

안드로이드 프로그래밍 및 실습 - 개인과제1

- 유용석 교수님, 최유진 조교님 -

B289070, 이상민

1. 목표

기본적인 모바일 안드로이드 애플리케이션을 직접 제작해봄으로써 모바일 안드로이드 프로그래밍에 익숙해지기 위함이다. 특히, 버튼, 라디오 Widgets을 추가 / 삭제하면서 UI의 감각을 익힐 수 있다.

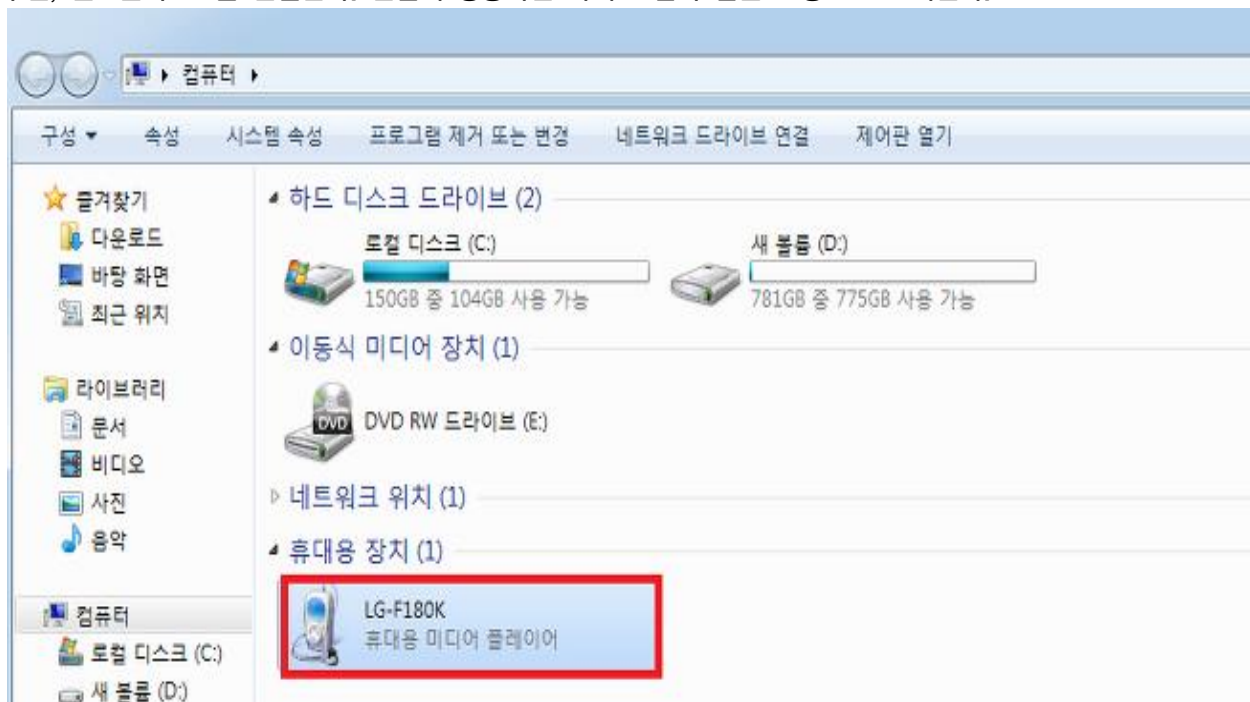
2. 기능 및 사용법

첫 번째 실습인 만큼, 모바일 안드로이드 애플리케이션(APK 파일)을 설치하는 방법에 대해서 소개하려고 한다. 그러한 이유로는, 이번 실습 과제가 특별한 기능이 있는 것이 아니며, 단순히 widgets을 추가하여 UI를 구성하기 때문이다.

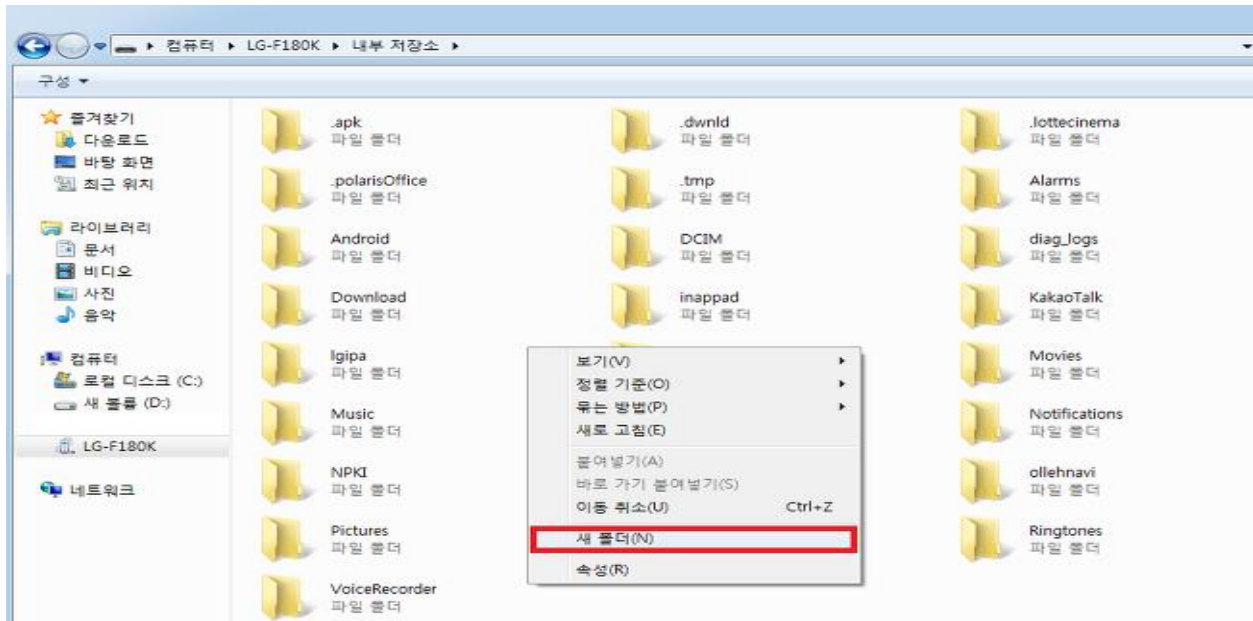
모바일 안드로이드 애플리케이션을 실행하는 방법은 크게 두 가지로 나눌 수 있다. 첫 번째는 실제 핸드폰에 설치하는 방법이다. 여기에는 공식 마켓(구글 플레이스토어, 앱스토어)을 통한 설치, 핸드폰을 PC에 연결하여 직접 제작한 파일 혹은 다운로드 받은 파일(APK, APP 등)을 핸드폰에 옮겨 설치하는 방법이 있다. 두 번째로는 안드로이드 가상 에뮬레이터를 사용하여 실행하는 방식을 말한다. 이 방식은 "3. 구현 방법 → 라. 애플리케이션 개발"을 참고하면 알 수 있다.

