

모바일 프로그래밍

10 브로드캐스트 리시버

2017 2학기

강승우

Broadcast (방송)

- 안드로이드 단말기에서 발생하는 다양한 이벤트/정보
 - 안드로이드 시스템 부팅
 - 배터리 부족
 - 전화/문자 수신
 - 네트워크 끊김
 - ✓ 설치된 앱이나 실행 중인 앱이 반드시 알아야 하는 중요한 정보들이 있음
 - 전화가 걸려오면 전화 앱이 실행되어야 함
 - 배터리가 부족하면 실행 중인 앱들은 중요한 정보를 미리 저장해두어야 함
- 브로드캐스트 이용
 - 안드로이드는 단말기에서 발생하는 일 중 앱이 알아야 하는 상황이 발생하면 방송을 해줌
 - 앱은 수신기(BroadcastReceiver)를 통해 해당 상황을 감지하고 적절한 작업을 수행 가능함

1

방송

시스템

시스템이 부팅되었다.
시스템이 부팅되었다.
시스템이 부팅되었다.

A 앱

파일을 내려받았다.
파일을 내려받았다.
파일을 내려받았다.

B 앱

...

... 앱

...

2

수신자

A 앱

배터리 부족 방송에
귀를 기울임

B 앱

시스템 부팅 방송에
귀를 기울임

... 앱

SD 카드 연결에
귀를 기울임

관심 없는
방송이군.
무시해야지.

시스템이 부팅되었다.

앗 시스템이
부팅되었다.
적절한 처리를
해야지...

관심 없는
방송이군.
무시해야지.

브로드캐스트 수신

브로드캐스트 리시버 (Broadcast Receiver)

- 안드로이드 애플리케이션을 구성하는 4대 component 중 하나
- 안드로이드 시스템 혹은 애플리케이션이 전송하는 브로드캐스트를 수신하는 기능 수행
- BroadcastReceiver 클래스를 상속받아서 구현
 - onReceive() 메소드를 재정의
 - 브로드캐스트가 수신되면 호출되는 콜백 메소드
 - 이 메소드에 브로드캐스트 수신 후 처리 작업을 수행하는 코드를 구현함
- 브로드캐스트 리시버를 등록하는 방법
 - 정적 등록 방법
 - 동적 등록 방법

브로드캐스트 리시버의 구조

```
public class MyBroadcastReceiver extends BroadcastReceiver {  
    public void onReceive(Context context, Intent intent) {  
        .....  
        .....  
    }  
}
```

- onReceive 메소드 매개 변수
 - Context 객체
 - 브로드캐스트 리시버가 실행되는 context
 - Intent 객체
 - 브로드캐스트의 자세한 내용
 - Intent action: 브로드캐스트의 종류를 나타냄
 - 추가 정보 (putExtra, putExtras 메소드를 통해 Intent 객체에 추가한 데이터)

브로드캐스트 리시버 등록 방법

- 정적 등록

- 애플리케이션 구현 시 매니페스트 파일(AndroidManifest.xml)에 등록
- 애플리케이션 설치 시 등록이 되며, 한 번 등록하면 해제할 수 없음
- `<receiver>` element를 추가
 - `<activity>`, `<service>`와 같은 레벨
 - `name` 속성 값으로 구현한 브로드캐스트 리시버 클래스 이름을 설정함
 - `intent-filter` 설정: 수신하고자 하는 방송의 종류 지정

- 동적 등록

- Activity와 같은 안드로이드 애플리케이션 component에서 등록 메소드를 이용하여 등록
 - `registerReceiver`
- 등록하여 사용하다 더 이상 방송 수신을 원하지 않을 경우 해제할 수 있음
 - `unregisterReceiver`

