모바일 프로그래밍

01 안드로이드 앱 개발 개요 2

2017 2학기

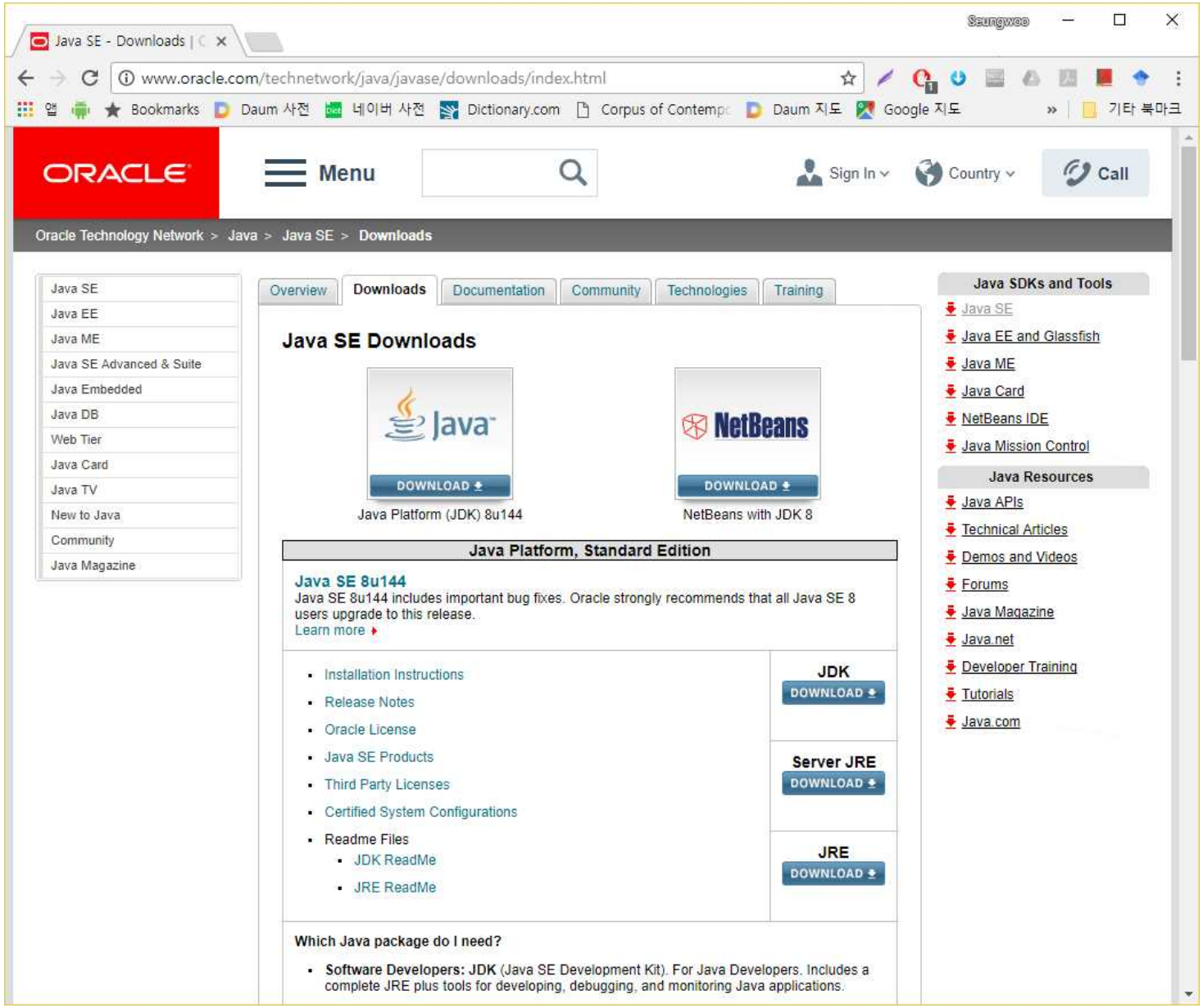
강승우

개발 도구 설치

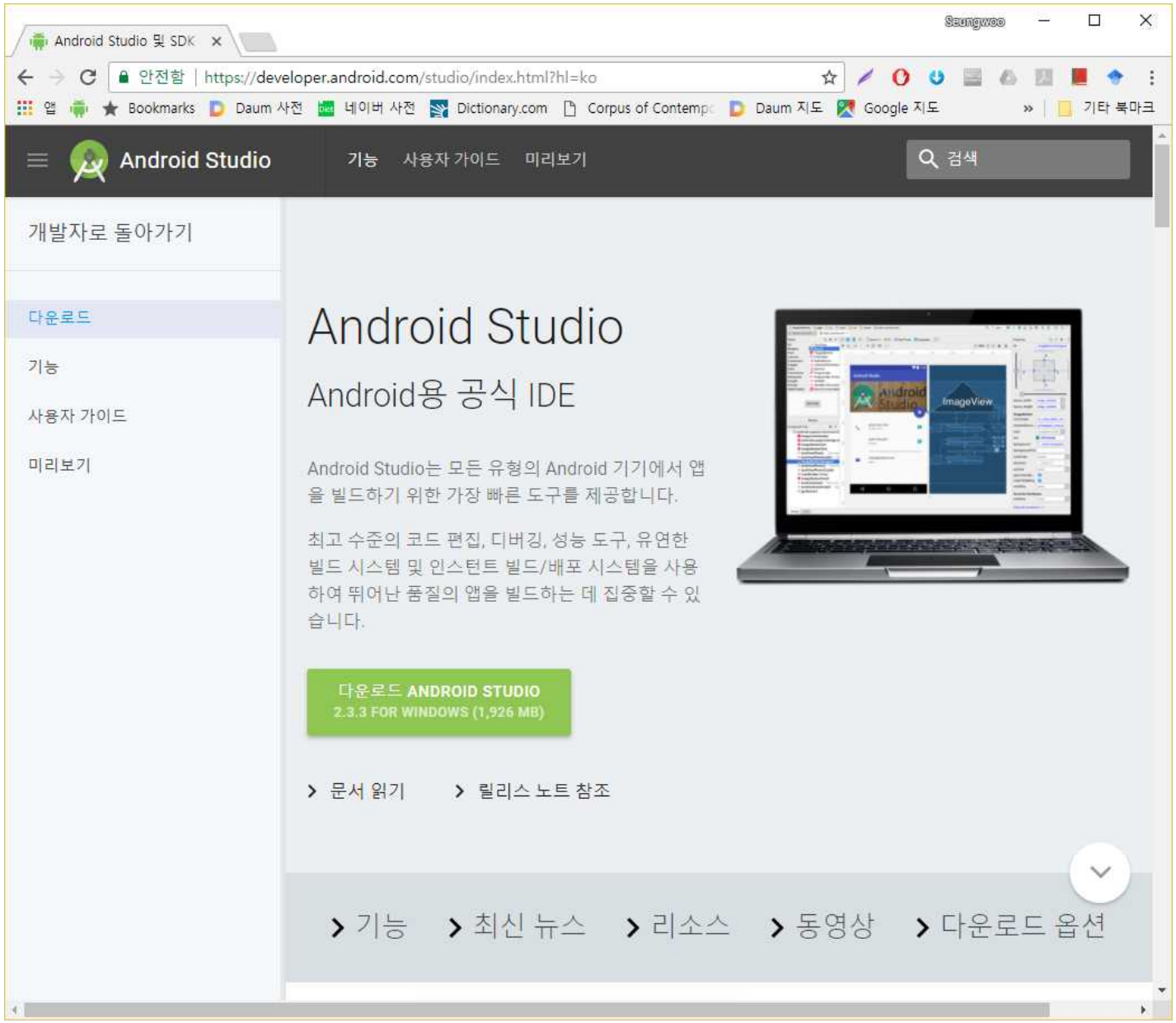
안드로이드 개발 도구

- JDK (Java Development Kit)
  - Java SE
  - http://www.oracle.com/technetwork/java/javase/downloads/index.html
    - 2017.09 현재 Version 8 Update 144
- Android Studio
  - https://developer.android.com/studio/index.html
    - https://developer.android.com/studio/releases/index.html (Release Note)
    - http://tools.android.com/recent (Recent Changes)
- Android SDK
  - Android Studio 안에 포함

- JDK 설치

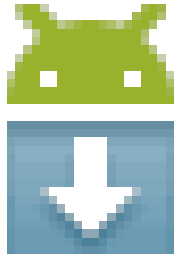

Java SE - Downloads
www.oracle.com/technetwork/java/javase/downloads/index.html
ORACLE
Menu
Sign In
Country
Call
Oracle Technology Network > Java > Java SE > Downloads
Java SE
Java EE
Java ME
Java SE Advanced & Suite
Java Embedded
Java DB
Web Tier
Java Card
Java TV
New to Java
Community
Java Magazine
Overview
Downloads
Documentation
Community
Technologies
Training
Java SE Downloads
DOWNLOAD
Java Platform (JDK) 8u144
NetBeans with JDK 8
Java Platform, Standard Edition
Java SE 8u144
Java SE 8u144 includes important bug fixes. Oracle strongly recommends that all Java SE 8 users upgrade to this release.
Learn more
Installation Instructions
Release Notes
Oracle License
Java SE Products
Third Party Licenses
Certified System Configurations
Readme Files
JDK ReadMe
JRE ReadMe
JDK
Server JRE
JRE
Which Java package do I need?
Software Developers: JDK (Java SE Development Kit). For Java Developers. Includes a complete JRE plus tools for developing, debugging, and monitoring Java applications.
Java SDKs and Tools
Java SE
Java EE and Glassfish
Java ME
Java Card
NetBeans IDE
Java Mission Control
Java Resources
Java APIs
Technical Articles
Demos and Videos
Forums
Java Magazine
Java.net
Developer Training
Tutorials
Java.com