여기서는 핸드폰과 PC를 연결하여 제작한 APK 파일을 설치하는 방법에 대해서 소개하고자 한다.

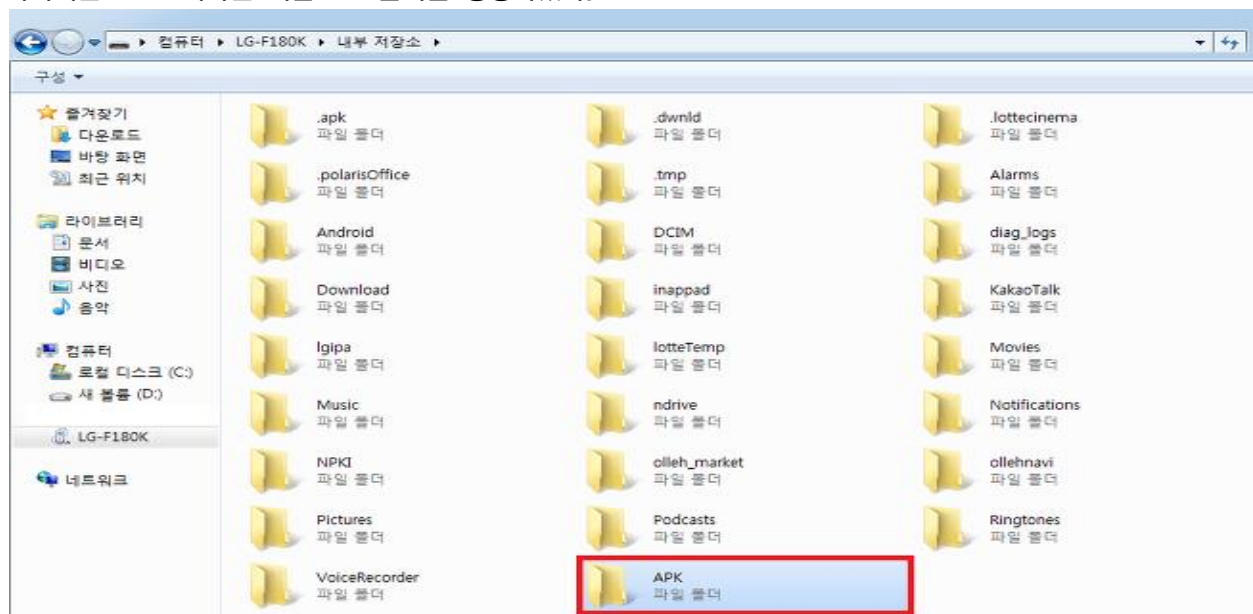
우선, 핸드폰과 PC를 연결한다. 연결에 성공하면 아래 그림과 같은 모양으로 표시된다.



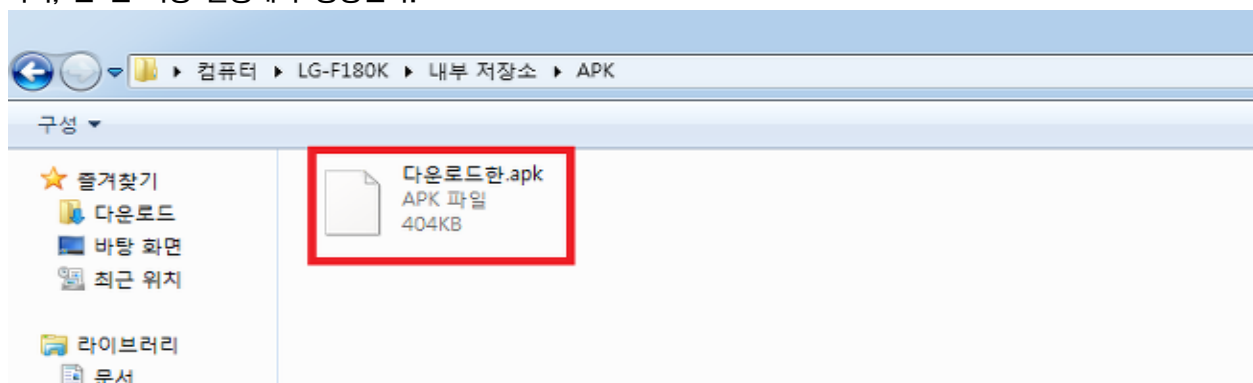
휴대용 장치(LG-F180K)를 더블 클릭하여 내부로 들어간 다음에, 마우스 오른쪽 버튼 → 새 폴더를 눌러 새로운 폴더를 생성한다.



여기서는 "APK"이라는 이름으로 폴더를 생성하였다.