예제: 전원 연결/연결 해제 브로드캐스트

- 안드로이드 단말기가 전원에 연결되거나 연결 해제 될 때 안드로이드 시스템은 브로드캐스트 전송
 - 전원이 연결되었을 때만 특정한 작업을 수행하도록 할 수 있음
 - 예: 전원이 연결되고 WiFi 연결 중일 때만 네트워크를 통하여 업데이트 다운로드 혹은 서버로 데이터 전송
- 예제 프로젝트 이름
 - 09_BroadcastReceiver\Ch12PowerBR
 - 정적 등록 방식의 예제 코드
- 위 예제를 기반으로 동적 등록 방식을 사용한 코드를 작성해보자

예제: 정적 등록 (1/2)

- 매니페스트 파일에 <receiver> 요소 추가

```
<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>
```

❖ activity와 같은 level로 <receiver> element를 추가한다

```
<receiver android:name=".PowerBroadcastReceiver">
  <intent-filter>
    <action android:name="android.intent.action.ACTION_POWER_CONNECTED"/>
    <action android:name="android.intent.action.ACTION_POWER_DISCONNECTED"/>
  </intent-filter>
</receiver>
```

✓ Intent filter 설정

예제: 정적 등록 (2/2)

- PowerBroadcastReceiver.java 구현

```
public class PowerBroadcastReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        // 수신한 Broadcast의 intent action이 무엇인지 알아냄
        String action = intent.getAction();

        // action의 종류에 따라서 Toast 메시지를 화면에 표시
        if(action == Intent.ACTION_POWER_CONNECTED) {
            Toast.makeText(context, "Power connected", Toast.LENGTH_SHORT).show();
        } else if(action == Intent.ACTION_POWER_DISCONNECTED) {
            Toast.makeText(context, "Power disconnected", Toast.LENGTH_SHORT).show();
        }
    }
}
```

예제: 동적 등록 (1/2)

- 매니페스트 파일에 <receiver> element 추가할 필요 없음
- Activity 클래스에서 BroadcastReceiver의 onReceive 콜백 메소드를 구현하여 BroadcastReceiver 객체를 생성한다

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

```
    BroadcastReceiver receiver = new BroadcastReceiver() {  
        @Override  
        public void onReceive(Context context, Intent intent) {  
            String action = intent.getAction();  
            if(action == Intent.ACTION_POWER_CONNECTED) {  
                Toast.makeText(context, "Power connected", Toast.LENGTH_SHORT).show();  
            } else if(action == Intent.ACTION_POWER_DISCONNECTED) {  
                Toast.makeText(context, "Power disconnected", Toast.LENGTH_SHORT).show();  
            }  
        }  
    };
```

.....

예제: 동적 등록 (2/2)

@Override

```
protected void onResume() {  
    super.onResume();
```

```
    IntentFilter filter = new IntentFilter();  
    filter.addAction(Intent.ACTION_POWER_CONNECTED);  
    filter.addAction(Intent.ACTION_POWER_DISCONNECTED);  
    registerReceiver(receiver, filter);
```

```
}
```

