

# Ch08. 파일 처리 2

모바일게임프로그래밍  
김지심 교수

# 학습목표



파일을 처리하는 방식을 이해한다.

3 종류 영역의 파일을 처리할 수 있다.

03



## 외부메모리 처리



# 외부메모리 처리

## 외부 메모리

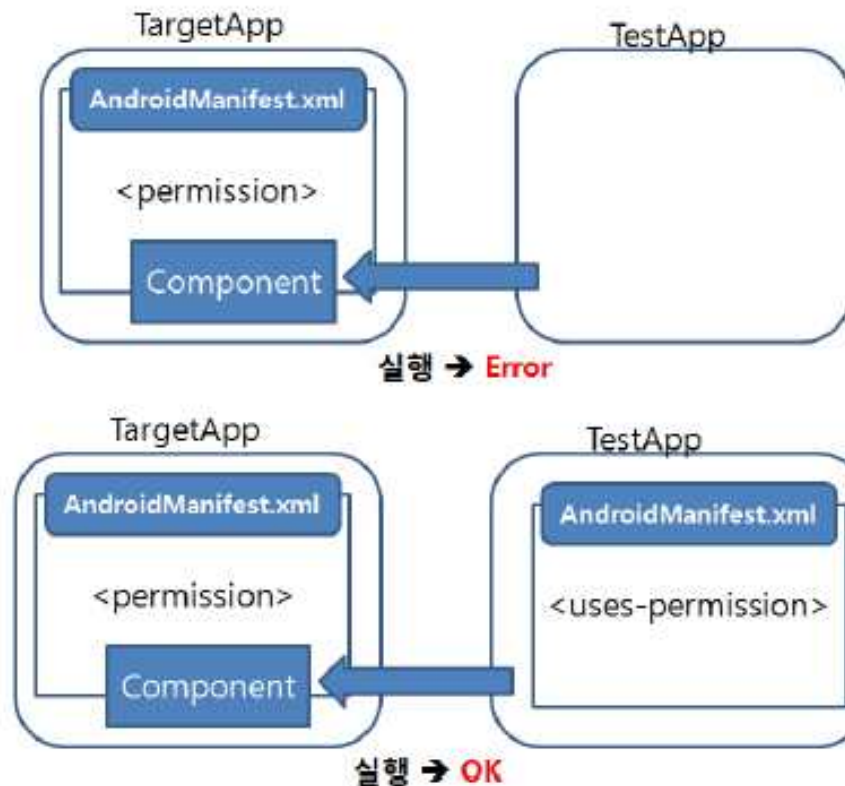
내부 저장소를 제외한 사용자 데이터 저장 공간  
앱 간 데이터를 공유하거나 사용자가 관리하는 파일 등(R/W 모두 가능)  
에뮬레이터에서는 /storage/emulated/0 (/sdcard),  
스마트폰에서는 /storage/시리얼넘버에 존재  
\* 스마트폰의 경우 모델마다 다름

# 외부메모리 처리

## 퍼미션 설정

AndroidManifest에 기본적인 퍼미션 설정

어떤 앱이 <permission>을 부여했다면, 그 앱을 이용하는 앱은 <use-permission>을 선언해야만 권한을 득할 수 있음



# 외부메모리 처리

## 퍼미션 설정

### AndroidManifest에 기본적인 퍼미션 설정

```
<permission android:name="com.test.permission.SOME_PERMISSION"  
    android:label="SOME Permission"  
    android:description="@string/permission"  
    android:protectionLevel="normal"/>
```

name: 퍼미션명

label, description: 퍼미션에 대한 설명(사용자에게 보임)

protectionLevel: 보호수준(normal, dangerous...)

