



Alarm 실습

배 희호 교수
경북대학교
스마트IT과



Alarm



- Alarm은 운영체제가 관리함
 - 응용 프로그램이 종료된 상태에서도, Alarm은 동작하며 시간이 되면 응용 프로그램을 가동시키기도 함
 - 장비가 슬립(Sleep) 상태이더라도 장비를 깨워 응용 프로그램을 실행시킨다는 면에서 아주 강력함
 - 한번 설정한 Alarm이 취소되는 경우는 장비를 재 부팅 (ROM reset)할 때 뿐임
- Alarm은 Alarm을 이용하여 사용자가 지정한 임의의 시점에 애플리케이션이 실행되도록 Scheduling이 가능
- 장비의 전원을 끄거나 명시적으로 Alarm을 취소하지 않는 한 Alarm은 어떤 조건에서도 정확히 동작함
 - 예) Morning Call, 약속 Alarm, 예약 다운로드 & 업로드, 디스크 조각모음, 바이러스 체크 등에 사용



Alarm

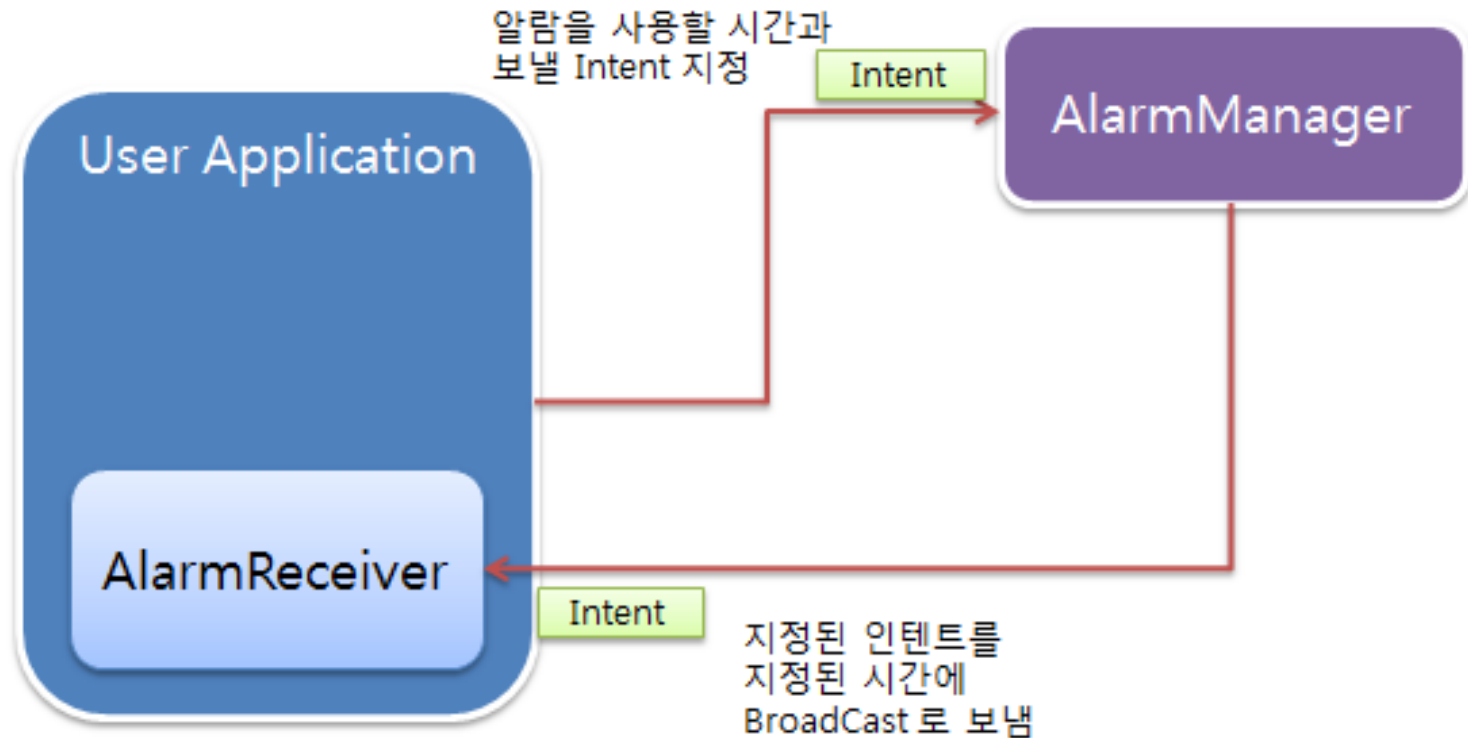


- 지정된 어떤 일을 보다 세련된 방법으로 사용자에게 알려줌으로써 다양한 애플리케이션에서 사용될 수 있음
- Alarm 기능은 Alarm을 Intent와 함께 등록시켜 줌으로써 동작되는데 정해진 시간이 되면 Intent를 Broadcasting 함(단말기가 슬립 모드인 경우에도 자동으로 수행됨)
- 미리 정해둔 시간과 간격에 따라 Intent를 발생시키는 기능
- 애플리케이션과 독립적으로 동작함
 - 애플리케이션이 종료되어도 설정된 Alarm은 정상적으로 수행될 뿐만 아니라 종료된 애플리케이션의 액티비티를 열수도 있음
- 애플리케이션의 리소스 요구사항을 줄이는데 매우 효과적
- 장치가 절전 상태에 있어도 활성화되어 있으며, 옵션으로 장치를 깨울 수도 있음



AlarmManager 클래스

- 사용자가 만든 애플리케이션이 AlarmManager에게 언제 깨워 달라고 부탁을 하면, AlarmManager가 지정된 시간에 Intent를 보내 알려주는 것





AlarmManager 클래스

- Alarm을 생성하고 취소하는 등 Alarm을 관리하는 Alarm Manager 클래스
- 객체 생성
 - 시스템이 제공하는 Service이므로 객체를 직접 생성할 필요 없음

```
AlarmManager manager = (AlarmManager)  
    getSystemService(Context.ALARM_SERVICE);
```



AlarmManager 클래스

■ 메소드

set(int type, long triggerAtTime, PendingIntent operation)

■ 일회성 알람을 발생

■ int type(Alarm type의 종류)

값	설명
RTC	지정된 시간에 펜딩 인텐트를 발생시키지만 장치를 깨우지는 않음
RTC_WAKEUP	RTC와 같지만 장치가 절전상태인 경우 장치를 깨움
ELAPSED_REALTIME	장치가 마지막으로 부팅된 이후로 경과된 시간을 기준으로 시간을 지정 장치를 깨우지는 않음
ELAPSED_REALTIME_WAKEUP	ELAPSED_REALTIME과 같지만 장치가 절정상태인 경우 장치를 깨움



