

Bloque 10 . Ejercicio Resuelto

BroadCast Alimentación

[Descargar este ejercicio](#)

Vamos a definir un broadcast receiver que se quede escuchando si se conecta/desconecta el telefono a la alimentación y que muestre un Toast indicando lo que ha ocurrido. Para que no se muestre una actividad durante la ejecución, la cerraremos en el instante de iniciar el receiver. El código del receiver será el siguiente, crearemos la clase **MiBroadcastReceiver.kt**:

```
class MiBroadcastReceiver : BroadcastReceiver() {  
    override fun onReceive(context: Context?, intent: Intent) {  
        val action = intent.action  
        if (Intent.ACTION_POWER_CONNECTED == action)  
            Toast.makeText(context, "Alimentación conectada",  
                Toast.LENGTH_LONG).show();  
        else if (Intent.ACTION_POWER_DISCONNECTED == action)  
            Toast.makeText(context, "Alimentación desconectada",  
                Toast.LENGTH_LONG).show();  
    }  
}
```

✂ Como se puede ver en el código, lo único que hacemos en el broadcastReceiver es controlar las acciones del intent de entrada, si coinciden con las especificadas para estos casos se muestra el mensaje correspondiente.

Por otro lado, en la actividad principal tendremos el siguiente código:

```
class MainActivity : AppCompatActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)  
        val filter = IntentFilter()  
        filter.addAction(Intent.ACTION_POWER_CONNECTED )  
        filter.addAction(Intent.ACTION_POWER_DISCONNECTED )  
        val rcv = MiBroadcastReceiver()  
        registerReceiver(rcv, filter)  
        finish()  
    }  
}
```

✦ A partir de las últimas versiones de android, [no se permite iniciar actividades desde segundo plano](#) por lo que lanzamos la actividad y la finalizamos al instante, el broadcast quedará en segundo plano hasta que se desinstale la actividad. Añadimos la acción al IntentFilter y registramos el BroadcastReceiver.