

모바일 프로그래밍

05 액티비티와 인텐트 1

2017 2학기

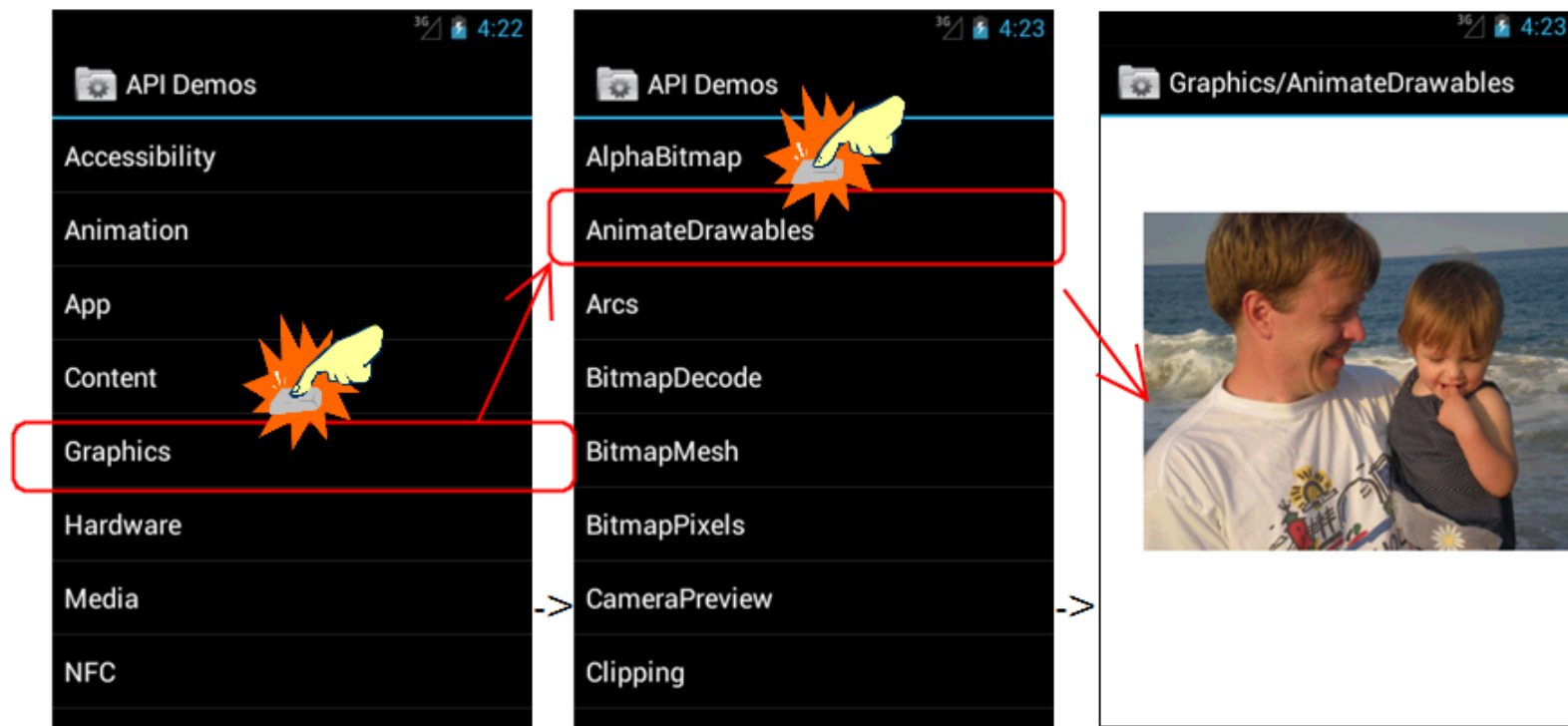
강승우

안드로이드 애플리케이션

- Android application components
 - Activity
 - Service
 - Intent/Broadcast Receiver
 - Content Provider
- 보통 한 개 이상의 액티비티(Activity)로 구성

액티비티 (Activity)

- 일반적으로 UI를 갖는 하나의 스크린을 나타냄
- 안드로이드 애플리케이션은 반드시 하나의 “launcher” (or “main”) activity를 가지고 있어야 함
 - Launcher activity: 애플리케이션의 시작 포인트가 되는 액티비티
- 각 액티비티는 AndroidManifest.xml 파일에 등록되어 있어야 함
- 하나 이상의 View를 가질 수 있음
- Activity lifecycle callback 메소드를 통해 실행이 관리됨



