

9주차 정리

[모바일프로그래밍 9주차 백찬은 2024.pptx.pdf](#)

내장 메모리 파일 처리

- 앱을 종료하고 다시 실행할 때, 사용한 곳에서 이어서 작업하고 싶은 경우 사용한다.
- 내장 메모리의 저장 위치: /data/data/패키지명/files 폴더
- **내장 메모리 파일 처리는 openFileInput, OutputStream등을 사용한다.**



그림 8-1 내장 메모리에서의 파일 처리

파일 처리 기본 Java 코드

예제 8-2 파일 처리의 기본 Java 코드

```
1 public void onCreate(Bundle savedInstanceState) {
2     super.onCreate(savedInstanceState);
3     setContentView(R.layout.activity_main);
4
5     Button btnRead, btnWrite;
6     btnRead = (Button) findViewById(R.id.btnRead);
7     btnWrite = (Button) findViewById(R.id.btnWrite);
8
9     btnWrite.setOnClickListener(new View.OnClickListener() {
10         public void onClick(View v) {
11             try {
12                 FileOutputStream outFs = openFileOutput("file.txt",
13                     Context.MODE_PRIVATE);
14                 String str = "북북 안드로이드";
```

```

15         outFs.write(str.getBytes());
16         outFs.close();
17         Toast.makeText(getApplicationContext(), "file.txt가 생성됨", Toast.
            LENGTH_SHORT).show();
18     } catch (IOException e) { }
19     }
20 });
21
22 btnRead.setOnClickListener(new View.OnClickListener() {
23     public void onClick(View v) {
24         try {
25             FileInputStream inFs = openFileInput("file.txt");
26             byte[] txt = new byte[30];
27             inFs.read(txt);
28             String str = new String(txt);
29             Toast.makeText(getApplicationContext(), str, Toast.LENGTH_SHORT).
                show();
30             inFs.close();
31         } catch (IOException e) {
32             Toast.makeText(getApplicationContext(), "파일 없음", Toast.LENGTH_
                SHORT).show(); }
33     }
34 });
35 }

```

FileOutputStream을 사용해서 내장 메모리에 저장할 때의 저장 위치는
/data/data/패키지명/files 이다.

여기서는 /data/data/패키지명/files/"files.txt"이다.

파일 쓰기 모드

Context.MODE_PRIVATE - 파일 내용 덮어쓰기

Context.MODE_APPEND - 파일 내용에 문자열 추가하기 (이어 쓰기)

내부 메모리 파일 처리 실습

예제 8-5 Java 코드 2

```

1  ~~~~ 생략 ~~~~
2  btnWrite = (Button) findViewById(R.id.btnWrite);
3
4  Calendar cal = Calendar.getInstance();
5  int cYear = cal.get(Calendar.YEAR);
6  int cMonth = cal.get(Calendar.MONTH);
7  int cDay = cal.get(Calendar.DAY_OF_MONTH);
8
9  dp.init(cYear, cMonth, cDay, new DatePicker.OnDateChangedListener() {
10     public void onDateChanged(DatePicker view, int year,
11         int monthOfYear, int dayOfMonth) {
12         fileName = Integer.toString(year) + "_"
13             + Integer.toString(monthOfYear + 1) + "_"
14             + Integer.toString(dayOfMonth) + ".txt";
15         String str = readDiary(fileName);
16         edtDiary.setText(str);
17         btnWrite.setEnabled(true);
18     }
19 });
20
21 }
22
23 String readDiary(String fName) {
24     return null;
25 }
26 }

```

맨 아래 버튼을 클릭했을 때 동작하는 내용을 onCreate() 내부에 완성

예제 8-7 Java 코드 4

```
1 btnWrite.setOnClickListener(new View.OnClickListener() {
2     public void onClick(View v) {
3         try {
4             FileOutputStream outFs = openFileOutput(fileName,
5                 Context.MODE_PRIVATE);
6             String str = edtDiary.getText().toString();
7             outFs.write(str.getBytes());
8             outFs.close();
9             Toast.makeText(getApplicationContext(), fileName+" 이 저장됨", Toast.
                LENGTH_SHORT).show();
10        } catch (IOException e) {
11        }
12    }
13 });
```

버튼을 누르면 editText에서 문자열을 가지고 오고 이를 저장.