AlarmManager 클래스

■ alarm type

- 반복되는 알람을 만들 때 가장 먼저 고려해야 할 사항 중 하나는 어떤 타입의 알람으로 만들지 결정하는 것
- 두 가지의 시간 타입
 - elapsed Real time
 - `SystemClock.elapsedRealtime()` 메소드로 구한 부팅된 이후의 경과 시간으로 지정
 - RTC
 - `System.currentTimeMillis()` 메소드로 구한 세계 표준시(UTC)로 지정
- elapsed Real time의 경우 타임존과 로케일에 영향을 받지 않기 때문에 시간 경과에 기반한 알람 (예 : 30초에 한 번 작동하는 알람)을 셋팅하는 데 적합
- RTC 타입은 현재 로케일에 의존하는 알람에 더 적합



AlarmManager 클래스



■ alarm type

■ ELAPSED_REALTIME

- 디바이스를 깨우지 않은 상태에서 디바이스가 부팅 된 후 시간의 경과에 따라 PendingIntent를 작동
- 경과 시간에는 디바이스가 'sleep' 상태였을 때도 포함

■ ELAPSED_REALTIME_WAKEUP

- 디바이스 부팅 후 일정 시간이 지난 후에 디바이스를 깨우고 PendingIntent를 작동

■ RTC

- 지정된 시간에 PendingIntent를 작동시키지만, 디바이스를 깨우지는 않음

■ RTC_WAKEUP

- 지정된 시간에 PendingIntent를 작동시키기 위해 디바이스를 깨움



AlarmManager 클래스



- PendingIntent Flag의 종류
 - FLAG_UPDATE_CURRENT
 - PendingIntent가 이미 존재할 경우, Extra Data를 모두 대체
 - FLAG_CANCEL_CURRENT
 - PendingIntent가 이미 존재할 경우, 기존 PendingIntent를 cancel 하고 다시 생성
 - FLAG_IMMUTABLE
 - 이름에서 알 수 있듯이, 기존 PendingIntent는 변경되지 않고, 새로 데이터를 추가한 PendingIntent를 보내도 무시함
 - FLAG_NO_CREATE
 - PendingIntent가 기존에 존재하지 않으면, Null을 return
 - FLAG_ONE_SHOT
 - 한번만 사용할 수 있는 PendingIntent



AlarmManager 클래스

■ 메소드

setRepeating(int type, long triggerAtTime, long interval, PendingIntent operation)

- 정확하게 지정된 시간간격으로 **반복적으로 알람을 발생**
- 배터리 소모가 상당히 클 수 있음

setInexactRepeating(int type, long triggerAtTime, long interval, PendingIntent operation)

- 정확한 간격 대신 AlarmManager에 정의된 상수들 중 하나를 사용하여 알람을 반복 발생 (배터리 효율이 좋음)
 - INTERVAL_FIFTEEN_MINUTES
 - INTERVAL_HALF_HOUR
 - INTERVAL_HOUR
 - INTERVAL_HALF_DAY
 - INTERVAL_DAY



AlarmManager 클래스

- 머쉬멜로(SDK 23)부터는 Doze 모드가 있어 **디바이스가 잠 들어버리면 AlarmManager가 작동하지 않음**
- Android에서는 Doze에서도 깨울 수 있는 setExactAndAllowWhileIdle와 setAndAllowWhileIdle를 추가
- API 19부터 배터리 효율을 위해 AlarmManager.set() 메소드는 정확한 시간에 알람이 동작하지 않을 수 있음
- 정확한 알람이 필요할 경우, API 19 이상에서는 setExact() 메소드를 사용

Doze 모드

휴대폰을 사용 안할 때 배터리 절약을 위해 최소한의 동작만을 하며 나머지 불필요한 백그라운드 작업은 모두 정지 시키는 기능



AlarmManager 클래스

- 공식적으로 API 19에서 모든 반복 알람은 정확하지 않다고 명시하고 있음
- 실제로 setRepeating() 메소드를 사용해서 반복 알람을 설정하면 최소 주기가 60초라는 것을 알 수 있음



AlarmManager 클래스



■ 메소드

메소드	설명
<code>void cancel(PendingIntent intent)</code>	일치하는 인텐트 알람 제거
<code>void set(int type, long triggerAtTime, PendingIntent operation)</code>	알람 설정
<code>void setRepeating(int type, long triggerAtTime, long interval, PendingIntent operation)</code>	반복 알람 설정
<code>void SetTimeZone(String TimeZone)</code>	알람을 위한 시간대 설정



AlarmManager 클래스



■ 메소드

```
void cancel(PendingIntent intent)
```

■ 알람을 취소



AlarmManager 클래스



- setInExactAnd() 메소드
 - 정확한 시간에 알람을 트리거 하지 않음
- setExact() 메소드
 - 알람 관리자를 설정하면 OS가 거의 정확한 시간에 알람을 트리거 할 수 있음
- setExactAndAllowWhileIdle() 메소드
 - 알람을 저전력 모드에서도 실행될 수 있음
- 알람 관리자는 제대로 처리되지 않으면 배터리를 소모 할 수 있으므로 주의
- 일반적으로 경보 관리자는 1분 전에 반복할 수 없음
- 저전력 모드에서는 지속 시간이 최대 15분까지 증가할 수 있음



AlarmManager 클래스



■ 시간 설정 방법

■ 현재 시간 기준으로 설정하기

```
Calendar calendar = Calendar.getInstance();  
calendar.setTimeInMillis(System.currentTimeMillis());  
calendar.add(Calendar.SECOND, 5); // 5초 후  
  
alarm.set(AlarmManager.RTC,  
          calendar.getTimeInMillis(), sender);
```

■ 절대 시간으로 설정하기

```
Calendar calendar = Calendar.getInstance();  
calendar.set(2017, 5, 30, 10, 30);  
  
alarm.set(AlarmManager.RTC,  
          calendar.getTimeInMillis(), sender);
```




AlarmManager 클래스

■ 시간 설정 방법

■ 오후 10시간 48분 8초로 예약

```
Calendar calendar = Calendar.getInstance();  
calendar.set(Calendar.HOUR_OF_DAY, 22);  
calendar.set(Calendar.MINUTE, 48);  
calendar.set(Calendar.SECOND, 8);  
  
alarm.set(AlarmManager.RTC,  
          calendar.getTimeInMillis(), sender);
```



AlarmManager 클래스

■ PendingIntent 만들기

■ 일반적인 방법(알람 1개만 발생시킬 때)

```
Intent intent = new Intent(AlarmTest.this, AlarmReceiver.class);  
intent.putExtra("tag", "onetime");
```

```
PendingIntent sender =  
    PendingIntent.getBroadcast(AlarmTest.this, 0, intent, 0);
```



