



## 3장 : 실습교재

옥 상 훈

Blog: <http://okgosu.tistory.com>

Web Site: <http://okgosu.net>

**twitter** @okgosu

# My Intent Exam

---

- ❑ 목표
  - ❑ 암시적 인텐트 호출 예제
  
- ❑ 작성할 파일
  - ❑ layout : my\_intent\_exam.xml
  - ❑ Activity : MyIntentExam.java



# My Intent Exam 1 단계 – 대상 : my\_intent\_exam.xml

---

## □ 다음과 같이 Table 레이아웃 생성

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:stretchColumns="*"
  android:padding="10dp"
  android:layout_width="fill_parent"
  android:layout_height="fill_parent">
</TableLayout>
```



## My Intent Exam 2 단계 – 대상 : my\_intent\_exam.xml

---

### ❑ TableLayout 안에 다음과 같은 TableRow 3개 추가

```
<TableRow android:layout_weight="1" android:layout_height="fill_parent">
  <Button android:id="@+id/btn_1" android:text="버튼1"
    android:layout_weight="1" android:layout_height="fill_parent"
    android:layout_width="fill_parent"/>
  <Button android:id="@+id/btn_2" android:text="버튼2"
    android:layout_weight="1" android:layout_height="fill_parent"
    android:layout_width="fill_parent"/>
  <Button android:id="@+id/btn_3" android:text="버튼3"
    android:layout_weight="1" android:layout_height="fill_parent"
    android:layout_width="fill_parent"/>
</TableRow>
```



## My Intent Exam 3 단계 – 대상 : MyIntentExam.java

---

- ❑ MyIntentExam 액티비티 생성
  - ❑ **public class** MyIntentExam **extends** Activity
  
- ❑ 멤버 변수 선언
  - ❑ Button btn;
  - ❑ Intent intent;
  
- ❑ onCreate() 함수 오버라이딩
  - ❑ setContentView(R.layout.my\_intent\_exam);



## My Intent Exam 4 단계 – 대상 : MyIntentExam.java

---

### ☐ onCreate() 함수 오버라이딩 (이어서)

- ☐ btn\_1을 누르면 전화걸기
  - ☐ act : Intent.ACTION\_CALL
  - ☐ uri : tel:010-1234-5678

```
btn = (Button)findViewById(R.id.btn_1);  
btn.setOnClickListener(new OnClickListener() {  
    public void onClick(View arg0) {  
        intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:010-1234-5678"));  
        startActivity(intent);  
    }  
});
```



# My Intent Exam 5 단계 – 대상 : MyIntentExam.java

---

## □ 각 버튼별로 다양한 인텐트 호출

```
// 전화 걸기 직전
intent = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:111-1111"));
// 전화 걸기
intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:111-1111"));
// 연락처보기
intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("content://contacts/people/"));
// 연락처만 따로 보여줌
intent = new Intent(Intent.ACTION_PICK,
Uri.parse("content://contacts/people"));
// 사진앨범
intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("content://media/internal/images/media"));
```



# My Intent Exam 6 단계 – 대상 : MyIntentExam.java

---

## □ 기타

```
// 음악재생
intent = new Intent (Intent.ACTION_VIEW);
Uri uri = Uri.parse("file:///sdcard/1.mp3");
intent.setDataAndType (uri, "audio/mp3");
// 음악선택
intent = new Intent(Intent.ACTION_GET_CONTENT);
intent.setType("audio/*");
//startActivityForResult(Intent.createChooser(intent7, "
Select Audio Source..."), 0);
// SMS
intent = new Intent (Intent.ACTION_SENDTO,
Uri.parse ( "smsto: 0800000123"));
intent.putExtra ( "sms_body", "The SMS text");
// MMS
//Uri uri = Uri.parse ( "file:///sdcard/s_1.jpg");
Uri uri1 = Uri.parse
("content://media/external/images/media/1");
```





# My Intent Caller

---

## ☐ 목표

- ☐ startActivityForResult와 getStringExtra를 이용한 인텐트호출후 값 전달 받기

## ☐ 작성할 파일

### ☐ layout

- ☐ 호출: my\_intent\_caller.xml
- ☐ 리턴: my\_intent\_receiver.xml

### ☐ Activity

- ☐ 호출: MyIntentCaller.java
- ☐ 리턴: MyIntentReceiver.java



# My Intent Caller 1 단계 – 대상 : my\_intent\_caller.xml

---

## □ 다음과 같이 레이아웃 생성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content">
    <EditText android:hint="My Intent Receiver로 전송할 메시지 입력"
        android:id="@+id/send_text" android:layout_width="fill_parent"
        android:layout_height="wrap_content"/>
    <Button android:text="인텐트호출!!!!" android:id="@+id/btn_caller"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```



## My Intent Caller 2 단계 – 대상 : my\_intent\_receiver.xml

---

### □ 다음과 같이 레이아웃 생성

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content">
    <EditText android:hint="My Intent Call로 결과 전송할 메시지 입력"
        android:id="@+id/result_text" android:layout_width="fill_parent"
        android:layout_height="wrap_content"/>
    <Button android:text="결과 전송" android:id="@+id/finish_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"/>
</LinearLayout>
```