[View] → [Tool Windows] - [Device Explorer]로 탐색기를 열어서
data/data/패키지명/files/"연_월_일.txt" 찾아서 열어볼 수 있음

Raw 폴더 파일 처리

- 프로젝트의 /res/raw 폴더에 필요한 파일을 저장해서 사용한다.
- openRawResource() 메서드를 사용해서 접근한다. (InputStream 클래스를 사용한다.)
- 프로젝트의 /res/raw는 프로젝트에 포함된 폴더이기에 읽기 전용만 사용 가능하다.

읽기만 수행하므로 res/raw 안에 파일을 미리 등록해야함

내장 메모리 vs raw 폴더 파일 처리 차이

- 내장 메모리에서 ~ FileInputStream 사용
- raw 폴더에서는 inputStream 사용

파일 생성 방법

- res 폴더 우 클릭 [Vew] - [Android Resource Directory]
- 창이 뜨면 Directory name: raw, resource type: raw 선택(또는 입력)
- 생성된 raw 폴더를 우 클릭한 후 [New] - [file] → raw_test.txt 파일 생성

Raw 폴더 파일 처리 Java 코드

```
btnRead.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        try {
            InputStream inputS = getResources().openRawResource(R.raw.raw_test);
            byte[] txt = new byte[inputS.available()];
            inputS.read(txt);
            edtRaw.setText(new String(txt));
            inputS.close();
        } catch (IOException e) {}
    }
});
```

getResources()의 openRawResource로 접근후 이를 InputStream 타입 클래스에 저장

inputS.available() 의미

inputS 스트림에서 읽을 수 있는 byte의 수(파일 길이에 따라 동적으로 byte수를 가져온다.)

1. SD카드 파일 처리

가상 SD 카드 확인

- MP3 파일처럼 여러 응용 프로그램에서 사용되는 경우 SD 카드에 저장해 활용한다.

- 안드로이드는 SD 카드에 저장된 데이터에 특별한 인증 절차 없이 접근 가능하다.
→ Android SDK 30 버전부터 처리 방법이 변경되어 접근 거부 가능성이 있으므로 버전을 29로 변경하면 된다. → [Gradle Scripts] - [build.gradle.kts(Model:app)] 파일의 targetSdk를 29로 변경하고 [Sync Now]를 클릭
- [Tools] - [Device Manager] 선택 → AVD 이름의 오른쪽 [:] 아이콘 클릭
→ [Edit] 선택 → [Android Virtual Device(AVD) 창에서 <Show Advanced Settings> 클릭
→ 아래로 스크롤 → 사용하는 AVD에 장착된 가상의 SD 카드 확인

2. SD카드에서 파일 읽기

1. SD 카드에서 파일 읽기

Device Explorer에서 **/sdcard** 폴더 또는 **/storage/emulated/0** 폴더에서 텍스트 파일 올리거나
ex /storage/emulated/0/Pictures (or /sdcard/Pictures) 각 폴더에서 파일 올리거나 읽어오기

2. 권한 지정 필요

AndroidManifest.xml 파일에 SD 카드를 사용할 수 있도록 퍼미션 및 application에 속성 추가

```
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<application
    android:requestLegacyExternalStorage="true"
```

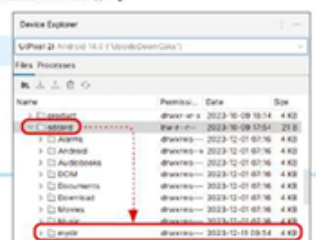
3. SD 카드에 폴더 및 파일 생성하기

예제 8-13 SD 카드에 폴더 및 파일 생성 Java 코드

```

1 public void onCreate(Bundle savedInstanceState) {
2     super.onCreate(savedInstanceState);
3     setContentView(R.layout.activity_main);
4     ActivityCompat.requestPermissions(this, new String[] {android.Manifest.
        permission.WRITE_EXTERNAL_STORAGE},MODE_PRIVATE);
5     Button btnMkdir, btnRmdir;
6     btnMkdir = (Button) findViewById(R.id.btnMkdir);
7     btnRmdir = (Button) findViewById(R.id.btnRmdir);
8     final String strSDpath =
9         Environment.getExternalStorageDirectory().getAbsolutePath();
10    final File myDir = new File(strSDpath + "/mydir");
11
12    btnMkdir.setOnClickListener(new View.OnClickListener() {
13        public void onClick(View v) {
14            myDir.mkdir();
15        }
16    });
17    btnRmdir.setOnClickListener(new View.OnClickListener() {
18        public void onClick(View v) {
19            myDir.delete();
20        }
21    });
22 }

```



SD 카드에 디렉터리 생성

SD 카드에서 디렉터리 삭제

1. Environment.getExternalStorageDirectory().getAbsolutePath()를 사용해서 외부저장소 절대 경로 찾기
2. 절대 경로 주소 + "/mydir" → 폴더를 추가하면서 파일 생성

