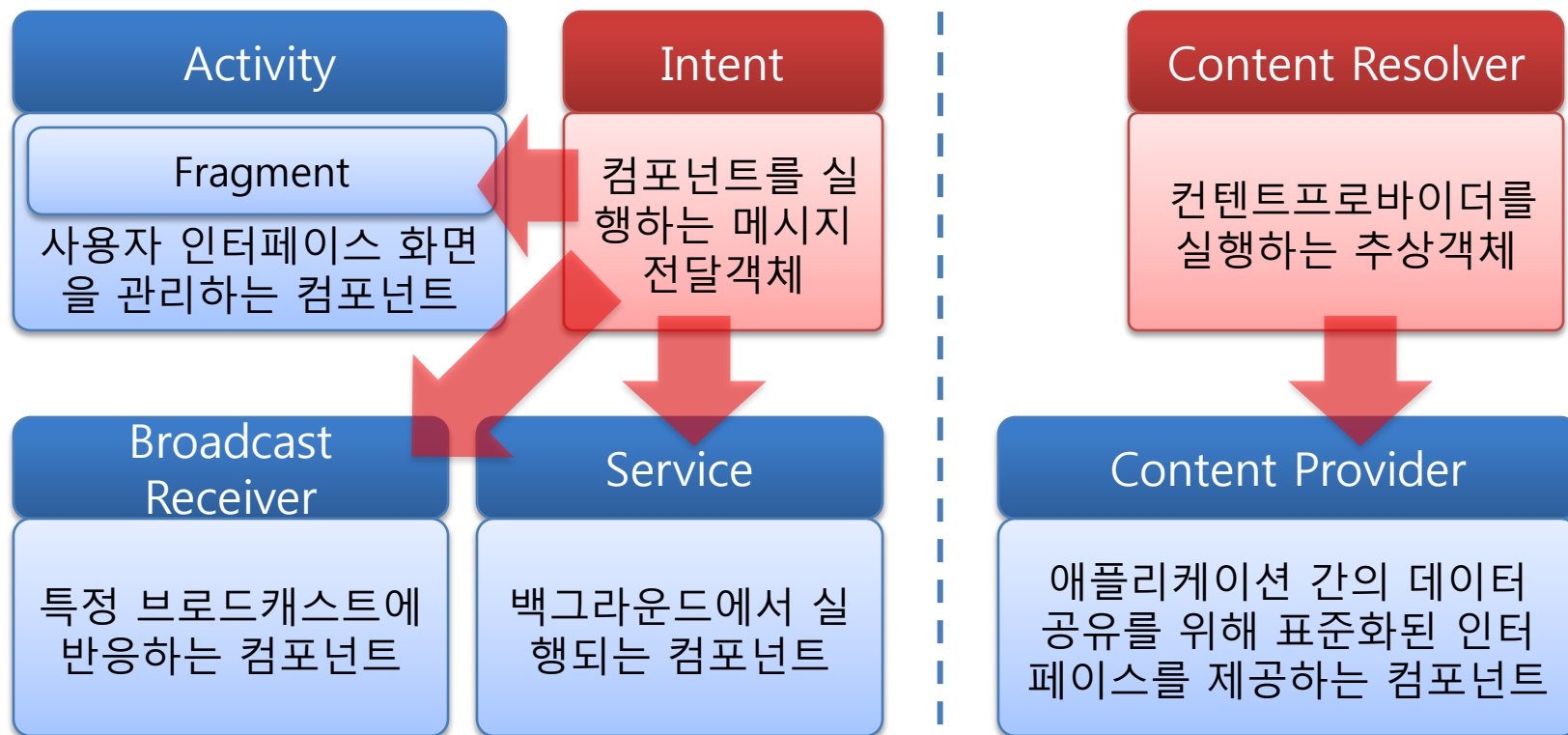


Android

007 Intent

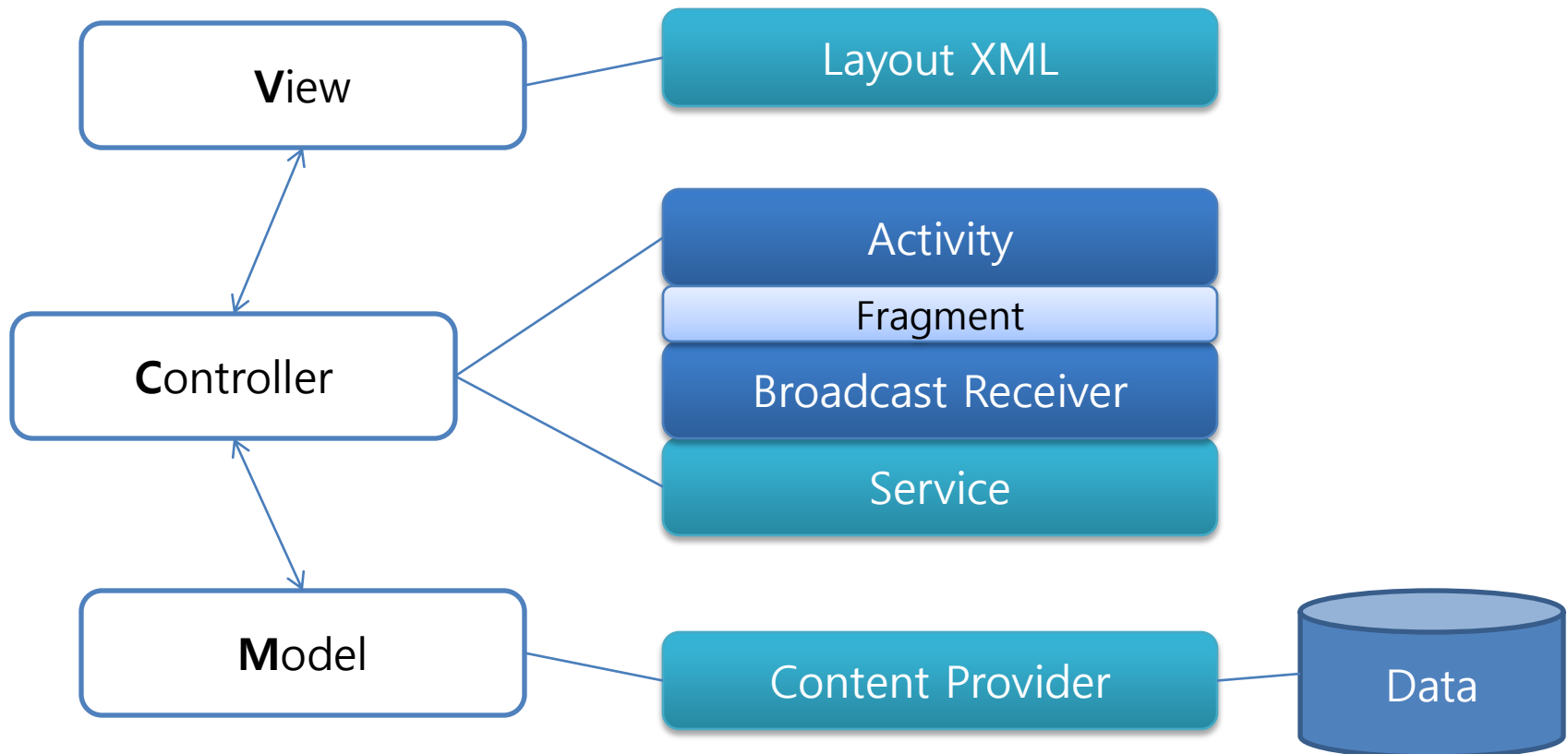
1. App Components

- 안드로이드는 App 을 구성하는 4개의 핵심 컴포넌트가 있으며 이들은 각각의 독립적인 Life Cycle 에 의해서 실행된다. 왼쪽의 세개의 컴포넌트는 Intent 에 의해, 나머지 한개인 Content Provider 는 Resolver에 의해 실행된다



