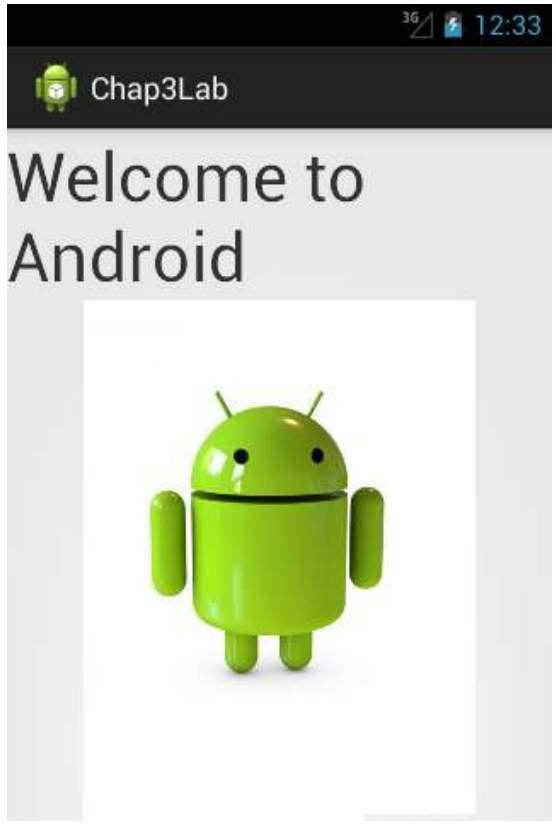
CHAP 2. 애플리케이션 기본구조

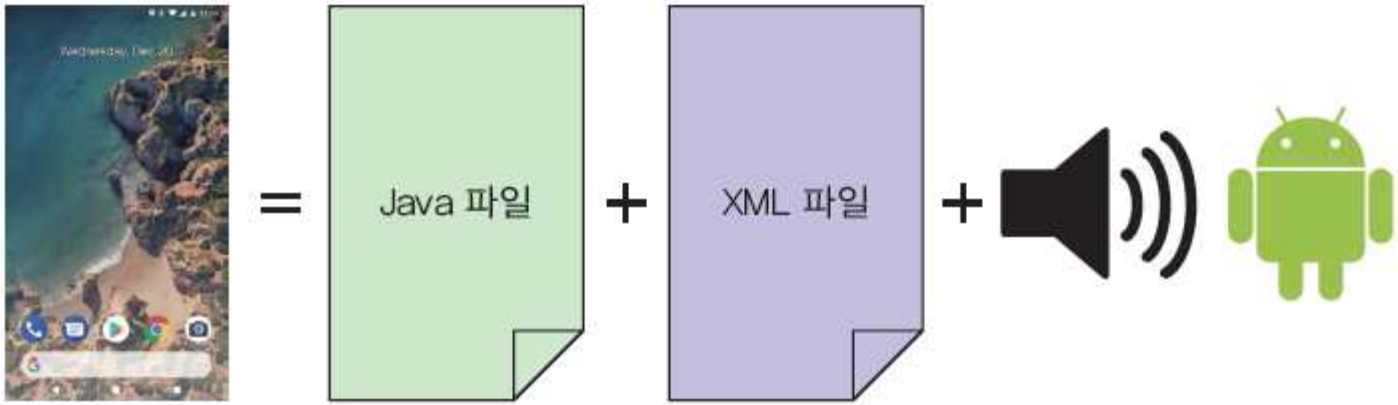
이번 주의 목표

- 비주얼 도구를 사용하여서 다음과 같은 앱을 작성하여 본다.



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

애플리케이션의 구성



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

일반적인 애플리케이션 작성 절차

① 사용자 인터페이스 작성**(XML)**

② 자바 코드 작성**(JAVA)**

③ 매니페스트 파일 작성**(XML)**

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

1. 사용자 인터페이스 작성

- 첫 번째 단계는 XML을 이용하여서 사용자 인터페이스 화면을 디자인하는 단계

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

2. 자바 코드 작성

- 두 번째 단계는 자바를 이용하여서 코드를 작성하는 단계

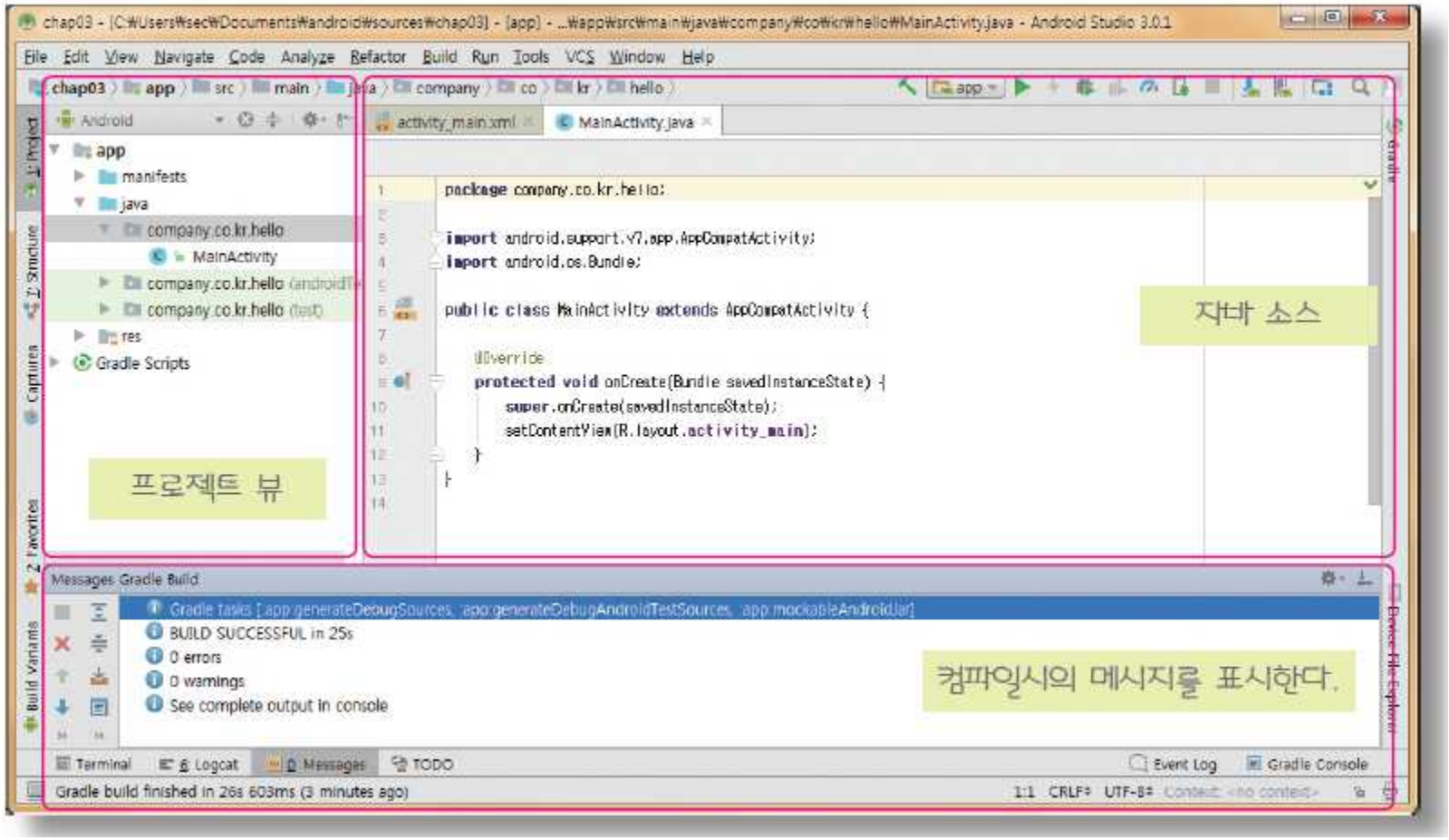
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

3. 매니페스트 파일 작성

- 애플리케이션을 구성하고 있는 컴포넌트를 기술하고 실행 시에 필요한 권한을 지정한다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

애플리케이션의 구성



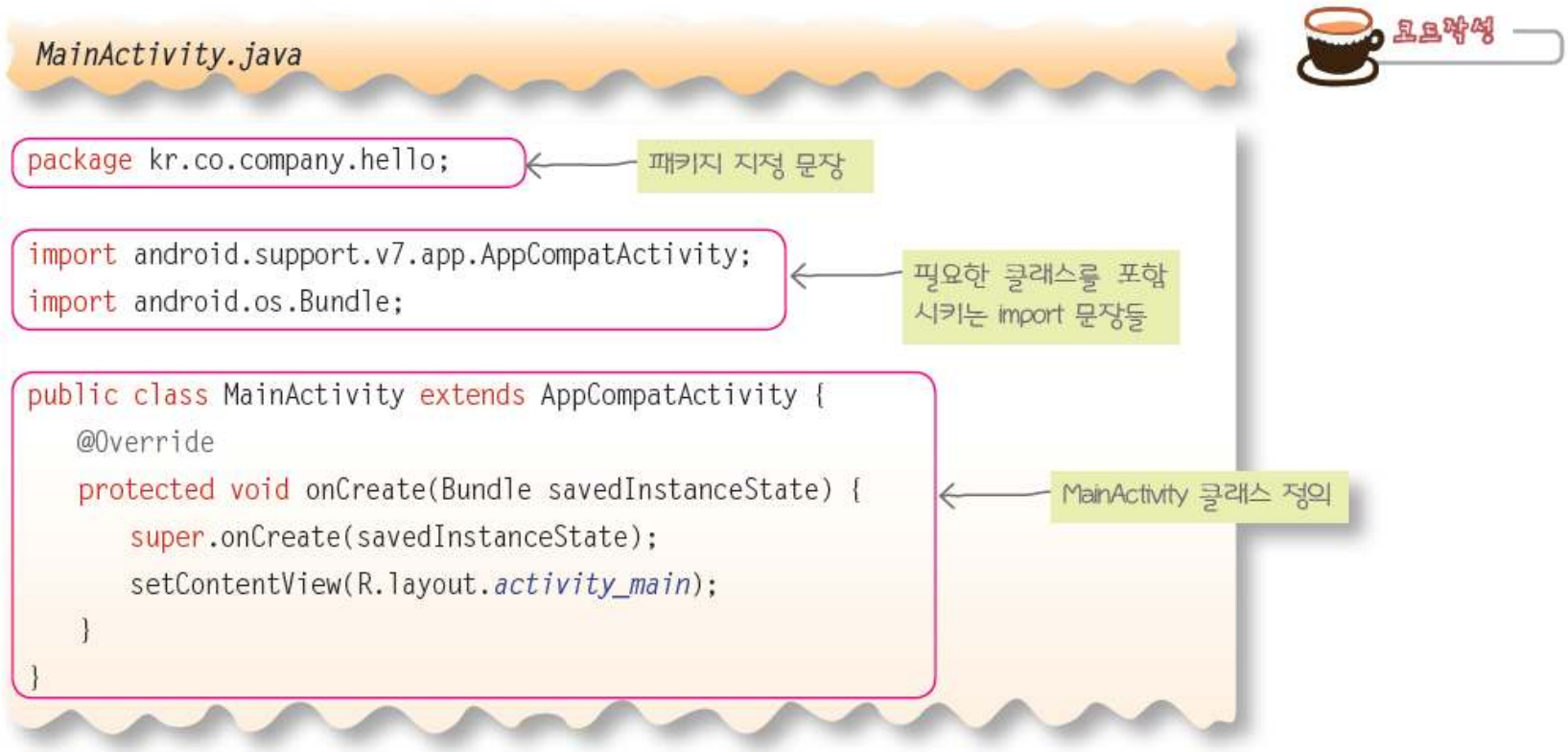
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