❖ IntentFilter를 설정한다

❖ BroadcastReceiver를 등록한다

@Override

```
protected void onPause() {  
    super.onPause();
```

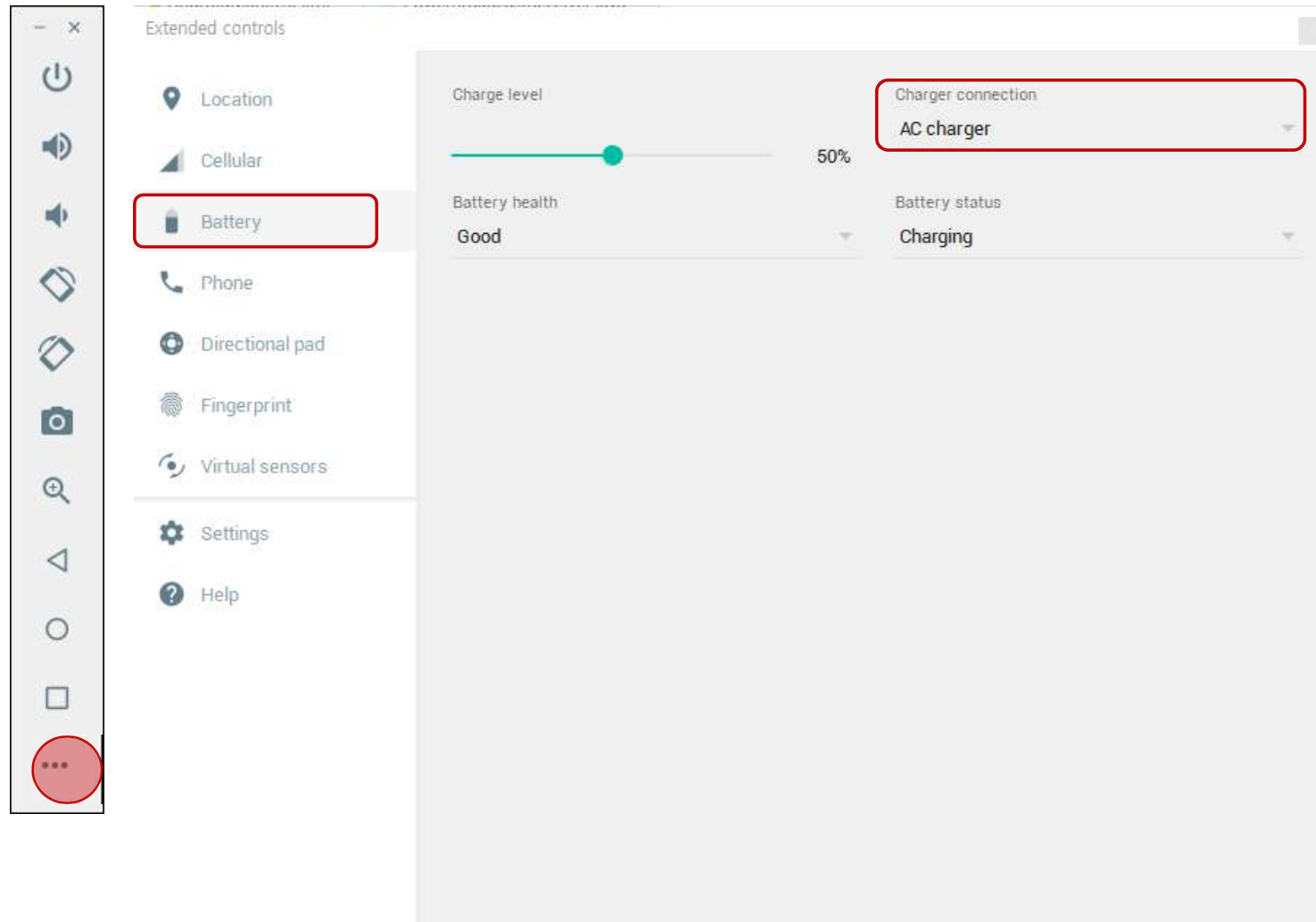
```
    unregisterReceiver(receiver);
```

```
}
```

❖ BroadcastReceiver를 등록 해제한다

Emulator에서 전원 제어하기

- 확장 컨트롤 윈도우
 - Battery 제어
- Emulator 옆의 메뉴 바에서 맨 하단 아이콘을 클릭하면 확장 컨트롤 윈도우가 나타남
- Battery 탭 클릭
- Charger connection 선택
 - None
 - AC charger



안드로이드 시스템의 주요 브로드캐스트

Intent 클래스에 정의된 action을 참고
(Broadcast Action으로 표시된 것)