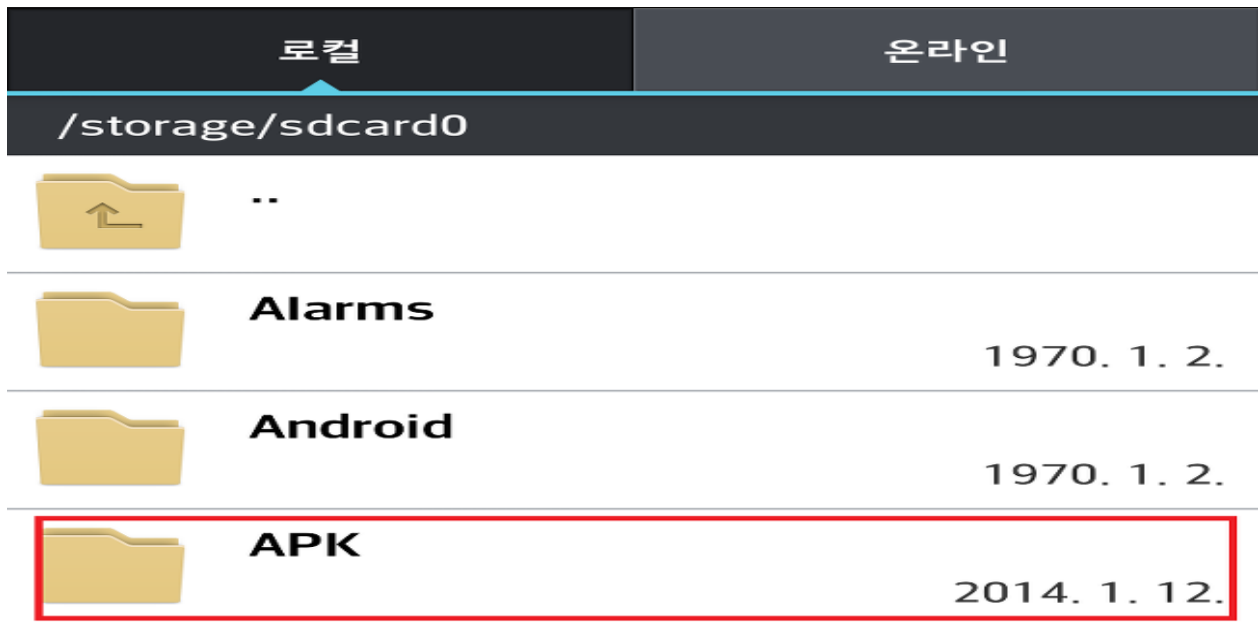
자신이 직접 제작한 APK 파일 또는 다운로드한 APK 파일을 마우스 드래그 앤 드롭하여 APK 폴더 안으로 넣는다. 우리가 제작한 APK 파일의 경로는 HelloAndroid → app → build → outputs → apk → app-debug.apk이며, 한 번 이상 실행해야 생성된다.



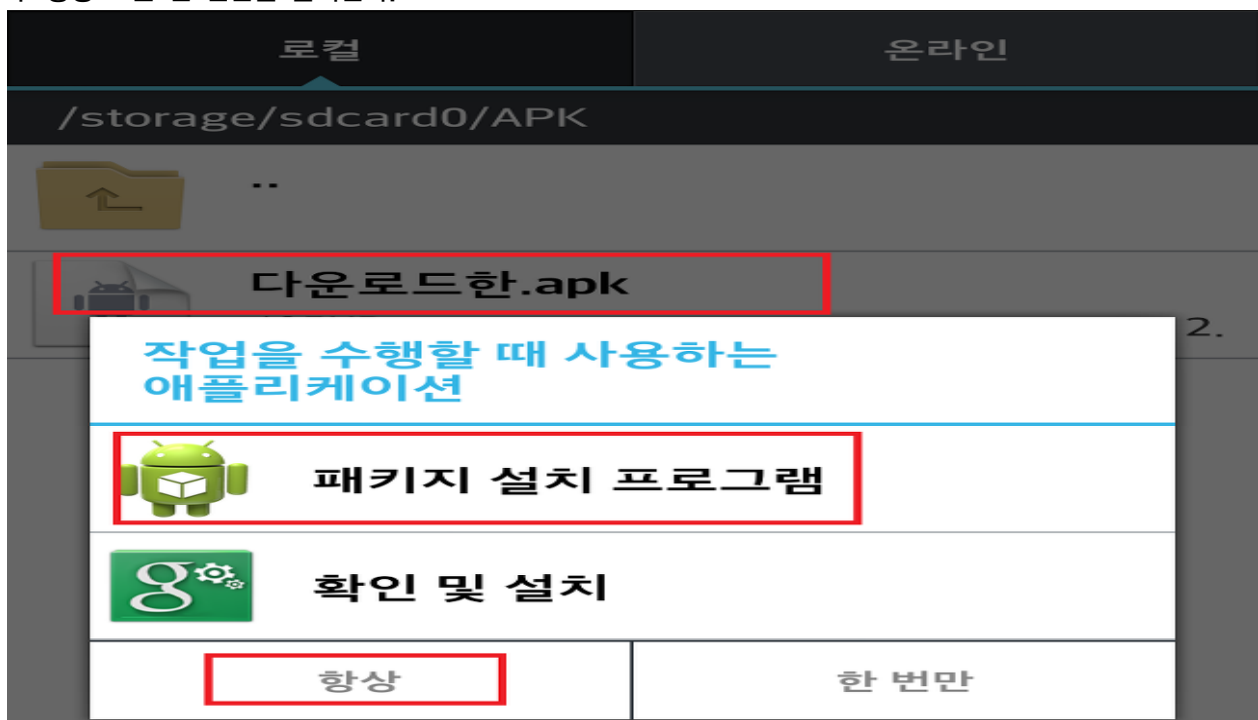
핸드폰을 PC에서 분리한 뒤, 핸드폰 내에 내장되어 있는 파일 관리자를 실행한다.



아까 만든 APK 폴더를 찾는다.



APK 폴더 안에는 PC에서 드래그 앤 드롭한 APK 파일이 있다. 한번 클릭한 뒤, 패키지 설치 프로그램 선택 후 항상 또는 한 번만을 클릭한다.



구글 플레이스토어와 같은 마켓에서 다운로드 하지 않은 애플리케이션은 보안상 휴대폰에서 설치가 안되도록 설정을 해 놓았다. 그리하여, 환경설정 → 보안 → 알 수 없는 소스 → 허용함 체크를 해줘야 설치가 가능하다.

  **보안**

USIM 설정

USIM 설정

비밀번호

입력 시 비밀번호 표시
비밀번호 입력 시 마지막 글자를 보여줍니다.

휴대폰 관리

휴대폰 관리자
휴대폰 관리자를 표시하거나 해제합니다.

알 수 없는 소스
허용하지 않음

자격증명 저장소

알 수 없는 소스

Play 스토어에 등록되지 않은 앱 설치를 허용합니다.

