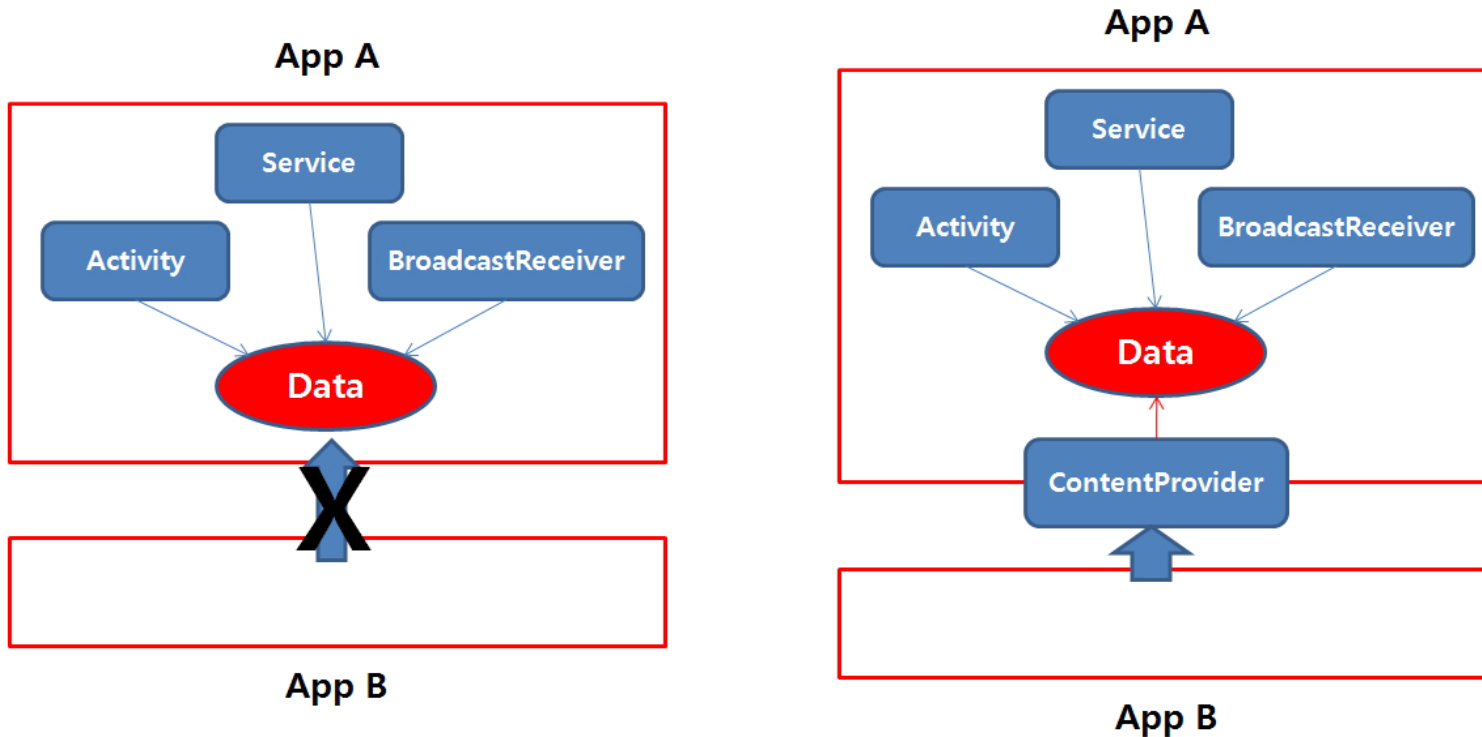


21.1 ContentProvider이해

21.1.1. 콘텐츠 프로바이더 구조

- 콘텐츠 프로바이더는 앱 간의 데이터 공유를 목적으로 사용 되는 컴포넌트



21.1 ContentProvider이해

21.1.2. 콘텐츠 프로바이더 작성법

```
public class MyContentProvider extends ContentProvider {  
    public MyContentProvider() { }  
    public int delete(Uri uri, String selection, String[] selectionArgs) {  
        return 0;  
    }  
    public String getType(Uri uri) {  
        throw new UnsupportedOperationException("Not yet implemented");  
    }  
    public Uri insert(Uri uri, ContentValues values) {  
        return null;  
    }  
    public boolean onCreate() {  
        return false;  
    }  
    public Cursor query(Uri uri, String[] projection, String selection,  
        String[] selectionArgs, String sortOrder) {  
        return null;  
    }  
    public int update(Uri uri, ContentValues values, String selection,  
        String[] selectionArgs) {  
        return 0;  
    }  
}
```



21.1 ContentProvider 이해

- AndroidManifest.xml에 등록

```
<provider  
    android:name=".MyContentProvider"  
    android:authorities="com.example.test.Provider"  
    android:enabled="true"  
    android:exported="true"></provider>
```

- URL은 규칙
- content:// com.example.test.Provider
- content:// com.example.test.Provider/user
- content:// com.example.test.Provider/user/1

content:// com.example.test.Provider/user/1

└──┬──┬──┘

Scheme host path



21.1 ContentProvider 이해

21.1.3. 콘텐츠 프로바이더 이용

```
Cursor cursor=getContentResolver().query(uri, null, null, null, null);
```

- query(Uri uri, String[] projection, String selection, String[] selectionArgs, String sortOrder)
- insert(Uri url, ContentValues values)
- update(Uri uri, ContentValues values, String where, String[] selectionArgs)
- delete(Uri url, String where, String[] selectionArgs)