<https://developer.android.com/reference/android/content/Intent.html>

액션	설명
ACTION_TIME_TICK	1분마다 보내진다.
ACTION_TIME_CHANGED	현재 시각 설정
ACTION_TIMEZONE_CHANGED	시간대 변경
ACTION_BOOT_COMPLETED	부트 완료
ACTION_PACKAGE_ADDED	패키지 추가
ACTION_PACKAGE_CHANGED	패키지 변경
ACTION_PACKAGE_REMOVED	패키지 삭제
ACTION_MEDIA_MOUNTED	외부 저장 장치 마운트 완료
ACTION_MEDIA_REMOVED	외부 저장 장치 제거
ACTION_BATTERY_CHANGED	배터리 상태 변경
ACTION_BATTERY_LOW	배터리 저충전
ACTION_POWER_CONNECTED	전원 연결
ACTION_POWER_DISCONNECTED	전원 연결 해제
ACTION_SHUTDOWN	파워 오프

안드로이드 시스템의 주요 브로드캐스트

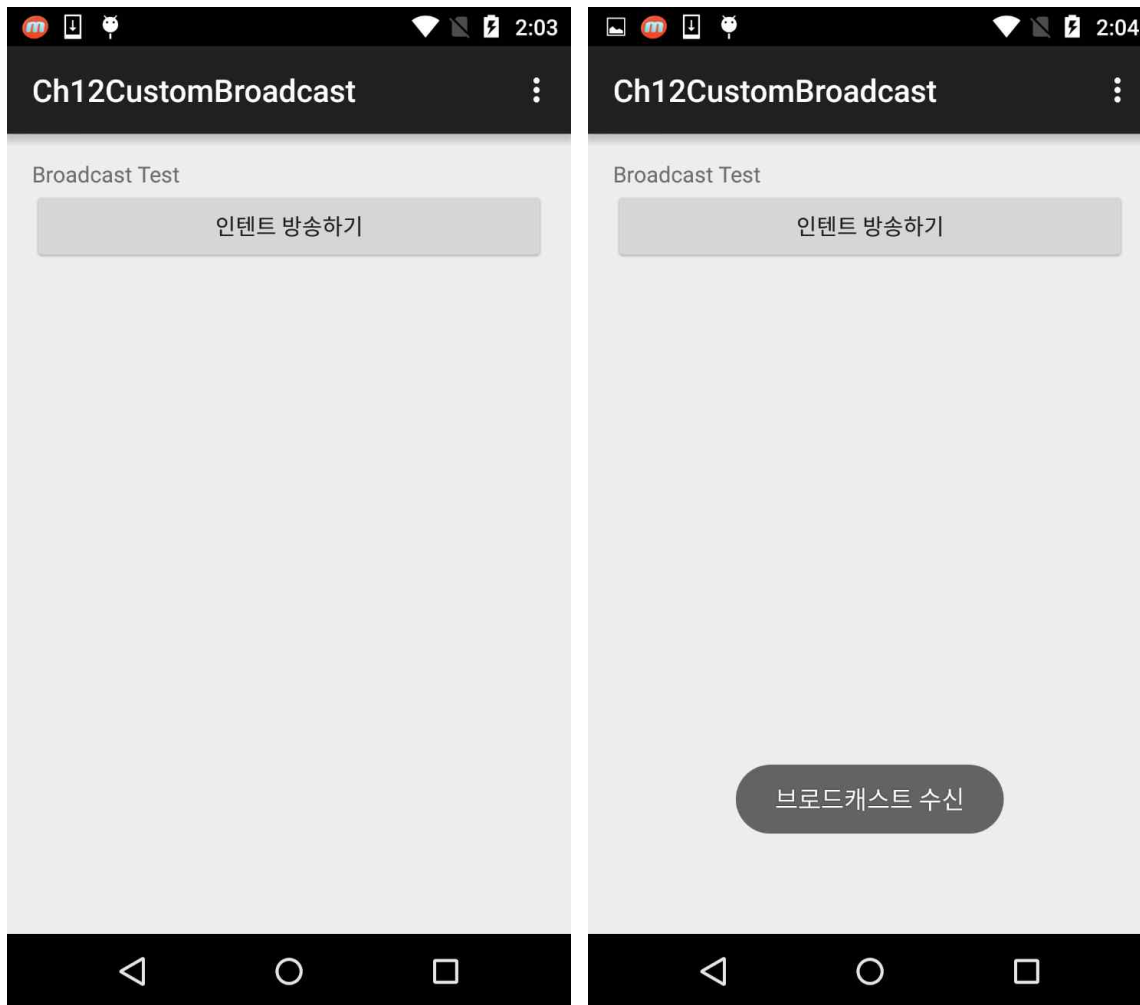
- ACTION_HEADSET_PLUG
 - Broadcast Action: Wired Headset plugged in or unplugged.
- ACTION_SCREEN_OFF
 - Broadcast Action: Sent when the device goes to sleep and becomes non-interactive.
- ACTION_SCREEN_ON
 - Broadcast Action: Sent when the device wakes up and becomes interactive.