허용하지 않음

항상 묻기

허용함

취소

3. 구현 방법

이번 실습에 관한 소스 코드, 레이아웃 XML의 링크를 첨부한다.

소스 코드

[https://github.com/carpfishz/Android-](https://github.com/carpfishz/Android-Study/blob/master/HelloAndroid/app/src/main/java/com/example/administrator/helloandroid/MainActivity.java)

[Study/blob/master/HelloAndroid/app/src/main/java/com/example/administrator/helloandroid/MainActivity.java](https://github.com/carpfishz/Android-Study/blob/master/HelloAndroid/app/src/main/java/com/example/administrator/helloandroid/MainActivity.java)

레이아웃 XML

[https://github.com/carpfishz/Android-](https://github.com/carpfishz/Android-Study/blob/master/HelloAndroid/app/src/main/res/layout/activity_main.xml)

[Study/blob/master/HelloAndroid/app/src/main/res/layout/activity_main.xml](https://github.com/carpfishz/Android-Study/blob/master/HelloAndroid/app/src/main/res/layout/activity_main.xml)

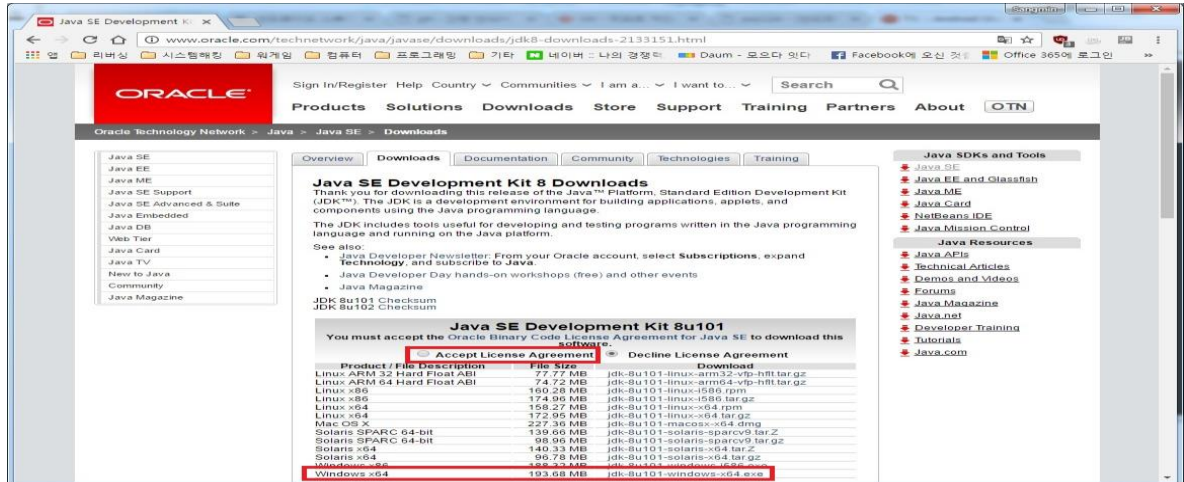
가. 테스트 환경 및 도구

- 1) Windows 7 64bit
- 2) Java 1.8.0_101
- 3) Android Studio 2.1
- 4) AVD : Nexus5, API23

나. 환경 구축

안드로이드 스튜디오를 설치하기 전에 자바가 설치되어 있어야 한다.

- 1) 자바 설치(<http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>)



라이선스 동의 버튼을 클릭한 뒤, 자신의 버전에 맞는 JDK 파일을 다운로드 한다.

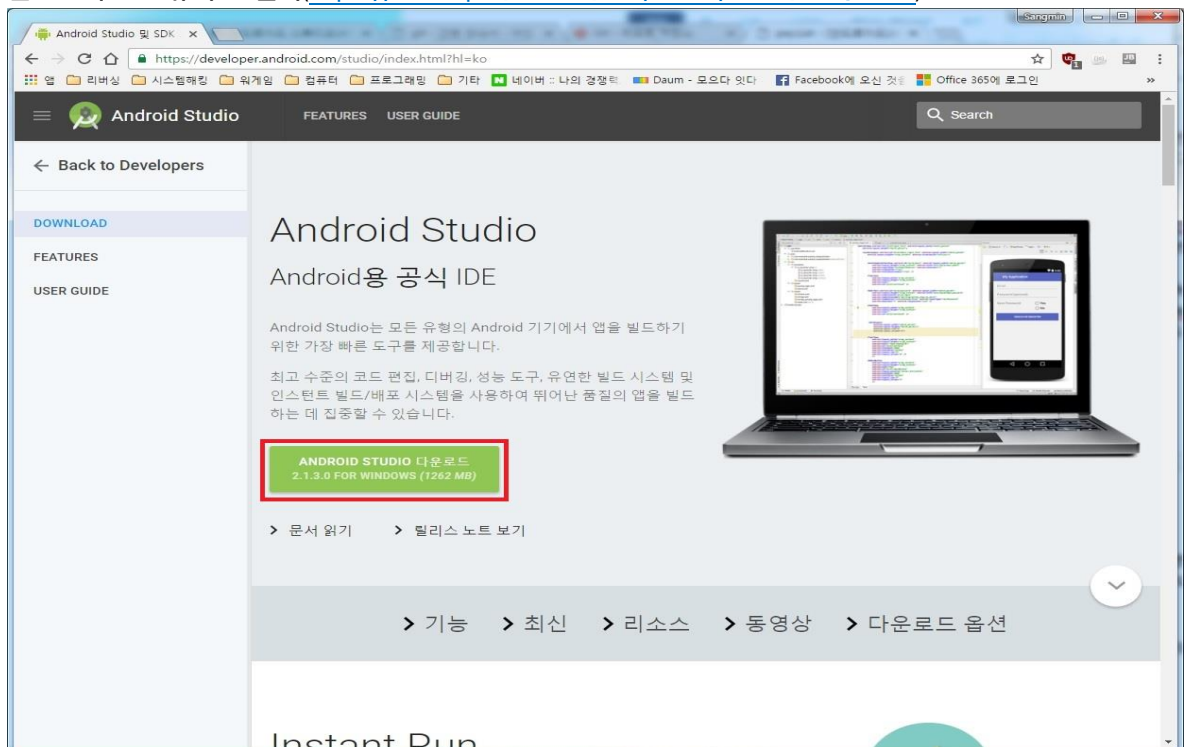
* 언어별 버전 확인 방법

- Windows : 내 컴퓨터 → 마우스 오른쪽 버튼 클릭 → 속성 → Windows 버전
- Linux : 리눅스 버전 확인(터미널 창 `cat /etc/issue`), 비트 확인(터미널 창 `getconf LONG_BIT` or `arch`)
- Mac : 위 사과모양 → 첫 번째 탭 → 이 Mac 정보 클릭

다운로드 받은 파일을 실행시킨 뒤, next 버튼을 누르면 설치가 진행된다.