AlarmManager 클래스

■ PendingIntent 만들기

■ 알람을 여러 개 발생시킬 때

```
Intent intent = new Intent(ToDoEdit.this, AlarmReceiver.class);  
PendingIntent sender = PendingIntent.getBroadcast  
    (ToDoEdit.this, mRowId.intValue(), intent, 0);
```

- 알람을 여러 개 발생시킬 때는 알람끼리 구분할 구분자가 필요한데 그 구분자가 getBroadcast() 메소드의 두 번째 인자



AlarmManager 클래스

■ PendingIntent 만들기

- 인텐트에 값을 넣고 그 값이 변경될 일이 많을 때

```
Intent intent = new Intent(ToDoEdit.this, AlarmReceiver.class);  
intent.putExtra("tag", tag);  
PendingIntent sender = PendingIntent.getBroadcast  
    (ToDoEdit.this, mRowId.intValue(), intent,  
        PendingIntent.FLAG_UPDATE_CURRENT);
```

- getBraodcast() 메소드의 마지막 인자에
PendingIntent.FLAG_UPDATE_CURRENT를 넣으면
변경된 intent의 extra 값을 제대로 받아오는 것을 보장



Alarm Receiver

- 알람 이벤트 Receiver 등록

- 알람은 지정된 시간이 되면 설정된 PendingIntent를 Broadcast하는데 이것을 수신하여 처리하기 위해 Receiver를 구현해야 함

- “AndroidManifest.xml”에 등록

```
<receiver android:name=".AlarmReceiver"/>
```



Alarm Receiver

- BroadcastReceiver를 확장한 Receiver 클래스 구현

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

- onReceive() 메소드 안에서 원하는 작업을 할 수 있으나 5초라는 시간제한이 존재하므로 시간이 오래 걸리는 작업의 경우 Service나 Thread를 이용
- Receiver는 목적에 따라 여러 개 만들 수도 있음



Alarm 예제 1



Alarm01

명시적 Intent ☒

암시적 Intent ☐

May 30 2021

Jun 01 2022

Jul 02 2023

6 06

7 : 07 AM

8 08 PM

RESET SET

Alarm01

May 30 2021

Jun 01 2022

Jul 02 2023

6 06

7 : 07 AM

8 08 PM

Alarm이 설정되었습니다

RESET SET

새로운 Activity

안녕하세요





Alarm 예제 1

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:orientation="vertical"
    tools:context=".MainActivity">

    <DatePicker
        android:id="@+id/date_picker"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:calendarViewShown="false"
        android:datePickerMode="spinner" />
```




Alarm 예제 1

■ 사용자 인터페이스

<TimePicker

```
    android:id="@+id/time_picker"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:timePickerMode="spinner" />
```

<LinearLayout

```
    android:layout_width="match_parent"  
    android:layout_height="wrap_content">
```

<Button

```
    android:id="@+id/reset"  
    android:layout_width="0dp"  
    android:layout_height="wrap_content"  
    android:layout_weight="1"  
    android:text="reset" />
```



Alarm 예제 1

■ 사용자 인터페이스

```
<Button
    android:id="@+id/set"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:text="set" />
</LinearLayout>
</LinearLayout>
```



Alarm 예제 1

■ menu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <group android:checkableBehavior="single">
        <item android:id="@+id/item1"
            android:title="명시적 Intent"
            android:checked="true"/>
        <item android:id="@+id/item2"
            android:title="암시적 Intent"/>
    </group>
</menu>
```



Alarm 예제 1

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity
    implements DatePicker.OnDateChangedListener,
        TimePicker.OnTimeChangedListener {

    GregorianCalendar calendar; // 설정 일시
    DatePicker datePicker;
    TimePicker timePicker;
    AlarmManager alarmManager;
    Intent intent;
    PendingIntent pendingIntent;
    int type = 1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```



Alarm 예제 1

■ MainActivity.JAVA

```
calendar = new GregorianCalendar();
datePicker = findViewById(R.id.date_picker);
datePicker.init(calendar.get(Calendar.YEAR), calendar.get(Calendar.MONTH),
    calendar.get(Calendar.DAY_OF_MONTH), this);
timePicker = findViewById(R.id.time_picker);
timePicker.setCurrentHour(calendar.get(Calendar.HOUR_OF_DAY));
timePicker.setCurrentMinute(calendar.get(Calendar.MINUTE));
timePicker.setOnTimeChangedListener(this);

alarmManager = (AlarmManager) getSystemService(
    Context.ALARM_SERVICE);
```



Alarm 예제 1

```
Button button1 = findViewById(R.id.set);
button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        if (type == 1) {
            intent = new Intent(MainActivity.this, Main2Activity.class);
            intent.putExtra("message", "안녕하세요");
        } else {
            intent = new Intent("Alarmtest");
            intent.putExtra("message", "AlarmTest 안녕하세요");
        }
        intent.putExtra("message", "안녕하세요");
        pendingIntent = PendingIntent.getActivity(MainActivity.this, 0, intent,
            PendingIntent.FLAG_CANCEL_CURRENT);
        alarmManager.set(AlarmManager.RTC_WAKEUP,
            calendar.getTimeInMillis(), pendingIntent);
        Toast.makeText(MainActivity.this, "Alarm이 설정되었습니다",
            Toast.LENGTH_SHORT).show();
    }
});
```



Alarm 예제 1

■ MainActivity.JAVA

```
Button button2 = findViewById(R.id.reset);
button2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        pendingIntent = PendingIntent.getActivity(MainActivity.this, 0, intent,
                                                    PendingIntent.FLAG_NO_CREATE);

        if (pendingIntent != null) {
            alarmManager.cancel(pendingIntent);
            Toast.makeText(getApplicationContext(), "Alarm이 해제되었습니다",
                                Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(MainActivity.this, "Alarm이 없습니다",
                                Toast.LENGTH_SHORT).show();
        }
    }
});
}
```



Alarm 예제 1

■ MainActivity.JAVA

```
public void onChanged (DatePicker view, int year, int monthOfYear,
                        int dayOfMonth) {
    calendar.set(year, monthOfYear, dayOfMonth,
        timePicker.getCurrentHour(), timePicker.getCurrentMinute());
}

public void onTimeChanged (TimePicker view, int hourOfDay, int minute) {
    calendar.set(datePicker.getYear(), datePicker.getMonth(),
        datePicker.getDayOfMonth(),
        hourOfDay, minute);
}
```




Alarm 예제 1

@Override

```
public boolean onCreateOptionsMenu(Menu menu) {  
    MenuInflater inflater = getMenuInflater();  
    inflater.inflate(R.menu.menu, menu);  
    return true;  
}
```

