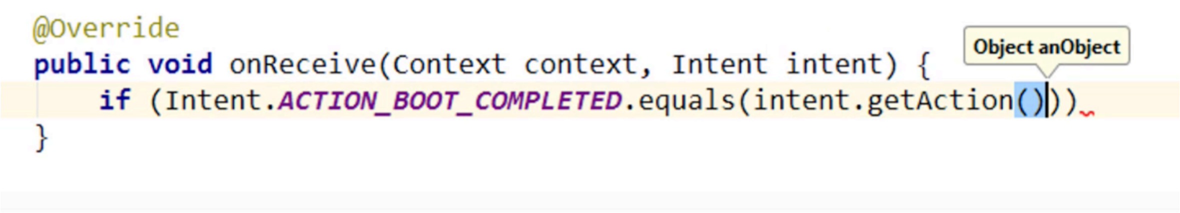


```
        </intent-filter>
    </activity>
    <receiver android:name=".ExampleBroadcastReceiver">
        <intent-filter>
            <action android:name="android.intent.action.BOOT_COMPLETED"
        </intent-filter>
    </receiver>
/application>
```

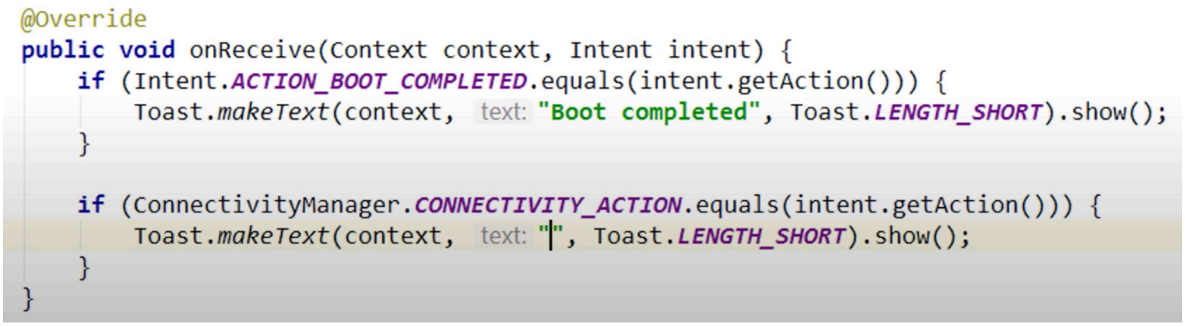
- for this example you may need to have permission:

android.permission.RECEIVE_BOOT_COMPLETED

- example intent actions
    - android.intent.action.BOOT_COMPLETED
- implicit broadcast
- explicit broadcast
- adding to manifest means registering it statically
- broadcasts can also be added dynamically
- look into job schedulers
    - register your broadcast receiver in onstart
    - deregister it in onstop

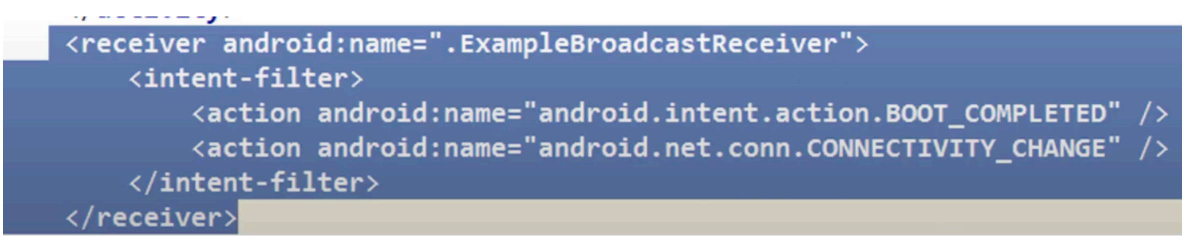


```
@Override
public void onReceive(Context context, Intent intent) {
    if (Intent.ACTION_BOOT_COMPLETED.equals(intent.getAction()))
}
```