자세한 설치 및 환경변수 설정 방법이 나와있는 링크를 첨부한다. <http://t2t2t.tistory.com/9>

- 2) 안드로이드 스튜디오 설치(<https://developer.android.com/studio/index.html?hl=ko>)



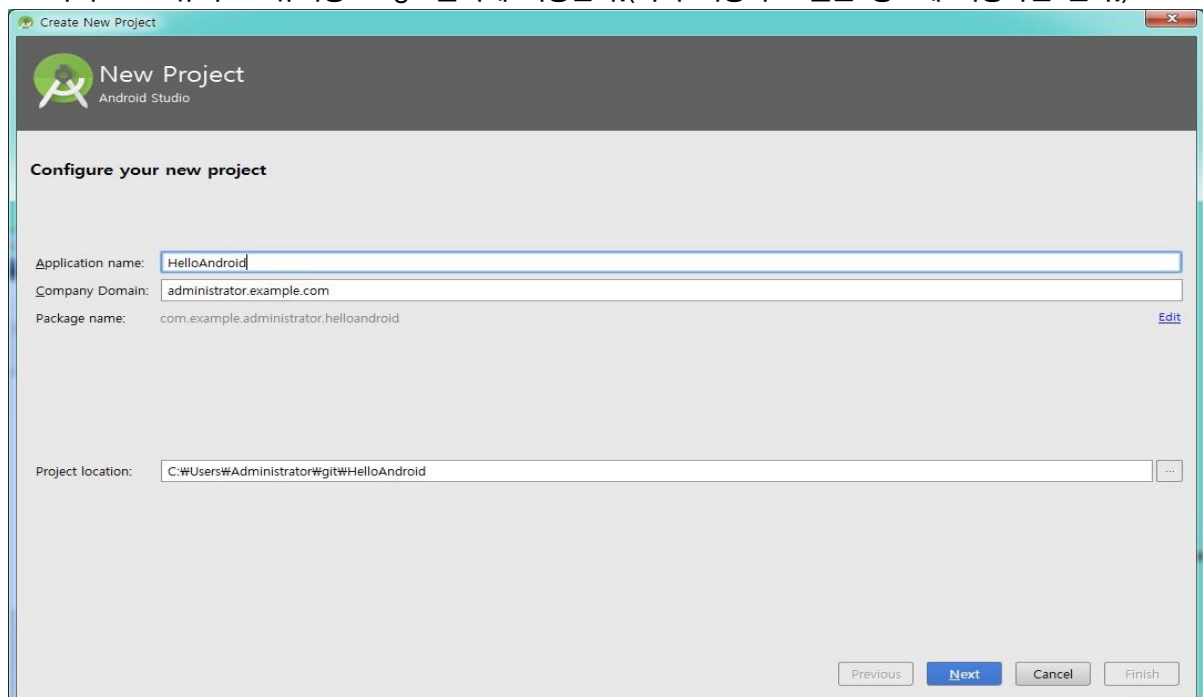
안드로이드 스튜디오 공식사이트에서 다운로드 받는다. 그 후, 다운로드 받은 파일을 실행시킨 뒤 설치 시 특별히 설정한 부분이 없어 next 버튼을 눌러 설치를 진행한다.
자세한 설치 방법이 나와있는 사이트 링크를 첨부한다. <http://withcoding.com/52>

다. 프로젝트 생성

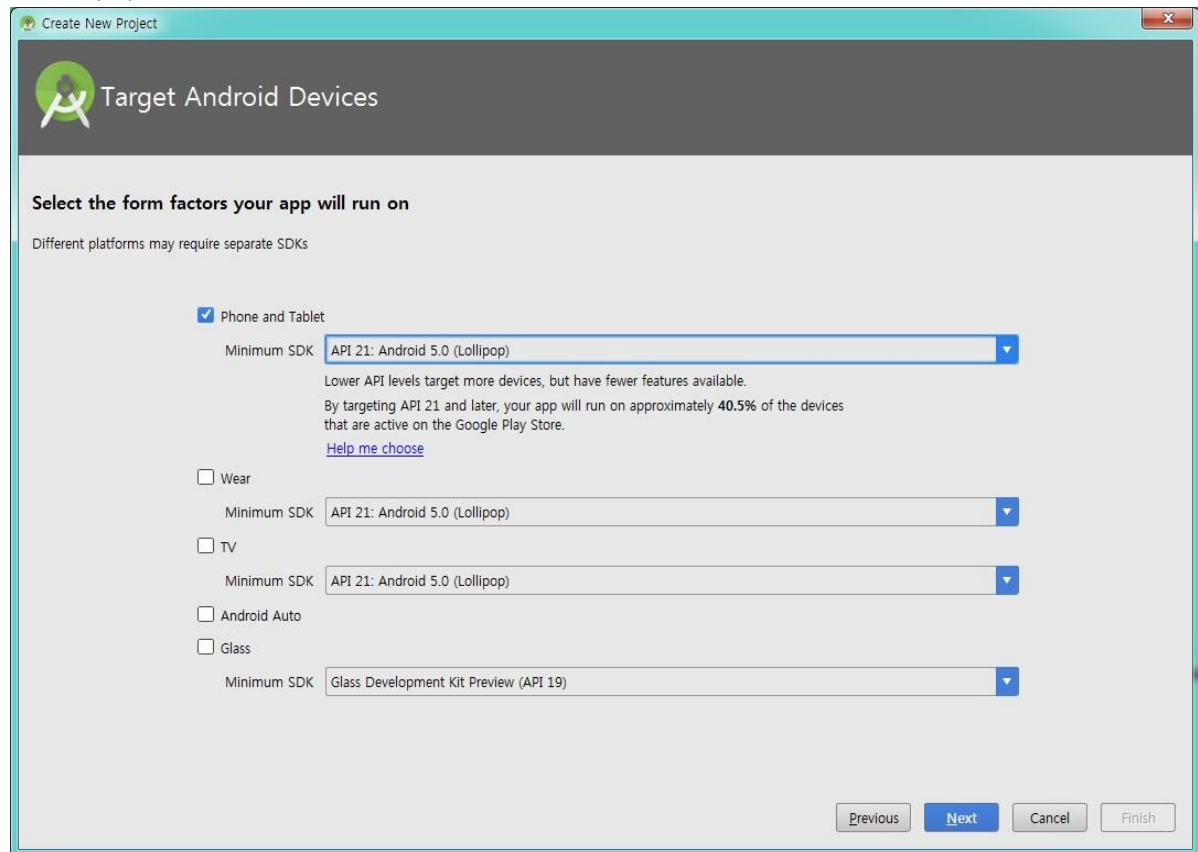
빨간색 박스 표시가 있는 곳을 선택하여 프로젝트를 생성한다.