첫 번째 액티비티 ->

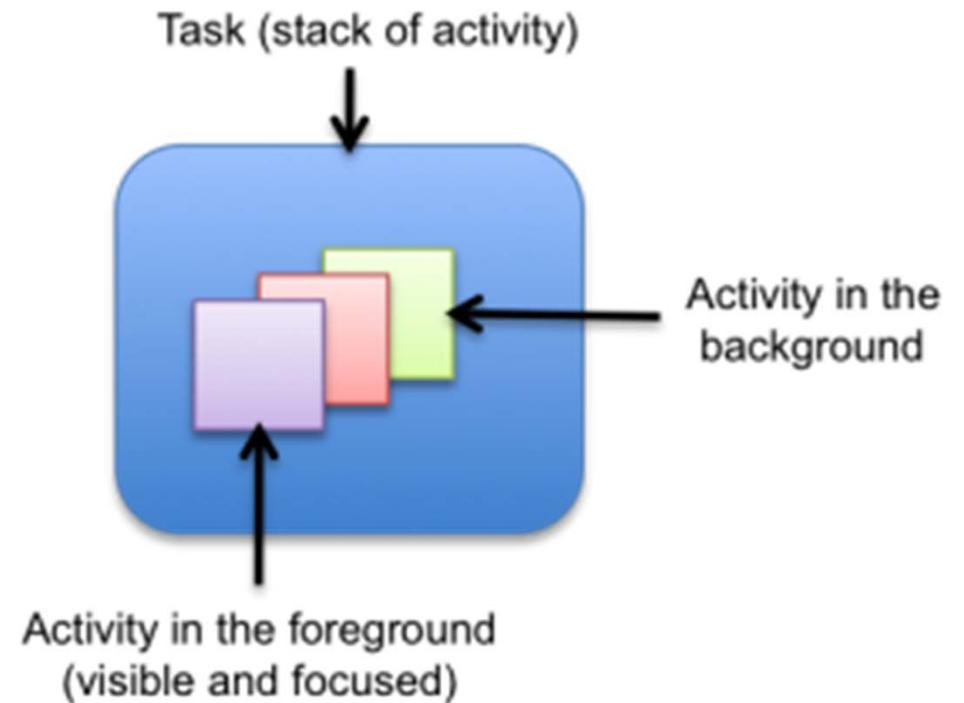
두 번째 액티비티 ->

세 번째 액티비티

Task

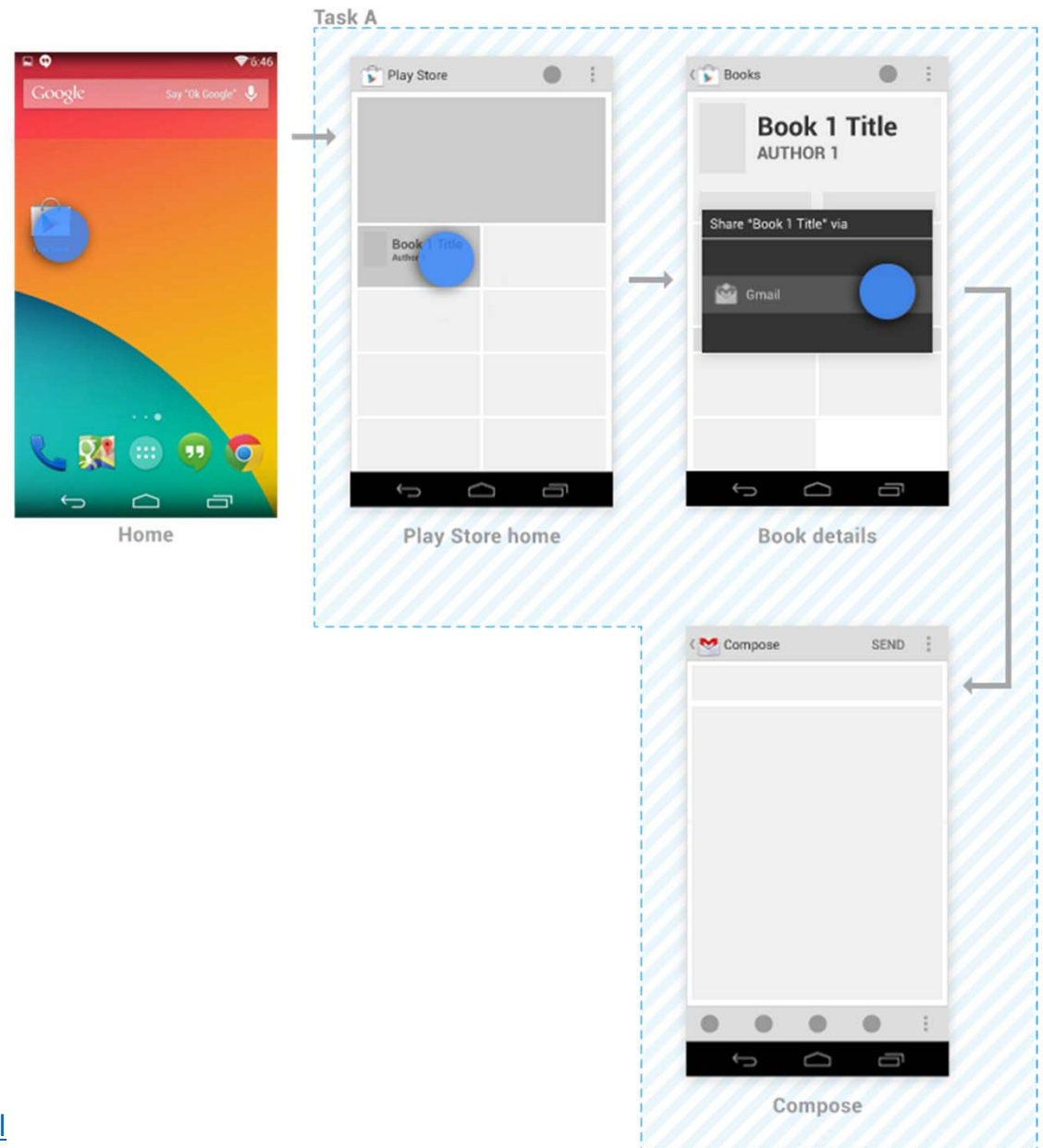
- 일련의 실행 흐름에 속한 액티비티들의 그룹
 - 사용자가 수행하고자 하는 작업을 위해 따르는 액티비티의 시퀀스
 - 서로 다른 애플리케이션의 액티비티도 하나의 태스크에 속할 수 있음

- Activity의 Stack으로 관리 됨
 - “Back Stack”

