

The screenshot shows the Android Studio interface with the project 'Broadcast_Receiver' open. The left sidebar displays the project structure under the 'Android' tab, showing the app module with its Java files. The main editor window shows the code for `AirPlaneModeChangeReceiver.java`. The code implements a `BroadcastReceiver` to handle changes in airplane mode.

```
2 import android.content.BroadcastReceiver;
3 import android.content.Context;
4 import android.content.Intent;
5 import android.provider.Settings;
6 import android.widget.Toast;
7
8 public class AirPlaneModeChangeReceiver extends BroadcastReceiver {
9     @Override
10    public void onReceive(Context context, Intent intent) {
11        if (isAirplaneModeOn(context.getApplicationContext())){
12            Toast.makeText(context, "Airplane Mode is On!!", Toast.LENGTH_SHORT).show();
13        }else{
14            Toast.makeText(context, "Airplane Mode is off!", Toast.LENGTH_SHORT).show();
15        }
16    }
17    1 usage
18    private static Boolean isAirplaneModeOn(Context context){
19        return Settings.System.getInt(context.getContentResolver(), Settings.Global.AIRPLANE_MODE_ON, def: 0) != 0;
20    }
21}
22}
```

The code includes imports for `BroadcastReceiver`, `Context`, `Intent`, `Settings`, and `Toast`. It defines a `BroadcastReceiver` named `AirPlaneModeChangeReceiver` that overrides the `onReceive` method. Inside `onReceive`, it checks if airplane mode is on using the `isAirplaneModeOn` helper method. If on, it shows a toast with the message "Airplane Mode is On!!". If off, it shows a toast with the message "Airplane Mode is off!". The `isAirplaneModeOn` method uses `Settings.System.getInt` to check the value of `Settings.Global.AIRPLANE_MODE_ON`.

The screenshot shows the Android Studio interface with the project 'Broadcast_Receiver' open. The code editor displays the file `MainActivity.java`. The code implements an `AppCompatActivity` and overrides `onCreate`, `onStart`, and `onStop` methods. A yellow circle highlights the `onCreate` method. The code also registers a broadcast receiver for the `ACTION_AIRPLANE_MODE_CHANGED` intent.

```
</> activity_main.xml  MainActivity.java  AirPlaneModeChangeReceiver.java
8
9 </> public class MainActivity extends AppCompatActivity {
10    1 usage
11    AirPlaneModeChangeReceiver airPlaneModeChangeReceiver = new AirPlaneModeChangeReceiver();
12
13    @Override
14    protected void onCreate(Bundle savedInstanceState) {
15        super.onCreate(savedInstanceState);
16        setContentView(R.layout.activity_main);
17    }
18
19
20
21    @Override
22    protected void onStart() {
23        super.onStart();
24        IntentFilter filter = new IntentFilter(Intent.ACTION_AIRPLANE_MODE_CHANGED);
25        registerReceiver(airPlaneModeChangeReceiver,filter);
26    }
27
28    @Override
29    protected void onStop() {
30        super.onStop();
31    }
32}
```

The screenshot shows the Android Studio interface with the project 'Broadcast_Receiver' open. The code editor displays `MainActivity.java`. The code implements an `AppCompatActivity` and overrides `onCreate`, `onStart`, and `onStop` methods. It initializes an `AirPlaneModeChangeReceiver` object and registers it for the `ACTION_AIRPLANE_MODE_CHANGED` intent filter.

```
</> activity_main.xml  MainActivity.java  AirPlaneModeChangeReceiver.java
</> MainActivity.java
9 </> public class MainActivity extends AppCompatActivity {
10     2 usages
11     AirPlaneModeChangeReceiver airPlaneModeChangeReceiver = new AirPlaneModeChangeReceiver();
12
13     @Override
14     protected void onCreate(Bundle savedInstanceState) {
15         super.onCreate(savedInstanceState);
16         setContentView(R.layout.activity_main);
17     }
18
19
20
21     @Override
22     protected void onStart() {
23         super.onStart();
24         IntentFilter filter = new IntentFilter(Intent.ACTION_AIRPLANE_MODE_CHANGED);
25         registerReceiver(airPlaneModeChangeReceiver,filter);
26     }
27
28     @Override
29     protected void onStop() {
30         super.onStop();
31         unregisterReceiver(airPlaneModeChangeReceiver);
32     }
33 }
```