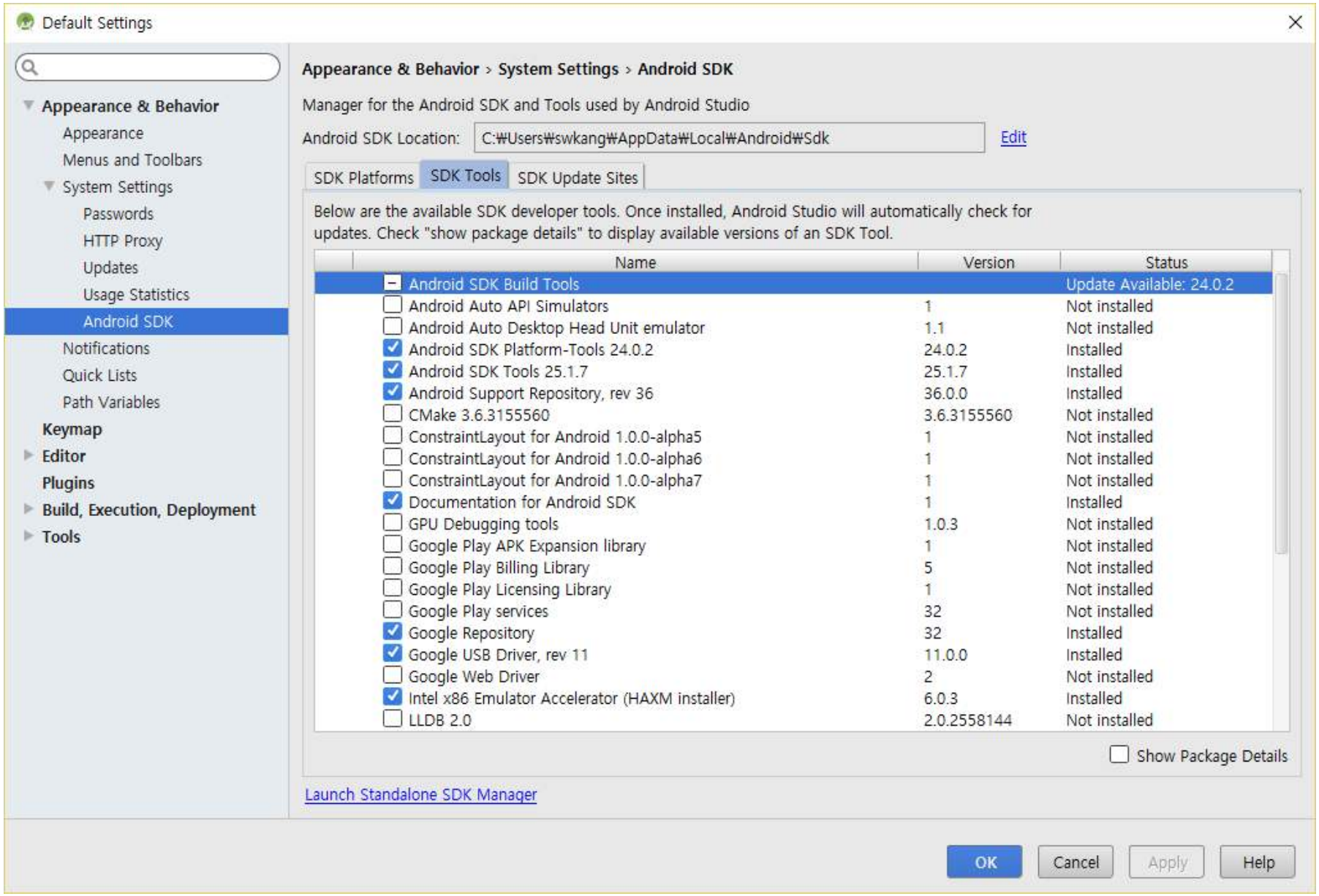
- Android Studio 설치



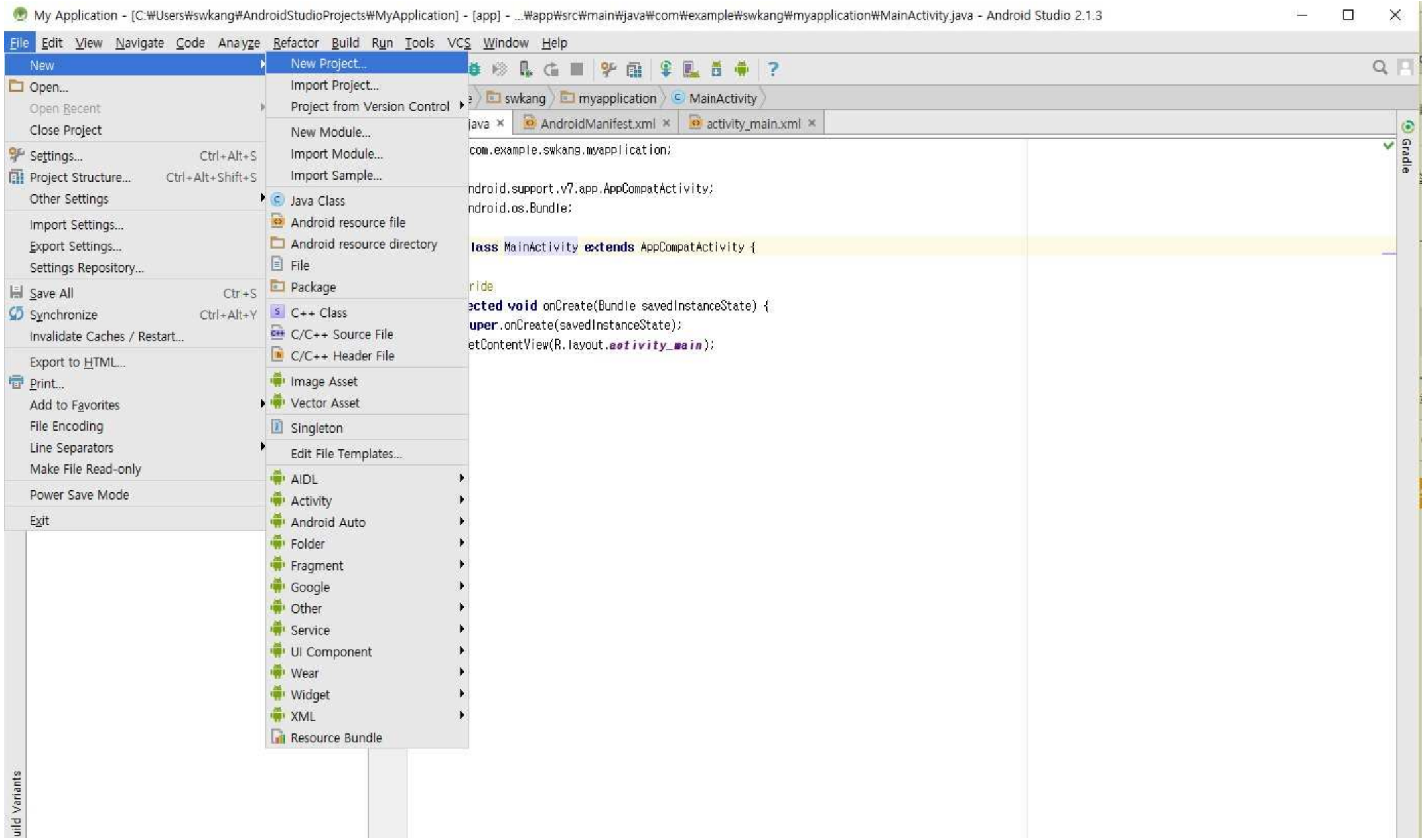
IDE 및 SDK 도구 업데이트

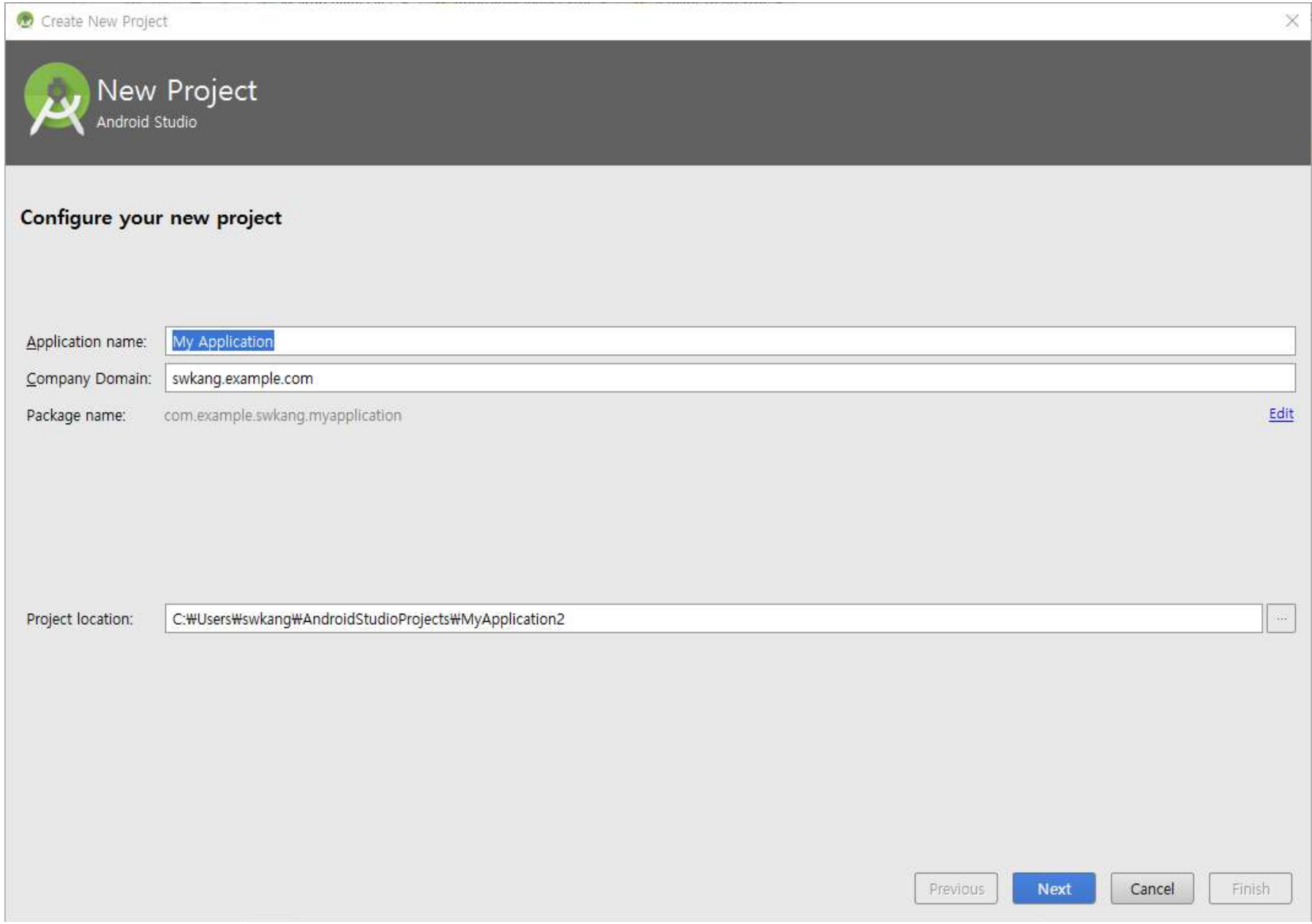
- IDE 업데이트
  - **Help** > **Check for Update**
  - https://developer.android.com/studio/intro/update.html
- SDK 도구 업데이트
  - SDK Manager 실행
    - **Tools > Android > SDK Manager**