## My Intent Caller 3 단계 – 대상 : **MyIntentCaller.java**

---

- ❑ MyIntentCaller 액티비티 생성
  - ❑ **public class** MyIntentCaller **extends** Activity
  
- ❑ 멤버 변수 선언
  - ❑ Button btn\_caller;
  - ❑ EditText ed;



## My Intent Caller 4 단계 – 대상 : [MyIntentCaller.java](#)

---

- ❑ onCreate() 함수 오버라이딩
  - ❑ setContentView(R.layout.my\_intent\_caller);
  - ❑ ed = (EditText)findViewById(R.id.send\_text);
  - ❑ *btn\_caller* = (Button)findViewById(R.id.btn\_caller);
  
- ❑ btn\_caller 클릭핸들러에서 인텐트 호출
  - ❑ Intent intent = new Intent("MyIntentReceiver");
  - ❑ intent.putExtra("mysend", ed.getText().toString());
  - ❑ startActivityForResult(intent, 1234);



## My Intent Caller 5 단계 – 대상 : [MyIntentCaller.java](#)

---

- ❑ onActivityResult() 함수 오버라이딩해서 결과 받기
  - ❑ `data.getExtras().getString("ACT_RESULT");`

```
if (requestCode == 1234) {  
    if (resultCode != RESULT_OK) {  
        ed.setText("Result Empty");  
    } else {  
        String result = data.getExtras().getString("ACT_RESULT");  
        ed.setText("Intent호출결과:" + result);  
    }  
} else {  
    super.onActivityResult(requestCode, resultCode, data);  
}
```



## My Intent Caller 6 단계 – 대상 : [MyIntentReceiver.java](#)

---

- ❑ MyIntentReceiver 액티비티 생성
  - ❑ **public class** MyIntentReceiver **extends** Activity



## My Intent Caller 7 단계 – 대상 : [MyIntentReceiver.java](#)

---

### ☐ onCreate() 함수 오버라이딩

- ☐ setContentView(R.layout.my\_intent\_receiver);
- ☐ Intent Parameter 수신
  - ☐ String input = getIntent().getStringExtra("mysend");
  - ☐ if (input != null) {
  - ☐     EditText edit = (EditText)findViewById(R.id.result\_text);
  - ☐     edit.setText(input);
  - ☐ }
- ☐ 버튼 이벤트 처리
  - ☐ Button finish = (Button)findViewById(R.id.finish\_button);
  - ☐ finish.setOnClickListener(new View.OnClickListener() {
  - ☐     public void onClick(View v) {
  - ☐         callFinish();
  - ☐     });





# My Intent Caller 8 단계 – 대상 : [MyIntentReceiver.java](#)

---

## ❑ callFinish() 함수 정의

```
private void callFinish() {  
    EditText edit = (EditText)findViewById(R.id.result_text);  
    String result = edit.getText().toString();  
    if (result.length() != 0) {  
        Intent i = new Intent();  
        i.putExtra("ACT_RESULT", result);  
        // 결과 코드(RESET_OK)와 함께 인텐트를 실행 이전으로 되돌림  
        setResult(RESET_OK, i);  
    } else {  
        setResult(RESET_CANCELED);  
    }  
    finish();  
}
```