패키지 폴더의 설명

폴더 또는 파일	설명
java	자바 소스 파일들이 들어있는 폴더이다. 폴더 안의 kr.co.company.hello는 패키지의 이름이다.
Gradle Scripts	그레이들(Gradle)은 빌드 시에 필요한 스크립트이다.
res	각종 리소스(자원)들이 저장되는 폴더이다. drawable에는 해상도 별로 아이콘 파일들이 저장된다. layout에는 화면의 구성을 정의한다. values에는 문자열과 같은 리소스가 저장된다. menu에는 메뉴 리소스들이 저장되어 있다.
manifest	XML 파일로 앱의 전반적인 정보 즉 앱의 이름이나 컴포넌트 구성과 같은 정보를 가지고 있다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

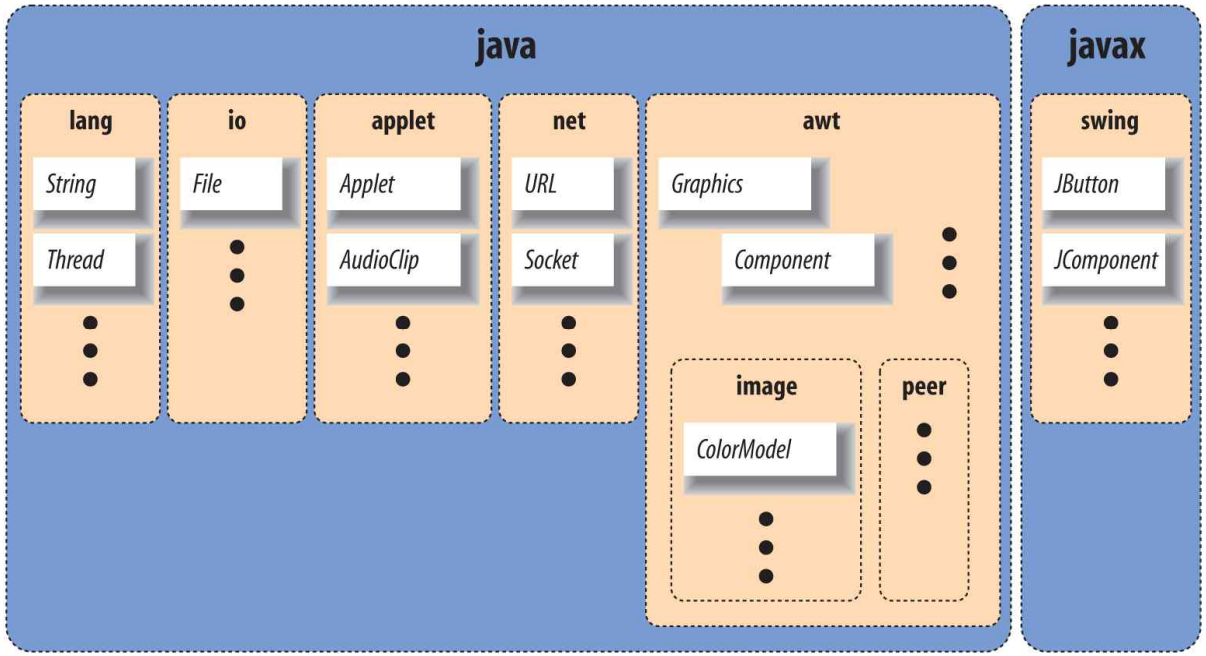
자동 생성된 소스 관찰



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

package kr.co.company.hello;

- 패키지(**package**)는 클래스들을 보관하는 컨테이너
- 일반적으로 인터넷의 도메인 이름을 역순으로 사용



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

import android.support.v7.app.App...

- import 문장은 외부에서 패키지나 클래스를 포함
- 앞에 android가 붙은 패키지는 안드로이드가 제공하는 패키지를 의미한다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

public class MainActivity extends AppCompatActivity { ... }

- 클래스의 정의
- **Activity**로부터 상속받았으므로 액티비티가 된다.
- **액티비티는** 안드로이드에서 애플리케이션을 구성하는 **4** 가지의 컴포넌트 중의 하나이다.

액티비티는 화면을 통하여 사용자와 상호작용하는 활동을 의미한다.

© Dept. of Comput... Engineering, ...

@Override

- 어노테이션의 하나
- 어노테이션은 컴파일러에게 추가적인 정보를 주는 것
- @Override은 메소드가 부모 클래스의 메소드를 재정의(오버라이드)하였다는 것을 나타낸다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

public void onCreate() { … }

- onCreate() 메소드는 액티비티가 생성되는 순간에 한번만 호출
- 모든 초기화와 사용자 인터페이스 설정이 여기에 들어간다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

super.onCreate(savedInstanceState);

- 위의 문장은 부모 클래스인 AppCompatActivity 클래스의 onCreate()를 호출하는 문장

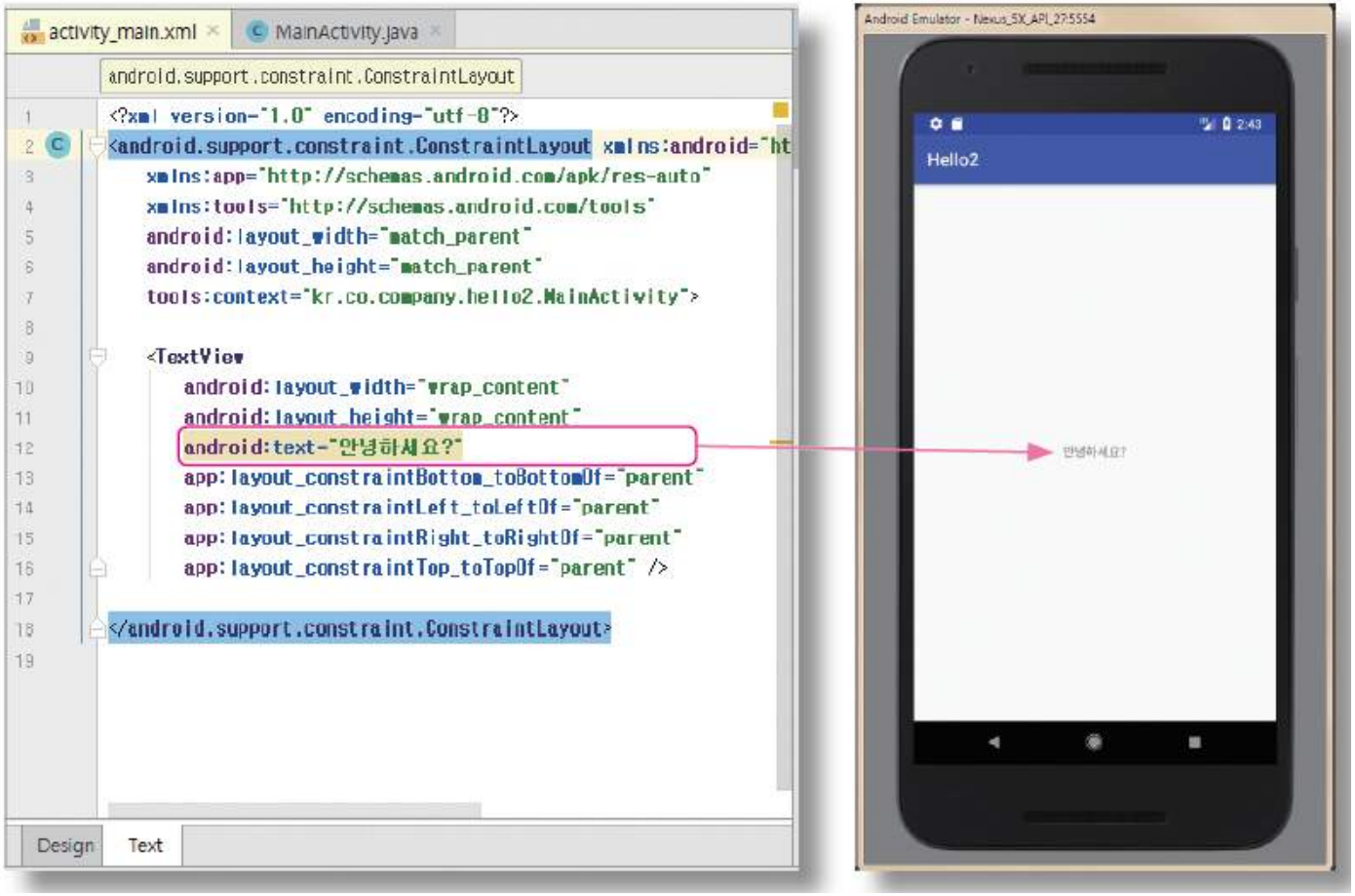
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

setContentView(R.layout.activity_main);

- setContentView()라는 함수는 액티비티의 화면을 설정하는 함수
- R.layout.activity_main은 activity_main.xml 파일을 나타낸다.