@Override

```
public boolean onOptionsItemSelected(MenuItem item) {  
    switch (item.getItemId()) {  
        case R.id.item1:  
            type = 1;  
            break;  
        case R.id.item2:  
            type = 2;  
    }  
    item.setChecked(true);  
    return true;  
}
```



Alarm 예제 1

■ activity_main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    tools:context=".Main2Activity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</LinearLayout>
```



Alarm 예제 1

■ Main2Activity.JAVA

```
public class Main2Activity extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main2);  
        setTitle("새로운 Activity");  
  
        TextView textView = findViewById(R.id.textView);  
        Intent intent = getIntent();  
        String message = intent.getStringExtra("message");  
        textView.setText(message);  
        // sound(getApplicationContext());  
        mediaplay(getApplicationContext());  
    }  
}
```



Alarm 예제 1

■ Main2Activity.JAVA

```
private void mediaplay(Context context) {  
    MediaPlayer player = MediaPlayer.create(context, R.raw.chicken);  
    player.start();  
}
```

```
private void sound(Context context) {  
    Uri uri = Uri.parse("android.resource://  
                        com.example.bae.alarm01/raw/chicken");  
    Ringtone ringtone = RingtoneManager.getRingtone(context, uri);  
    ringtone.play();  
}  
}
```



Alarm 예제 1

■ AlarmTest.JAVA

```
public class AlarmTest extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
  
        TextView textView = new TextView(this);  
        Intent intent = getIntent();  
        String message = intent.getStringExtra("message");  
        textView.setText(message);  
        textView.setTextSize(20);  
        setContentView(textView);  
        startSound(this);  
    }  
}
```



Alarm 예제 1

■ AlarmTest.JAVA

```
public void startSound(Context context) {  
    // Uri uri = RingtoneManager.getDefaultUri(RingtoneManager.  
                                           TYPE_NOTIFICATION);  
    // Uri uri = RingtoneManager.getDefaultUri(RingtoneManager.  
                                           TYPE_ALARM);  
    Uri uri = Uri.parse("android.resource://  
                        com.example.bae.alarm01/raw/chicken");  
    Ringtone ringtone = RingtoneManager.getRingtone(context, uri);  
    ringtone.play();  
}  
}
```



Alarm 예제 1

■ 메니페스트 파일

```
.....  
<activity android:name=".AlarmTest"  
    android:exported="false">  
    <intent-filter>  
        <action android:name="Alarmtest" />  
        <category android:name="android.intent.category.DEFAULT" />  
    </intent-filter>  
</activity>  
  
<activity android:name=".Main2Activity" />  
<activity android:name=".MainActivity">  
.....
```



Alarm 예제 1

■ 알람 취소하기

- 이미 설정된 알람을 취소하기 위해서는
PendingIntent.FLAG_NO_CREATE 플래그를 사용하여
PendingIntent.getActivity, PendingIntent.getBroadcast,
PendingIntent.getService 등을 호출해 보면 됨
- 이미 설정된 알람이 없다면 null을 반환
- AlarmManager와 PendingIntent를 cancel() 해주면 되는
데, 이 때, PendingIntent를 cancel()하지 않으면, 다음 번
에 위와 같이 알람 확인할 때 null을 반환하지 않게 됨
- 따라서, Alarm 해제 시에는 PendingIntent도 반드시
cancel() 하는 것을 잊지 말 것



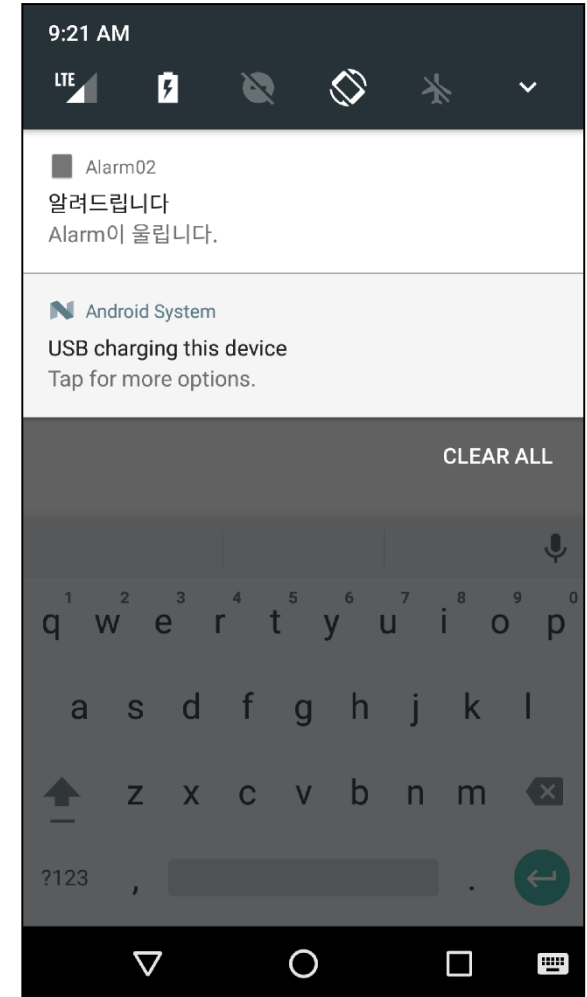
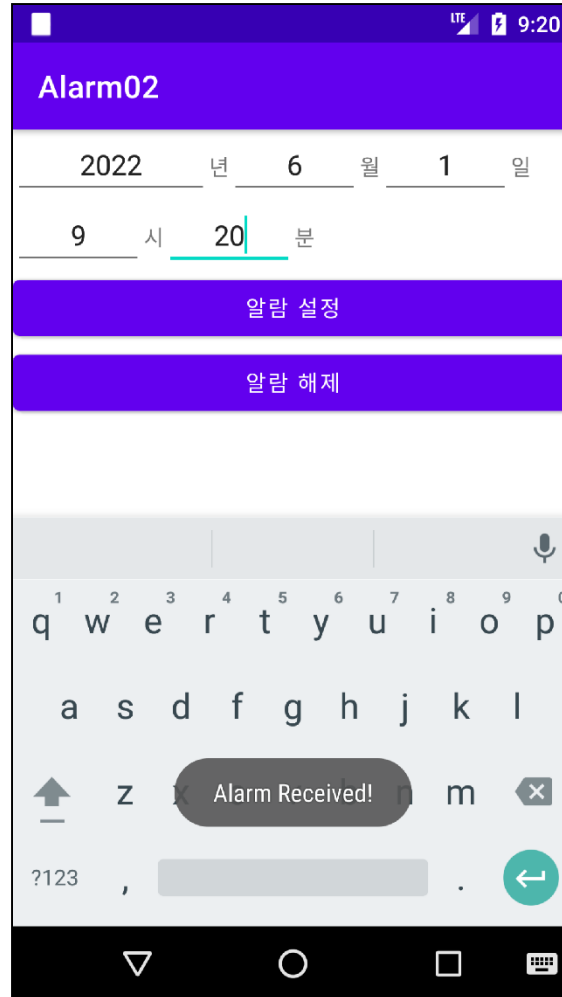
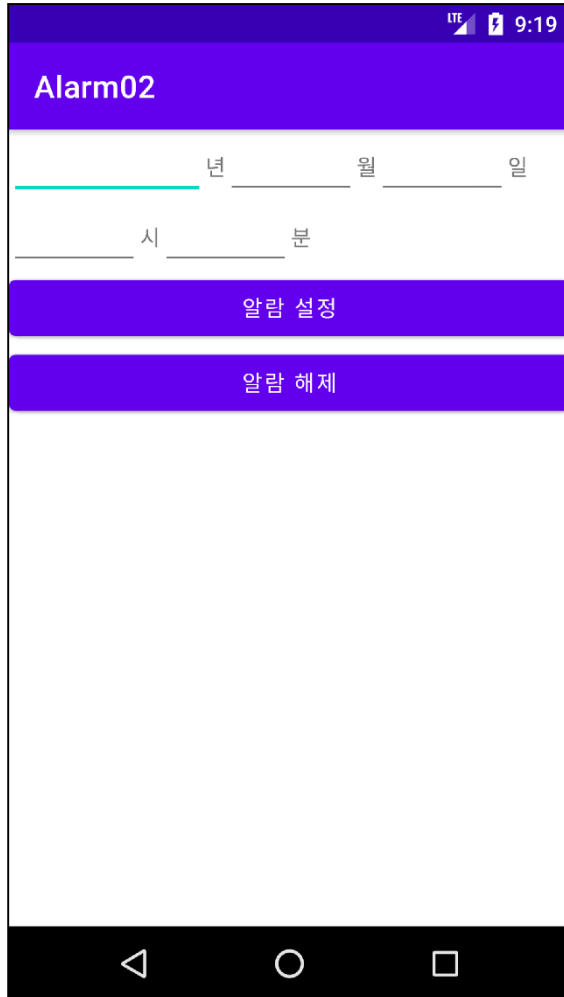
Alarm 예제 1

■ 실습 방법

- 실행 후 알람을 설정한 후 응용 프로그램을 종료
- 해당 시간이 되면 알람이 작동함
- 새로운 Activity가 열리면서 닭 울음 소리가 남



Alarm 예제 2





Alarm 예제 2

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8" ?>
<LinearLayout 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:orientation="vertical"
    tools:context=".MainActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <EditText
            android:id="@+id/edit1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:ems="6"
            android:gravity="center" />
```