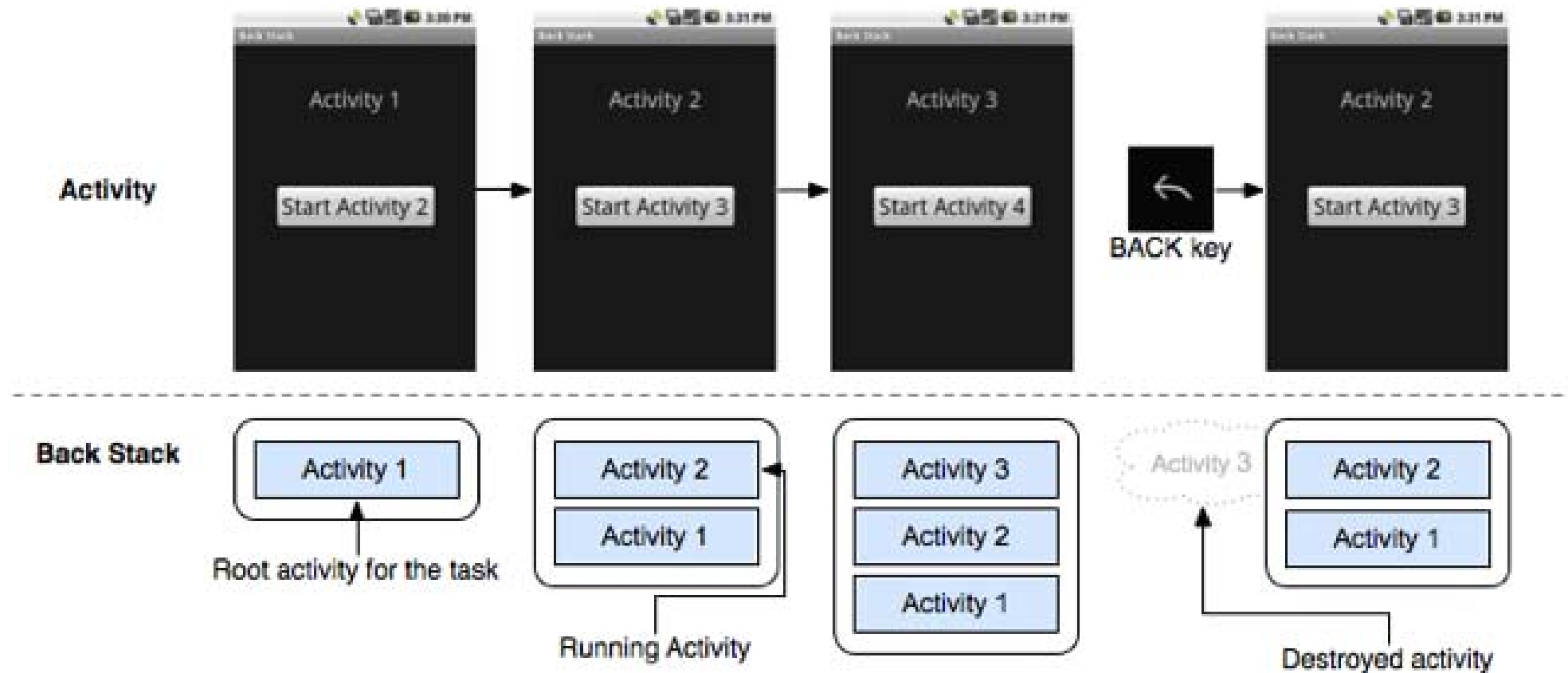


Example

- 서로 다른 애플리케이션과 액티비티의 실행 흐름



액티비티 스택

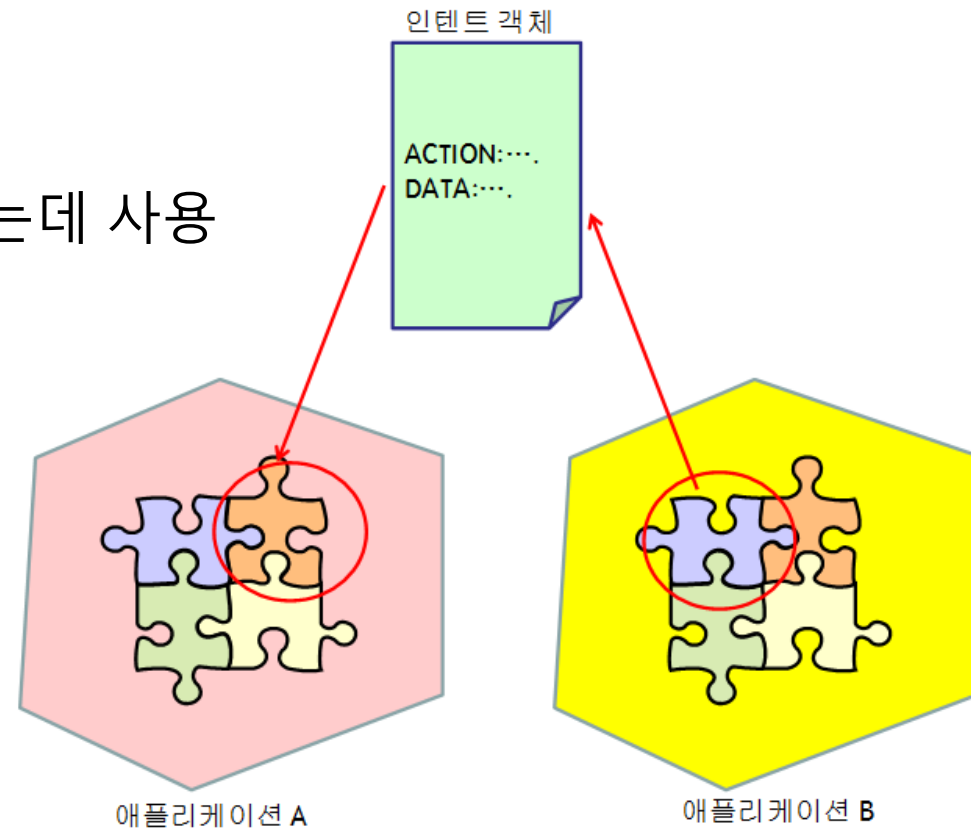


액티비티 전환

- 한 액티비티(화면)에서 다른 액티비티(화면)으로 전환하려면 어떻게 해야 하나?