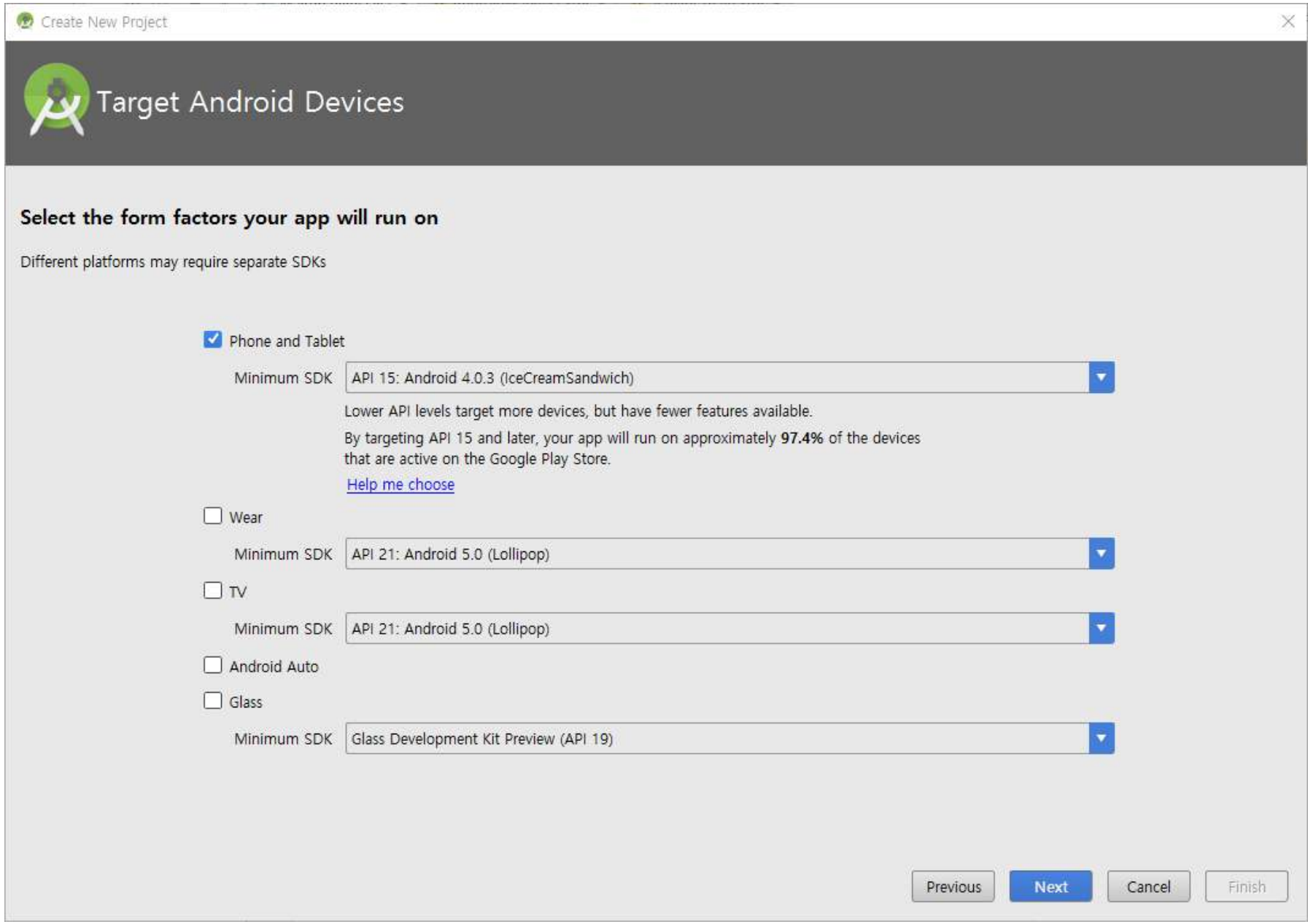


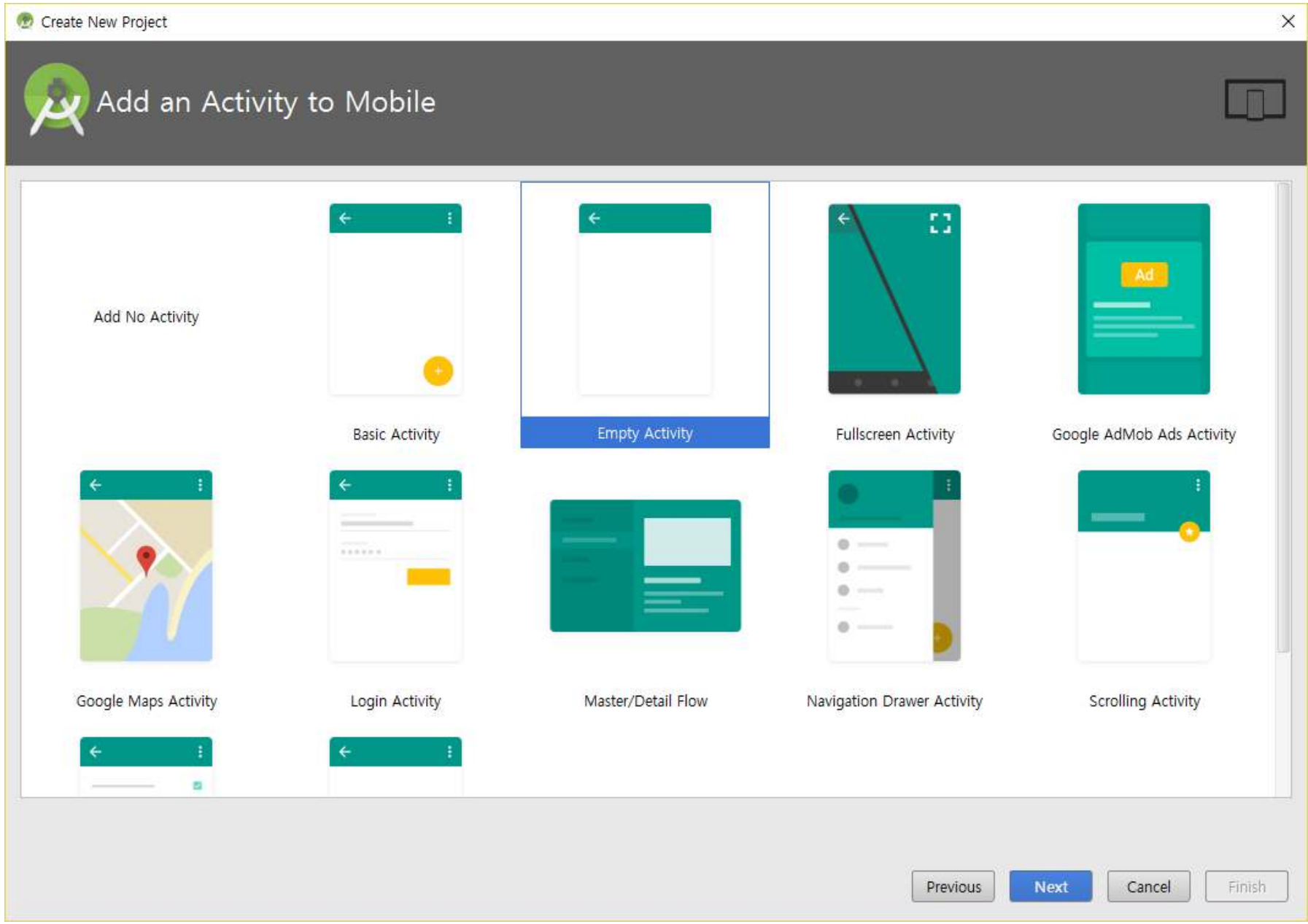

Default Settings
Appearance & Behavior
Appearance
Menus and Toolbars
System Settings
Passwords
HTTP Proxy
Updates
Usage Statistics
Android SDK
Notifications
Quick Lists
Path Variables
Keymap
Editor
Plugins
Build, Execution, Deployment
Tools
Appearance & Behavior › System Settings › Android SDK
Manager for the Android SDK and Tools used by Android Studio
Android SDK Location: C:₩Users₩swkang₩AppData₩Local₩Android₩Sdk
Edit
SDK Platforms
SDK Tools
SDK Update Sites
Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.
Name
Version
Status
Android SDK Build Tools
Update Available: 24.0.2
Android Auto API Simulators
1
Not installed
Android Auto Desktop Head Unit emulator
1.1
Not installed
Android SDK Platform-Tools 24.0.2
24.0.2
Installed
Android SDK Tools 25.1.7
25.1.7
Installed
Android Support Repository, rev 36
36.0.0
Installed
CMake 3.6.3155560
3.6.3155560
Not installed
ConstraintLayout for Android 1.0.0-alpha5
1
Not installed
ConstraintLayout for Android 1.0.0-alpha6
1
Not installed
ConstraintLayout for Android 1.0.0-alpha7
1
Not installed
Documentation for Android SDK
1
Installed
GPU Debugging tools
1.0.3
Not installed
Google Play APK Expansion library
1
Not installed
Google Play Billing Library
5
Not installed
Google Play Licensing Library
1
Not installed
Google Play services
32
Not installed
Google Repository
32
Installed
Google USB Driver, rev 11
11.0.0
Installed
Google Web Driver
2
Not installed
Intel x86 Emulator Accelerator (HAXM installer)
6.0.3
Installed
LLDB 2.0
2.0.2558144
Not installed
Show Package Details
Launch Standalone SDK Manager
OK
Cancel
Apply
Help

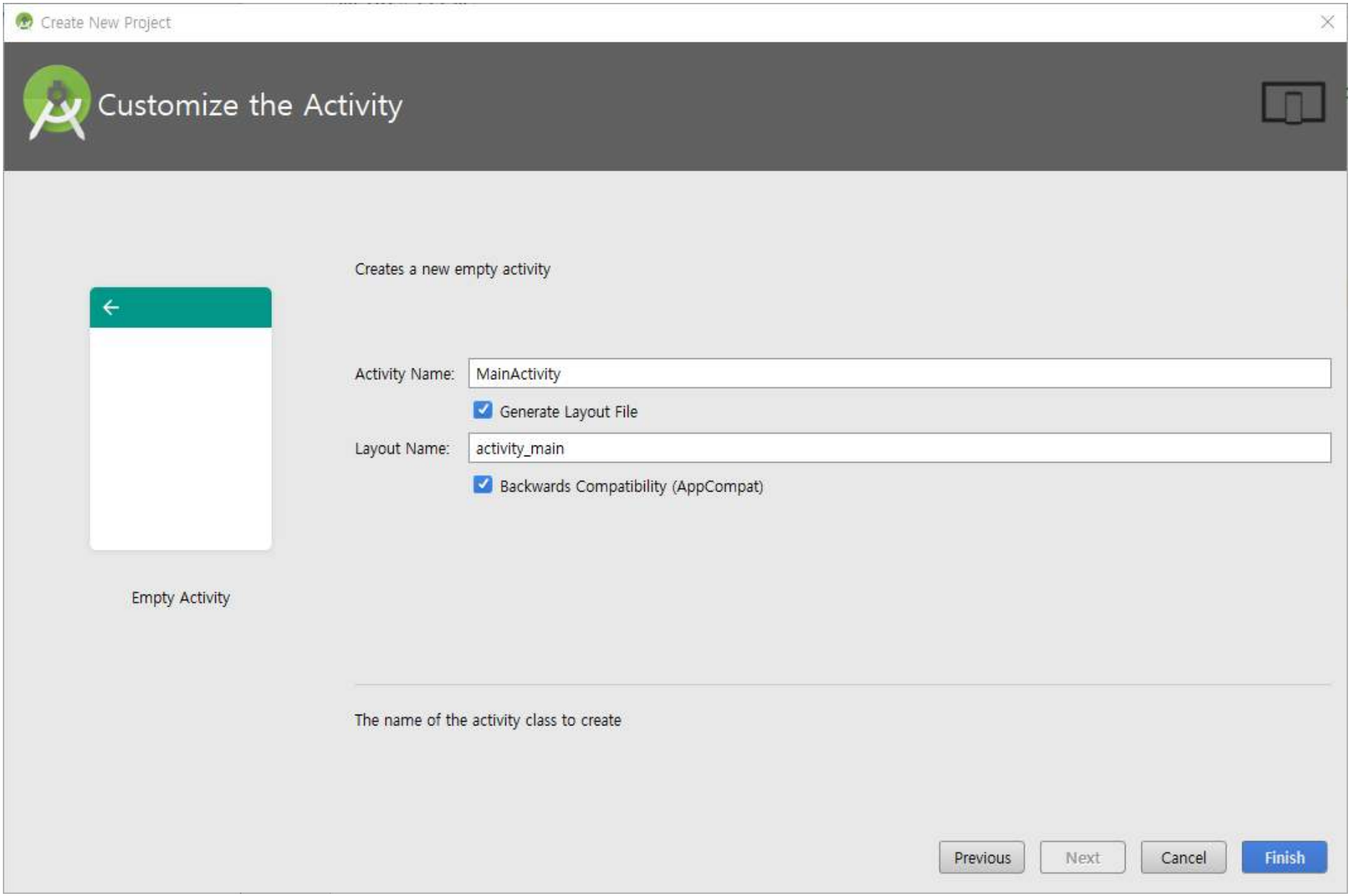
- Android project 생성
  - http://developer.android.com/training/basics/firstapp/creating-project.html