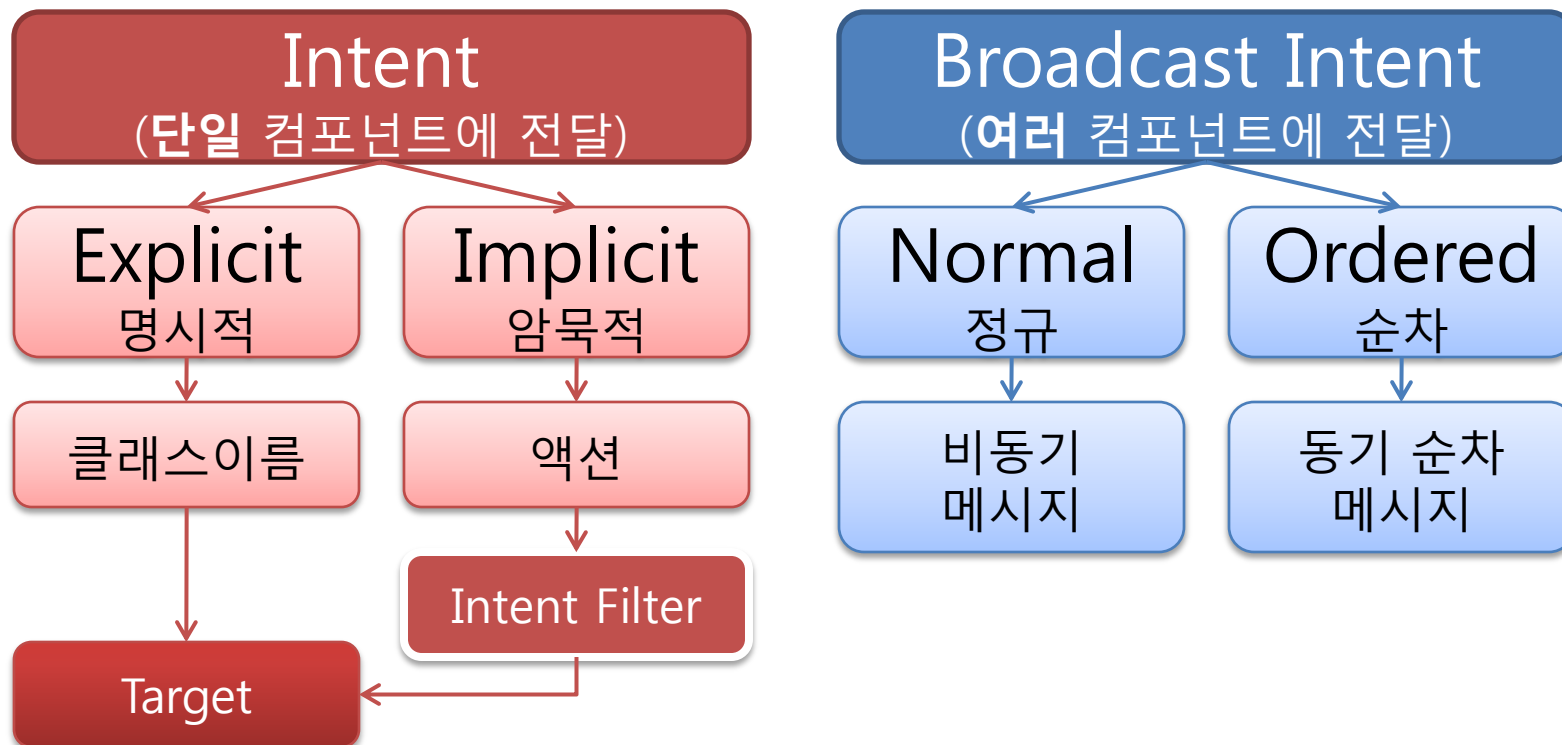
2. App Architecture

- 안드로이드 어플리케이션은 기본적으로 Model(데이터)-View(화면)-Controller(제어) 의 MVC 아키텍처로 구성된다



3. Intent

- 하나의 컴포넌트가 **다른 컴포넌트를 실행시킬 수 있는 메커니즘**
- 일반적으로 명시적인테트와 암묵적인테트 두 가지가 있으며, 다른 분류로는 Broadcast Receiver 에서 사용되는 Broadcast Intent가 있다. Broadcast 인테트는 단일 컴포넌트가 아닌 **시스템 전역에 메시지를 전달한다**



4. Explicit Intent

- 명시적 인텐트는 호출하고자 하는 컴포넌트를 지정해서 메시지를 전달한다
- 아래와 같이 버튼 클릭시에 특정 Activity를 지정해서 호출할 수 있다

```
@Override
public void onClick(View v) {
    //Intent를 생성하면서 호출대상이 되는 클래스를 명시적으로 지정한다
    Intent intent = new Intent(this, 클래스이름.class);
    //putExtra api를 이용해 값을 전달한다. 변수명은 msg 이다
    intent.putExtra("msg", et_msg.getText().toString());
    //위에서 지정한 클래스이름을 가진 Activity를 실행한다
    startActivity(intent);
}
```