인텐트 (Intent)

- 한 액티비티에서 다른 액티비티를 시작하려면 액티비티 실행에 필요한 정보를 보내주어야 함
- 인텐트
 - 여러 가지 필요한 정보를 담아서 보내는데 사용



인텐트 (Intent)

- 다른 application component가 어떤 action을 수행하도록 요청하기 위해 사용하는 메시징 객체
- 대표적인 use cases
 - Activity를 시작
 - Service를 시작
 - Broadcast를 전달
- 종류
 - 명시적 인텐트 (Explicit intent): 실행할 component 이름을 명확하게 지정
 - 암시적 인텐트 (Implicit intent): 수행해야 하는 일반적인 액션만 선언 (지도 상의 위치를 표시하고자 할 때 지도를 보여줄 수 있는 component면 어떤 것이라도 좋다)

<https://developer.android.com/guide/components/intents-filters.html>

<https://developer.android.com/reference/android/content/Intent.html>

명시적 인텐트 (Explicit Intent)

- 코드 예제

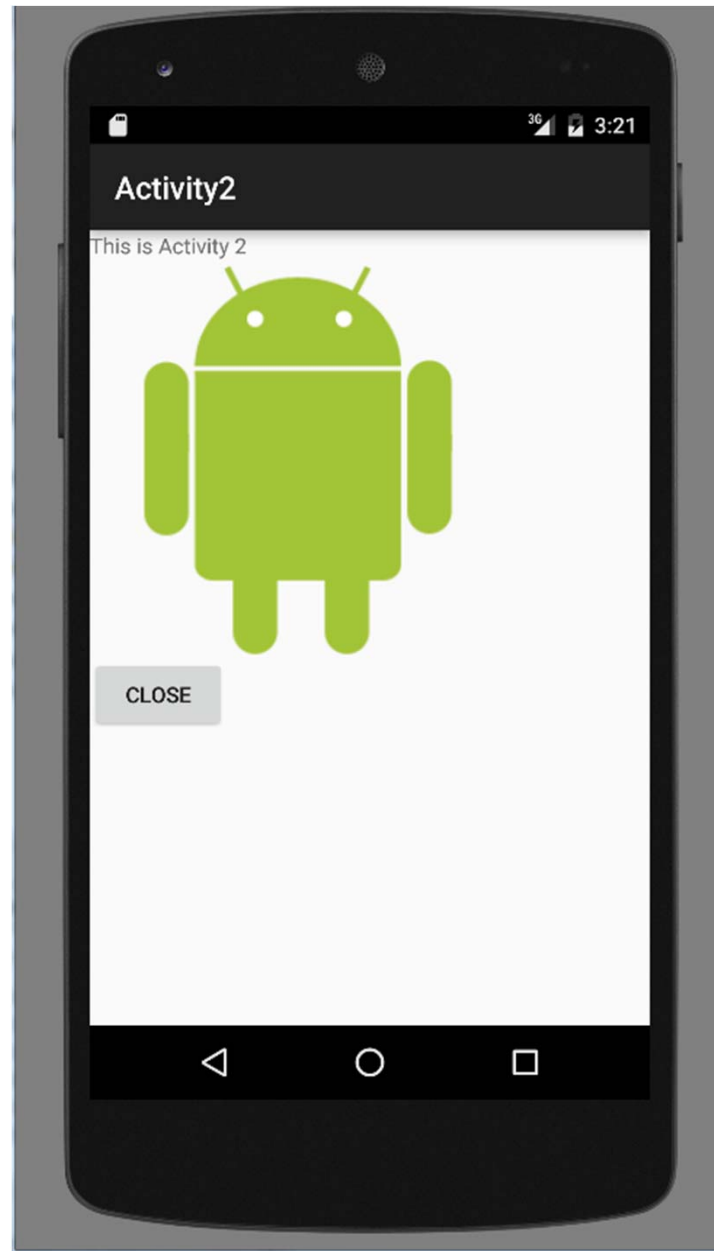
```
Intent intent = new Intent(this, NextActivity.class);  
startActivity(intent);  
// Executed in an Activity, so 'this' is the Context
```

- Intent(Context, Class)

- 특정 component(class 객체)에 대한 인텐트 생성

- startActivity(Intent)