Alarm 예제 2

■ 사용자 인터페이스

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="년" />
```

```
<EditText  
    android:id="@+id/edit2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:ems="4"  
    android:gravity="center" />
```

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="월" />
```



Alarm 예제 2

■ 사용자 인터페이스

```
<EditText  
    android:id="@+id/edit3"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:ems="4"  
    android:gravity="center" />
```

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="일" />
```

```
</LinearLayout>
```

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content">
```



Alarm 예제 2

■ 사용자 인터페이스

```
<EditText
    android:id="@+id/edit4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="4"
    android:gravity="center" />
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="시" />
```

```
<EditText
    android:id="@+id/edit5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="4"
    android:gravity="center" />
```



Alarm 예제 2

■ 사용자 인터페이스

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="분" />
</LinearLayout>

<Button
    android:id="@+id/button1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="알람 설정" />

<Button
    android:id="@+id/button2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="알람 해제" />
</LinearLayout>
```



Alarm 예제 2

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {  
    PendingIntent pender;
```

@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);
```

```
    EditText edit1 = findViewById(R.id.edit1);  
    EditText edit2 = findViewById(R.id.edit2);  
    EditText edit3 = findViewById(R.id.edit3);  
    EditText edit4 = findViewById(R.id.edit4);  
    EditText edit5 = findViewById(R.id.edit5);
```

```
    AlarmManager alarm = (AlarmManager) getSystemService(  
                                                                    Context.ALARM_SERVICE);  
    Intent intent = new Intent(MainActivity.this, AlarmReceiver.class);
```




Alarm 예제 2

■ MainActivity.JAVA

```
Button button1 = findViewById(R.id.button1);
button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        String year = edit1.getText().toString();
        String month = edit2.getText().toString();
        String day = edit3.getText().toString();
        String hour = edit4.getText().toString();
        String minute = edit5.getText().toString();

        if (year.equals("") || month.equals("") || day.equals("") ||
            hour.equals("") || minute.equals("")) {
            Toast.makeText(getApplicationContext(), "입력해주세요",
                Toast.LENGTH_SHORT).show();
        } else {
```



Alarm 예제 2

■ MainActivity.JAVA

```
Calendar calendar = Calendar.getInstance();
calendar.set(Integer.parseInt(year), Integer.parseInt(month) - 1,
             Integer.parseInt(day), Integer.parseInt(hour),
                               Integer.parseInt(minute));
pender = PendingIntent.getBroadcast(MainActivity.this, 0,
                                   intent, PendingIntent.FLAG_CANCEL_CURRENT);
alarm.set(AlarmManager.RTC, calendar.getTimeInMillis(), pender);
Toast.makeText(getBaseContext(), calendar.get(Calendar.YEAR) +
              "/" + (calendar.get(Calendar.MONTH) + 1) +
              "/" + calendar.get(Calendar.DAY_OF_MONTH) +
              "/" + calendar.get(Calendar.HOUR) +
              "/" + calendar.get(Calendar.MINUTE) + " 알람 설정",
              Toast.LENGTH_SHORT).show();
    }
}
```



Alarm 예제 2

```
Button button2 = findViewById(R.id.button2);
button2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        pender = PendingIntent.getBroadcast(MainActivity.this,
            0, intent, PendingIntent.FLAG_NO_CREATE);
        if (pender != null) {
            if (AlarmReceiver.manager != null)
                AlarmReceiver.manager.cancel(1234);
            alarm.cancel(pender);
            Toast.makeText(getApplicationContext(), "Alarm이 해제되었습니다",
                Toast.LENGTH_SHORT).show();
        } else {
            Toast.makeText(MainActivity.this, "Alarm이 없습니다",
                Toast.LENGTH_SHORT).show();
        }
    }
});
}
```



Alarm 예제 2

■ AlarmReceiver.JAVA

```
public class AlarmReceiver extends BroadcastReceiver {
    static NotificationManager manager;