브로드캐스트 전송

애플리케이션에서 브로드캐스트 전송

- 다른 애플리케이션(혹은 동일 앱의 다른 component)에서 수신할 수 있는 브로드캐스트를 만들어 전송할 수 있음
 - 애플리케이션 간 혹은 애플리케이션 component 간 통신 수단으로 활용 가능
- 사용 메소드
 - `sendBroadcast(Intent intent)`
 - `sendBroadcast(Intent intent, String receiverPermission)`
 - `intent`: 브로드캐스트 할 액션을 지정
 - `receiverPermission`: 브로드캐스트 수신을 위해 수신자가 가져야만 하는 permission 지정
 - `sendOrderedBroadcast(Intent intent, String receiverPermission)`
 - 한 번에 한 개의 수신자로 차례대로 전달됨. 수신하는 순서는 `<intent-filter>`에서 `android:priority` 속성으로 설정
 - 특정 리시버에서 수신 후에 다른 리시버가 수신하지 못 하도록 브로드캐스트를 중지시킬 수 있다. `abortBroadcast()`

예제



- 예제 프로젝트 이름
 - 09_BroadcastReceiver\Ch12CustomBroadcast
- Activity 화면에서 버튼을 누르면 브로드캐스트 전송
- Receiver에서 이를 수신하면 토스트 메시지 표시

- Layout의 Button (activity_main.xml)

```
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/button"
    android:onClick="broadcastIntent"
    android:text="인텐트 방송하기"/>
```

- Activity 클래스 (MainActivity.java)

```
public void broadcastIntent(View view) {
    Intent intent = new Intent();
    intent.setAction("kr.ac.koreatech.CUSTOM_INTENT");
    sendBroadcast(intent);
}
```

- BroadcastReceiver 클래스 구현 (MyReceiver.java)

```
public class MyReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        Toast.makeText(context, "브로드캐스트 수신", Toast.LENGTH_LONG).show();
    }
}
```

- 매니페스트 파일

```
<receiver android:name=".MyReceiver">
    <intent-filter>
        <action
            android:name="kr.ac.koreatech.CUSTOM_INTENT"/>
        </intent-filter>
    </receiver>
```

❖ Intent filter 설정

✓ 정적 등록 방식 예제

❖ 브로드캐스트로 전송할 Intent 객체 생성 및 Intent action 설정

❖ 브로드캐스트 전송

실습

- 배터리에 대한 브로드캐스트를 수신하고 배터리 레벨에 따라 경고 메시지를 표시하도록 만들어보자
 - 다음 예제 프로젝트를 이용한다
 - 09_BroadcastReceiver\Ch12BatteryBR
- 메시지
 - 배터리 레벨 20-30% → 배터리가 부족합니다.
 - 배터리 레벨 20% 미만 → 배터리를 충전해야 합니다.
 - 배터리 레벨 100% → 배터리가 완충 되었습니다.