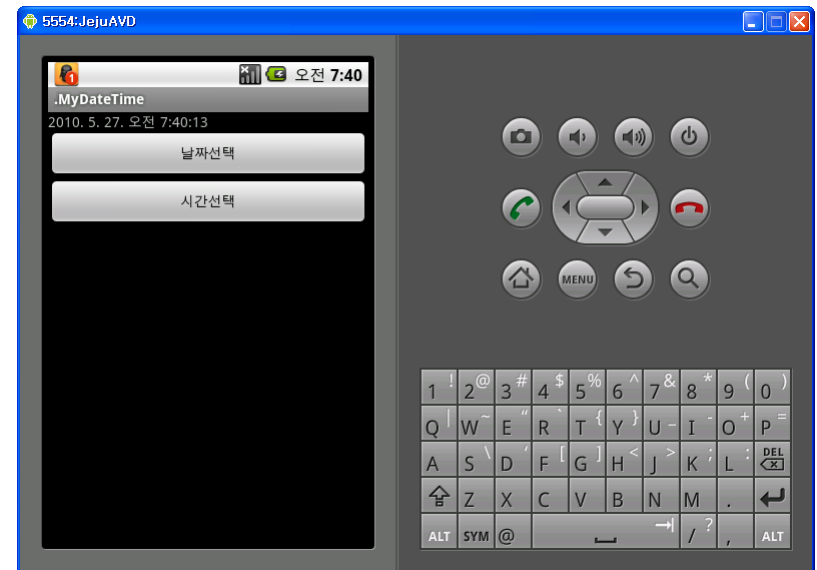
# My Alarm (지정된 시간에 알림 메시지)

## □ 목표

- Date, Time 선택 대화상자 위젯 사용 및 이벤트 처리 방법
- NotificationManager 사용

## □ 작성할 파일

- layout: my\_alarm.xml
- Activity: MyAlarm.java



## My Alarm 1단계 – 대상: my\_alarm.xml

---

- ❑ LinearLayout 생성
  - ❑ android:orientation= *"vertical"*
  - ❑ android:layout\_width= *"fill\_parent"*
  - ❑ android:layout\_height= *"fill\_parent"*



## My Alarm 2단계 – 대상: my\_alarm.xml

---

- 1단계의 LinearLayout안에 TextView, Button, Button 추가

```
<TextView android:id="@+id/dateAndTime"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"/>
```

```
<Button android:text="날짜선택" android:id="@+id/btn_date"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"/>
```

```
<Button android:text="시간선택" android:id="@+id/btn_time"  
android:layout_width="fill_parent"  
android:layout_height="wrap_content"/>
```



## My Alarm 3단계 – 대상: MyAlarm.java

---

- ❑ MyAlarm 액티비티 생성
  - ❑ **public class** MyAlarm **extends** Activity
  
- ❑ 멤버변수 선언
  - ❑ **private** Timer timer = **new** Timer();
  - ❑ **private** DateFormat dtf = DateFormat.*getDateTimeInstance*();
  - ❑ **private** TextView dateAndTime;
  - ❑ **private** Calendar cal = Calendar.*getInstance*();



# My Alarm 4단계 – 대상: MyAlarm.java

---

## □ DatePickerDialog, TimePickerDialog 리스너 설정

```
DatePickerDialog.OnDateSetListener d = new OnDateSetListener() {  
    @Override  
    public void onDateSet(DatePicker view, int year, int monthOfYear,  
        int dayOfMonth) {  
        cal.set(Calendar.YEAR, year);  
        cal.set(Calendar.MONTH, monthOfYear);  
        cal.set(Calendar.DAY_OF_MONTH, dayOfMonth);  
        updateLabel();  
    }  
};  
  
TimePickerDialog.OnTimeSetListener t = new OnTimeSetListener() {  
    @Override  
    public void onTimeSet(TimePicker view, int hourOfDay, int minute) {  
        cal.set(Calendar.HOUR_OF_DAY, hourOfDay);  
        cal.set(Calendar.MINUTE, minute);  
        updateLabel();  
    }  
};
```



## My Alarm 5단계 – 대상: MyAlarm.java

---

- ❑ onCreate() 함수 오버라이딩
  - ❑ setContentView(R.layout.*my\_alarm*);
  - ❑ Button btn\_date = (Button)findViewById(R.id.*btn\_date*);
  - ❑ Button btn\_time = (Button)findViewById(R.id.*btn\_time*);
  - ❑ dateAndTime = (TextView)findViewById(R.id.*dateAndTime*);
  - ❑ updateLabel();