    @Override
    public void onReceive(Context context, Intent intent) {
        Toast.makeText(context, "Alarm Received!", Toast.LENGTH_LONG).show();
        Uri uri = Uri.parse("android.resource://com.example.alarm01/raw/chicken");
        Ringtone ringtone = RingtoneManager.getRingtone(context, uri);
        ringtone.play();
        manager = (NotificationManager) context.getSystemService(
            Context.NOTIFICATION_SERVICE);
        Notification.Builder builder = new Notification.Builder(context)
            .setSmallIcon(R.drawable.icon)
            .setContentTitle("알려드립니다")
            .setContentText("Alarm이 울립니다.");
        Notification notification = builder.build();
        manager.notify(1234, notification);
    }
}
```



Alarm 예제 2

■ 메니페스트

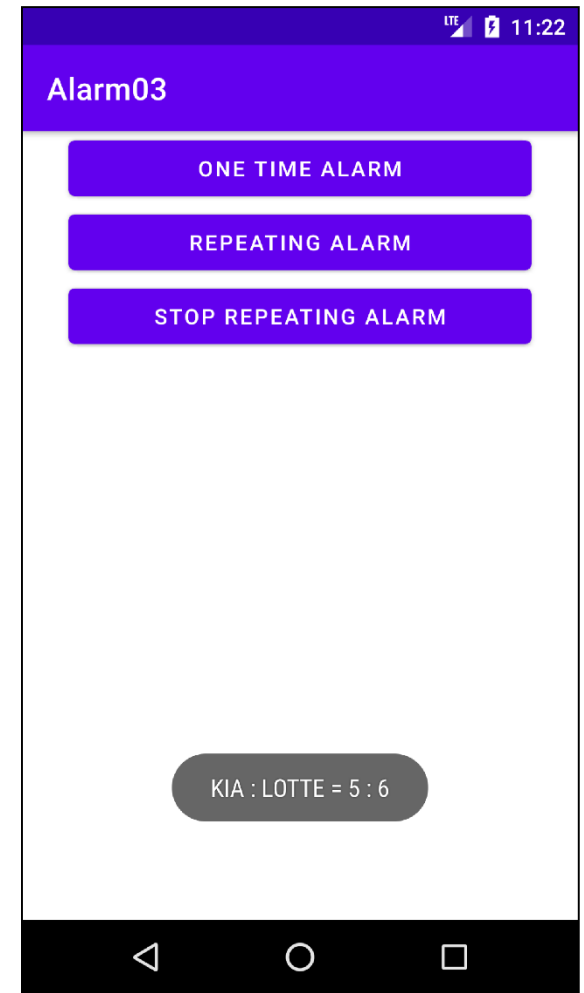
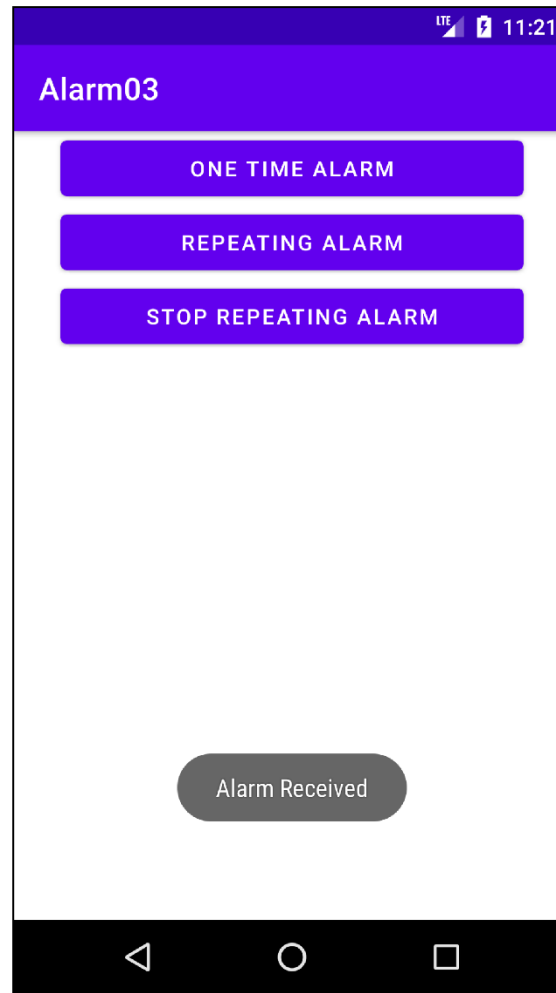
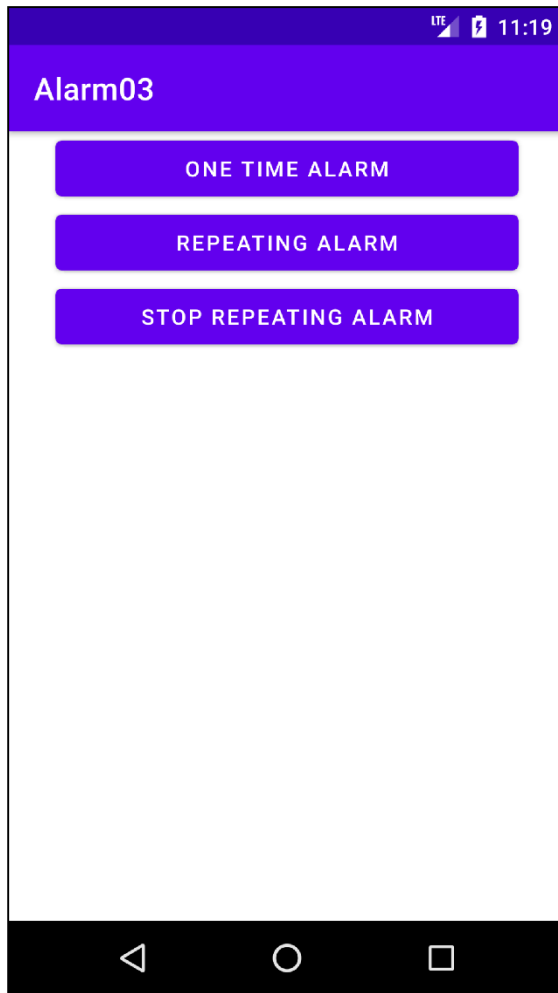
```
android:theme="@style/Theme.Alarm01">
<receiver
    android:name="com.example.alarm02.AlarmReceiver"
    android:enabled="true"
    android:exported="true"/>

<activity
    android:name="com.example.alarm02.MainActivity"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

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



Alarm 예제 3





Alarm 예제 3

■ activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:gravity="center|top"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/onetime"
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:onClick="buttonOnClicked"
        android:text="One Time Alarm" />
```



Alarm 예제 3

■ activity_main.xml