# 외부메모리 처리

## 퍼미션 설정

### 대표적 퍼미션

- ACCESS\_FINE\_LOCATION: 정확한 위치 정보 액세스
- ACCESS\_NETWORK\_STATE: 네트워크에 대한 정보 액세스
- ACCESS\_WIFI\_STATE: 와이파이 네트워크에 대한 정보 액세스
- BATTERY\_STATS: 배터리 통계 수집
- BLUETOOTH: 연결된 블루투스 장치에 연결
- BLUETOOTH\_ADMIN: 블루투스 장치를 검색하고 페어링
- CALL\_PHONE: 다이얼 UI를 거치지 않고 전화를 시작
- CAMERA: 카메라 장치에 액세스
- INTERNET: 네트워크 연결
- READ\_CONTACTS: 사용자의 연락처 데이터 읽기
- READ\_EXTERNAL\_STORAGE: 외부 저장소에서 파일 읽기
- READ\_PHONE\_STATE: 장치의 전화번호, 네트워크 정보, 진행 중인 통화 상태 등 전화 상태에 대한 읽기
- READ\_SMS: SMS 메시지 읽기
- RECEIVE\_BOOT\_COMPLETED: 부팅 완료 시 수행
- RECEIVE\_SMS: SMS 메시지 수신
- RECORD\_AUDIO: 오디오 녹음
- SEND\_SMS: SMS 메시지 발신
- VIBRATE: 진동 울리기
- WRITE\_CONTACTS: 사용자의 연락처 데이터 쓰기
- WRITE\_EXTERNAL\_STORAGE: 외부 저장소에 파일 쓰기



# 외부메모리 처리

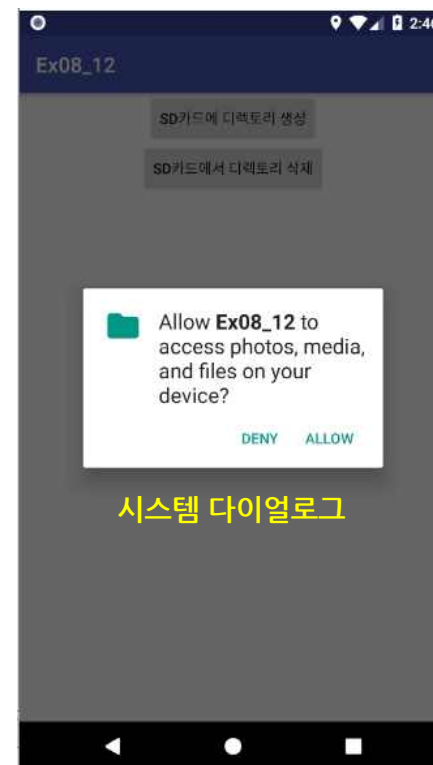
## 퍼미션 설정

5.1까지는 권한 화면에서 어떤 퍼미션이 사용되는지 정보 성격으로만 알려주지만(신고제), 6.0(Marshmallow, API23) 이상부터는 사용자에게 해당 권한을 부여할 것인지 선택

\* **매니페스트**에 퍼미션을 설정할 뿐 아니라, 강화된 퍼미션 모델에 따라 사용자에게 **런타임시** 권한을 요청



5.1 이하, 매니페스트에 설정



알림  
저장소 권한이 거부되었습니다. 사용을 원하시면 설정에서 해당 권한을 직접 허용하셔야 합니다.  
설정      확인

6.0 이상, 클래스에서  
동적으로 권한 요청



# 외부메모리 처리

## 퍼미션 설정

CLASS	METHOD/DEFINITION
ActivityCompat	<p>Activity의 기능을 사용할 수 있도록 해주는 Helper 클래스 * Helper: 어떤 클래스의 작업을 도와주는 역할을 하는 클래스</p> <p><b>void requestPermissions(Activity activity, String[] permissions, int requestCode)</b>      퍼미션 요청 메소드 permissions: 퍼미션명, 배열로 여러 퍼미션 항목을 한번에 요청 가능 requestCode: 개발자 임의의 숫자값. 결과를 처리할 때 요청코드 확인 시 사용</p> <p><b>public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults)</b>      퍼미션 결과에 따라 요청을 처리</p>