# My Alarm 6단계 – 대상: MyAlarm.java

---

## □ btn\_date, btn\_time 클릭 리스너 설정

```
btn_date.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View v) {  
        DatePickerDialog d1 =  
            new DatePickerDialog(MyAlarm.this, d, cal.get(Calendar.YEAR), cal.get(Calendar.MONTH),  
                cal.get(Calendar.DAY_OF_MONTH));  
        d1.show();  
    }  
});  
btn_time.setOnClickListener(new View.OnClickListener() {  
    public void onClick(View v) {  
        TimePickerDialog t1 =  
            new TimePickerDialog(MyAlarm.this, t, cal.get(Calendar.HOUR_OF_DAY), cal.get(Calendar.MINUTE), true);  
        t1.show();  
    }  
});
```





# My Alarm 7단계 – 대상: MyAlarm.java

---

## ❑ updateLabel() 함수 정의

```
private void updateLabel() {  
    dateAndTime.setText(dtf.format(cal.getTime()));  
    runTimer();  
}
```

## ❑ runTimer() 함수 정의

```
private void runTimer() {  
    TimerTask task = new TimerTask() {  
        @Override  
        public void run() {  
            // TODO Auto-generated method stub  
            notifyMe();  
        }  
    };  
    timer.schedule(task,cal.getTime());  
}
```



# My Alarm 8단계 – 대상: MyAlarm.java

---

## □ notifyMe() 함수 정의

```
private int count = 0;  
private void notifyMe() {  
    final NotificationManager mgr =  
        (NotificationManager) getSystemService(NOTIFICATION_SERVICE);  
    Notification note = new Notification(R.drawable.icon, "okgosu 메시지",  
        System.currentTimeMillis());  
    PendingIntent i = PendingIntent.getActivity(this, 0, new Intent(this, MyImageView.class),  
        0);  
    note.setLatestEventInfo(this, "알림제목", "본문이오~", i);  
    note.number = ++count;  
    mgr.notify(12345, note);  
}
```