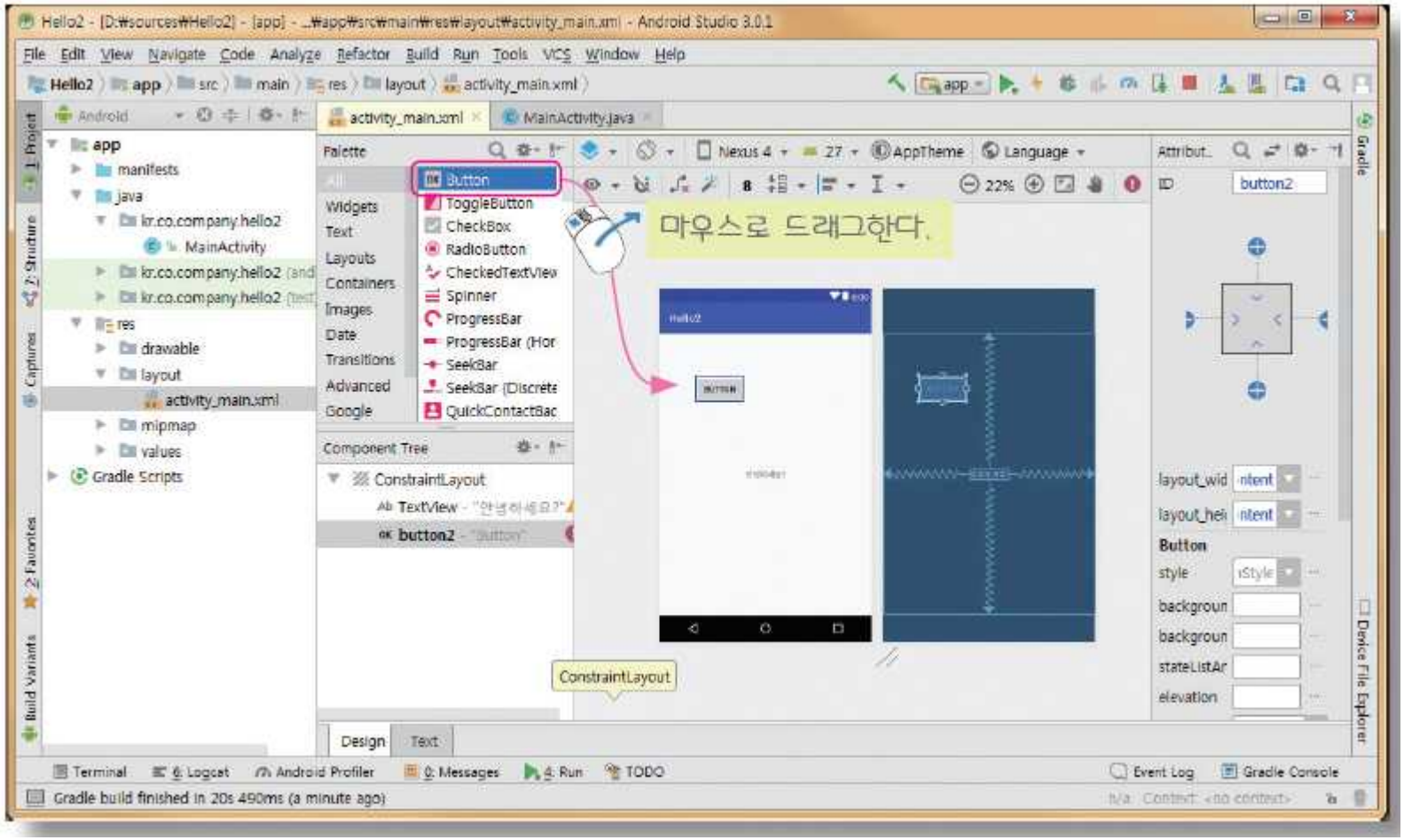
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

한글 사용하기



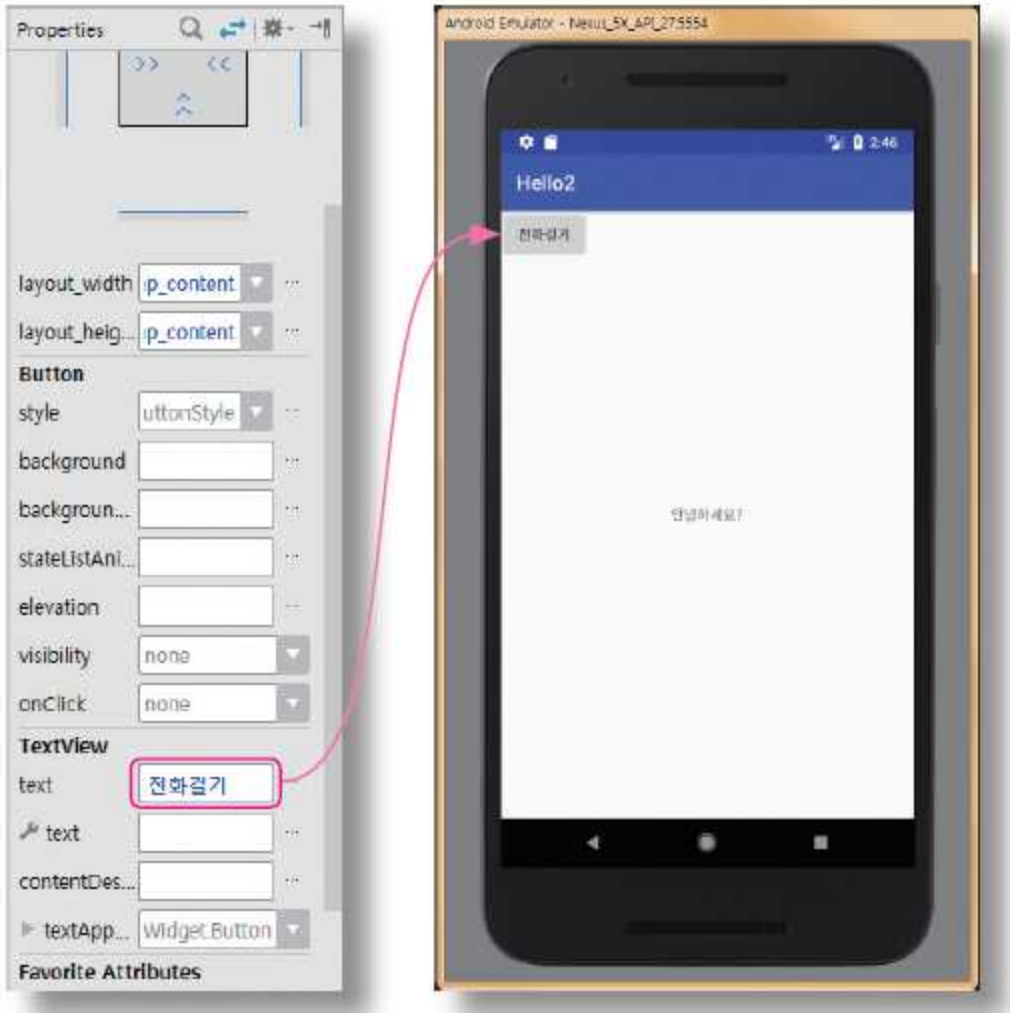
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

비주얼 도구 사용하기



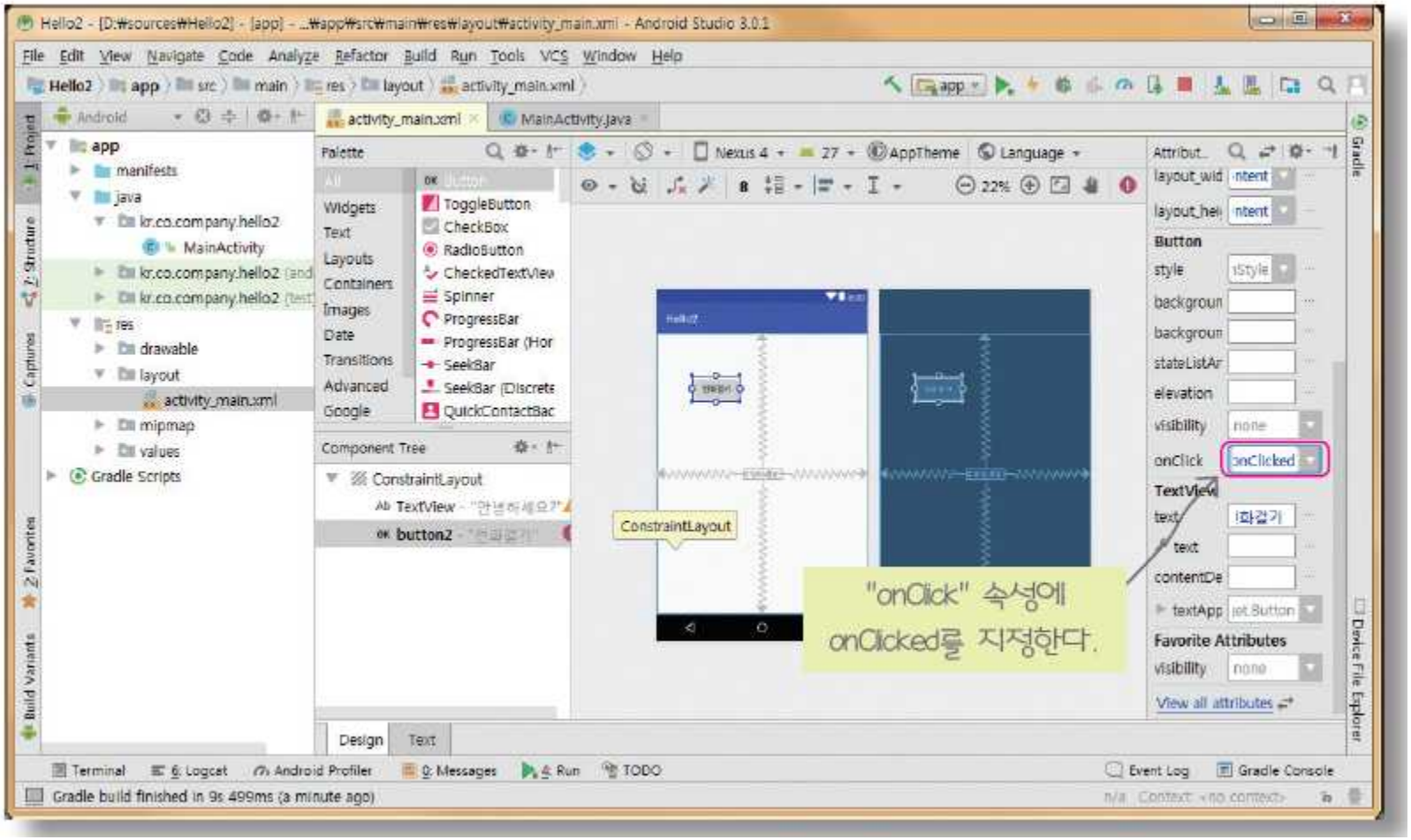
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