# 외부메모리 처리

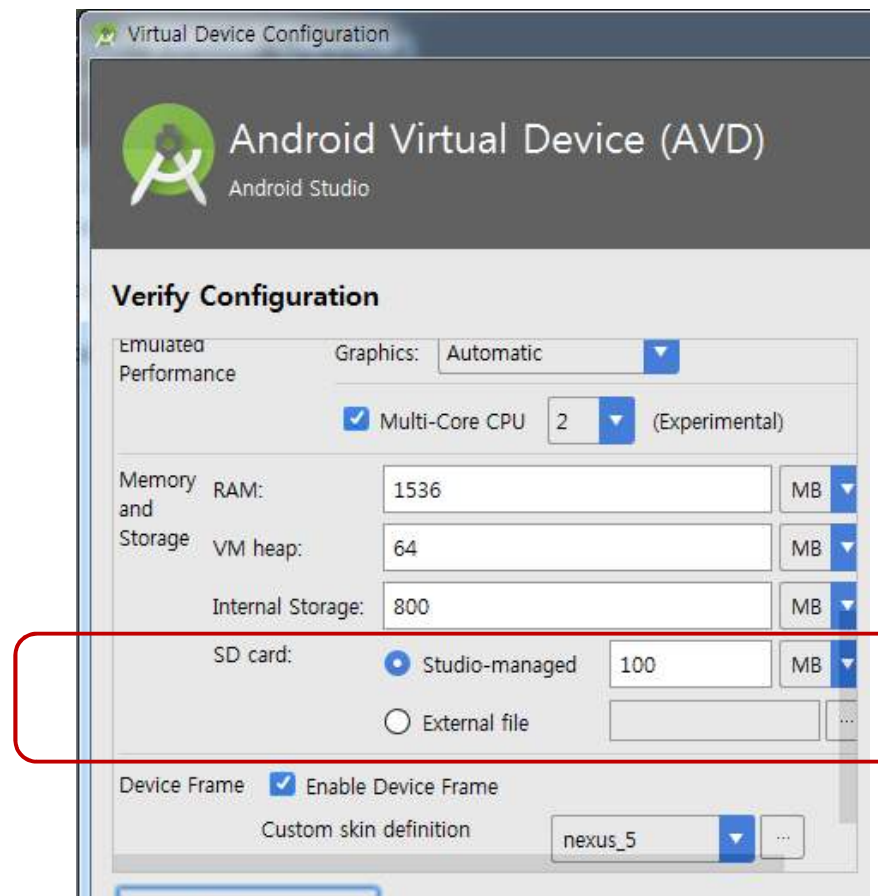
## 외부메모리 처리

CLASS	METHOD/DEFINITION
Environment	환경 변수에 접근할 때 사용하는 클래스
	File Environment.getExternalStorageDirectory() 외부 저장공간 디렉토리를 File로 리턴
	String Environment.getExternalStorageDirectory().getAbsolutePath() 외부 저장공간 경로
File	파일 및 디렉토리를 지칭하는 클래스
	boolean exists() 디렉토리 존재 여부 판단
	boolean mkdir() 디렉토리 생성
	boolean delete() 디렉토리 삭제