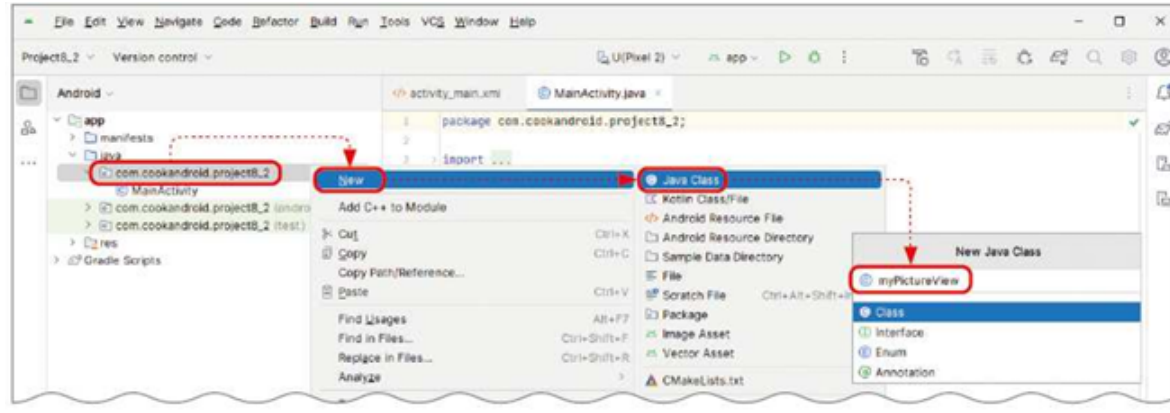
4. 특정 폴더의 하위 폴더 및 파일 목록

특정 폴더의 하위 폴더 및 파일 목록은 **File.listFiles()** 메소드를 사용하여 접근한다.
반환 값은

File[]형 즉, 배열 형태이다.

- File.listFiles() - 특정 폴더의 하위 폴더, 폴더 목록에 접근할 때 사용한다.
- File 클래스의 isDirectory()를 사용해서 디렉토리인지 파일인지 확인

커스텀 위젯



예제 8-16 myPictureView 클래스의 Java 코드 1

```

1  ~~~~ 생략(import문) ~~~~
2  public class myPictureView extends View {
3
4      public myPictureView(Context context, @Nullable AttributeSet attrs) {
5          super(context, attrs);
6

```

```

<com.cookandroid.project8_2.myPictureView
    android:id="@+id/myPictureView1" />

```

1. 프로젝트의 [java] - [패키지명] 에서 우클릭
2. [New] - [Java Class] 선택 - [Create New Class]의 Name에 원하는 입력 후 OK
3. extends View
4. 커스텀 클래스 내부에서 View(context:Context, attrs:AttributeSet)를 오버라이딩

[Code] - [Override Methods]에서 파라미터 2개 있는 것 선택해서 오버라이딩

5. 커스텀 위젯을 xml에서 사용할 때, <패키지명.클래스명 ~/> 형식으로 불러와 사용한다.

교재 요약본

- 내장 메모리의 저장 위치는 /data/data/패키지명/files 이다.
- 프로젝트의 /res/raw 폴더에 필요한 파일을 저장하여 사용할 수 있으나 읽기 전용이다.
- Device Explorer는 **AVD나 실제 안드로이드 폰 내의 폴더와 파일에 접근하는 기능**을 제공한다.
- SD카드의 기본적인 경로에는 /sdcard를 사용하고 [Device Explorer]에서 /storage/emulated/0 에 접근하면 된다.
- SD 카드의 파일을 좀 더 정밀하게 처리하려면 Environment 클래스의 정적 메소드를 이용해야한다. → Environment.getExternalStorageDirectory().getAbsolutePath()
- SD 카드에 읽기, 스기를 하려면 AndroidManifest.xml에 퍼미션을 추가해야한다.
- 지정한 폴더의 하위 폴더 및 파일 목록에 접근하려면 File.listFiles() 메소드를 사용한다.
- 커스텀 위젯을 만들려면 먼저 Java 코드로 작성한 다음 activity_main.xml에서 <패키지명.클래스 ~ /> 형식을 사용한다.
- Android SDK 버전을 29로 사용하기 위해 변경하는 파일 이름은 build.gradle.kts이다.
- 이미지 뷰어를 작성하기 위해 새로운 클래스를 생성하고 View 클래스를 상속 받는다.