버튼의 텍스트 변경



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

onclick 속성 변경



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

소스 추가

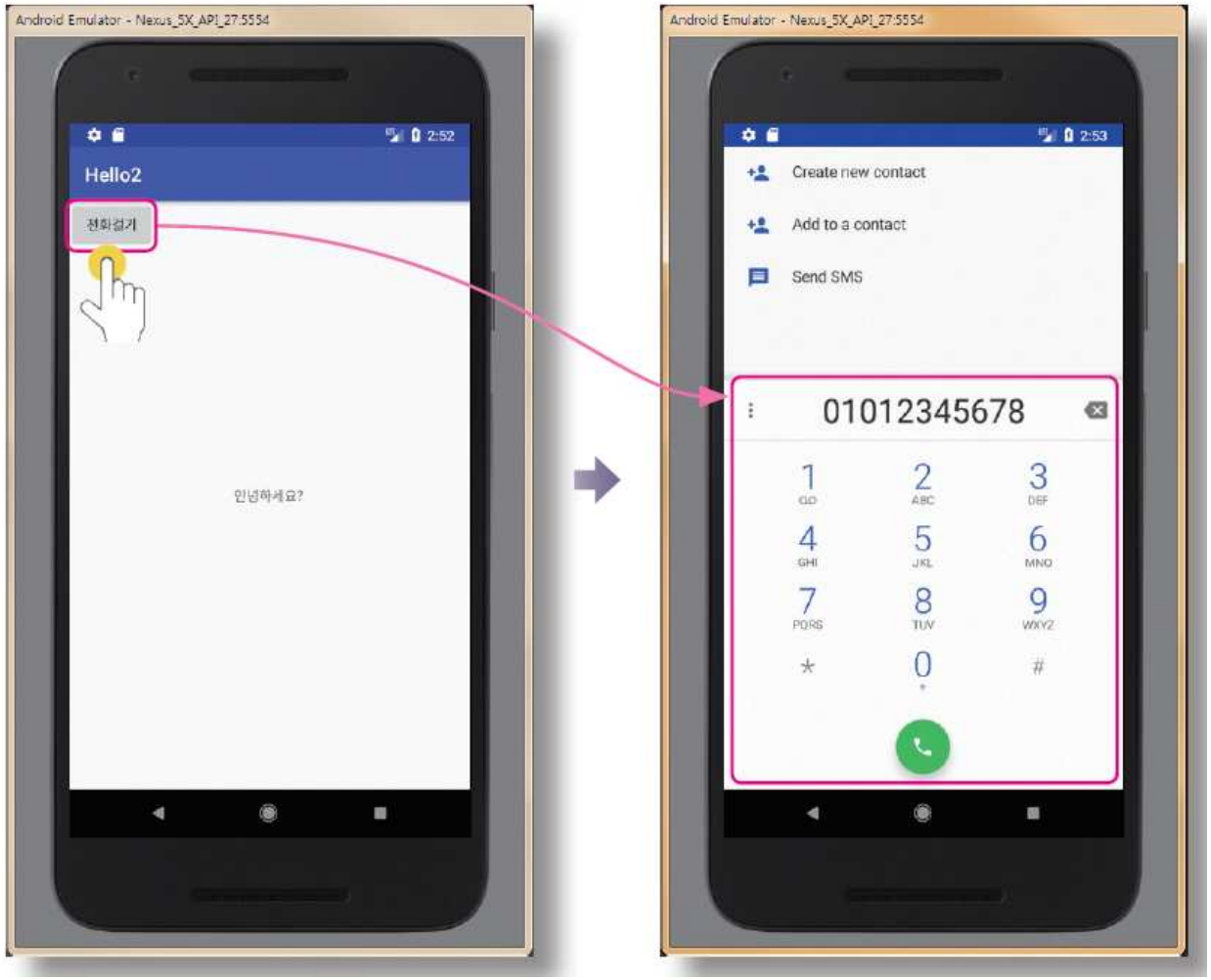
activity_main.xml | MainActivity.java

```
1  package kr.co.company.hello;
2
3  import androidx.appcompat.app.AppCompatActivity;
4
5  import android.os.Bundle;
6
7  public class MainActivity extends AppCompatActivity {
8
9      @Override
10     protected void onCreate(Bundle savedInstanceState) {
11         super.onCreate(savedInstanceState);
12         setContentView(R.layout.activity_main);
13     }
14
15     public void onClicked(View v) {
16         Inter                    ACTION_VIEW, Uri.parse("tel:010-1234-5678"));
17         start
18     }
19 }
20
```

Cannot resolve symbol 'View'

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

변경된 앱 실행



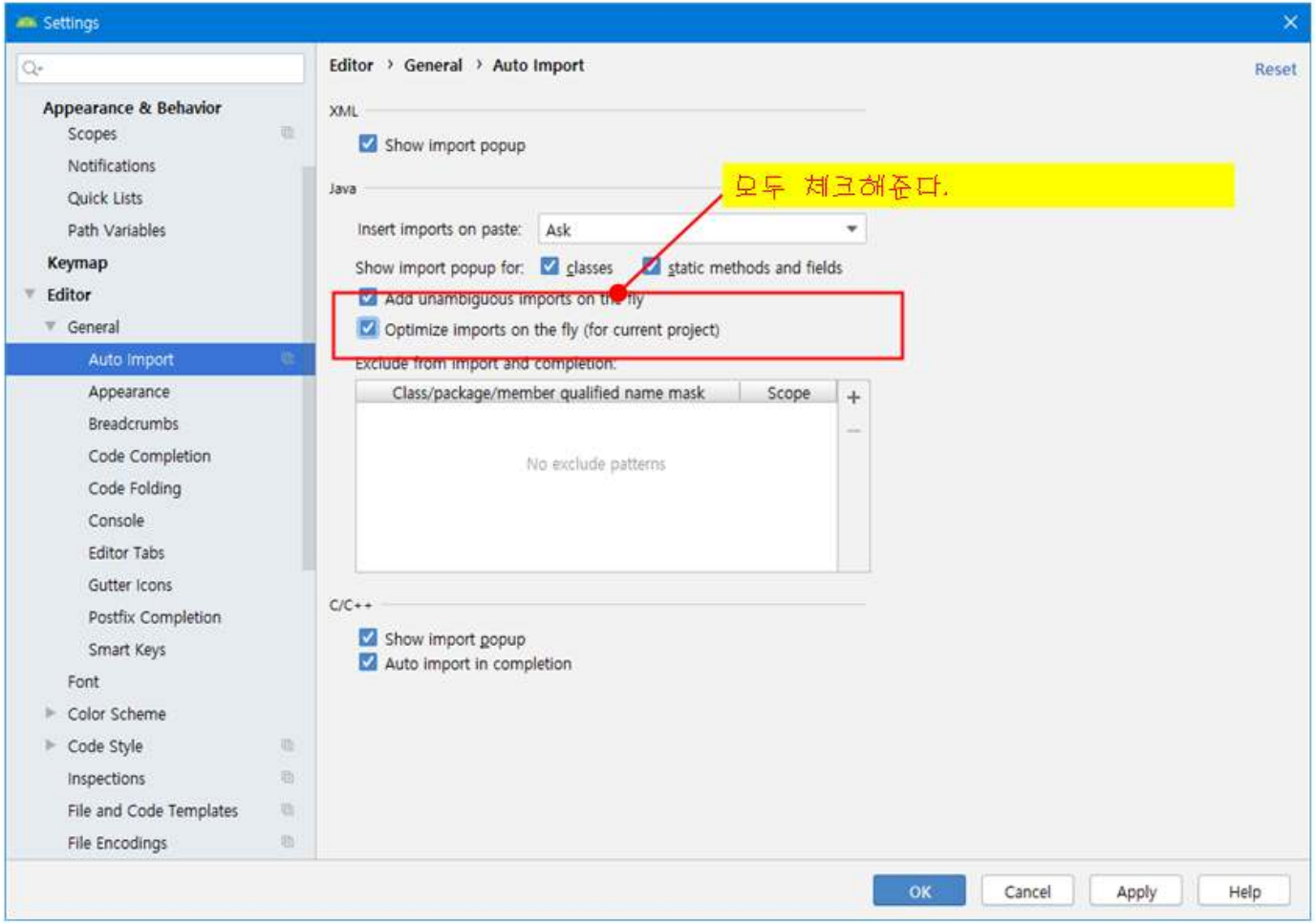
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