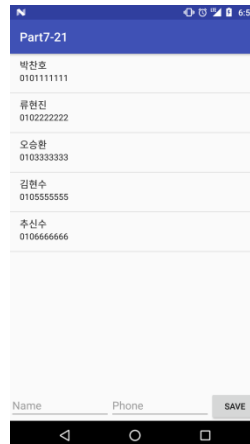
Step by Step 21-1 - ContentProvider

ContentProvider 테스트

두개의 앱을 만들어 테스트

하나의 앱이 데이터를 DBMS 방식으로 가지고 있고 다른 앱이 이 데이터를 ContentProvider로 이용한다는 가정

1. Module 생성
2. 파일 복사
3. Part7-21-provider 작업 - ContentProvider 작성
4. Part7-21 작업 - AndroidManifest.xml 작업
5. 실행



21.2 구글 기본 앱의 ContentProvider 이용

21.2.1. 주소록 앱 데이터 획득

```
<uses-permission android:name="android.permission.READ_CONTACTS"/>
```

- 주소록의 목록 액티비티

```
Intent intent = new Intent(Intent.ACTION_PICK);  
intent.setData(Uri.parse("content://com.android.contacts/data/phones"));  
startActivityForResult(intent, 10);
```

- 콘텐츠 프로바이더가 이용

```
Cursor cursor = getContentResolver().query(  
    ContactsContract.Data.CONTENT_URI,  
    new String[] {  
        ContactsContract.Contacts.DISPLAY_NAME,  
        ContactsContract.CommonDataKinds.Phone.NUMBER },  
    ContactsContract.Data._ID + "=" + id, null, null);
```



21.2 구글 기본 앱의 ContentProvider 이용

21.2.2. 갤러리 앱 데이터 획득

- API Level 16 하위 버전

```
if (Build.VERSION.SDK_INT < Build.VERSION_CODES.JELLY_BEAN) {  
    //jelly bean 하위에서는 하나만 선택하게..  
    Intent intent = new Intent(Intent.ACTION_PICK);  
    intent.setType(MediaStore.Images.Media.CONTENT_TYPE);  
    intent.setData(MediaStore.Images.Media.EXTERNAL_CONTENT_URI);  
    startActivityForResult(intent, 20);  
}
```

- onActivityResult () 함수

```
String[] projection = {MediaStore.Images.Media.DATA};  
Cursor cursor = getContentResolver().query(data.getData(), projection, null, null, null);  
cursor.moveToFirst();  
String filePath = cursor.getString(0);
```



21.2 구글 기본 앱의 ContentProvider 이용

- API Level 16 이상 버전
- 인텐트를 발생

```
Intent intent = new Intent();
intent.setType("image/*");
intent.putExtra(Intent.EXTRA_ALLOW_MULTIPLE, true);
intent.setAction(Intent.ACTION_GET_CONTENT);
startActivityForResult(intent, 30);
```

- 결과 획득

```
if (data.getClipData() != null) {
    ClipData clipData = data.getClipData();
    for (int i = 0; i < clipData.getItemCount(); i++) {
        ClipData.Item item = clipData.getItemAt(i);
        Uri uri = item.getUri();
        if ("com.android.providers.media.documents".equals(uri.getAuthority()) &&
            Build.VERSION.SDK_INT
            >= 19) {
            //...
        } else if ("external".equals(uri.getPathSegments().get(0))) {
            //...
        }
    }
} else {
}
```

21.2 구글 기본 앱의 ContentProvider 이용

- ClipData 타입으로 전달되었을 때 API Level 19(Android 4.4)부터 결과 데이터의 Uri 표현식이 이전 버전과 달라져서 이 또한 구분해서 처리
- ① 도큐먼트식 경로:
content://com.android.providers.media.documents/document/image:3A35260
- ② 세그먼트식 경로:
content://media/external/images/media/....
- 세그먼트식 이미지 파일 경로 획득

```
String selection = MediaStore.Images.Media._ID + "=?";
String[] selectionArgs = new String[] {uri.getLastPathSegment()};

String column = "_data";
String[] projection = {column};

Cursor cursor = getContentResolver().query(MediaStore.Images.Media.EXTERNAL_CONTENT_URI, projection, selection,
selectionArgs, null);
String filePath=null;
if (cursor != null && cursor.moveToFirst()) {
    int column_index = cursor.getColumnIndexOrThrow(column);
    filePath = cursor.getString(column_index);
}
cursor.close();
```



21.2 구글 기본 앱의 ContentProvider 이용

- 도큐먼트식 이미지 파일 경로 획득

```
String docId = DocumentsContract.getDocumentId(uri);
String[] split = docId.split(":");
String type = split[0];
Uri contentUri = null;
if ("image".equals(type)) { //video, audio type 도 가능
    contentUri = MediaStore.Images.Media.EXTERNAL_CONTENT_URI;
}

String selection = MediaStore.Images.Media._ID + "=?";
String[] selectionArgs = new String[]{split[1]};

String column = "_data";
String[] projection = {column};

Cursor cursor = context.getContentResolver().query(contentUri, projection, selection, selectionArgs, null);

String filePath = null;
if (cursor != null && cursor.moveToFirst()) {
    int column_index = cursor.getColumnIndexOrThrow(column);
    filePath = cursor.getString(column_index);
}
cursor.close();
```



Step by Step 21-2 – 주소록, 갤러리 앱 연동

주소록, 갤러리 앱 연동을 테스트

두 앱 모두 목록 화면을 띄우고 사용자가 선택한 데이터를 단순 화면 출력하는 형태로 테스트

1. Activity 생성
2. 파일 복사
3. AndroidManifest.xml 작업
4. Lab21_2Activity – insertImageView () 작성
5. Lab21_2Activity – getFileFromDocumentUri() 작성
6. Lab21_2Activity – getFileFromUriSegment() 작성
7. Lab21_2Activity – onClick() 작성
8. Lab21_2Activity – onActivityResult() 작성
9. Lab21_2Activity.java 실행