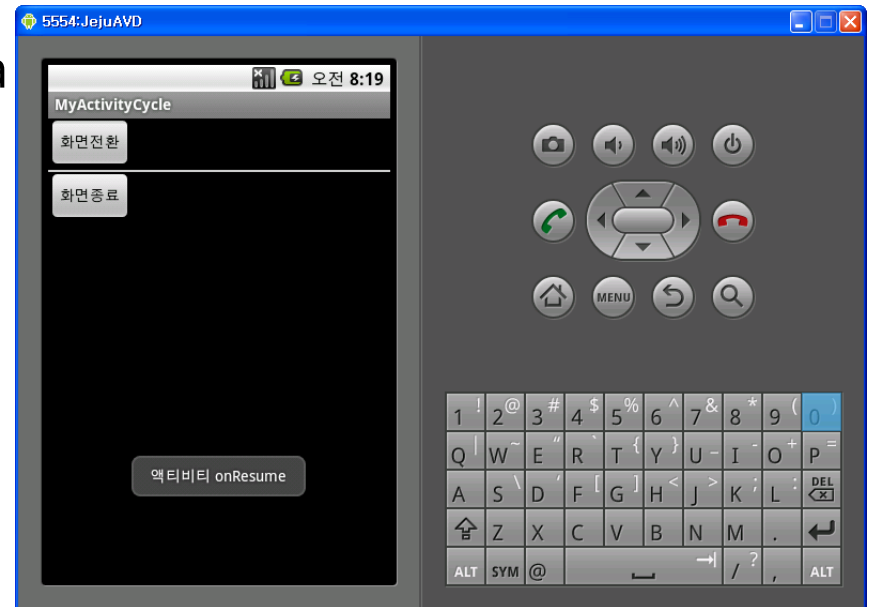
# My Activity Cycle

## □ 목표

- Activity 라이프 사이클의 이해
- Activity 라이프 사이클별 함수 오버라이딩

## □ 작성할 파일

- layout: my\_activity\_cycle.xml
- Activity: MyActivityCycle.java



# My Activity Cycle 1단계 – 대상: my\_activity\_cycle.xml

---

## □ Table 레이아웃 작성

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">
    <TableRow>
        <Button android:text="화면전환" android:id="@+id/btn_go"
            android:layout_width="fill_parent" android:layout_height="wrap_content"/>
    </TableRow>
    <View android:background="#CCCCCC" android:layout_width="fill_parent"
        android:layout_height="2sp"/>
    <TableRow>
        <Button android:text="화면종료" android:id="@+id/btn_end"
            android:layout_width="fill_parent" android:layout_height="wrap_content"/>
    </TableRow>
</TableLayout>
```



## My Activity Cycle 2단계 – 대상 : MyActivityCycle.java

---

- ❑ MyActivity 액티비티 생성
  - ❑ **public class** MyActivityCycle **extends** Activity
- ❑ onCreate() 함수 오버라이딩
  - ❑ setContentView(R.layout.*my\_activity\_cycle*);
  - ❑ Button btn\_go = (Button)findViewById(R.id.*btn\_go*);
  - ❑ Button btn\_end = (Button)findViewById(R.id.*btn\_end*);



# My Activity Cycle 3단계 – 대상 : MyActivityCycle.java

---

## □ 버튼 클릭 리스너 설정

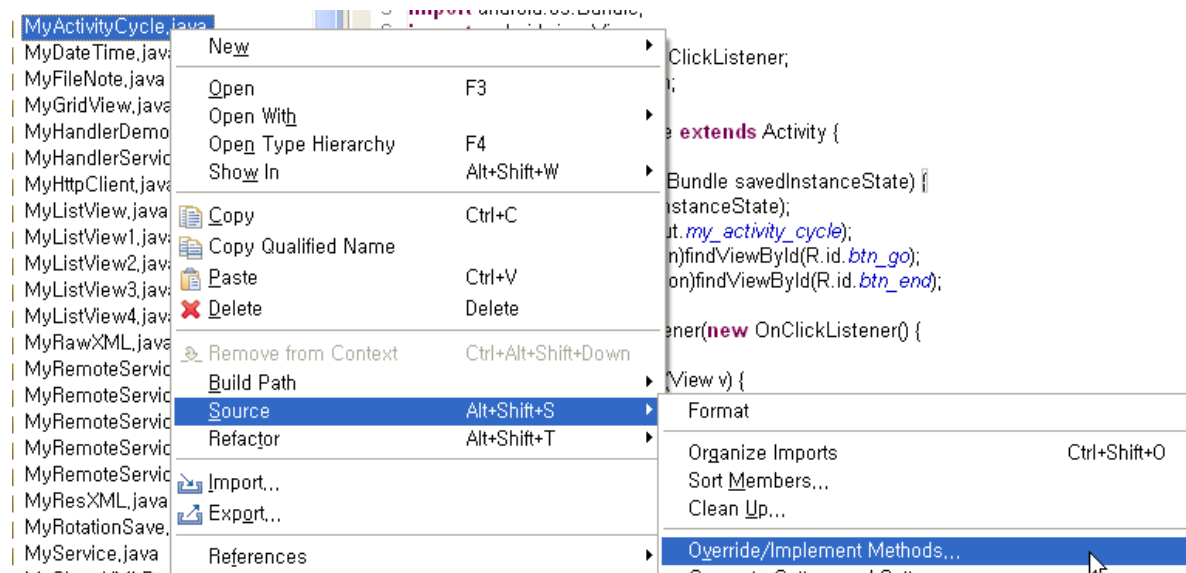
```
btn_go.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent i = new Intent(MyActivityCycle.this, MyListView.class);  
        startActivity(i);  
    }  
});
```

```
btn_end.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        finish();  
    }  
});
```



# My Activity Cycle 4단계 – 대상 : MyActivityCycle.java

- ❑ MyActivityCycle.java에서 Override/Implement Methods 선택
  - ❑ onDestroy
  - ❑ onPause
  - ❑ onResume
  - ❑ onStart



# My Activity Cycle 5단계 – 대상 : MyActivityCycle.java

---

## □ Toast로 각 사이클별 메시지 출력설정

```
@Override
protected void onDestroy() {
    Toast.makeText(this, "액티비티 onDestory", Toast.LENGTH_SHORT).show();
    super.onDestroy();
}
```

```
@Override
protected void onPause() {
    Toast.makeText(this, "액티비티 onPause", Toast.LENGTH_SHORT).show();
    super.onPause();
}
```

```
@Override
protected void onRestart() {
    Toast.makeText(this, "액티비티 onRestart", Toast.LENGTH_SHORT).show();
    super.onRestart();
}
```

```
@Override
protected void onResume() {
    Toast.makeText(this, "액티비티 onResume", Toast.LENGTH_SHORT).show();
    super.onResume();
}
```





# My Photo Album

- 목표
  - Bundle savedInstanceState을 이용한 상태 유지
- 작성할 파일
  - layout: my\_photo\_album.xml
  - Activity: MyPhotoAlbum.java



# My Photo Album 1단계 – 대상: my\_photo\_album.xml

---

- ❑ LinearLayout 생성
  - ❑ `android:orientation= "vertical"`
  - ❑ `android:layout_width= "fill_parent"`
  - ❑ `android:layout_height= "fill_parent"`
  - ❑ `android:background="@drawable/background"`
- ❑ res/drawable 폴더에 background.png 복사



# My Photo Album 3단계 – 대상: my\_photo\_album.xml

---

## □ 1단계의 LinearLayout 안에 LinearLayout 2개 추가