TIP

- 필요한 패키지를 가장 쉽게 프로젝트에 추가하려면 [File]->[Settings]->[Editor]->[General]->[Auto Import]로 가서 “Add unambigious imports on the fly” 옵션과 "Optimize imports on the fly“ 옵션을 체크하면 된다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

TIP



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

안드로이드 애플리케이션의 실행이 시작되는 곳

- 안드로이드에는 main()이 없음.
- 액티비티별로 실행된다.
- 액티비티 중에서는 onCreate() 메소드가 가장 먼저 실행된다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

XML을 이용하여서 사용자 인터페이스 기술

- 사용자 인터페이스 작성 방법
  - 코드를 사용하는 방법(기존의 자바)
  - XML을 사용하는 방법(안드로이드 선호 방법)

- 안드로이드에서는 UI 화면의 구성을 XML을 이용하여서 선언적으로 나타내는 방법을 선호
  - 애플리케이션의 외관과 애플리케이션의 로직을 서로 분리
  - 빠르게 UI를 구축

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

XML을 이용한 사용자 인터페이스 작성

- 코드로 작성하였던 UI를 XML로 표현하면

```
TextView tv = new TextView(this);
tv.setText("Hello, world!");
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/textview"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Hello, world!" />
```

XML을 이용한 사용자 인터페이스 작성

- UI 컴포넌트 들은 XML의 하나의 요소로 표현된다.
- TextView 컴포넌트는 <TextView ... /> 요소로 표현된다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

XML 파일의 분석

```
<?xml version="1.0" encoding="utf-8"?>
<TextView
xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/textview"
    android:layout_width= "match_parent"
    android:layout_height="wrap_content"
    android:text="Hello, world!" />
```

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

<TextView>의 속성 설명

속성	의미
xmlns:android	XML 이름공간의 선언으로 안드로이드 도구에 안드로이드 이름공간에 정의된 속성들을 참조하려고 한다는 것을 암시한다. XML 파일에서 항상 최외곽 태그는 이 속성을 정의하여야 한다.
android:id	TextView 요소에 유일한 아이디를 할당한다. 이 아이디를 이용하여서 소스 코드에서 이 텍스트 뷰를 참조할 수 있다.
android:layout_width	화면에서 얼마나 폭을 차지할 것인지를 정의한다. "match_parent"는 전체 화면의 폭을 다 차지하는 것을 의미한다.
android:layout_height	화면에서 길이를 얼마나 차지할 것인지를 정의한다. "wrap_content"는 콘텐츠를 표시할 정도만 차지하는 것을 의미한다.
android:text	화면에 표시하는 텍스트를 설정한다. 이 속성은 예제와 같이 하드코딩될 수도 있고 아니면 문자열 리소스의 개념을 사용할 수도 있다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

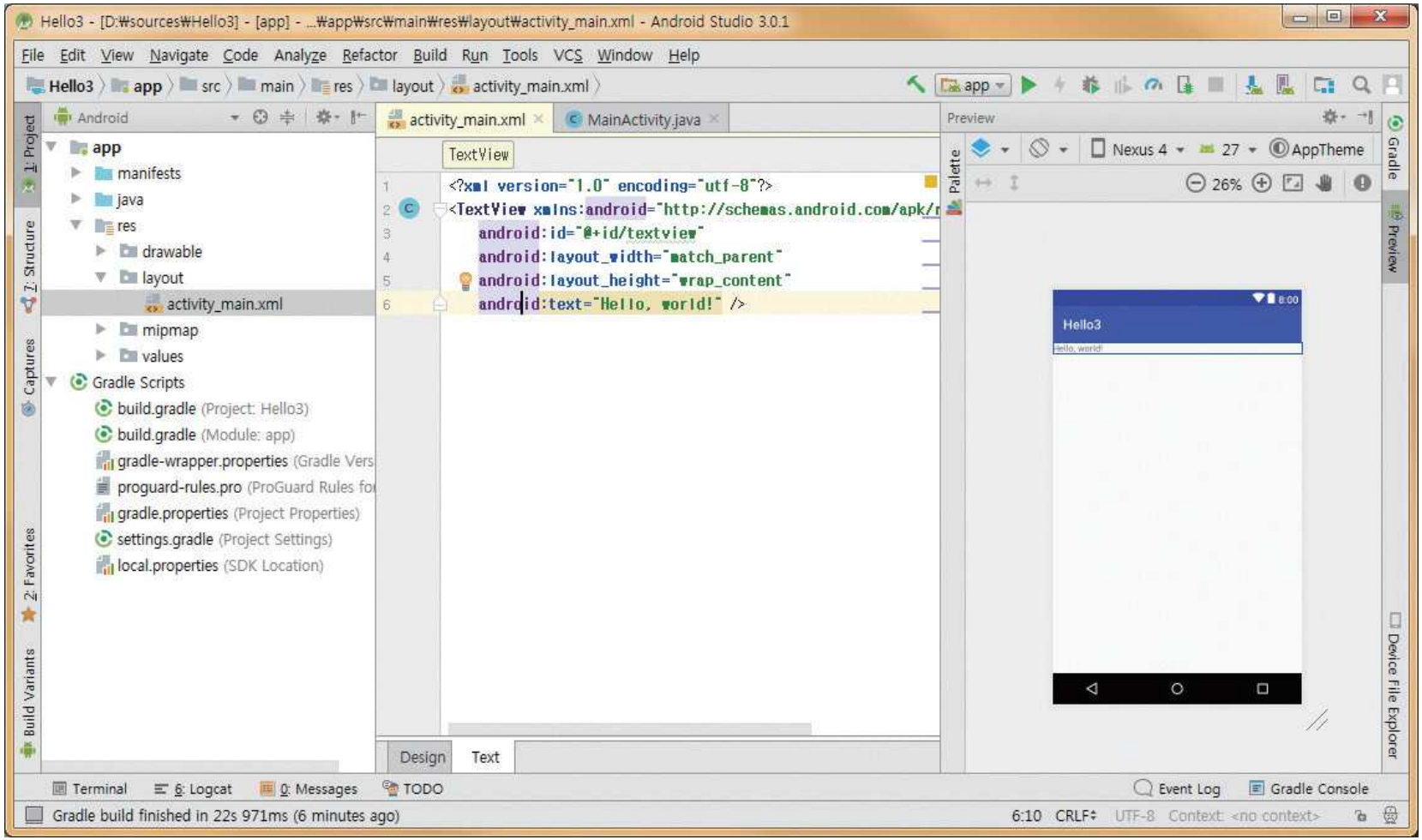
XML

- 시작 태그로 시작되어 종료 태그로 끝나는 논리적인 구성 요소를 **요소(element)**
- <Greeting>Hello, world.</Greeting>가 요소

- **속성(attribute)**는 요소의 속성으로서 "이름/값"의 쌍으로 구성
- 에서는 img 요소는 src와 alt라는 2개의 속성을 가진다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