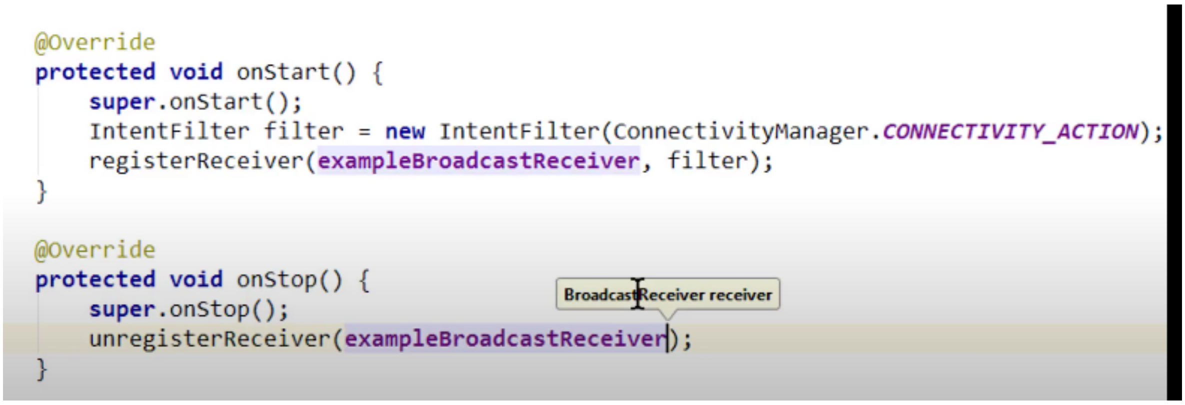


```
@Override
public void onReceive(Context context, Intent intent) {
    if (Intent.ACTION_BOOT_COMPLETED.equals(intent.getAction())) {
        Toast.makeText(context, text: "Boot completed", Toast.LENGTH_SHORT).show();
    }

    if (ConnectivityManager.CONNECTIVITY_ACTION.equals(intent.getAction())) {
        Toast.makeText(context, text: "", Toast.LENGTH_SHORT).show();
    }
}
```

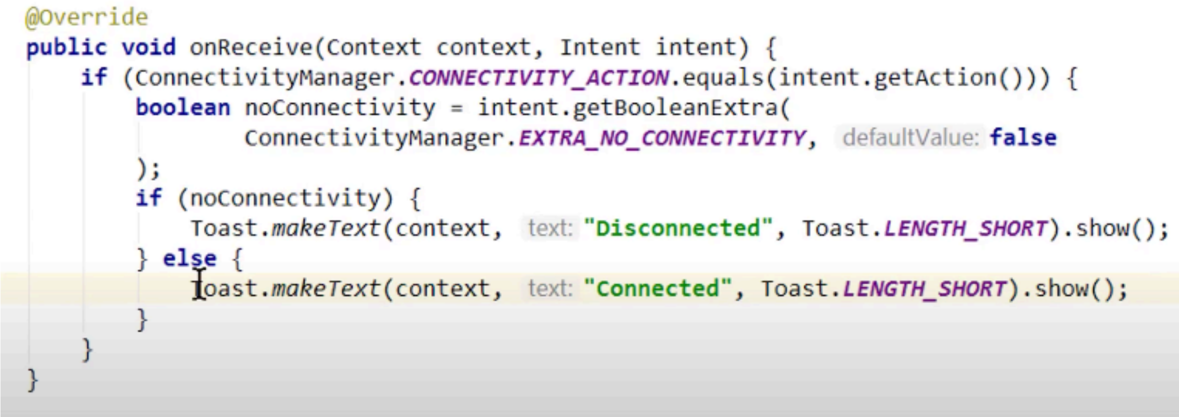


```
<receiver android:name=".ExampleBroadcastReceiver">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED" />
        <action android:name="android.net.conn.CONNECTIVITY_CHANGE" />
    </intent-filter>
</receiver>
```



```
@Override
protected void onStart() {
    super.onStart();
    IntentFilter filter = new IntentFilter(ConnectivityManager.CONNECTIVITY_ACTION);
    registerReceiver(exampleBroadcastReceiver, filter);
}

@Override
protected void onStop() {
    super.onStop();
    unregisterReceiver(exampleBroadcastReceiver);
}
```



```
@Override
public void onReceive(Context context, Intent intent) {
    if (ConnectivityManager.CONNECTIVITY_ACTION.equals(intent.getAction())) {
        boolean noConnectivity = intent.getBooleanExtra(
                ConnectivityManager.EXTRA_NO_CONNECTIVITY,  defaultValue: false
        );
        if (noConnectivity) {
            Toast.makeText(context,  text: "Disconnected", Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(context,  text: "Connected", Toast.LENGTH_SHORT).show();
        }
    }
}
```