

פרק 20 - Broadcast Receiver

Broadcast Receiver הינה מחלקה אשר יכולה להאזין למסרים (intent) שמשודרים.
על מנת ליצור Broadcast Receiver צריך ליצור class ולרשת מ - BroadcastReceiver וליישם את הפונקציה onReceive.

דוגמאות לשימוש

האזנה לבטריה

האזנה לשיחה נכנסת או יוצאת

האזנה לשעון

האזנה ל alarm manager

האזנה לכניסה של sms

האזנה לכניסה של push notification

ועוד

שימו לב : במעגל החיים של BroadcastReceiver ישנה רק
פונקציה אחת: OnReceive
לאחר הפונקציה OnReceive ה-BroadcastReceiver נהרס.

דוגמה ל-BroadcastReceiver שמאזין לבטריה:

בכל פעם שאחוז הסוללה ישתנה, הטקסט ב TextView ישתנה
בהתאם.

שלב 1 - עיצוב MainActivity:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:paddingBottom="@dimen/activity_vertical_margin"
    tools:context=".MainActivity">
```

```
<TextView
    android:text="Broadcast Java"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/tv"
    android:textSize="30sp"
/>
```

```
</RelativeLayout>
```

שלב 2 - עריכת MainActivity:

```
public class MainActivity extends AppCompatActivity {
```

```
    TextView tv;
    BroadcastReceiver broadcastBattery;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        tv=(TextView)findViewById(R.id.tv);
        broadcastBattery=new BroadcastReceiver();
    }
```

הצהרות

בהמשך נממש את המחלקה
BroadcastBattery

הפניות

שלב 3 - יצירת class של Broadcast Receiver:

(נוכל ליצור את הקלאס בתוך קובץ של קלאס אחר, לא חובה ליצור קלאס חדש לשם כך)

```
private class BroadcastBattery extends BroadcastReceiver {
```

יורש מ BroadcastReceiver

```
    @Override
    public void onReceive(Context context, Intent intent) {
        int battery = intent.getIntExtra("level",0);
        tv.setText(String.valueOf(battery) + " %");
    }
}
```

בהמשך נעביר ב intent את
מצב הסוללה

נשנה את הטקסט ב
TextView להיות מצב הסוללה

שלב 4 - מימוש פונקציות שיעירו את BroadcastReceiver:

```

@Override
protected void onResume() {
    super.onResume();
    registerReceiver(broadcastBattery, new IntentFilter(Intent.ACTION_BATTERY_CHANGED));
}

@Override
protected void onPause() {
    super.onPause();
    unregisterReceiver(broadcastBattery);
}
}

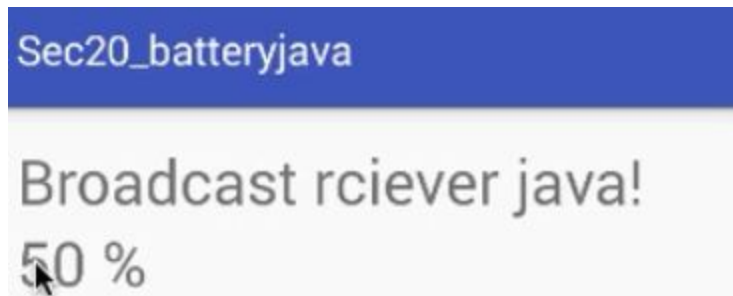
```

בכל פעם שנחזור למסך
נפעיל שוב את
broadcastReceiver

בכל פעם שנצא מהמסך נעצור
את ה broadcastReceiver

הריצו את האפליקציה (:

בשביל לראות את השינוי בבטריה בפעולה עדיף להריץ את
האפליקציה על טלפון, ולא על האימולטור.



דוגמה מספר 2 : יצירת Broadcast באמצעות xml שמאזין לבטריה

שלב 1 - ניצור class BroadcastReceiver: (הפעם בקובץ java נפרד)

```
public class BroadcastReceiver extends BroadcastReceiver {
```

```
@Override
```

```
public void onReceive(Context context, Intent intent) {
```

```
    int battery = intent.getIntExtra("level",0);
```

```
    Toast.makeText(context,String.valueOf(battery),Toast.LENGTH_SHORT).show();
```

```
}
```

```
}
```

הפעם נציג ב toast את מצב
הסוללה בכל פעם שהוא ישתנה

שלב 2 - נרשום את ה class בקובץ ה - manifest, ונוסיף את הפילטר המתאים

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.apps.appschool.broadcastreceiverpart2xml" >
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <receiver android:name=".BroadcastBattery">
            <intent-filter>
                <action android:name="android.intent.action.BATTERY_CHANGED"/>

            </intent-filter>

        </receiver>
    </application>
</manifest>

```

כעת בכל פעם שיהיה שינוי בבטריה, לא
 משנה באיזה מסך יהיה המשתמש, יופעל
 ה BroadcastReceiver