XML 파일의 위치



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

XML파일과 코드와의 연결

MainActivity.java

```
package kr.co.company.hello3;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

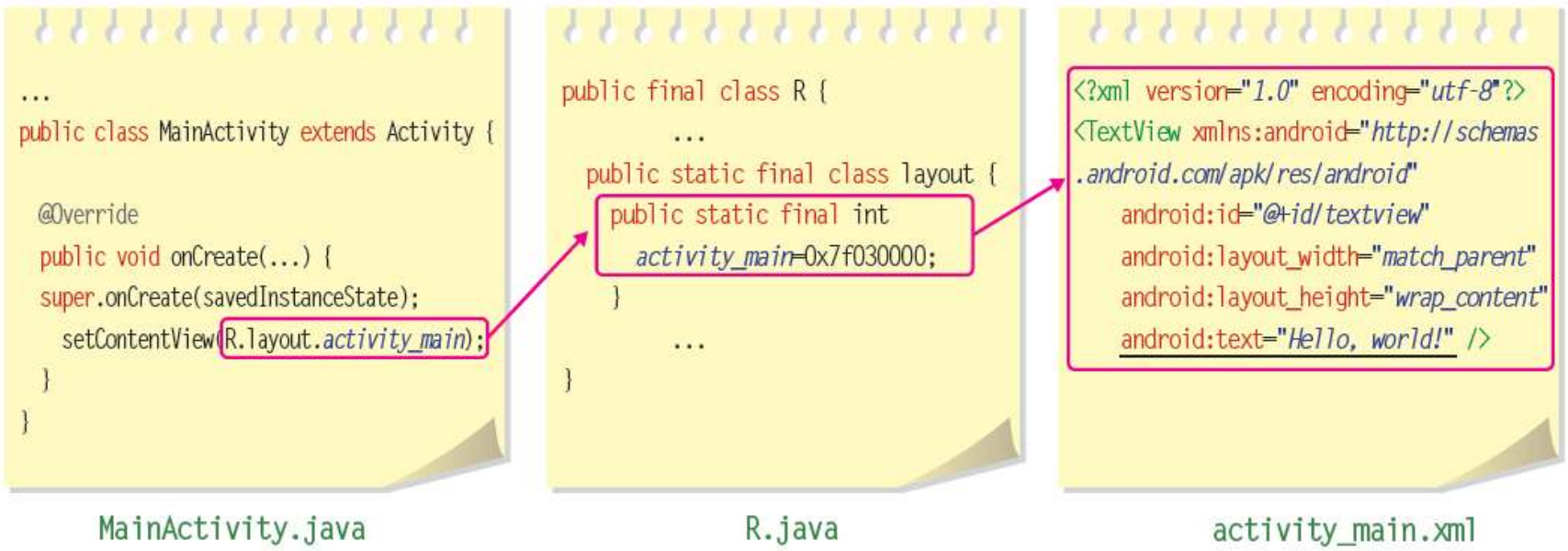
액티비티의 화면을 activity_main.xml로 설정

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

실행 결과

© Dept. of Computer Engineering, SeoKyeong Univ. 2021

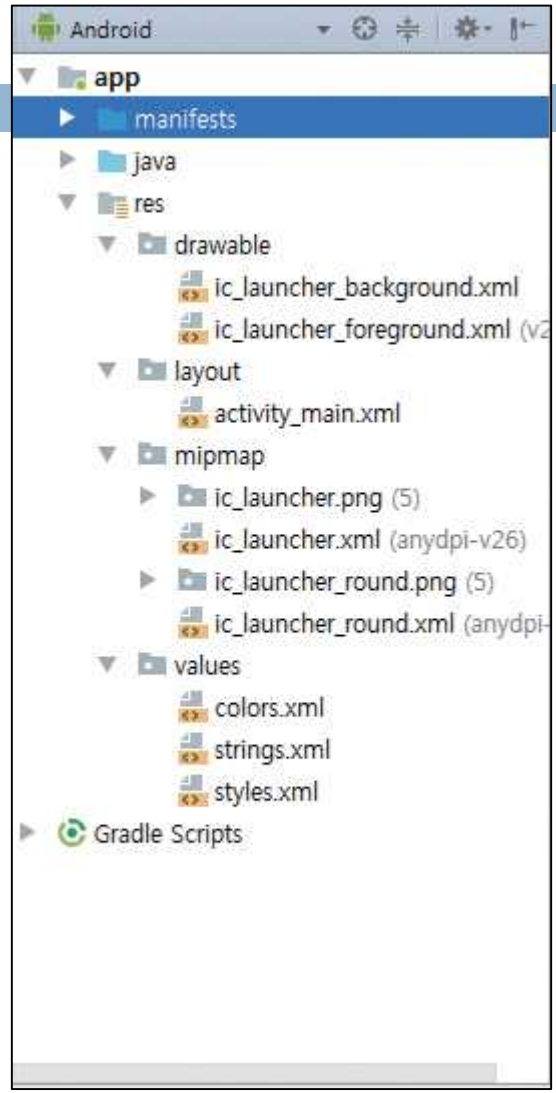
코드에서 리소스를 참조하는 방법



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

리소스

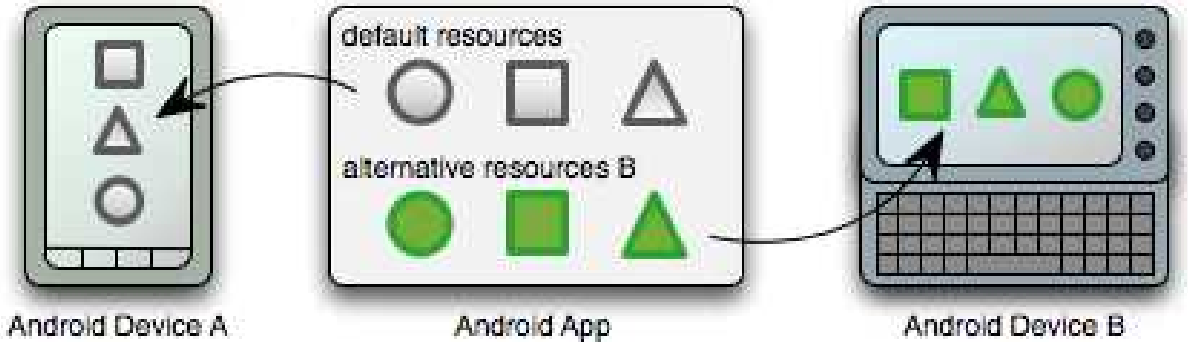
- 안드로이드에서 레이아웃, 이미지, 문자열 들을 리소스로 취급



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

코드와 리소스를 분리하는 이유

- 안드로이드가 탑재된 장치들이 다양해지면서 언어나 화면 크기에 따라서, 리소스를 다르게 하는 것이 필요



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

문자열 리소스

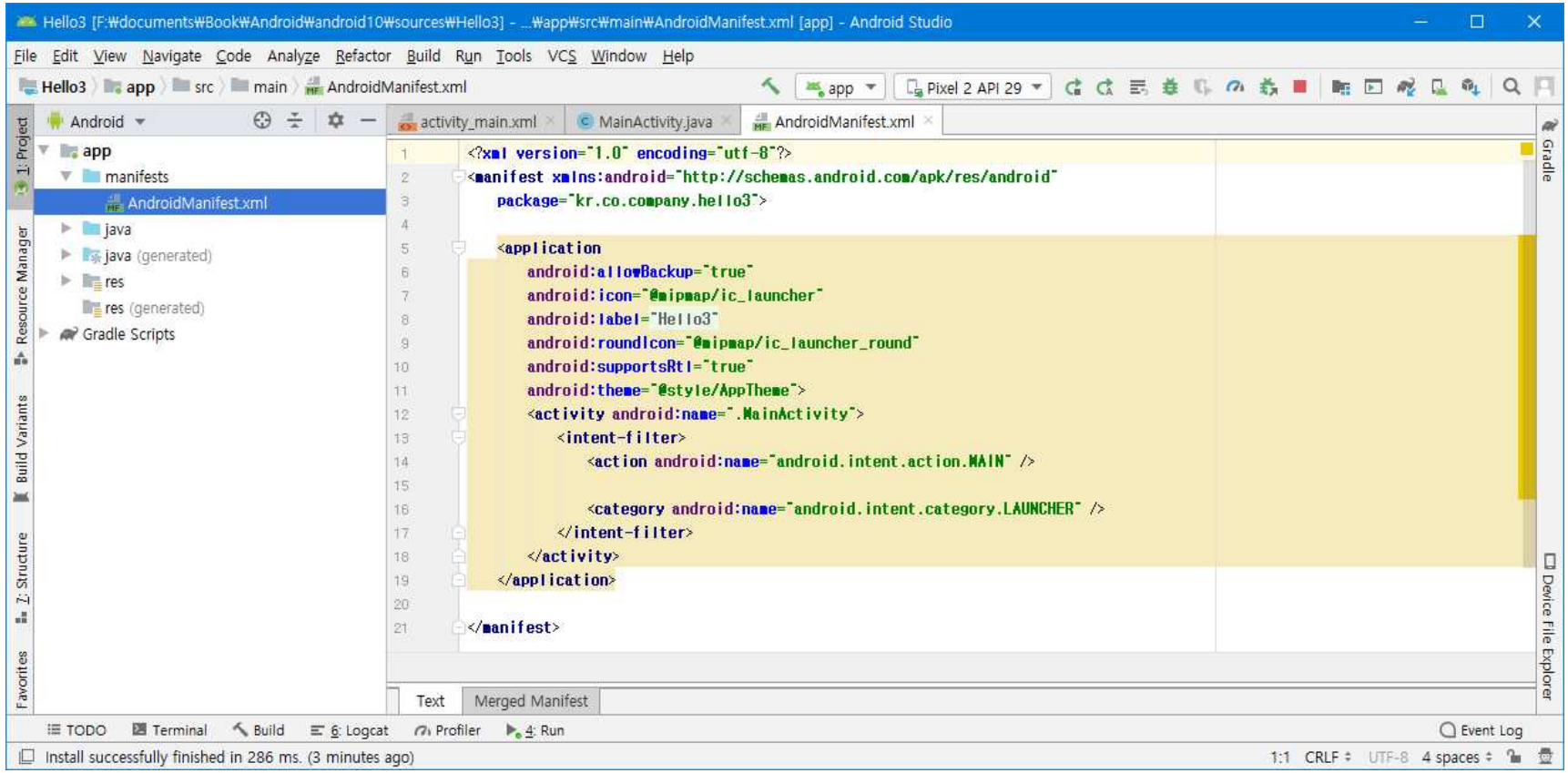
- 문자열도 XML로 기술하는 것이 바람직하다.
- 영어 버전, 한국어 버전, ...

```xml
<?xml version="1.0" encoding="utf-8"?>
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/textview"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="@string/hello_world" />
```