# 외부메모리 처리

## 가상 SD카드 확인

[AVD Manager]에서 사용하는 AVD의 [Edit] > [Show Advanced settings] > 설정된 SD카드를 확인



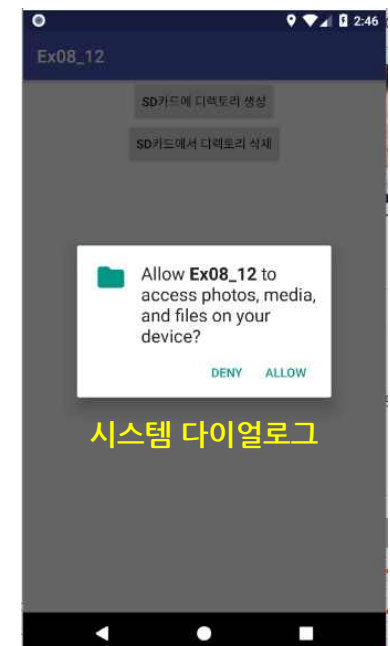
# 외부메모리 처리

## 예제8-12: Ex08\_12

SD카드에 디렉토리 생성/삭제

activity\_main.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <android.support.constraint.ConstraintLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent">
7
8     <Button
9         android:id="@+id/btnCreateDir"
10        android:layout_width="wrap_content"
11        android:layout_height="wrap_content"
12        android:text="SD카드에 디렉토리 생성"
13        android:textAllCaps="false"
14        app:layout_constraintLeft_toLeftOf="parent"
15        app:layout_constraintRight_toRightOf="parent"
16        app:layout_constraintTop_toTopOf="parent" />
17
18     <Button
19         android:id="@+id/btnDeleteDir"
20        android:layout_width="wrap_content"
21        android:layout_height="wrap_content"
22        android:text="SD카드에서 디렉토리 삭제"
23        app:layout_constraintLeft_toLeftOf="parent"
24        app:layout_constraintRight_toRightOf="parent"
25        app:layout_constraintTop_toBottomOf="@+id/btnCreateDir" />
26
27 </android.support.constraint.ConstraintLayout>
```



# 외부메모리 처리

예제8-12: Ex08\_12

AndroidManifest.xml

SD카드에 접근할 수 있도록 퍼미션 지정

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3          package="kjs.mjc.ackr.ex08_12">
4
5      <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
6
7      <application
8          android:allowBackup="true"
9          android:icon="@mipmap/ic_launcher"
10         android:label="Ex08_12"
11         android:roundIcon="@mipmap/ic_launcher_round"
12         android:supportsRtl="true"
13         android:theme="@style/AppTheme">
14         <activity android:name=".MainActivity">
15             <intent-filter>
16                 <action android:name="android.intent.action.MAIN" />
```

Read/Write 모드로 구분되어 있으나,  
Write 권한 추가 시 Read 권한 포함됨

# 외부메모리 처리

예제8-12: Ex08\_12

MainActivity.java

퍼미션 모델에 따라 런타임 권한 요청

```
14 public class MainActivity extends AppCompatActivity {  
15  
16     String strSDpath;  
17     File myDir;  
18  
19     @Override  
20     protected void onCreate(Bundle savedInstanceState) {  
21         super.onCreate(savedInstanceState);  
22         setContentView(R.layout.activity_main);  
23  
24         ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE}, MODE_PRIVATE);  
25  
26         strSDpath = Environment.getExternalStorageDirectory().getAbsolutePath();  
27         myDir = new File(strSDpath, "mydir");  
28     }
```

# 외부메모리 처리

## 예제8-12: Ex08\_12 MainActivity.java

```
29 (findViewById(R.id.btnCreateDir)).setOnClickListener(new View.OnClickListener() {  
30     @Override  
31     public void onClick(View view) {  
32         if (!myDir.exists()) {  
33             myDir.mkdir();  
34             Toast.makeText(MainActivity.this, strSDpath + "/mydir 디렉토리를 생성했습니다.", Toast.LENGTH_SHORT).show();  
35         }  
36     }  
37 });  
38  
39 (findViewById(R.id.btnDeleteDir)).setOnClickListener(new View.OnClickListener() {  
40     @Override  
41     public void onClick(View view) {  
42         myDir.delete();  
43         Toast.makeText(MainActivity.this, strSDpath + "/mydir 디렉토리가 삭제되었습니다.", Toast.LENGTH_SHORT).show();  
44     }  
45 });  
46 }  
47 }
```

에뮬레이터에서 앱을 실행하면서 adb shell에서 [mydir]의 생성, 삭제 등 확인  
/storage/emulated/0 혹은 /sdcard