```
<LinearLayout  
    android:layout_weight= "1"  
    android:layout_width= "fill_parent"  
    android:layout_height= "fill_parent">  
</LinearLayout>
```

```
<LinearLayout  
    android:layout_width= "fill_parent"  
    android:layout_height= "wrap_content"  
    android:layout_marginTop= "10dip"  
    android:gravity= "center">  
</LinearLayout>
```



# My Photo Album 4단계 – 대상: my\_photo\_album.xml

---

## □ 2단계 첫번째 LinearLayout 안에 FrameLayout 추가

```
<FrameLayout
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:id="@+id/image_container"
    android:background="@drawable/image_container">
    <ImageView
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:id="@+id/image_view"
        android:scaleType="fitCenter"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom|center_horizontal"
        android:id="@+id/status_text"
        android:textColor="@android:color/primary_text_dark"
        android:textSize="16sp"
        android:shadowDx="1.0"
        android:shadowDy="1.0"
        android:shadowRadius="1"
        android:layout_margin="5dip"
        android:shadowColor="@android:color/background_dark"/>
</FrameLayout>
```



# My Photo Album 5단계 – 대상: my\_photo\_album.xml

---

## □ 2단계 두번째 LinearLayout 안에 Button 추가

```
<Button  
    android:text= "다음"  
    android:id= "@+id/next_button"  
    android:layout_width= "wrap_content"  
    android:layout_height= "wrap_content"  
    android:drawableLeft= "@drawable/icon"  
    android:textSize= "24sp"/>
```



## My Photo Album 6단계 – 대상: MyPhotoAlbum.java

---

- ❑ MyPhotoAlbum액티비티 생성
  - ❑ **public class** MyPhotoAlbum **extends** Activity
  
- ❑ 멤버변수 선언
  - ❑ **private int** mCurrentPhotoIndex = 0;
  - ❑ **private int[]** mPhotoIds = **new int[]** { R.drawable.s\_1, R.drawable.s\_2, R.drawable.s\_3, R.drawable.s\_4, R.drawable.s\_5, R.drawable.s\_6, R.drawable.s\_7};



## My Photo Album 7단계 – 대상 : MyPhotoAlbum.java

---

### ☐ onCreate() 함수 오버라이딩

- ☐ setContentView(R.layout.my\_photo\_album);
- ☐ showPhoto(mCurrentPhotoIndex);
  
- ☐ Button nextButton = (Button) findViewById(R.id.next\_button);
- ☐ nextButton.setOnClickListener(**new** View.OnClickListener() {
- ☐       **public void** onClick(View v) {
- ☐           mCurrentPhotoIndex = (mCurrentPhotoIndex +
- ☐       1)
- ☐           % mPhotoIds.length;
- ☐       showPhoto(mCurrentPhotoIndex);
- ☐       }
- ☐     });



# My Photo Album 8단계 – 대상 : MyPhotoAlbum.java

---

## □ showPhoto 메서드 구현

```
private void showPhoto(int photoIndex) {  
    ImageView imageView = (ImageView) findViewById(R.id.image_view);  
    imageView.setImageResource(mPhotoIds[photoIndex]);  
  
    TextView statusText = (TextView) findViewById(R.id.status_text);  
    statusText.setText(String.format("%d/%d", photoIndex + 1,  
        mPhotoIds.length));  
}
```





# My Photo Album 9단계 – 대상 : MyPhotoAlbum.java

---

## □ onSaveInstanceState, onRestoreInstanceState 구현

@Override

```
protected void onSaveInstanceState(Bundle outState) {  
    outState.putInt("photo_index", mCurrentPhotoIndex);  
    super.onSaveInstanceState(outState);  
}
```

@Override

```
protected void onRestoreInstanceState(Bundle savedInstanceState) {  
    mCurrentPhotoIndex = savedInstanceState.getInt("photo_index");  
    showPhoto(mCurrentPhotoIndex);  
    super.onRestoreInstanceState(savedInstanceState);  
}
```



---

수고하셨습니다 ^^/



by OkGosu.Net