- Intent 객체에 명시된 액티비티를 시작
- 액티비티가 종료될 때 아무런 반환값을 받지 않게 됨



- 예제 프로젝트:
Ch10TwoActivities

예제 소스 구조

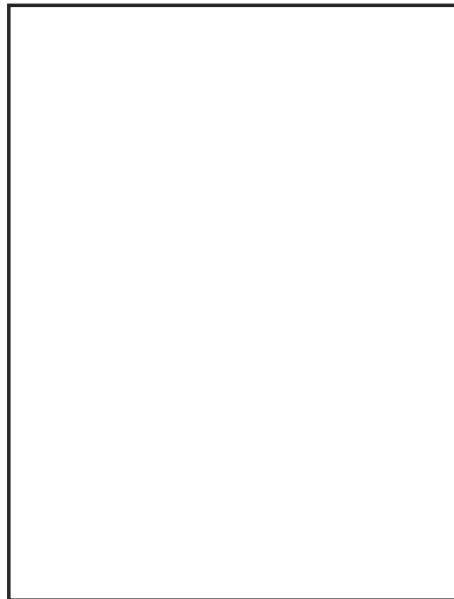
- activity_main.xml
 - TextView / Button
- MainActivity.java
 - Button 객체 onClick 메소드
 - Intent 객체 생성
 - startActivity() 호출

- activity_second.xml
 - TextView / ImageView / Button
- SecondActivity.java
 - Button 객체 onClick 메소드
 - finish() 호출

- AndroidManifest.xml
 - <activity> element 추가: SecondActivity 설정

액티비티 결과 받기

Activity1 (main activity)



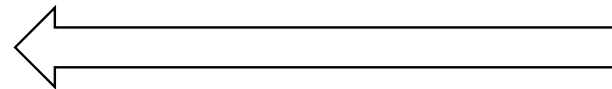
start

1. Activity1에 의해
Activity2가 시작

Activity2



2. Activity2 실행



3. Activity2의 실행 종료 시
Activity1에 결과를 전달

How?

액티비티 결과 받기를 위한 메소드

다른 activity 실행 시 호출해야 하는 메소드 [startActivity() 메소드 대신 사용]

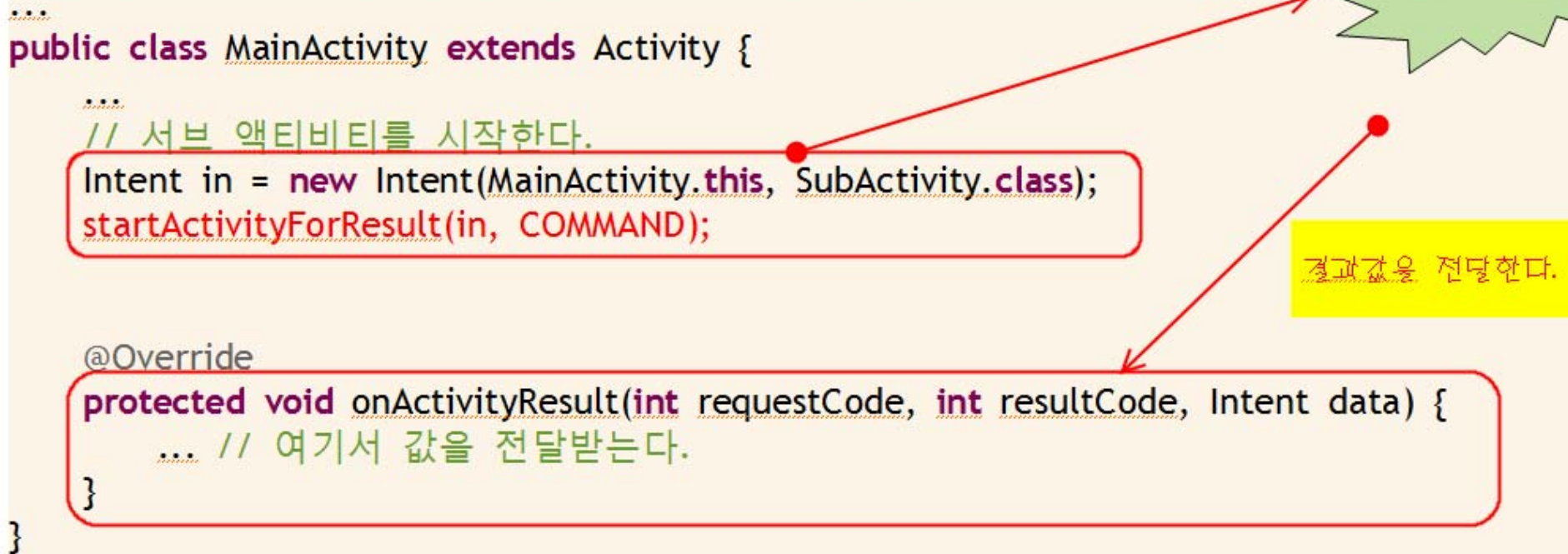
- `startActivityForResult(Intent intent, int requestCode)`
 - `intent`
 - 실행하고자 하는 액티비티 정보를 담고 있는 인텐트 객체
 - `requestCode`
 - 시작된 액티비티가 종료될 때 `onActivityResult()` 메소드에서 반환됨