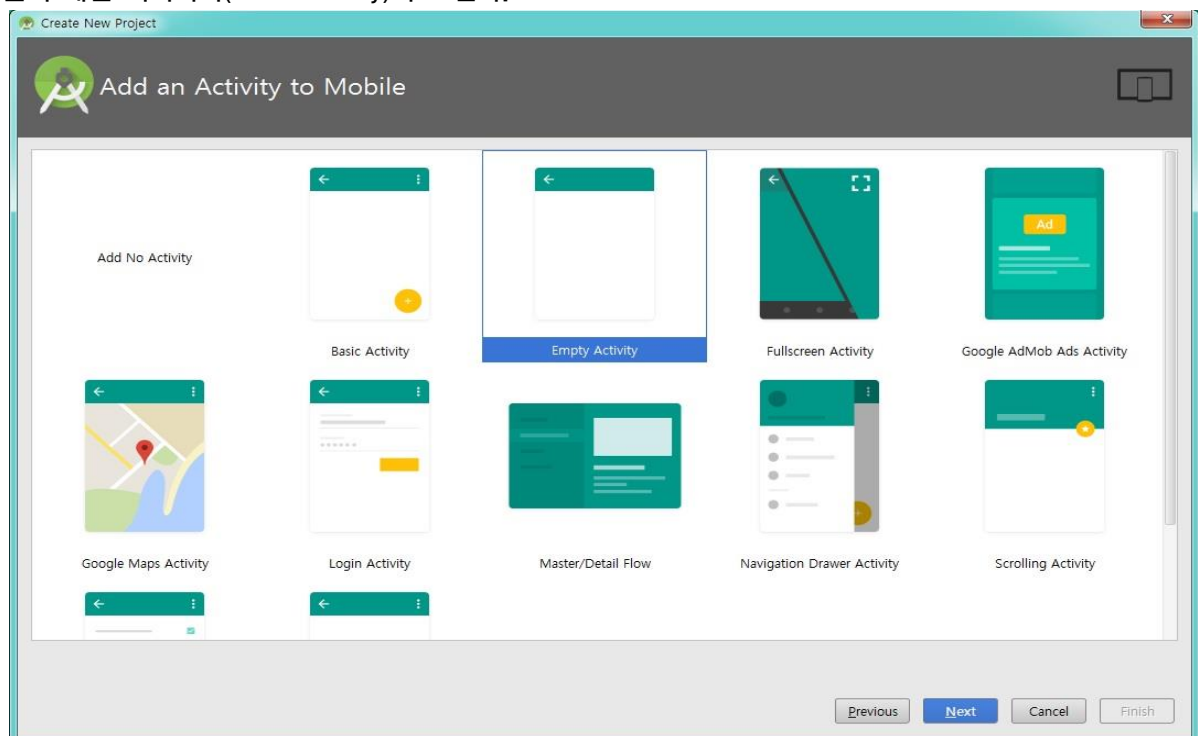
프로젝트명은 Application name과 동일하게 설정된다. 우린 HelloAndroid라고 작성한다.
Company Domain 부분은 회사의 도메인을 작성하는 부분인데, 이는 나중에 역순으로 패키지명이 된다. 아무거나 적어도 상관없다(test.com 이런 것도 가능), 우린 기본값인 계정명.example.com으로 둔다.
마지막으로 프로젝트 위치를 설정해야 하는데, Git을 사용해 소스 코드 형상 관리를 할 것이므로, C드라이브 → 유저 → 유저명 → git 폴더에 저장한다.(각자 저장하고 싶은 경로에 저장하면 된다.)