- 호출되는 Activity의 onCreate() 함수에서 전달된 값을 받을 수 있다

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    Intent intent = this getIntent();
    String msg = intent.getStringExtra("msg"); // 변수명 msg 로 값을 꺼낼 수 있다
}
```

5. Implicit Intent

- 암묵적 인텐트는 기 정의되어 있는 Action으로 호출할 때 사용된다

```
@Override
public void onClick(View v) {
    // Intent를 생성하면서 호출 액션을 지정하고 필요한 값을 입력한다
    // 아래는 전화를 거는 인텐트 액션예제이다
    Intent intent = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:01012341234"));
    // Action_DIAL에 해당하는 Activity를 실행한다
    startActivity(intent);
}
```

- 안드로이드에는 이외에도 아래와 같은 Built-in Action들이 있다

```
// 전화 걸기
Intent i = new Intent(Intent.ACTION_DIAL, Uri.parse("tel:전화번호"));
// 문자 보내기
Intent i = new Intent(Intent.ACTION_SENDTO, Uri.parse("smsto:전화번호"));
// 웹 브라우저 실행
Intent i = new Intent(Intent.ACTION_VIEW, Uri.parse("http://www.naver.com"));
```

6. Intent Filter

- 안드로이드에 내장된 Action 말고도 Intent Filter를 이용해서 암묵적으로 컴포넌트를 호출할 수 있다. 안드로이드 manifest.xml 에 아래와 같이 추가한다

```
<activity android:name=".클래스이름">
    <intent-filter>
        <action android:name="action.customaction"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
```

- 코드에서 위의 action android:name 에 정의된 이름으로 호출할 수 있다