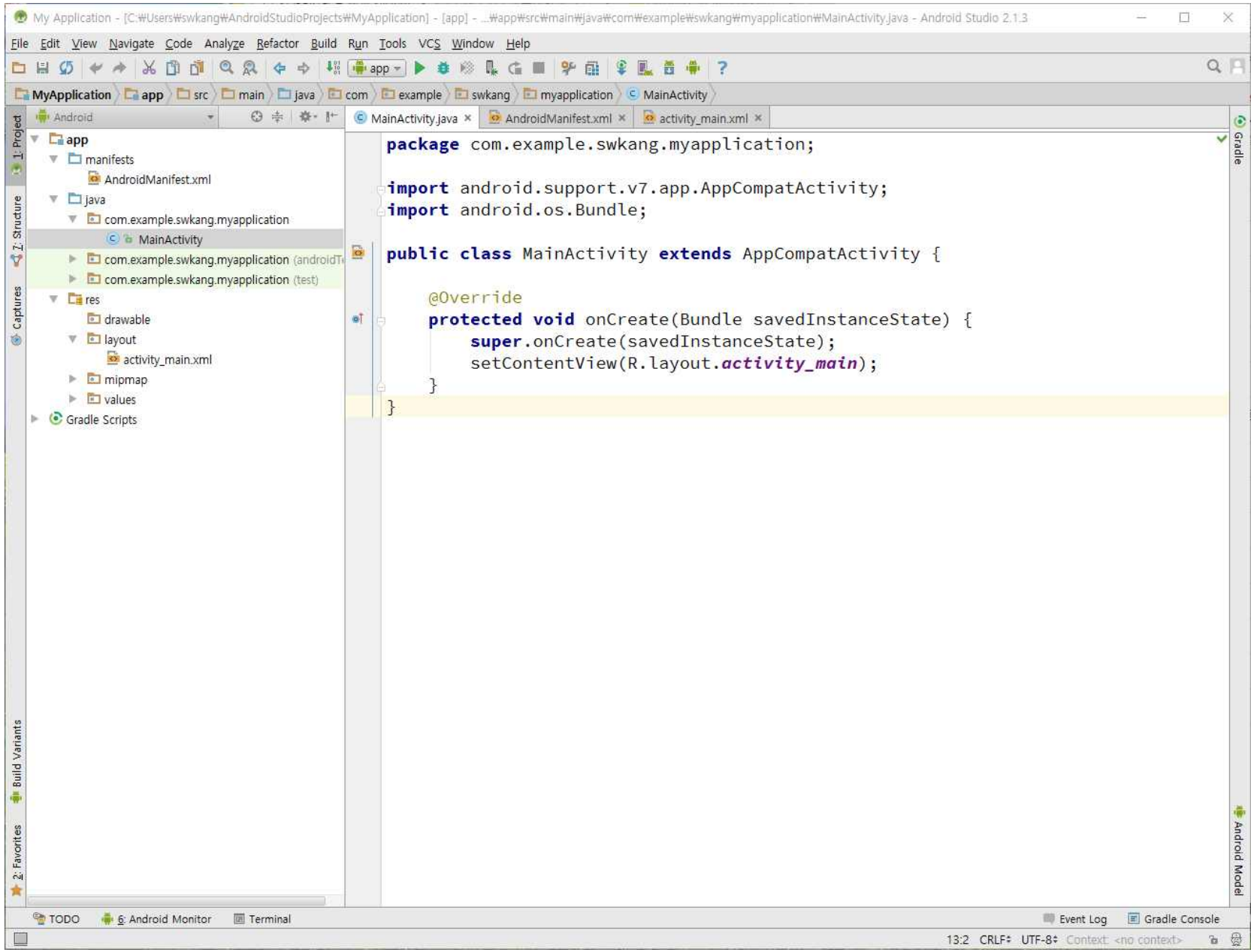

My Application - [C:₩Users₩swkang₩AndroidStudioProjects₩MyApplication] - [app] - ...₩app₩src₩main₩java₩com₩example₩swkang₩myapplication₩MainActivity.java - Android Studio 2.1.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
New
Open...
Open Recent
Close Project
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
Other Settings
Import Settings...
Export Settings...
Settings Repository...
Save All Ctr+S
Synchronize Ctrl+Alt+Y
Invalidate Caches / Restart...
Export to HTML...
Print...
Add to Favorites
File Encoding
Line Separators
Make File Read-only
Power Save Mode
Exit
New Project...
Import Project...
Project from Version Control
New Module...
Import Module...
Import Sample...
Java Class
Android resource file
Android resource directory
File
Package
C++ Class
C/C++ Source File
C/C++ Header File
Image Asset
Vector Asset
Singleton
Edit File Templates...
AIDL
Activity
Android Auto
Folder
Fragment
Google
Other
Service
UI Component
Wear
Widget
XML
Resource Bundle
swkang
myapplication
MainActivity
AndroidManifest.xml
activity_main.xml
com.example.swkang.myapplication;
ndroid.support.v7.app.AppCompatActivity;
ndroid.os.Bundle;
lass MainActivity extends AppCompatActivity {
ride
ected void onCreate(Bundle savedInstanceState) {
uper.onCreate(savedInstanceState);
etContentView(R.layout.activity_main);
Gradle
uild Variants


Create New Project
New Project
Android Studio
Configure your new project
Application name:
My Application
Company Domain:
swkang.example.com
Package name:
com.example.swkang.myapplication
Edit
Project location:
C:₩Users₩swkang₩AndroidStudioProjects₩MyApplication2
Previous
Next
Cancel
Finish


Create New Project
Target Android Devices
Select the form factors your app will run on
Different platforms may require separate SDKs
Phone and Tablet
Minimum SDK
API 15: Android 4.0.3 (IceCreamSandwich)
Lower API levels target more devices, but have fewer features available.
By targeting API 15 and later, your app will run on approximately 97.4% of the devices that are active on the Google Play Store.
Help me choose
Wear
Minimum SDK
API 21: Android 5.0 (Lollipop)
TV
Minimum SDK
API 21: Android 5.0 (Lollipop)
Android Auto
Glass
Minimum SDK
Glass Development Kit Preview (API 19)
Previous
Next
Cancel
Finish

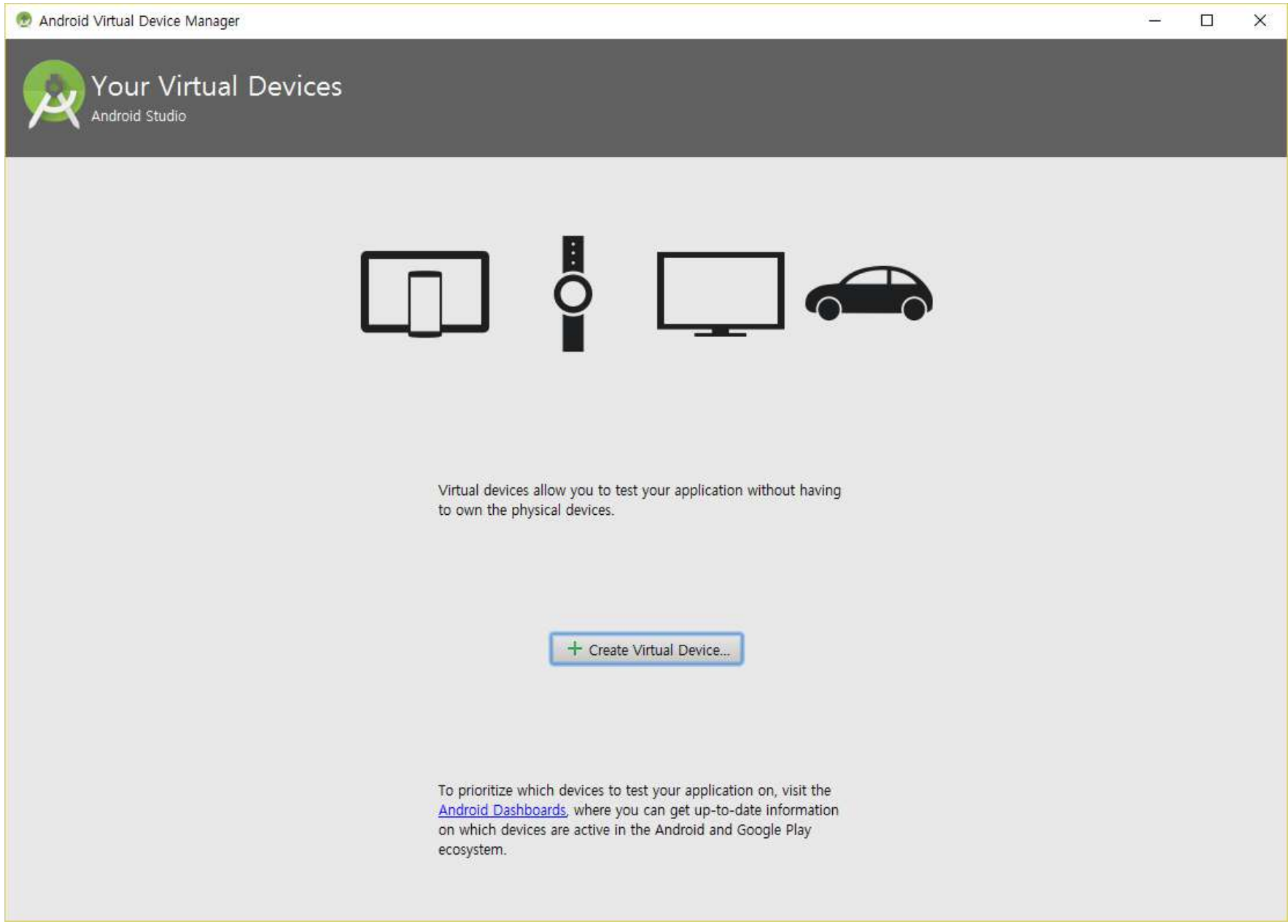

Create New Project
Add an Activity to Mobile
Add No Activity
Basic Activity
Empty Activity
Fullscreen Activity
Ad
Google AdMob Ads Activity
Google Maps Activity
Login Activity
Master/Detail Flow
Navigation Drawer Activity
Scrolling Activity
Previous
Next
Cancel
Finish