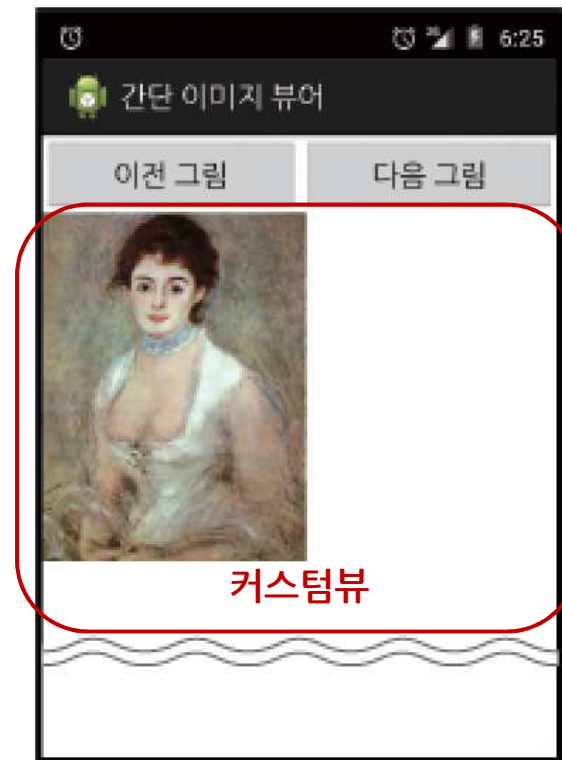
# 외부메모리 처리

## 실습8-2: Project08\_02

SD카드에 저장된 이미지를 커스텀뷰에 출력

### \* 커스텀뷰 만들기

View 클래스를 상속받아서 커스텀뷰 클래스를 구현



# 외부메모리 처리

## 실습8-2: Project08\_02

SD카드에 저장된 이미지를 커스텀뷰에 출력

### MainActivity 클래스



- SD카드에 접근하기 위한 런타임 권한 요청
- 이미지 파일의 경로 획득, 이미지 파일 배열 생성
- Bitmap 객체를 이용하여 해당 경로의 이미지 파일출력
- 버튼 클릭 시, 이미지를 MyPictureView에 출력

### MyPictureView 클래스



- 생성자: View를 상속받아 뷰 생성
- Bitmap 객체를 이용하여 해당 경로의 이미지 파일을 출력
- activity\_main.xml에  
kjs.mjc.ackr.project08\_02.MyPictureView 추가

# 외부메모리 처리

## 실습8-2: Project08\_02

### 이미지를 AVD의 외부메모리에 복사

에뮬레이터를 가동한 후, 스튜디오의 Terminal에서 이미지가 있는 폴더로 이동하여(cd ~), 이미지 5개를 복사

```
adb push img01.png /storage/emulated/0/Pictures/
```

다른 앱이 접근할 수 있는 공용디렉토리(Music, Ringtones, ...)

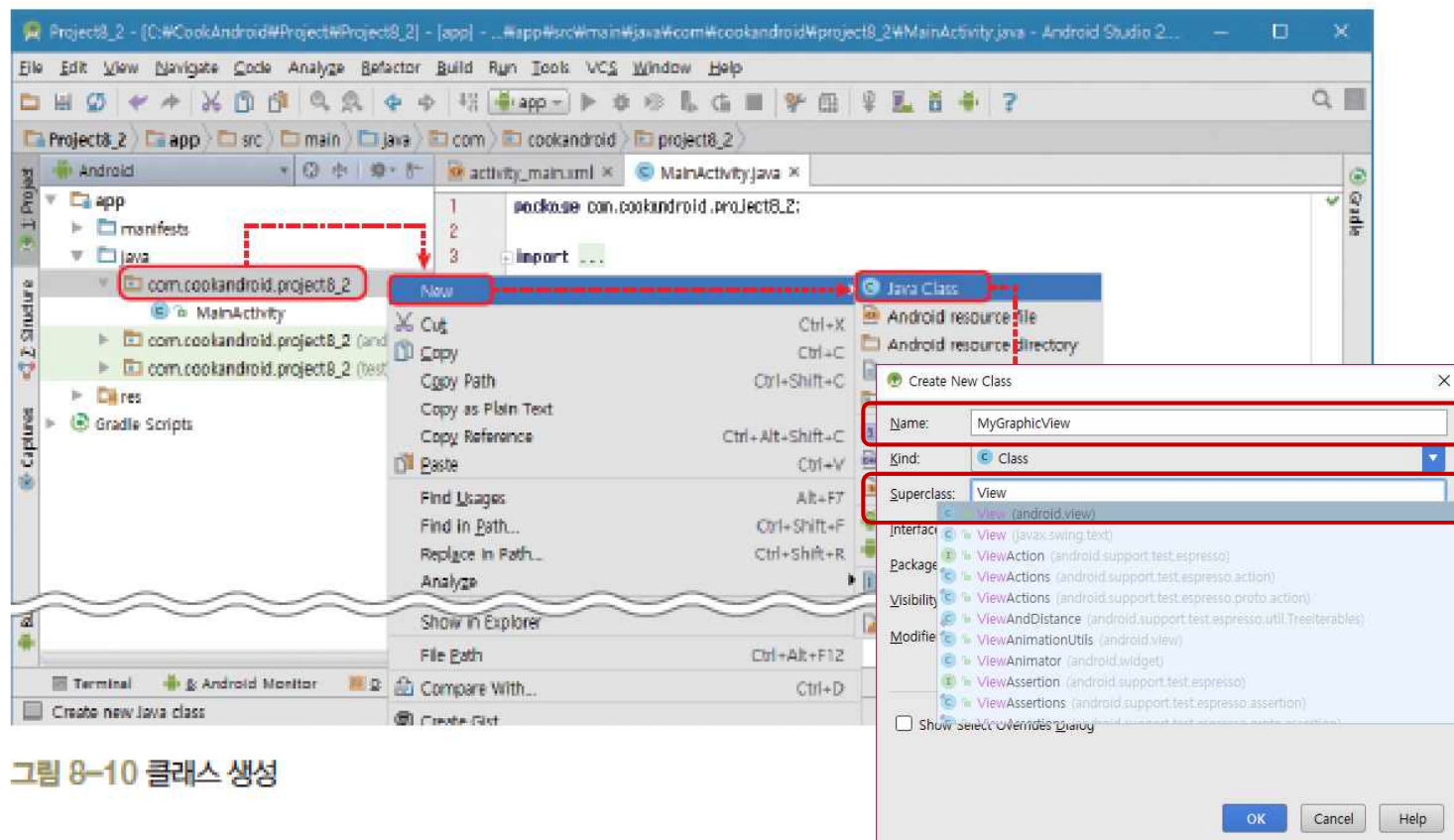
```
D:\mobile_home\mobile201801\Ch08_File>adb push img1.png /storage/emulated/0/Pictures
img1.png: 1 file pushed. 47.2 MB/s (546093 bytes in 0.011s)
```