```
@Override
public void onClick(View v) {
    //Intent를 생성하면서 manifest에 정의된 액션명으로 호출한다
    Intent intent =new Intent("action.customaction");
    startActivity(intent);
}
```

6.1. Intent Filter 구성요소

- 인텐트 필터는 action 과 category 뿐만이 아니라 아래와 같이 다양한 구성요소를 가지고 있다

구성요소	설명
Action	수행할 액션의 이름
Category	수행할 액션에 대한 추가정보
Data	수행할 데이터의 URI (tel , http, smsto 등)
Type	수행할 인텐트 데이터의 타입(application/pdf)
Component name	대상 클래스의 이름
Extras	컴포넌트에 추가적으로 전달할 키/값

6.2. Intent Filter 주요 액션

- 안드로이드에는 이미 정의된 내장 액션이 있다. (아래는 자주사용되는 액션)

Action	설명
ACTION_MAIN	시작 액티비티를 지정하기 위한 액션
ACTION_VIEW	데이터의 URL로 가장 적절한 액티비티를 호출하는 액션 content://contacts/people/1
ACTION_DEFAULT	ACTION_VIEW와 동일
ACTION_EDIT	수정하기 위해 호출하는 액션
ACTION_DELETE	삭제하기 위해 호출하는 액션
ACTION_DIAL	전화 다이얼 액티비티를 호출하는 액션 content://contacts/people/1 tel:0101235678
ACTION_CALL	전화 액티비티를 호출하는 액션 tel:0101235678
ACTION_WEB_SEARCH	웹 검색 액티비티를 호출하는 액션
ACTION_SEARCH	검색 액티비티를 호출하는 액션
ACTION_SENDTO	이메일 등의 메시지 전송을 지정하는 액션
ACTION_ANSWER	전화 착신을 위한 액션

7.1. 표준 Category

카테고리는 인텐트 대상에 대한 분류 정보를 지정하는 속성이다. `addCategory(String)` 메소드로 추가할 수 있으며 카테고리를 지정하여 원하는 대상을 명확히 지정할 수 있다.

- CATEGORY_DEFAULT 가장 기본이 되는 액티비티
- CATEGORY_BROWSABLE 웹 브라우징을 할 수 있는 액티비티
- CATEGORY_TAB TabActivity 내에서 사용되는 탭을 지정
- CATEGORY_ALTERNATIVE 사용자가 현재 보고 있는 데이터의 대체 액션
- CATEGORY_SELECTED_ALTERNATIVE 사용자가 현재 선택한 데이터의 대체 액션
- CATEGORY_LAUNCHER 애플리케이션에서 가장 먼저 실행되는 액티비티
- CATEGORY_INFO 패키지에 대한 정보를 제공하는 액티비티를 지정
- CATEGORY_HOME 가장 처음 실행되는 액티비티를 지정
- CATEGORY_PREFERENCE 프리퍼런스 액티비티
- CATEGORY_TEST 테스트로 사용할 경우 지정
- CATEGORY_CAR_DOCK 기기가 차량용 거치대에 장착되었을 때 호출되는 액티비티를 지정
- CATEGORY_DESK_DOCK 기기가 책상용 거치대에 장착되었을 때 호출되는 액티비티를 지정
- CATEGORY_CAR_MODE 차량에서 사용할 수 있는 액티비티를 지정

7.2. 표준 Activity Action

액티비티 액션은 액티비티를 어떻게 실행할 지를 지정하는 속성이다. 이 액션들은 액티비티를 실행할 때 사용하는 `startActivity(Intent)` 메소드에 지정하는 인텐트 객체에 설정할 수 있다.

- ACTION_MAIN 시작 액티비티를 지정하는 액션
- ACTION_VIEW 데이터를 사용자에게 보여주는 액션
- ACTION_ATTACH_DATA 데이터가 첨부되었다는 것을 알려주는 액션
- ACTION_EDIT 데이터 수정을 요청하는 액션
- ACTION_PICK 데이터에서 원하는 것을 선택하는 액션
- ACTION_CHOOSER 액티비티 선택 화면을 보여주는 액션
- ACTION_GET_CONTENT 사용자가 선택한 데이터를 반환하는 액션
- ACTION_DIAL 전화 다이얼 액티비티를 호출하는 액션
- ACTION_CALL 전화를 거는 액션
- ACTION_SEND 데이터를 전송하는 액션
- ACTION_SENDTO 데이터를 지정된 곳에 전송하는 액션
- ACTION_ANSWER 수신중인 전화를 다루는 액션
- ACTION_INSERT 주어진 컨테이너로 아이템을 삽입하는 액션
- ACTION_DELETE 주어진 컨테이너의 데이터를 삭제하는 액션
- ACTION_RUN 데이터를 실행하는 액션
- ACTION_SYNC 데이터 동기화를 수행하는 액션
- ACTION_PICK_ACTIVITY 액티비티를 골라주는 액션
- ACTION_SEARCH 검색을 수행하는 액션
- ACTION_WEB_SEARCH 웹 검색을 수행하는 액션
- ACTION_FACTORY_TEST 공장 테스트를 나타내는 액션

7.3. 표준 Broadcast Action

다음은 브로드캐스트 리시버가 받을 수 있는 액션들이다. 이 액션들은 `registerReceiver(BroadcastReceiver, IntentFilter)` 메소드의 `IntentFilter`에 정의하거나 `AndroidManifest.xml`의 태그 안에 정의할 수 있다.

`ACTION_TIME_TICK` 현재 시간이 변경되었음을 알려주는 액션

`ACTION_TIME_CHANGED` 시간이 설정되었음을 알려주는 액션