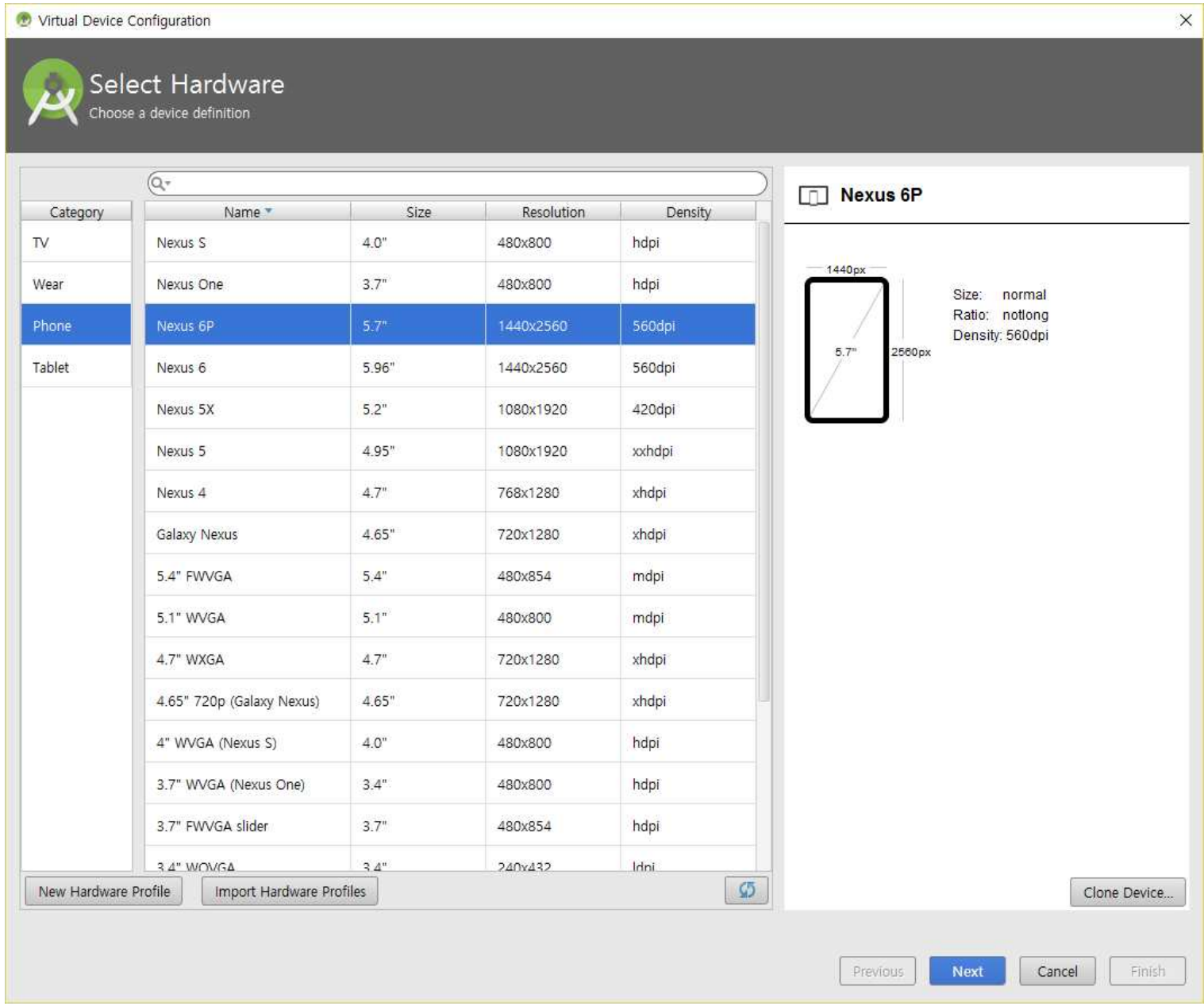

Create New Project
Customize the Activity
Creates a new empty activity
Activity Name:
MainActivity
Generate Layout File
Layout Name:
activity_main
Backwards Compatibility (AppCompat)
Empty Activity
The name of the activity class to create
Previous
Next
Cancel
Finish

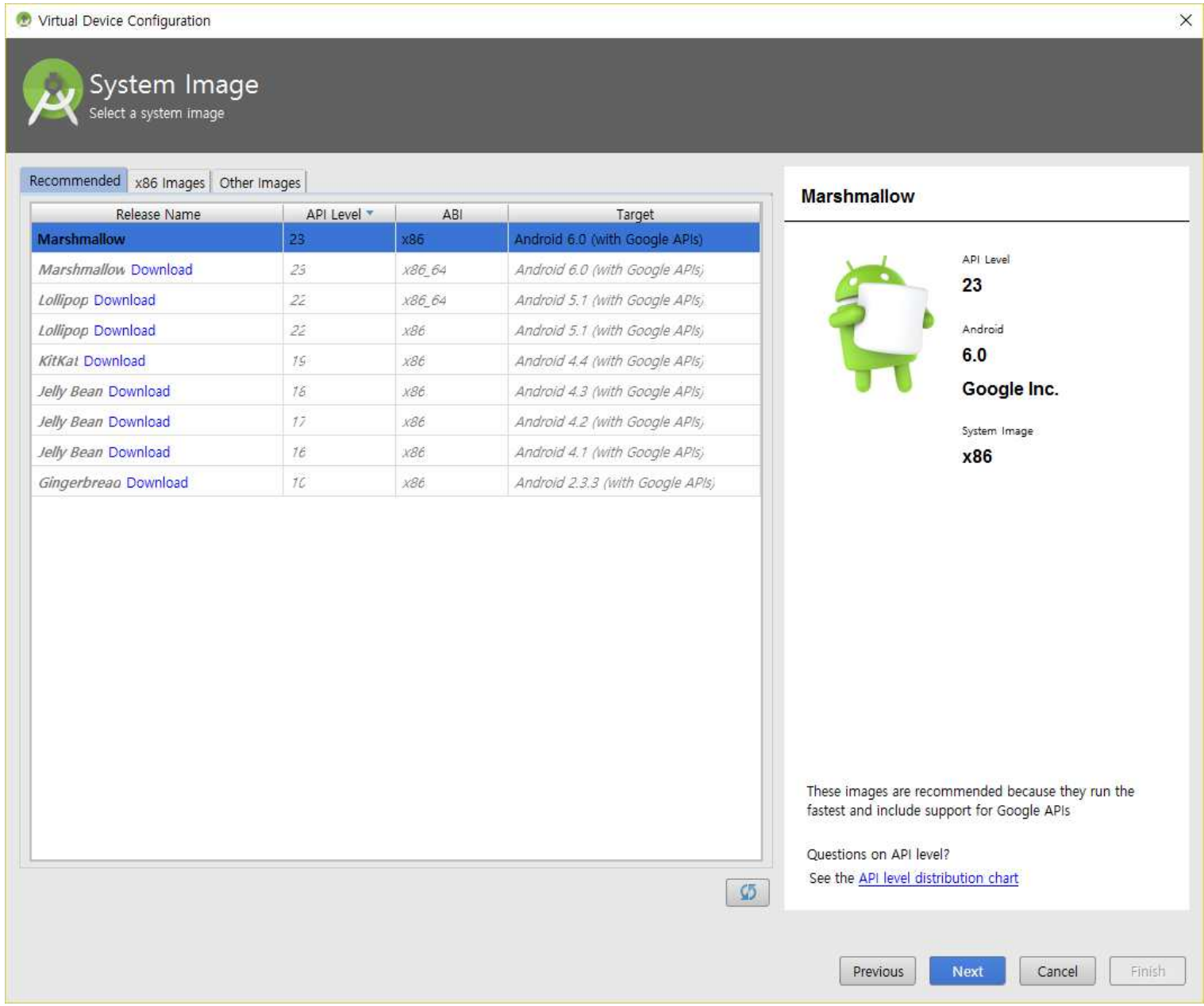

My Application - [C:₩Users₩swkang₩AndroidStudioProjects₩MyApplication] - [app] - ...₩app₩src₩main₩java₩com₩example₩swkang₩myapplication₩MainActivity.java - Android Studio 2.1.3
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication app src main java com example swkang myapplication MainActivity
Android
app
manifests
AndroidManifest.xml
java
com.example.swkang.myapplication
MainActivity
com.example.swkang.myapplication (androidT
com.example.swkang.myapplication (test)
res
drawable
layout
activity_main.xml
mipmap
values
Gradle Scripts
MainActivity.java
AndroidManifest.xml
activity_main.xml
package com.example.swkang.myapplication;
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);
}
}
1: Project
7: Structure
Captures
Build Variants
2: Favorites
Gradle
Android Model
TODO
6: Android Monitor
Terminal
Event Log
Gradle Console
13:2 CRLF UTF-8 Context: <no context>

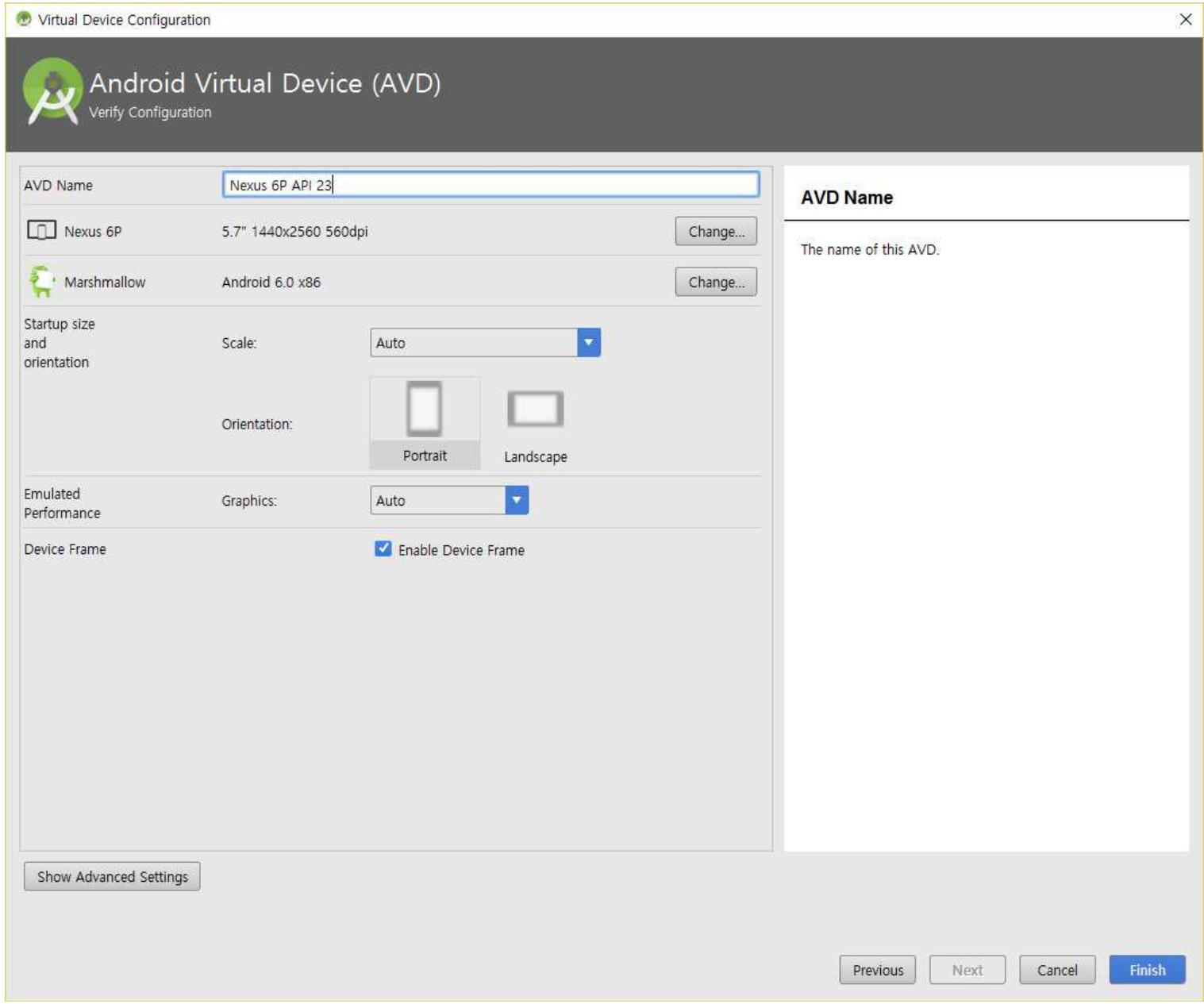
- App 실행
  - https://developer.android.com/training/basics/firstapp/running-app.html
  - 안드로이드 스마트폰에서 실행하기
  - 에뮬레이터에서 실행하기
    - AVD (Android Virtual Device)