```
<Button
    android:id="@+id/repeat"
    android:layout_width="300dp"
    android:layout_height="wrap_content"
    android:onClick="buttonOnClicked"
    android:text="Repeating Alarm" />

<Button
    android:id="@+id/stop"
    android:layout_width="300dp"
    android:layout_height="wrap_content"
    android:onClick="buttonOnClicked"
    android:text="Stop Repeating Alarm" />
</LinearLayout>
```




Alarm 예제 3

■ MainActivity.JAVA

```
public class MainActivity extends AppCompatActivity {
    AlarmManager manager;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        manager = (AlarmManager) getSystemService(Context.ALARM_SERVICE);
    }

    public void buttonOnClicked(View v) {
        Intent intent;
        PendingIntent pending;
        Calendar calendar = Calendar.getInstance();
```



Alarm 예제 3

■ MainActivity.JAVA

```
switch (v.getId()) {  
    case R.id.onetime:  
        intent = new Intent(this, AlarmReceiver.class);  
        pending = PendingIntent.getBroadcast(this, 0, intent,  
                                           PendingIntent.FLAG_CANCEL_CURRENT);  
        calendar.setTimeInMillis(System.currentTimeMillis());  
        calendar.add(Calendar.SECOND, 5);  
        manager.set(AlarmManager.RTC, calendar.getTimeInMillis(), pending);  
        break;  
    case R.id.repeat:  
        intent = new Intent(this, DisplayReceiver.class);  
        intent.putExtra("kia", 5);  
        intent.putExtra("lotte", 6);  
        pending = PendingIntent.getBroadcast(this, 1, intent,  
                                           PendingIntent.FLAG_UPDATE_CURRENT);  
        manager.setRepeating(AlarmManager.ELAPSED_REALTIME,  
                             SystemClock.elapsedRealtime(), 3000, pending);  
        break;
```



Alarm 예제 3

■ MainActivity.JAVA

```
case R.id.stop:
    intent = new Intent(this, DisplayReceiver.class);
    pending = PendingIntent.getBroadcast(this, 1, intent,
                                         PendingIntent.FLAG_NO_CREATE);
    if (pending != null) {
        manager.cancel(pending);
        Toast.makeText(this, "Alarm 해제", Toast.LENGTH_SHORT).show();
    }else {
        Toast.makeText(this, "Alarm이 없습니다",
                       Toast.LENGTH_SHORT).show();
    }
}
```



Alarm 예제 3

■ AlarmReceiver.JAVA

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



Alarm 예제 3

■ DisplayReceiver.JAVA

```
public class DisplayReceiver extends BroadcastReceiver {  
  
    @Override  
    public void onReceive(Context context, Intent intent) {  
        int kia = intent.getIntExtra("kia", 0);  
        int lotte = intent.getIntExtra("lotte", 0);  
        Toast.makeText(context, "KIA : LOTTE = " + kia + " : " + lotte,  
            Toast.LENGTH_SHORT).show();  
    }  
}
```



Alarm 예제 3

■ 메니페스트

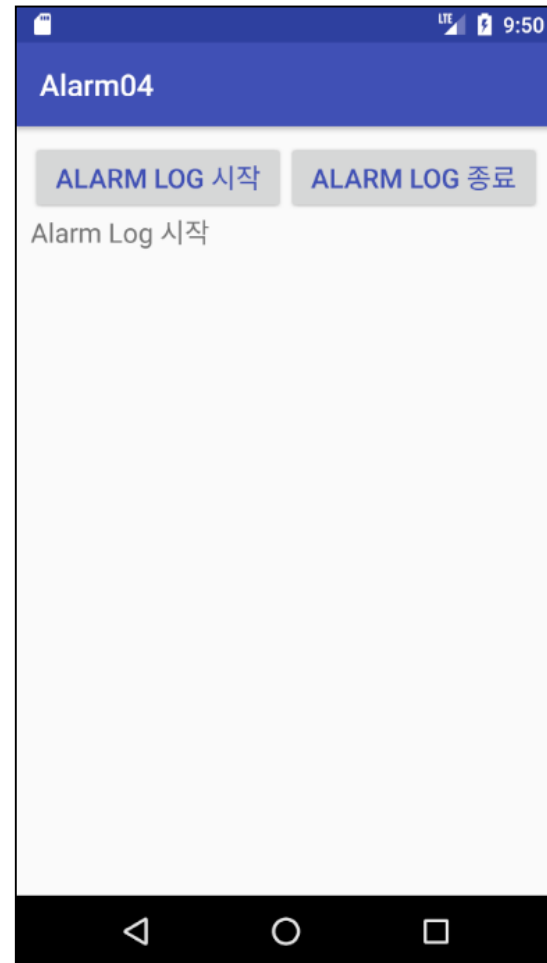
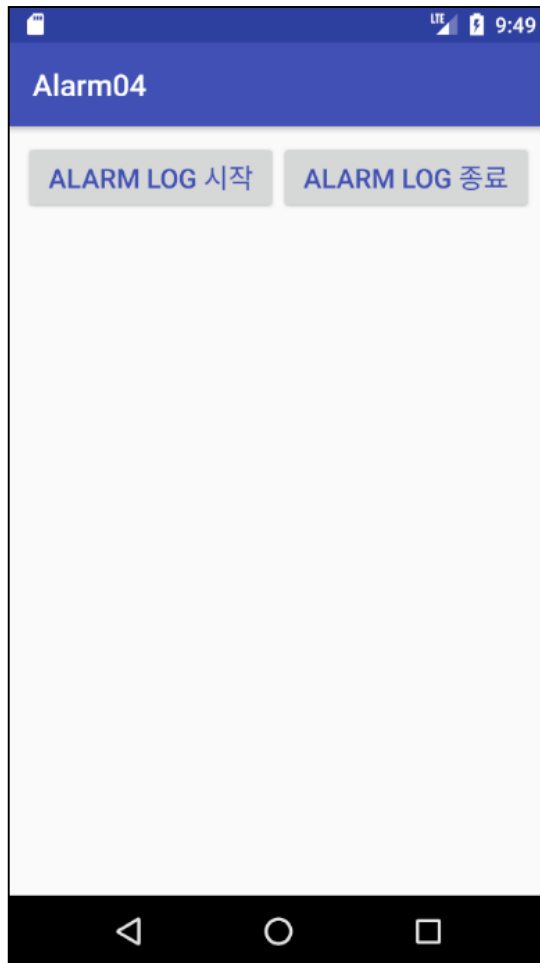
```
<receiver
    android:name="com.example.alarm03.DisplayReceiver"
    android:enabled="true"
    android:exported="true"/>
<receiver
    android:name="com.example.alarm03.AlarmReceiver"
    android:enabled="true"
    android:exported="true" />