`ACTION_TIMEZONE_CHANGED` 시간대가 변경되었음을 알려주는 액션

`ACTION_BOOT_COMPLETED` 기기 부팅이 완료되었음을 알려주는 액션

`ACTION_PACKAGE_ADDED` 패키지가 추가되었음을 알려주는 액션

`ACTION_PACKAGE_CHANGED` 패키지가 변경되었음을 알려주는 액션

`ACTION_PACKAGE_REMOVED` 패키지가 제거되었음을 알려주는 액션

`ACTION_PACKAGE_RESTARTED` 패키지가 다시 시작되었음을 알려주는 액션

`ACTION_PACKAGE_DATA_CLEARED` 사용자가 패키지 데이터를 정리했음을 알려주는 액션

`ACTION_UID_REMOVED` 시스템에서 사용자 아이디가 제거되었음을 알려주는 액션

`ACTION_BATTERY_CHANGED` 배터리 정보가 변경되었음을 알려주는 액션

`ACTION_POWER_CONNECTED` 외부 전원이 연결되었음을 알려주는 액션

`ACTION_POWER_DISCONNECTED` 외부 전원 연결이 해지되었음을 알려주는 액션

`ACTION_SHUTDOWN` 기기가 종료될 것이라는 것을 알려주는 액션

7.4. 표준 Extra data

다음은 추가 데이터로 사용할 수 있는 필드이다. `putExtra(String, Bundle)` 메소드와 함께 사용된다.

EXTRA_ALARM_COUNT AlarmManager가 사용하는 알람 개수

EXTRA_BCC 이메일의 숨은 참조 리스트

EXTRA_CC 이메일의 참조 리스트

EXTRA_DATA_REMOVED 데이터 전부 제거인지 아닌지를 알려주는 불린 값
(`ACTION_PACKAGE_REMOVED`에서 사용)

EXTRA_DOCK_STATE 거치대 상태를 요청할 때 사용하는 데이터(`ACTION_DOCK_EVENT`에서 사용)

EXTRA_DOCK_STATE_HE_DESK 거치대가 디지털인지를 알려주는 숫자 값

EXTRA_DOCK_STATE_LE_DESK 거치대가 아날로그인지를 알려주는 숫자 값

EXTRA_DOCK_STATE_CAR 거치대가 차량용인지를 알려주는 숫자 값

EXTRA_DOCK_STATE_DESK 거치대가 책상용인지를 알려주는 숫자 값

EXTRA_DOCK_STATE_UNDOCKED 기기가 거치대에 없다는 것을 알려주는 숫자 값

EXTRA_DONT_KILL_APP 애플리케이션을 재시작하는 기본 액션을 재정의하기 위해

`ACTION_PACKAGE_REMOVED`나 `ACTION_PACKAGE_CHANGED`에서 사용하는 여분의 불린 값

EXTRA_EMAIL 전송할 이메일 주소

EXTRA_INITIAL_INTENTS `ACTION_CHOOSER`에 의해 보여질 인텐트들

EXTRA_INTENT `ACTION_PICK_ACTIVITY`에 사용할 인텐트 정보

EXTRA_KEY_EVENT 인텐트에 의해 호출될 이벤트 정보를 나타내는 `KeyEvent` 객체

EXTRA_PHONE_NUMBER 전화 액션 관련해 저장할 전화번호

EXTRA_REMOTE_INTENT_TOKEN 원격 인텐트에 사용할 추가 정보

EXTRA_REPLACING 패키지가 교체를 알리기 위해 `ACTION_PACKAGE_REMOVED`에 사용하는 불린 값

7.5. 표준 Flag

인텐트에 사용할 수 있는 플래그(Flags)이다. 인텐트에 플래그를 설정해서 특정 정보를 전달할 수 있으며 대부분의 플래그는 액티비티의 동작 방식을 변경하는 용도로 쓰인다. `setFlags(int)`와 `addFlags(int)` 메소드가 함께 사용된다.

`FLAG_GRANT_READ_URI_PERMISSION` URI 읽기 권한 플래그

`FLAG_GRANT_WRITE_URI_PERMISSION` URI 쓰기 권한 플래그

`FLAG_FROM_BACKGROUND` 백그라운드 동작임을 알려주는 플래그

`FLAG_ACTIVITY_BROUGHT_TO_FRONT` 태스크에 있는 액티비티가 다시 실행되었을 때 시스템이 설정하는 플래그

`FLAG_ACTIVITY_CLEAR_TASK` 태스크를 초기화하는 플래그

`FLAG_ACTIVITY_CLEAR_TOP` 실행할 액티비티가 태스크에 이미 있다면 태스크에 있는 동일한 액티비티부터 최상단의 액티비티까지 모두 제거하고 새로운 액티비티를 실행하는 플래그

`FLAG_ACTIVITY_EXCLUDE_FROM_RECENTS` "최근 실행 액티비티 목록"에 보이지 않도록 하는 플래그

`FLAG_ACTIVITY_FORWARD_RESULT` 액티비티 처리 결과를 전달할 수 있는 플래그

`FLAG_ACTIVITY_MULTIPLE_TASK` 액티비티를 항상 새로운 태스크에서 실행하는 플래그

`FLAG_ACTIVITY_NEW_TASK` 실행한 액티비티와 관련된 태스크가 존재하면 동일한 태스크 내에서 실행하고 그렇지 않으면 새로운 태스크에서 액티비티를 실행하는 플래그

`FLAG_ACTIVITY_NO_HISTORY` 실행된 액티비티를 태스크 스택에서 관리하지 않도록 해주는 플래그

`FLAG_ACTIVITY_REORDER_TO_FRONT` 실행된 액티비티가 태스크에 이미 존재하는 액티비티라면 새로 실행하지 않고 태스크에 있는 액티비티를 스택 최상단으로 옮기는 플래그

`FLAG_ACTIVITY_SINGLE_TOP` 실행할 액티비티가 태스크 스택 최상단에 이미 있다면 액티비티를 다시 실행하지 않는 플래그

`FLAG_ACTIVITY_CLEAR_TASK` 태스크를 초기화하는 플래그