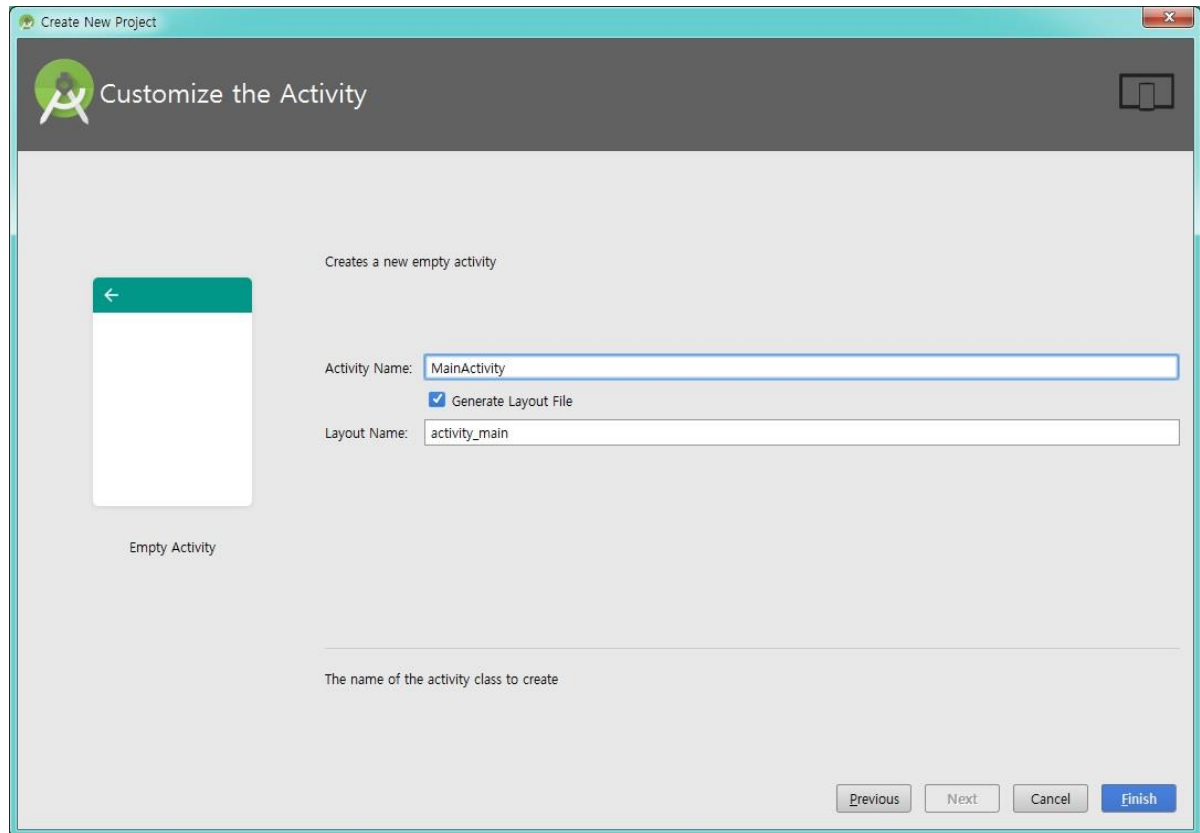
웨어러블, IoT시대로 접어들면서, 다양한 곳에 안드로이드가 적용되고 있다. 우리는 모바일 안드로이드 앱 리케이션을 제작하고 있으므로, Phone and Tablet을 선택한다. 또한, 최소 버전으로 API 21: Android 5.0(Lollipop)을 설정하였다.



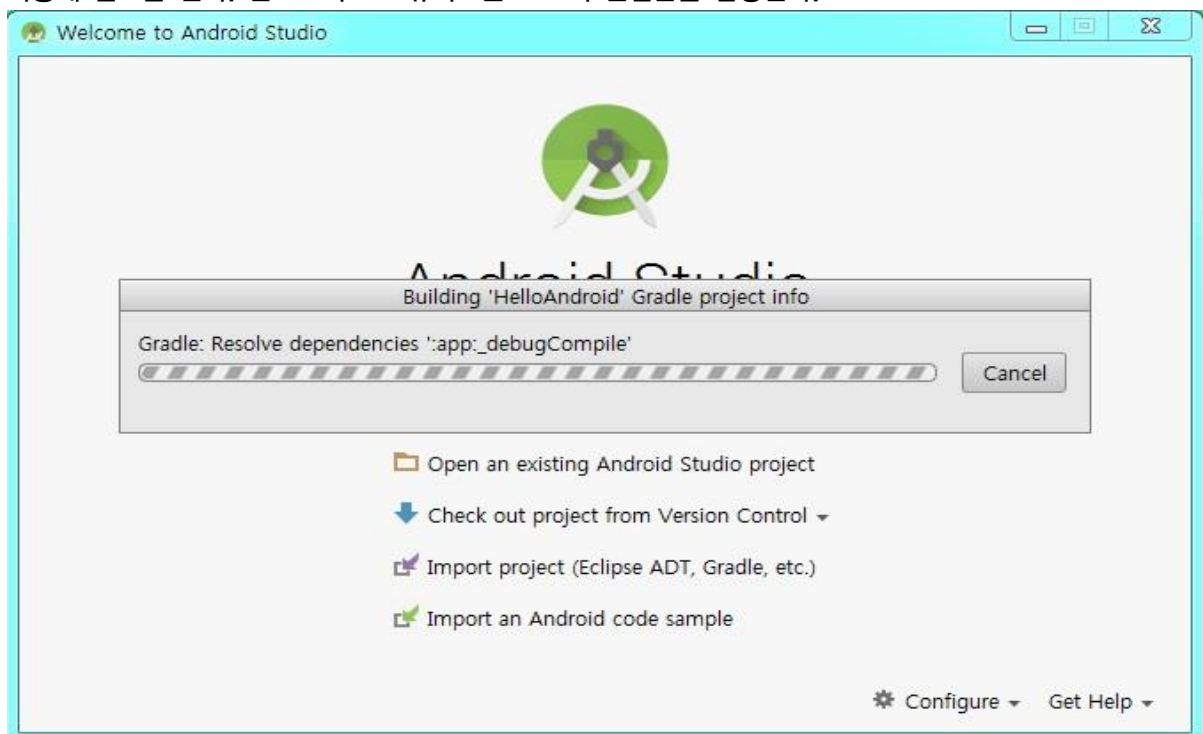
액티비티 선택 화면에선 우선 기본적으로 선택되어 있는 Empty Activity를 선택한다. 안드로이드 시스템에서 액티비티 컴포넌트는 사용자와 애플리케이션 간의 상호작용을 할 수 있도록 다이얼이나 웹브라우저 같은 가시적인 화면을 제공하는 역할을 한다. 애플리케이션이 실행되면 가장 먼저 실행되는 액티비티를 흔히 메인 액티비티(Main Activity)라고 한다.