실행 시킨 activity로부터 결과를 받기 위해 구현해야 하는 메소드

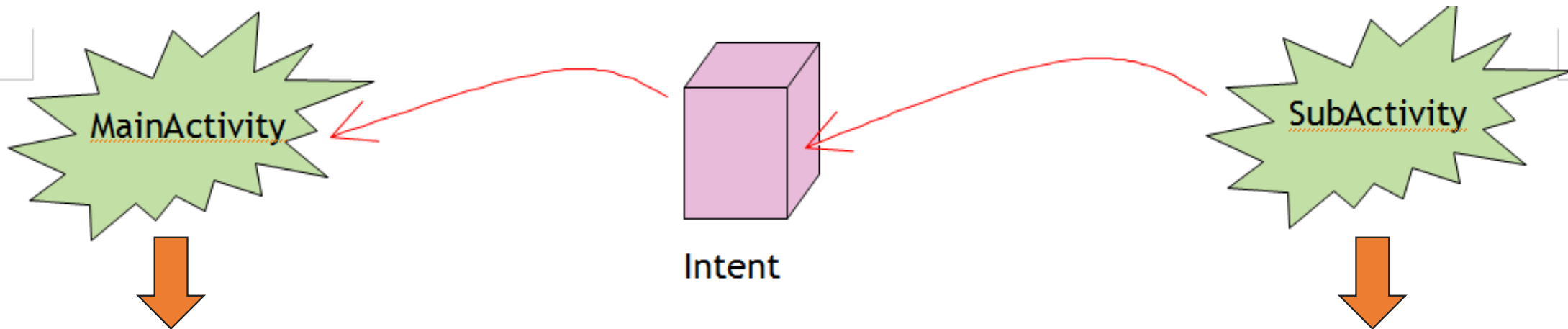
- `onActivityResult(int requestCode, int resultCode, Intent data)`
 - `requestCode`
 - `startActivityForResult()` 호출할 때 넘겨준 request code
 - `resultCode`
 - 실행된 액티비티에서 `setResult()` 메소드 호출을 통해 반환되는 result code
 - `data`
 - 부가 데이터를 담는 Intent 객체

액티비티 결과 받기 예

- SubActivity를 시작한 MainActivity의 경우



인텐트를 통해 데이터 전달하기



```
int  getIntExtra(String name, int defaultValue)
float  getFloatExtra(String name, float defaultValue)
int[]  getIntArrayExtra(String name)
String getStringExtra(String name)
```

```
putExtra(String name, int value)
putExtra(String name, float value)
putExtra(String name, int[] value)
putExtra(String name, String value)
```