```xml
<resources>
    <string name="app_name">첫번째 프로그램</string>
    <string name="hello_world">안녕하세요 !</string>
    <string name="menu_settings">Settings</string>
    <string name="title_activity_main">첫번째 액티비티</string>
</resources>
```

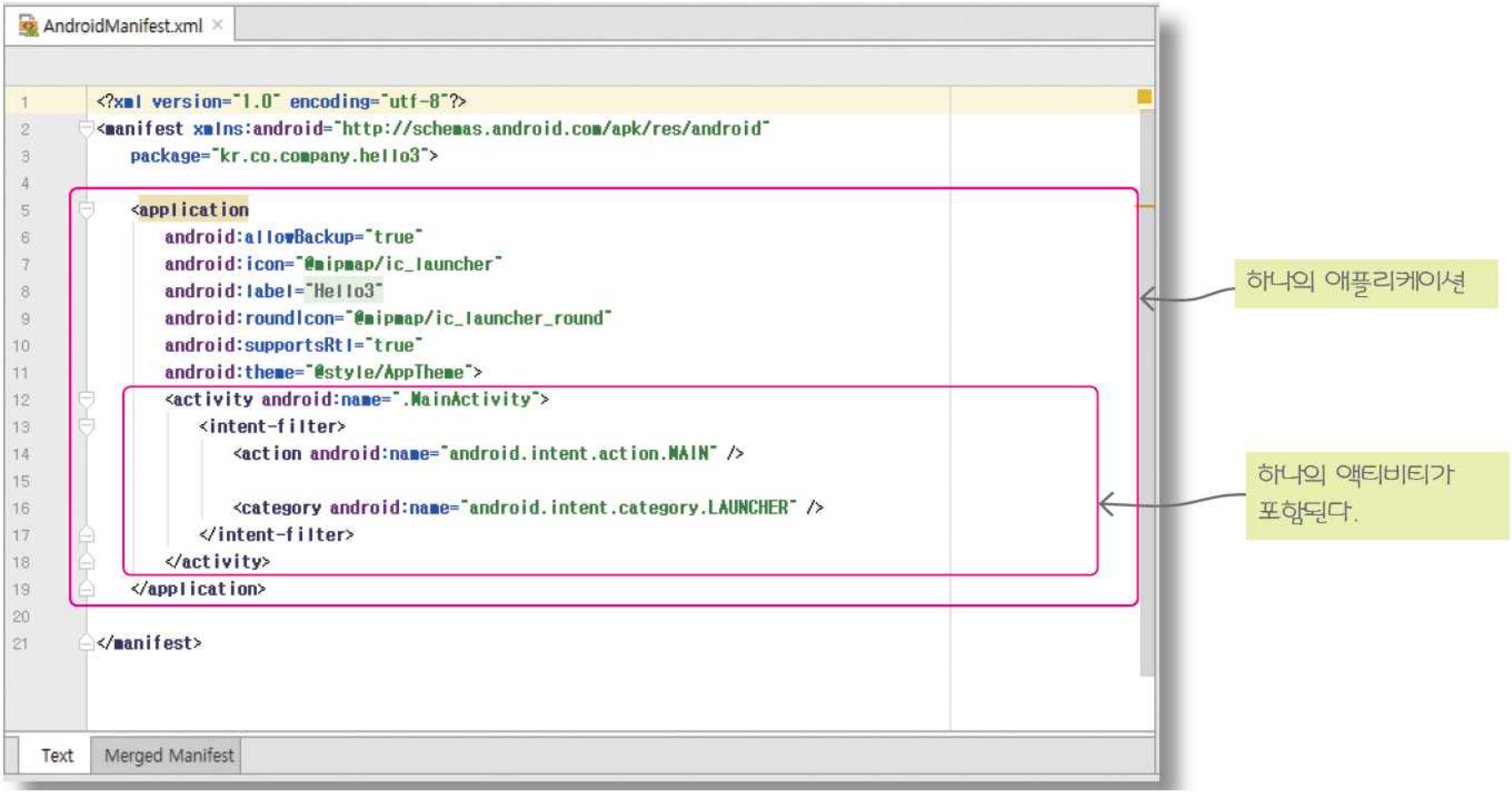
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

매니페스트 파일



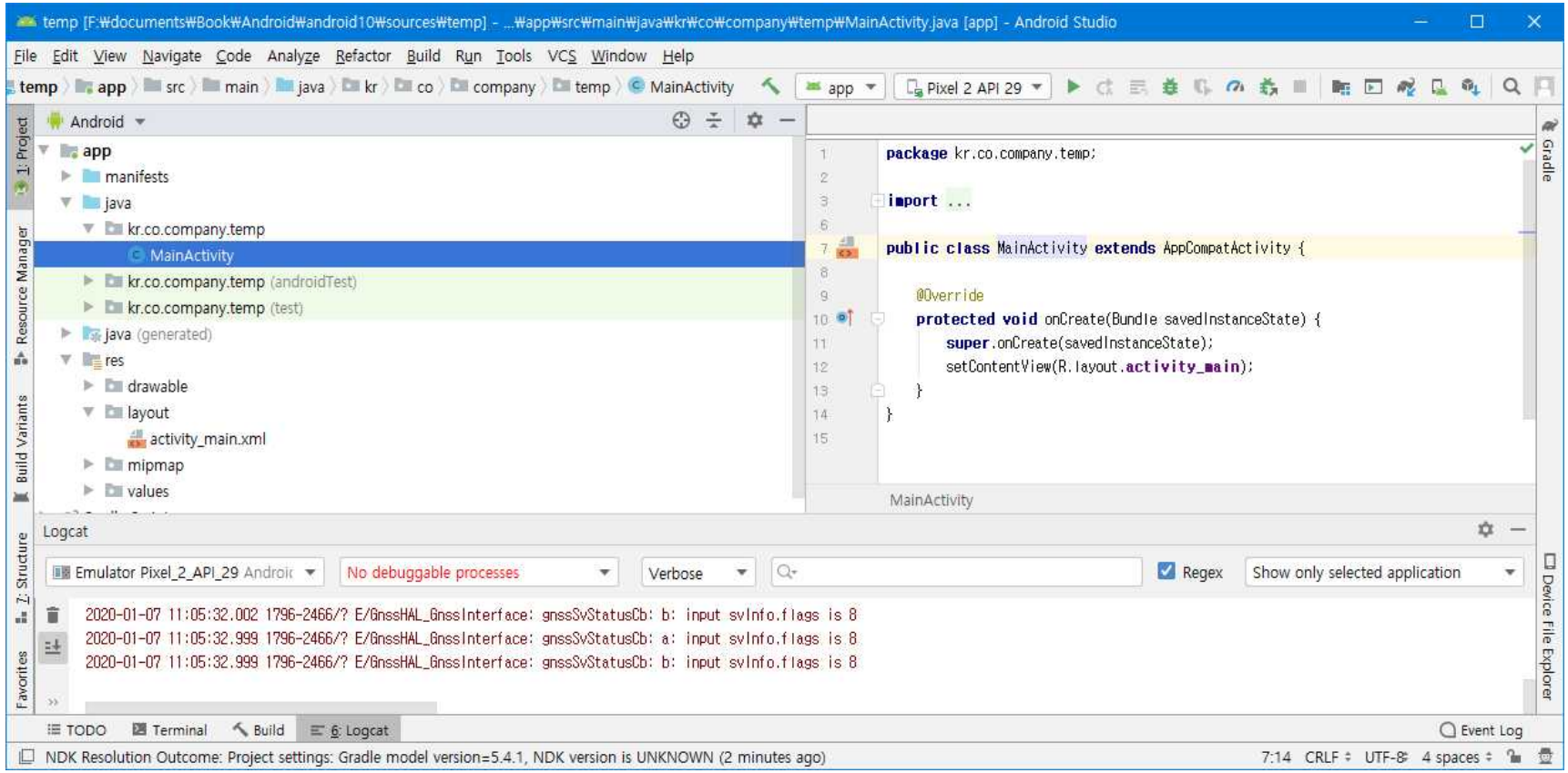
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

매니페스트 파일의 분석



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

에뮬레이터 로그캣



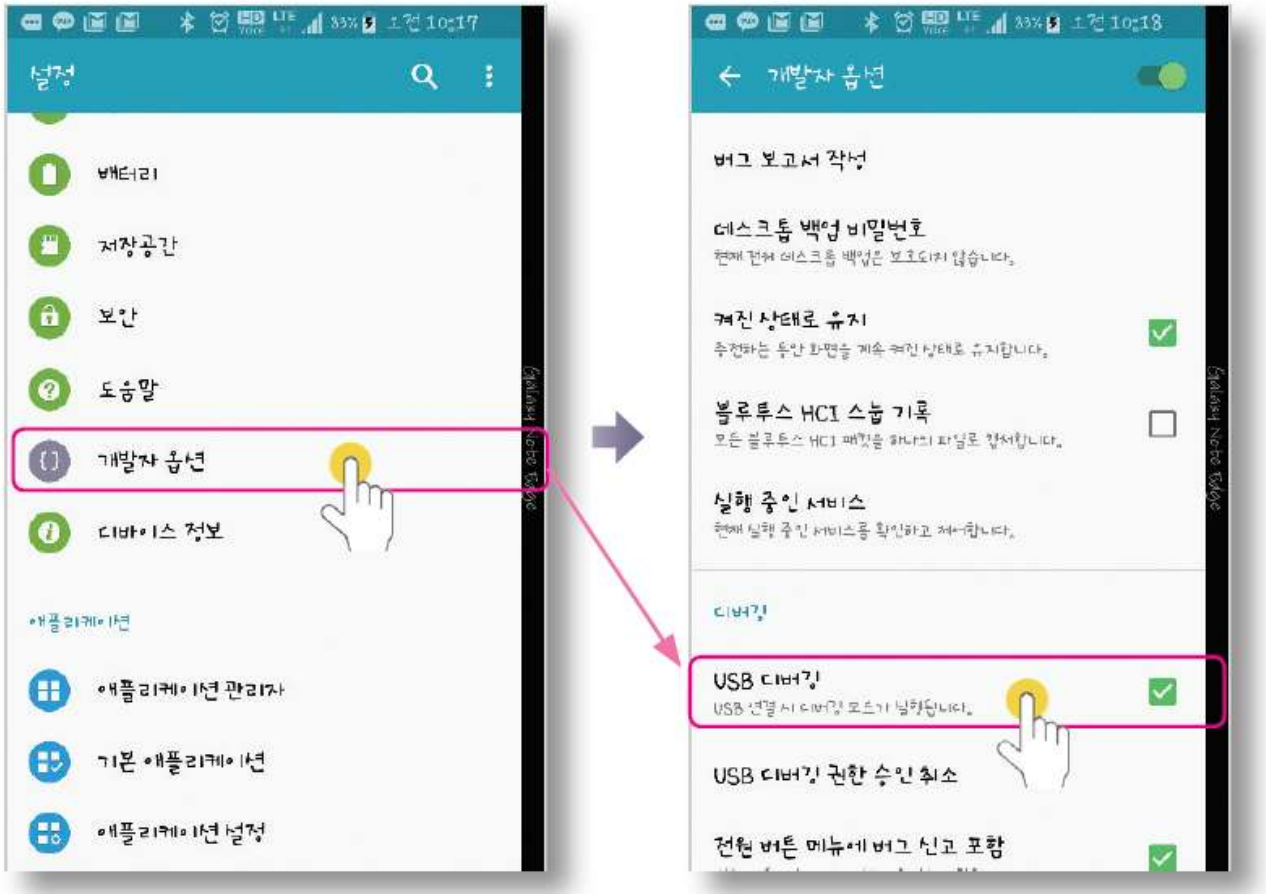
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

실제 장치와의 연결

1. Usb 드라이버 설치
2. 장치에서 "USB 디버깅"을 항목을 켠다.