# 외부메모리 처리

## 실습8-2: Project08\_02

### MyPictureView.java

MyPictureView로 커스텀 뷰(Custom Widget)를 직접 구현하여 activity\_main.xml에서 사용



# 외부메모리 처리

## 실습8-2: Project08\_02 MyPictureView.java

```
18 public class MyPictureView extends View {
19     String imagePath = null;
20
21     public MyPictureView(Context context, AttributeSet attrs) {
22         super(context, attrs);
23     }
24
25     @Override
26     protected void onDraw(Canvas canvas) {
27         super.onDraw(canvas);
28
29         if (imagePath != null) {
30
31             Bitmap bitmap = BitmapFactory.decodeFile(imagePath);
32             canvas.drawBitmap(bitmap, 0, 0, null);
33
34             bitmap.recycle();
35         }
36     }
37 }
38
```

XML로 View를 설정할 때  
뷰의 속성들을 사용하기 위하여 설정

커스텀뷰 생성시 자동으로 호출됨

이미지를 Bitmap 객체로 처리하여 출력(Ch.9)

# 외부메모리 처리

## 실습8-2: Project08\_02 activity\_main.xml



```
<?xml version="1.0" encoding="utf-8" ?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <Button
        android:id="@+id/btnPrev"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Previous"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toLeftOf="@+id/btnNext"
        app:layout_constraintTop_toTopOf="parent" />
    <TextView
        android:id="@+id/tvNumber"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="몇 번째"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="@id/btnPrev"
        app:layout_constraintLeft_toRightOf="@+id/btnPrev"
        app:layout_constraintRight_toLeftOf="@id/btnNext"
        app:layout_constraintTop_toTopOf="@+id/btnPrev" />
    <Button
        android:id="@+id/btnNext"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Next"
        app:layout_constraintLeft_toRightOf="@+id/btnPrev"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
    <android.support.constraint.Guideline
        android:id="@+id/guideline"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        app:layout_constraintGuide_percent="0.12" />
    <kjs.mjc.ackr.project08_02.MyPictureView
        android:id="@+id/myPictureView"
        android:layout_width="0dp"
        android:layout_height="0dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/guideline" />
</android.support.constraint.ConstraintLayout>
```