<activity
    android:name="com.example.alarm03.MainActivity"
    android:exported="true">
```



Alarm 예제 4

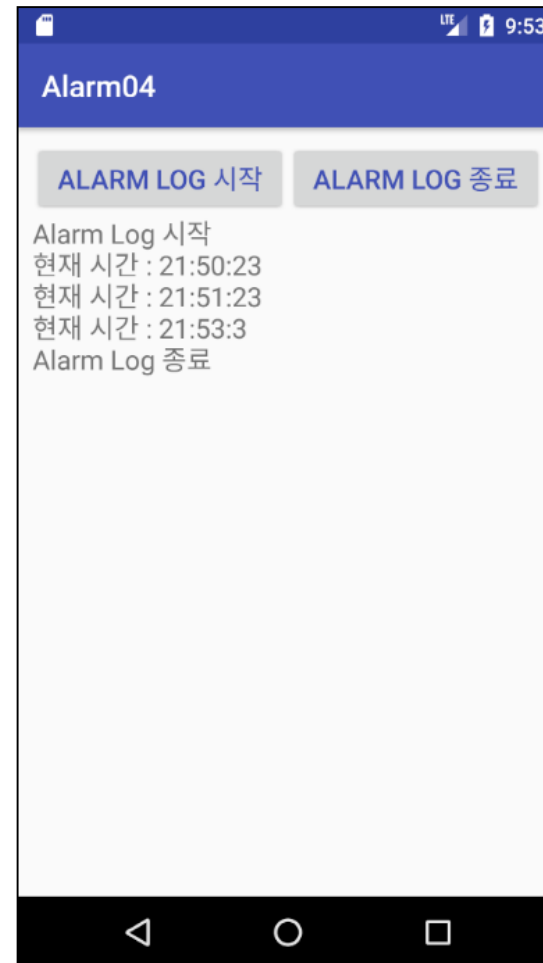
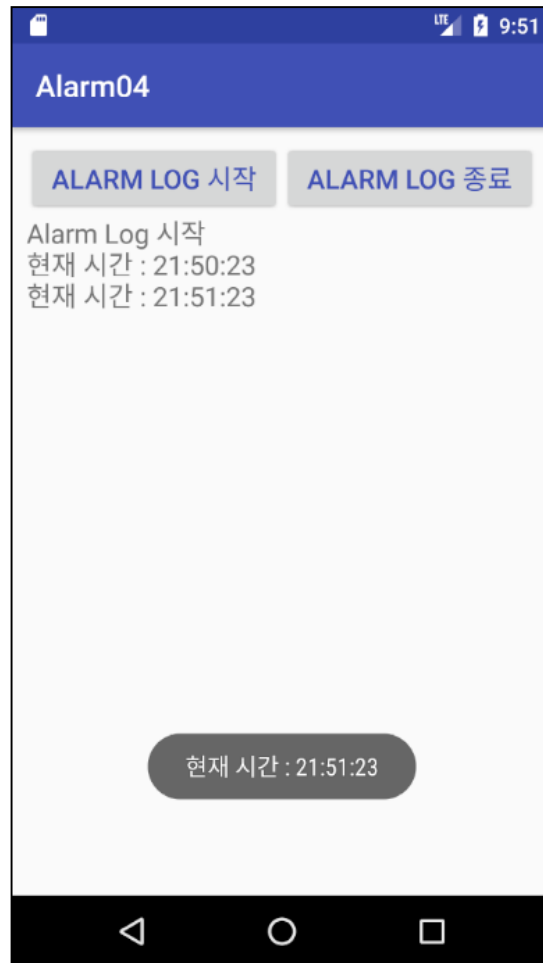
- 여러 개 알람 설정 예제 프로그램을 만들어보자





Alarm 예제 4

■ 실행 결과





Alarm 예제 4

■ 사용자 인터페이스

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:orientation="vertical"
    android:padding="10dp"
    tools:context=".MainActivity">
```



Alarm 예제 4

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <Button
        android:id="@+id/start"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:textColor="@color/colorPrimary"
        android:textSize="18dp"
        android:text="Alarm log 시작" />
    <Button
        android:id="@+id/stop"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:textColor="@color/colorPrimary"
        android:textSize="18dp"
        android:text="Alarm log 종료" />
</LinearLayout>
```



Alarm 예제 4

■ 사용자 인터페이스

```
<TextView
    android:id="@+id/log"
    android:textSize="18dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
</LinearLayout>
```



Alarm 예제 4

■ 프로그램

```
public class MainActivity extends AppCompatActivity {  
    static TextView textView;  
    PendingIntent pending;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        Intent intent = new Intent(MainActivity.this, AlarmReceiver.class);  
        AlarmManager manager = (AlarmManager) getSystemService(  
                                                                    ALARM_SERVICE);  
  
        textView = findViewById(R.id.log);
```



알람(Alarm) 예제 4

■ 프로그램

```
Button button = findViewById(R.id.start);
button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        textView.append("Alarm Log 시작 \n");
        pending = PendingIntent.getBroadcast(MainActivity.this,
            0, intent, PendingIntent.FLAG_UPDATE_CURRENT);
        long start = SystemClock.elapsedRealtime();

        start += (5 * 1000);
        manager.setRepeating(AlarmManager.ELAPSED_REALTIME_WAKEUP,
            start, 60 * 1000, pending);
    }
});
```



Alarm 예제 4

■ 프로그램

```
Button button1 = findViewById(R.id.stop);
button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        pending = PendingIntent.getBroadcast(MainActivity.this,
                                           0, intent, PendingIntent.FLAG_NO_CREATE);
        if (pending != null) {
            textView.append("Alarm Log 종료 \n");
            manager.cancel(pending);
            pending.cancel();
        } else {
            Toast.makeText(getBaseContext(),
                          "알람이 없습니다", Toast.LENGTH_SHORT).show();
        }
    }
});
}
```



Alarm 예제 4

■ AlarmReceiver.JAVA

```
public class AlarmReceiver extends BroadcastReceiver {
    @Override
    public void onReceive(Context context, Intent intent) {
        final Calendar calendar = Calendar.getInstance();
        int hour = calendar.get(Calendar.HOUR_OF_DAY);
        int minute = calendar.get(Calendar.MINUTE);
        int second = calendar.get(Calendar.SECOND);

        MainActivity.textView.append("현재 시간 : " + hour + ":"
                                   + minute + ":" + second + "\n");
        Toast.makeText(context, "현재 시간 : " + hour + ":" +
                               minute + ":" + second, Toast.LENGTH_LONG).show();
    }
}
```



Alarm 예제 4

■ 메니페스트 파일

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.bae.alarm03">
    <application
        android:allowBackup="true"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <receiver android:name=".AlarmReceiver" />
    </application>
</manifest>
```