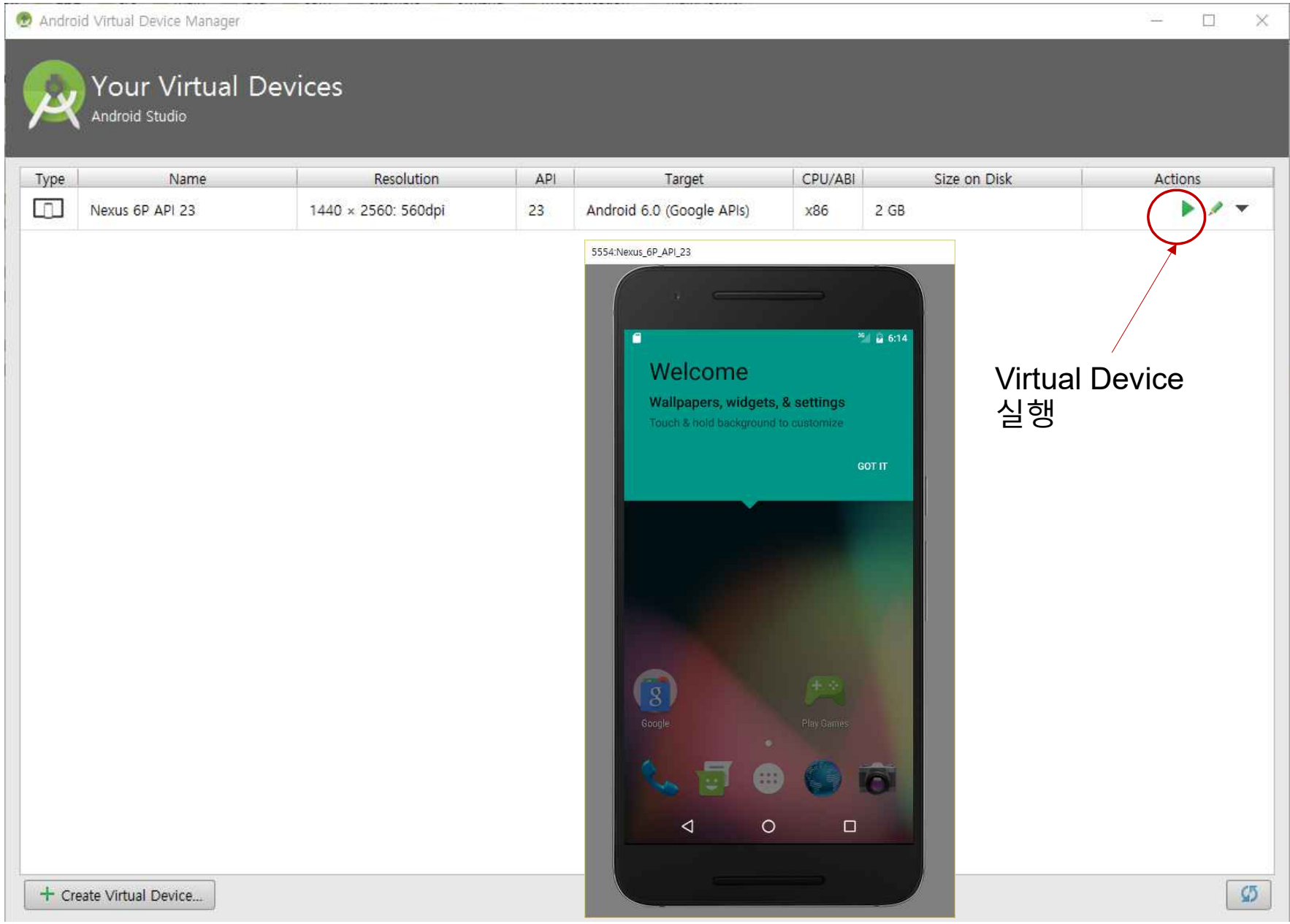
- 에뮬레이터 생성 및 실행
  - Tools → Android → AVD Manager


Android Virtual Device Manager
Your Virtual Devices
Android Studio
Virtual devices allow you to test your application without having to own the physical devices.
Create Virtual Device...
To prioritize which devices to test your application on, visit the Android Dashboards, where you can get up-to-date information on which devices are active in the Android and Google Play ecosystem.


Virtual Device Configuration
Select Hardware
Choose a device definition
Category
Name
Size
Resolution
Density
TV
Wear
Phone
Tablet
Nexus S 4.0" 480x800 hdpi
Nexus One 3.7" 480x800 hdpi
Nexus 6P 5.7" 1440x2560 560dpi
Nexus 6 5.96" 1440x2560 560dpi
Nexus 5X 5.2" 1080x1920 420dpi
Nexus 5 4.95" 1080x1920 xxhdpi
Nexus 4 4.7" 768x1280 xhdpi
Galaxy Nexus 4.65" 720x1280 xhdpi
5.4" FWVGA 5.4" 480x854 mdpi
5.1" WVGA 5.1" 480x800 mdpi
4.7" WXGA 4.7" 720x1280 xhdpi
4.65" 720p (Galaxy Nexus) 4.65" 720x1280 xhdpi
4" WVGA (Nexus S) 4.0" 480x800 hdpi
3.7" WVGA (Nexus One) 3.4" 480x800 hdpi
3.7" FWVGA slider 3.7" 480x854 hdpi
3.4" WQVGA 3.4" 240x432 ldpi
New Hardware Profile
Import Hardware Profiles
Nexus 6P
1440px
5.7"
2560px
Size: normal
Ratio: notlong
Density: 560dpi
Clone Device...
Previous
Next
Cancel
Finish

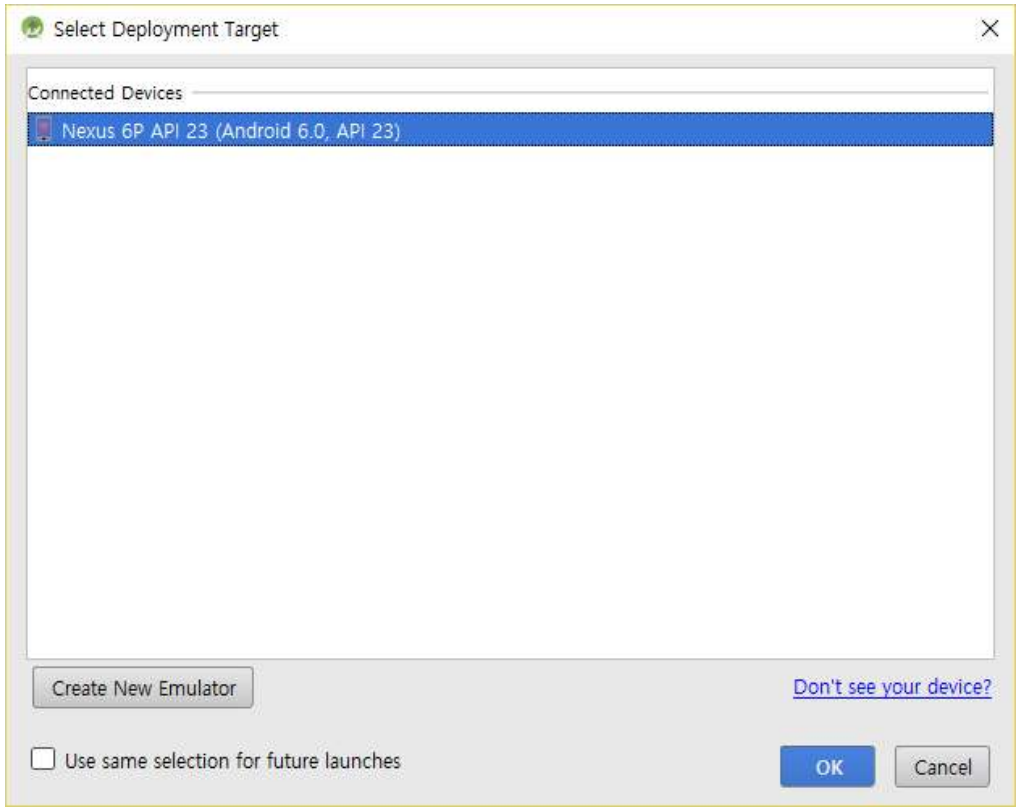

Virtual Device Configuration
System Image
Select a system image
Recommended
x86 Images
Other Images
Release Name
API Level
ABI
Target
Marshmallow 23 x86 Android 6.0 (with Google APIs)
Marshmallow Download 23 x86_64 Android 6.0 (with Google APIs)
Lollipop Download 22 x86_64 Android 5.1 (with Google APIs)
Lollipop Download 22 x86 Android 5.1 (with Google APIs)
KitKat Download 19 x86 Android 4.4 (with Google APIs)
Jelly Bean Download 18 x86 Android 4.3 (with Google APIs)
Jelly Bean Download 17 x86 Android 4.2 (with Google APIs)
Jelly Bean Download 16 x86 Android 4.1 (with Google APIs)
Gingerbread Download 10 x86 Android 2.3.3 (with Google APIs)
Marshmallow
API Level
23
Android
6.0
Google Inc.
System Image
x86
These images are recommended because they run the fastest and include support for Google APIs
Questions on API level?
See the API level distribution chart
Previous
Next
Cancel
Finish