데이터 전달 예

- 결과를 전달하고자 하는 SubActivity의 경우

1. 전달할 결과를 담은 Intent 객체 생성

2. Intent 객체에 데이터 저장

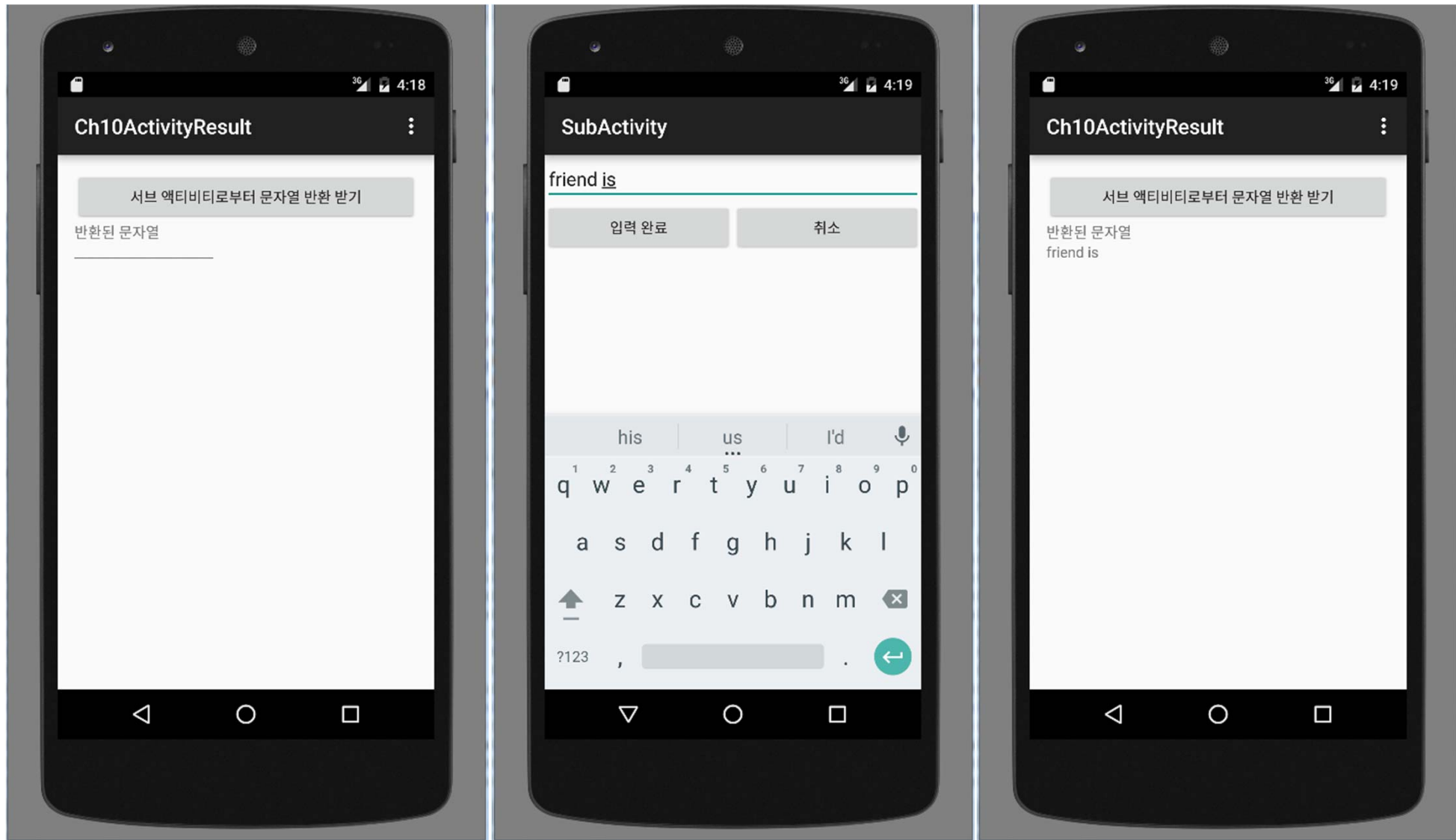
3. Intent 객체를 전달할 결과로 설정

```
Intent intent = new Intent();
```

```
intent.putExtra("INPUT_TEXT", edit.getText().toString());
```

```
setResult(RESULT_OK, intent);
```

- 예제 프로젝트: Ch10ActivityResult



예제 소스 구조

- activity_main.xml
 - Button / TextView / TextView
- MainActivity.java
 - Button onClick() 메소드
 - Intent 객체 생성
 - startActivityForResult() 호출
 - onActivityResult() 구현
 - 결과로 반환된 Intent의 문자열을 TextView의 텍스트로 설정

- activity_sub.xml
 - EditText / Button / Button
- SubActivity.java
 - Button_ok onClick() 메소드
 - Intent 객체 생성 / EditText에 입력된 문자열을 intent에 넣음 / setResult() 호출 / finish() 호출
 - Button_cancel onClick() 메소드
 - setResult() 호출 / finish() 호출

- AndroidManifest.xml
 - <activity> element 추가: SubActivity 설정