3. 안드로이드 스튜디오에서 실행하기를 원하는 장치 선택

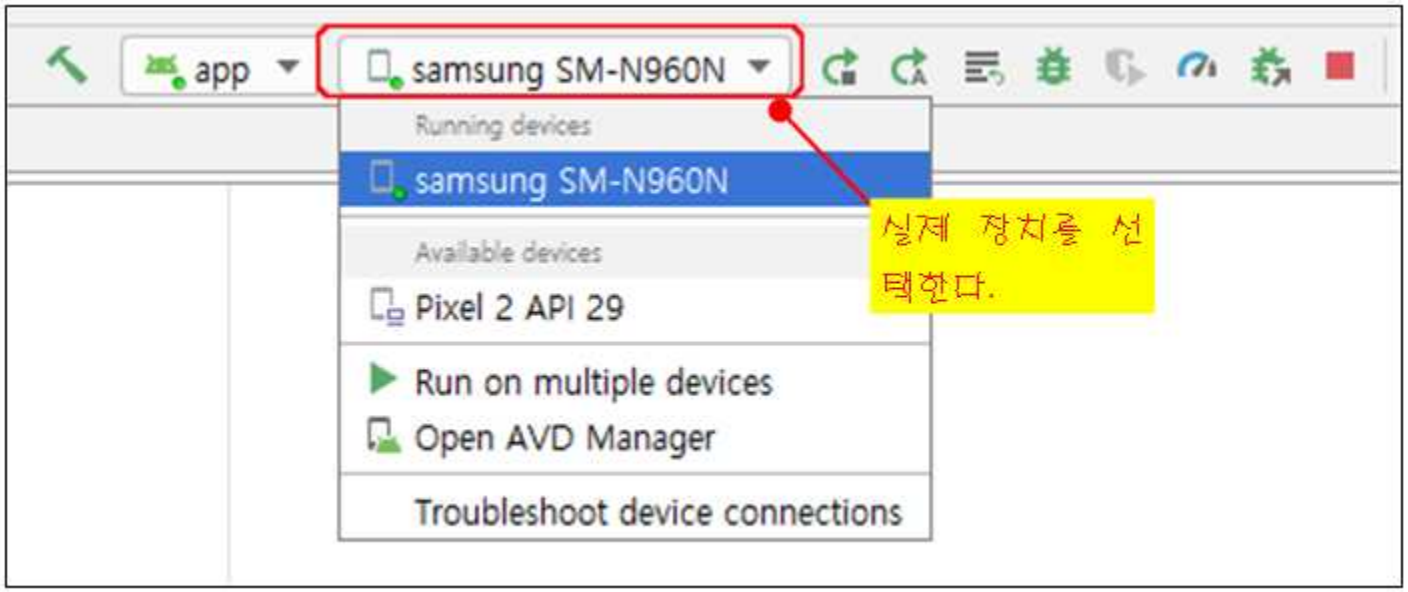
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

장치에서 USB 디버깅 켜기



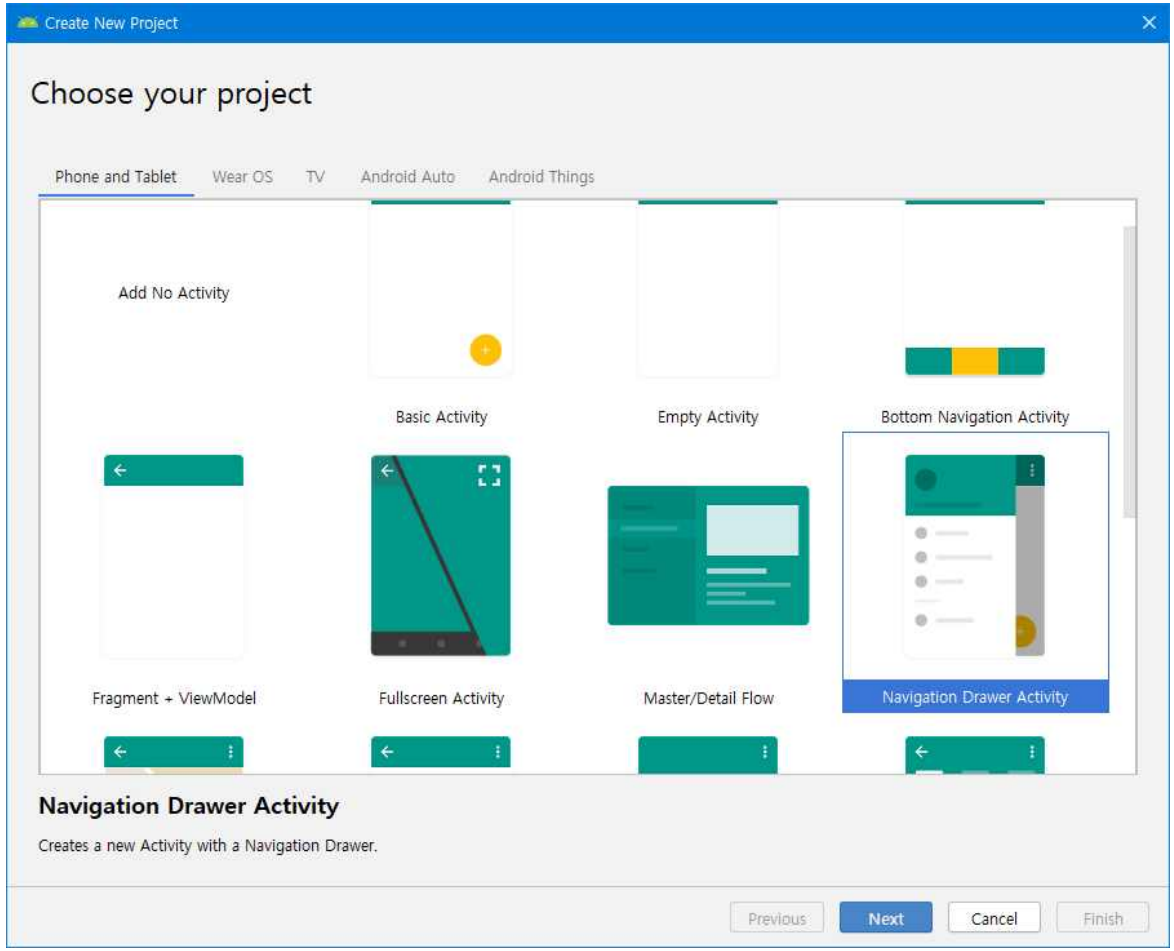
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

실행할 장치 선택 화면



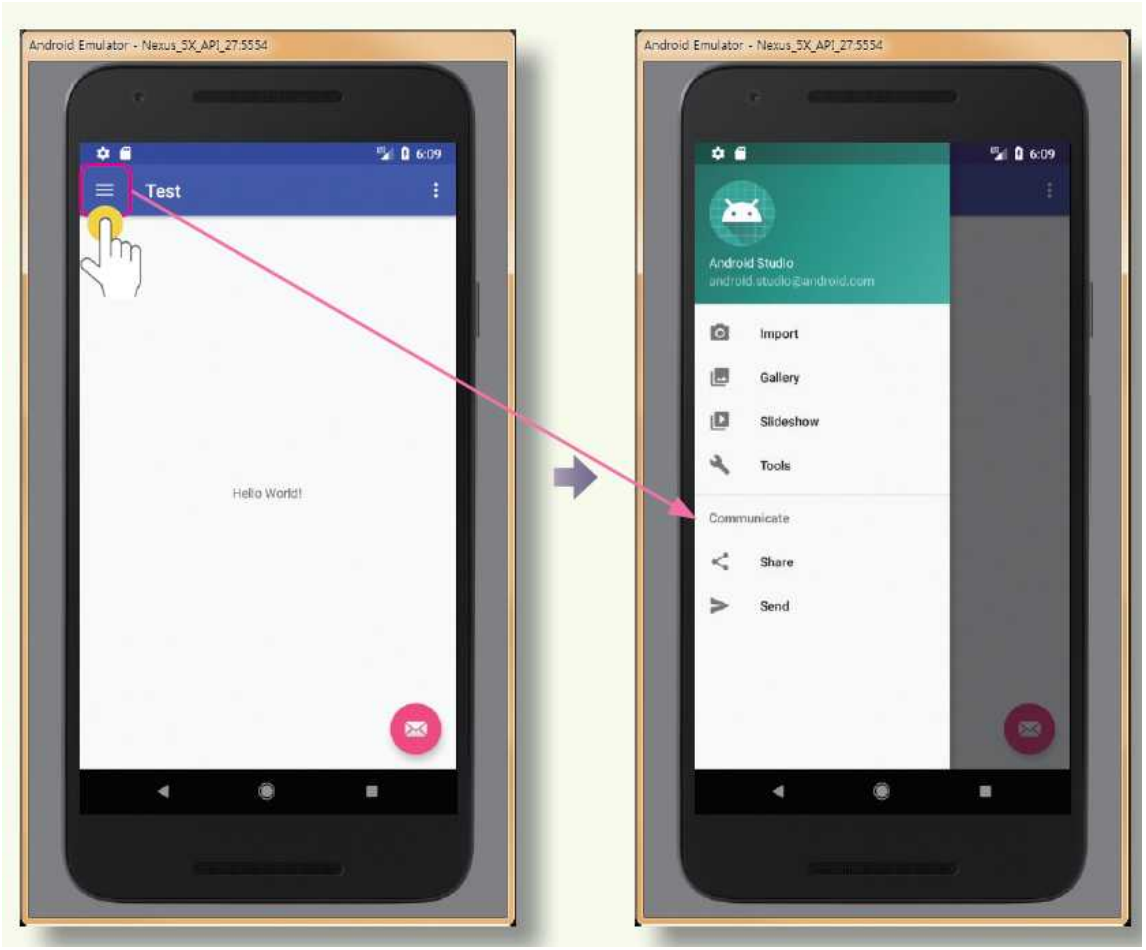
© Dept. of Computer Engineering, SeoKyeong Univ. 2021

Lab: 여러 가지 형식의 앱



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

실행 결과



© Dept. of Computer Engineering, SeoKyeong Univ. 2021

Summary

- 애플리케이션은 **컴포넌트**들의 조합으로 만들어진다.
- **코드**와 **리소스**는 철저하게 분리된다.
- 코드와 리소스는 개발 도구에 의하여 자동으로 생성되는 **R.java**에 의하여 서로 연결된다.

© Dept. of Computer Engineering, SeoKyeong Univ. 2021