Virtual Device Configuration
Android Virtual Device (AVD)
Verify Configuration
AVD Name
Nexus 6P API 23
Nexus 6P
5.7" 1440x2560 560dpi
Change...
Marshmallow
Android 6.0 x86
Change...
Startup size and orientation
Scale:
Auto
Orientation:
Portrait
Landscape
Emulated Performance
Graphics:
Auto
Device Frame
Enable Device Frame
Show Advanced Settings
AVD Name
The name of this AVD.
Previous
Next
Cancel
Finish


Android Virtual Device Manager
Your Virtual Devices
Android Studio
Type
Name
Resolution
API
Target
CPU/ABI
Size on Disk
Actions
Nexus 6P API 23
1440 × 2560: 560dpi
23
Android 6.0 (Google APIs)
x86
2 GB
5554:Nexus_6P_API_23
6:14
Welcome
Wallpapers, widgets, & settings
Touch & hold background to customize
GOT IT
Google
Play Games
Create Virtual Device...
Virtual Device
실행

- Run ‘app’

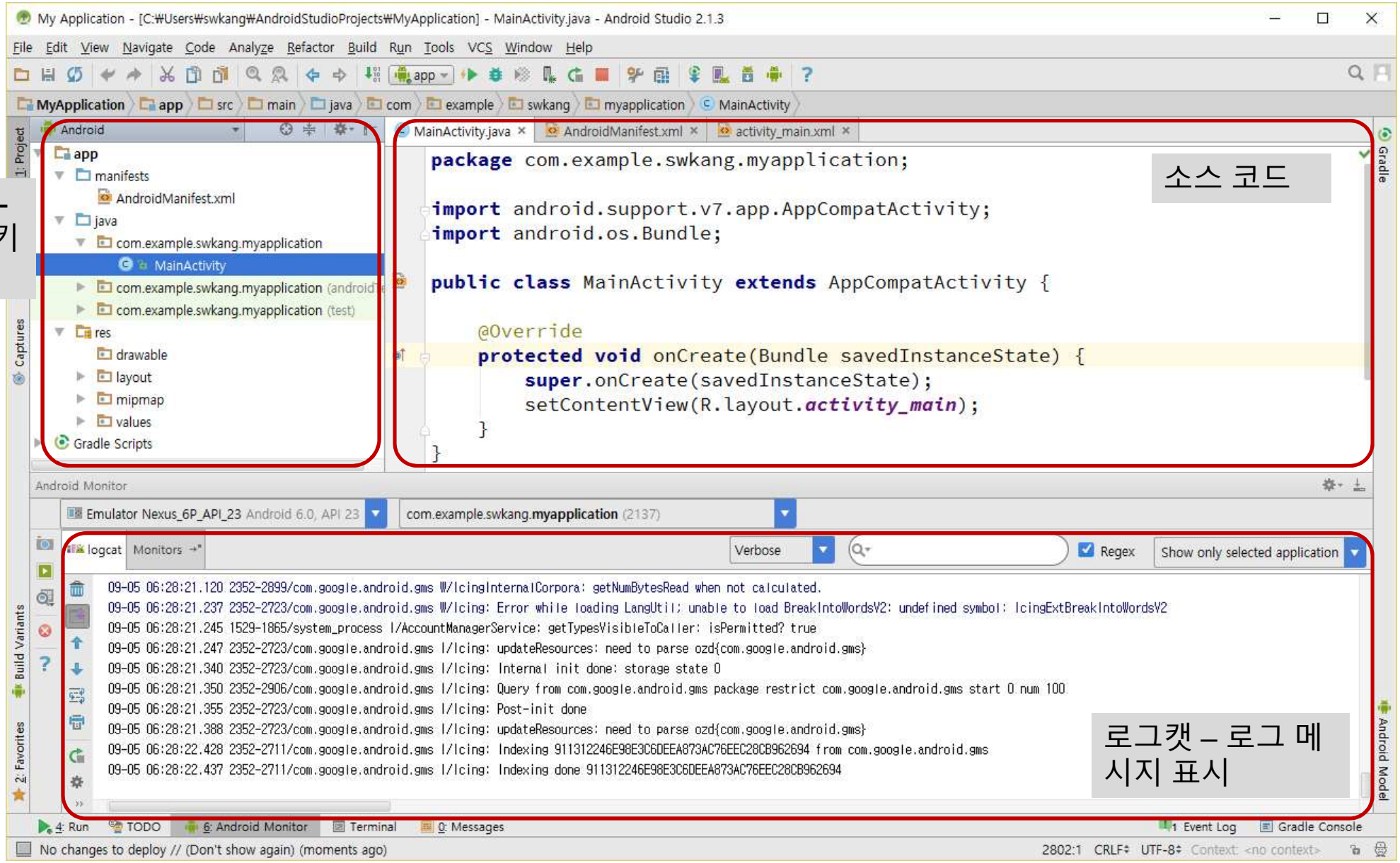
Emulator 선택



실행 화면

첫번째 애플리케이션

프로젝트 뷰 - 프로젝트 패키지 구성 요소



프로젝트 뷰 폴더 구성

- manifests
  - AndroidManfest.xml
- java
  - 자바 소스 파일
- res
  - 앱에 사용되는 각종 리소스들이 저장되는 폴더
  - drawable
    - 각 해상도 별 아이콘 이미지 파일
  - layout
    - 화면 레이아웃 정의하는 XML 파일
  - values
    - 앱에서 사용하는 문자열 값 등을 XML 파일로 명시
- Gradle Scripts
  - 프로젝트 빌드와 관련된 스크립트 파일들

자동 생성 소스 분석

- MainActivity.java

- 안드로이드 프로그램에는 main()이 없음
- 액티비티 별로 실행
- 액티비티 중에서는 onCreate() 메소드가 가장 먼저 실행

```
package com.example.swkang.myapplication;

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);
    }
}
```

package 이름 선언

```
package com.example.swkang.myapplication;
```

- package
  - 클래스들을 모아서 보관하는 컨테이너
  - Package 이름
    - 일반적으로 인터넷 도메인 이름을 역순으로 사용
    - unique한 namespace를 지정하는 역할

    → 동일한 이름의 클래스를 만들더라도 package 이름이 다르게 되면 구분 가능

import

```
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
```

- 외부 패키지나 클래스를 소스 프로그램에서 사용하고자 할 때
  - android로 시작하는 패키지는 안드로이드 SDK에서 제공하는 패키지

class 정의

```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

- AppCompatActivity를 상속 받는 액티비티 클래스

@Override

- Java annotation 중의 하나
  - 컴파일러에게 추가적인 정보를 제공
- 메소드가 부모 클래스의 메소드를 재정의(오버라이드) 하였음을 표시

onCreate()

```
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
}
```

- onCreate() 메소드
  - 액티비티가 처음 생성되어 실행되는 순간에 딱 한번 호출
  - 초기화와 사용자 인터페이스 (User Interface, UI) 설정 수행
- super.onCreate();
  - 부모 클래스의 onCreate() 메소드 호출
- setContentView(R.layout.activity_main);
  - 액티비티의 UI 화면 설정
  - R.layout.activity_main : res/layout/activity_main.xml 파일을 나타냄

안드로이드 애플리케이션 작성 절차

- 사용자 인터페이스 작성
  - XML
  - 레이아웃, 텍스트뷰, 버튼, 이미지뷰, 메뉴 등 정의
- 코드 작성
  - Java 소스 코드 작성
  - 코드에서 사용하는 리소스 준비
- 메니페스트 파일 작성
  - XML

사용자 인터페이스 정의

- XML을 이용하여 정의
  - 코드를 이용하여 UI를 설정할 수 있지만 XML로 표현하는 것이 바람직
  - 애플리케이션의 외관 (presentation)과 로직을 분리함
  - UI 구축 작업과 애플리케이션 로직 구현을 더 효과적이고 유연하게 할 수 있음
  - https://ko.wikipedia.org/wiki/XML

- UI 요소들은 XML의 하나의 element로 표현
  - TextView, EditText, Button, …

- Android Studio에서 제공되는 GUI 기반 Design 툴 이용 가능
- 이 강의에서는 XML을 이용하여 직접 기술하는 것으로 함

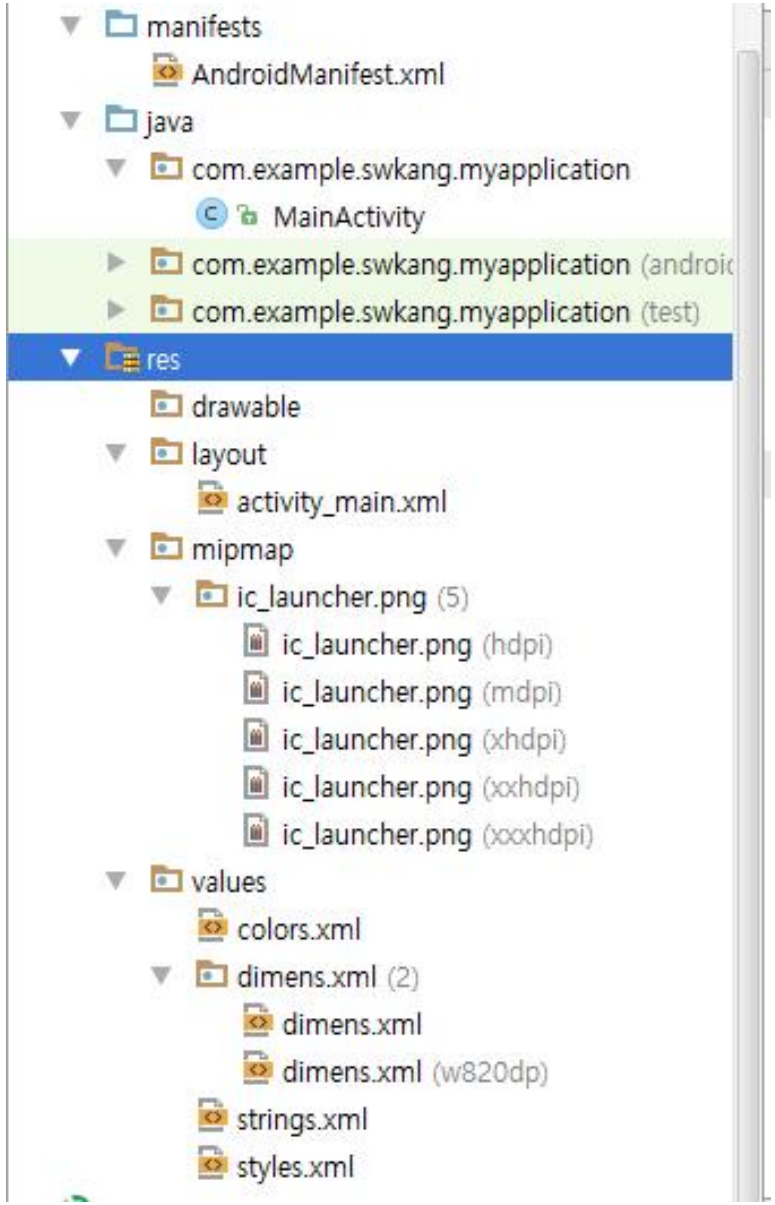
예제