# 외부메모리 처리

## 실습8-2: Project08\_02 MainActivity.java

```
15 public class MainActivity extends AppCompatActivity {  
16  
17     //전제: storage/emulated/0/Picture에 이미지 파일만 있음  
18     Button btnPrev, btnNext;  
19     MyPictureView myPicture;  
20     int curNum;  
21     File[] imageFiles;  
22     String imageFname;  
23  
24     @Override  
25     public void onCreate(Bundle savedInstanceState) {  
26         super.onCreate(savedInstanceState);  
27         setContentView(R.layout.activity_main);  
28         setTitle("간단 이미지 뷰어");  
29  
30         ActivityCompat.requestPermissions(this, new String[]{android.Manifest.permission.WRITE_EXTERNAL_STORAGE}, MODE_PRIVATE);  
31  
32         btnPrev = findViewById(R.id.btnPrev);  
33         btnNext = findViewById(R.id.btnNext);  
34         myPicture = findViewById(R.id.myPictureView);  
35  
36         imageFiles = new File(Environment.getExternalStorageDirectory().getAbsolutePath() + "/Pictures").listFiles();  
37  
38         imageFname = imageFiles[0].toString();  
39         myPicture.imagePath = imageFname;  
40     }  
41 }
```

외부 메모리 경로의 Pictures 폴더 내 이미지 파일들을 파일배열로 저장



# 외부메모리 처리

## 실습8-2: Project08\_02 MainActivity.java

```
41 btnPrev.setOnClickListener(new View.OnClickListener() {  
42     public void onClick(View v) {  
43         if (curNum <= 0) {  
44             Toast.makeText(MainActivity.this, "첫번째 그림입니다.", Toast.LENGTH_SHORT).show();  
45         } else {  
46             curNum--;  
47         }  
48         imageFname = imageFiles[curNum].toString();  
49         myPicture.imagePath = imageFname;  
50         myPicture.invalidate(); //반드시 UI스레드에서 호출해야 함. Worker스레드에서 호출하려면 postInvalidate()를 이용  
51     }  
52 });  
53  
54 btnNext.setOnClickListener(new View.OnClickListener() {  
55     public void onClick(View v) {  
56         if (curNum >= imageFiles.length - 1) {  
57             Toast.makeText(MainActivity.this, "마지막 그림입니다.", Toast.LENGTH_SHORT).show();  
58         } else {  
59             curNum++;  
60         }  
61         imageFname = imageFiles[curNum].toString();  
62         myPicture.imagePath = imageFname;  
63         myPicture.invalidate();  
64     }  
65 });  
66 }  
67 }
```

화면 갱신

# 외부메모리 처리

## 실습: Test08\_02

TextView를 추가하고  
아래 코드의 //TODO를 완성해보자.

### ▶ 직접 풀어보기 8-2

[실습 8-2]를 다음과 같이 수정하자.

- 버튼 사이에 '현재 그림 번호/전체그림 개수' 텍스트뷰가 나타나도록 한다.
- 토스트 메시지를 없애고, 첫 번째 그림에서 <이전 그림>을 클릭하면 마지막 그림이, 마지막 그림에서 <다음 그림>을 클릭하면 첫 번째 그림이 나오도록 한다.



```
45 btnPrev.setOnClickListener(new View.OnClickListener() {
46     public void onClick(View v) {
47         if (curNum <= 0) {
48             //TODO
49         } else {
50             curNum--;
51         }
52         imageFname = imageFiles[curNum].toString();
53         myPicture.imagePath = imageFname;
54         myPicture.invalidate(); //반드시 UI스레드에서 호출해야 함. Worker스레드에서 호출하려면 postInvalidate()를 이용
55         //TODO
56     }
57 });

58
59 btnNext.setOnClickListener(new View.OnClickListener() {
60     public void onClick(View v) {
61         if (curNum >= imageFiles.length - 1) {
62             //TODO
63         } else {
64             curNum++;
65         }
66         imageFname = imageFiles[curNum].toString();
67         myPicture.imagePath = imageFname;
68         myPicture.invalidate();
69         //TODO
70     }
71 });
72 }
73 }
```

# 참고문헌

## 참고문헌

강성운(2017). 깡삼의 안드로이드 프로그래밍, 루비페이퍼.