```
<?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!" />
```

Tag, element, attribute (name, value)

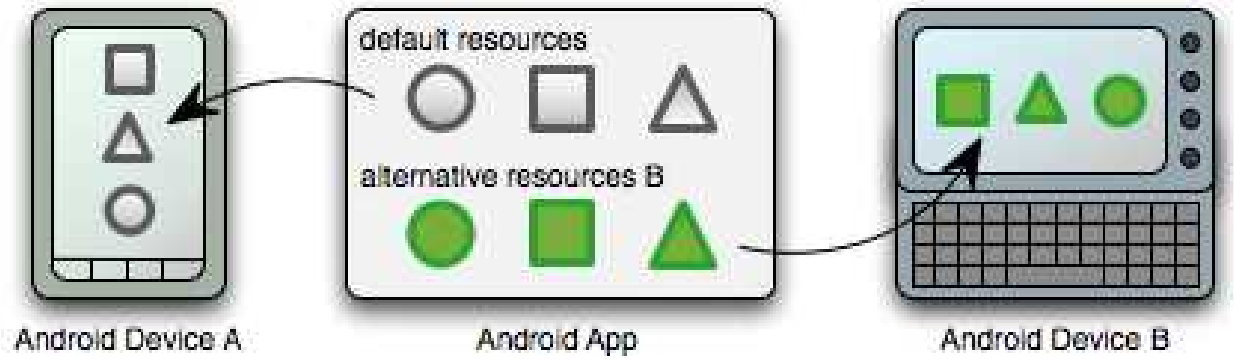
리소스

- 안드로이드에서 레이아웃, 이미지, 문자열 들을 리소스로 취급



코드와 리소스를 분리하는 이유

- 다양한 종류의 기기를 효과적으로 지원하기 위함
  - 해상도 / 화면 크기 / 언어
- 동작하는 애플리케이션 로직은 동일하지만, 리소스를 다르게 하는 것이 필요한 경우
  - 각 기기에 맞게 리소스를 준비하여 사용



문자열 리소스

- 문자열도 XML로 기술하는 것이 바람직
  - 국가별로 다른 언어 표시
  - 여러 번 사용되는 동일한 문자

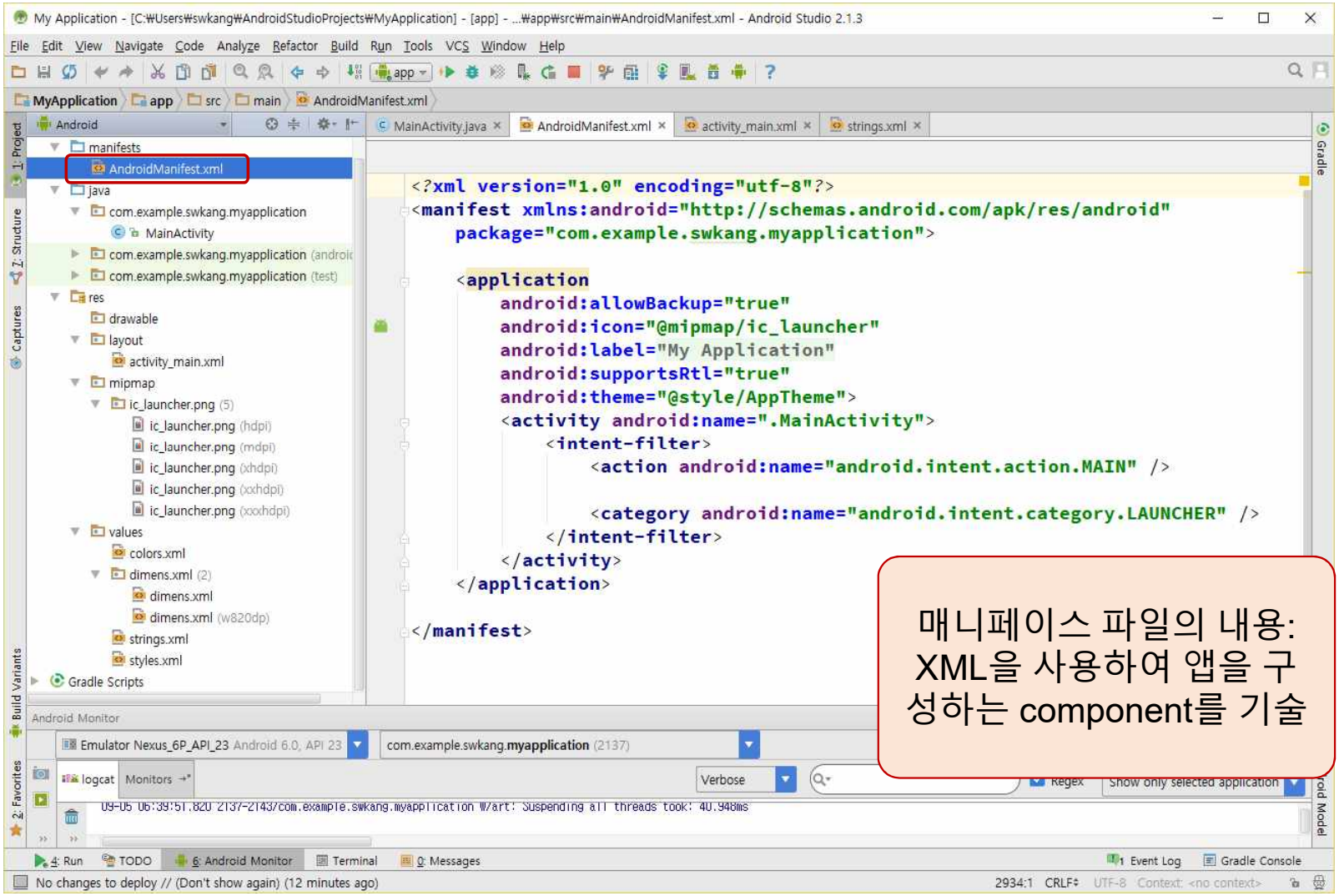
res/layout/activity_main.xml

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

res/values/strings.xml

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

매니페스트 파일



매니페이스 파일의 내용: XML을 사용하여 앱을 구성하는 component를 기술

매니페스트 파일 분석

ApplicationManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="kr.co.company.hello"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="18" />

    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
            android:name="kr.co.company.hello.MainActivity"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

하나의 애플리케이션

하나의 액티비티가 포함된다.

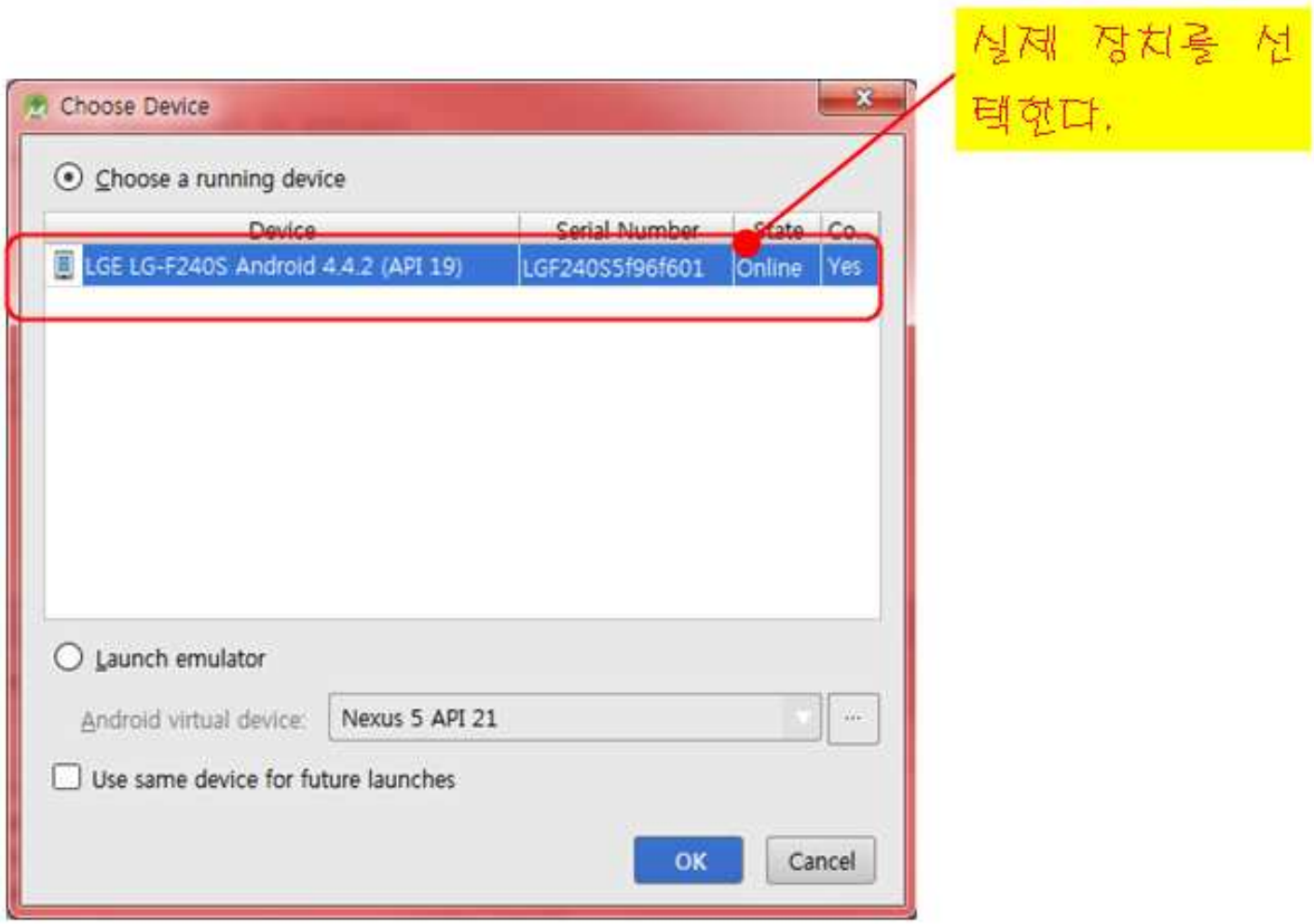
실제 장치와의 연결

1. USB 드라이버 설치
2. 장치에서 “USB 디버깅” 항목 활성화
3. 안드로이드 스튜디오에서 실행하기 원하는 장치 선택

장치에서 USB 디버깅 켜기

- 안드로이드 4.2 이상 기기에서 개발자 옵션 표시 방법
  - 설정 – 휴대전화 정보 – 빌드 번호 항목을 연속으로 일곱번 터치

실행할 장치 선택 화면


실제 장치를 선택한다.
Choose Device
Choose a running device
Device
Serial Number
State
Co...
LGE LG-F240S Android 4.4.2 (API 19)
LGF240S5f96f601
Online
Yes
Launch emulator
Android virtual device:
Nexus 5 API 21
Use same device for future launches
OK
Cancel

Tip

- Android Studio 폰트 변경
  - File -> Settings -> 좌측 Editors / Colors & Fonts / Font -> Save As 클릭, 이름 입력
  - Primary font 드랍다운 리스트에서 선택 (Source Code Pro) --> OK
- 필요한 패키지를 쉽게 프로젝트에 추가하는 방법
  - [File]->[Settings]->[Editor]->[General]->[Auto Import]
  - “Add unambigious imports on the fly” 옵션과 "Optimize imports on the fly“ 옵션을 체크