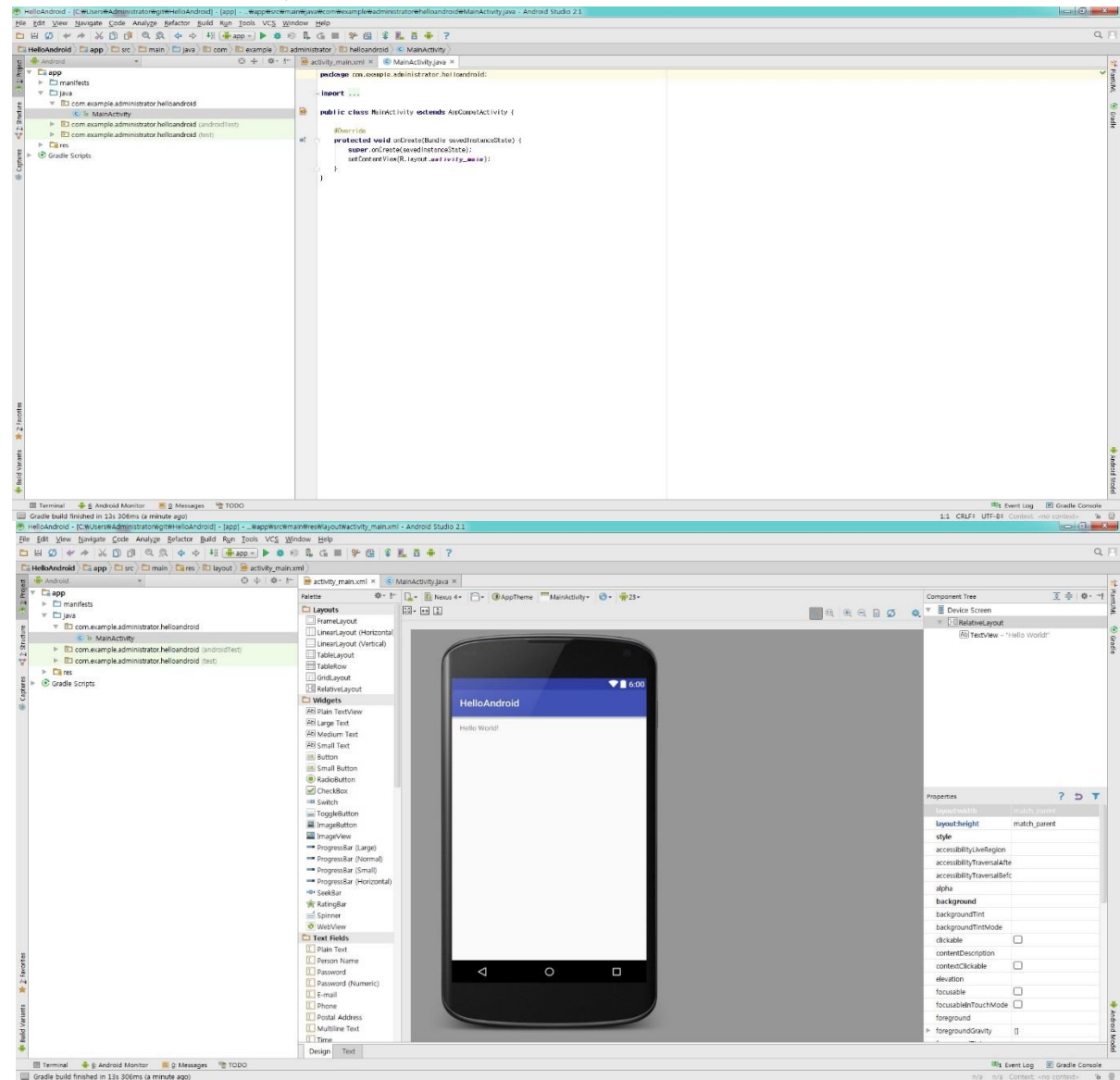
액티비티 컴포넌트 중에서 가장 먼저 실행되는 액티비티 컴포넌트를 흔히 메인 액티비티라고 부르고, 그에 해당하는 Layout을 액티비티 메인(activity_main)이라고 부른다. 두 개 모두 이름을 바꿔도 상관없으나, 유지보수 측면을 고려해 기본값으로 설정하는게 좋다. 우리는 기본값을 사용하므로 finish 버튼을 눌러 프로젝트를 생성한다.



프로젝트가 생성될 때 Gradle이라는 빌드도구를 사용하는 것을 볼 수 있다. Gradle에 대해 간단하게 말하면, "빌드 자동화 도구" 이다. 안드로이드 스튜디오(IDE)와 빌드 시스템은 서로 독립적이므로 Gradle을 사용해 빌드를 한다. 안드로이드 스튜디오는 코드의 편집만을 담당한다.

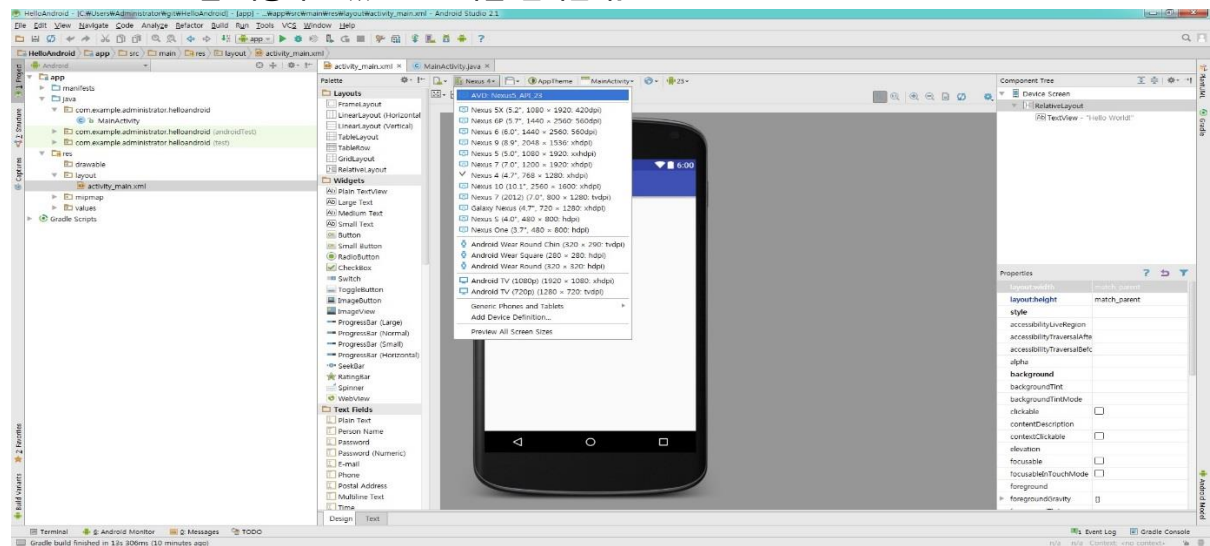


처음 프로젝트를 생성하면, 아래 그림과 같이 소스 코드와 레이아웃 화면을 볼 수 있다.

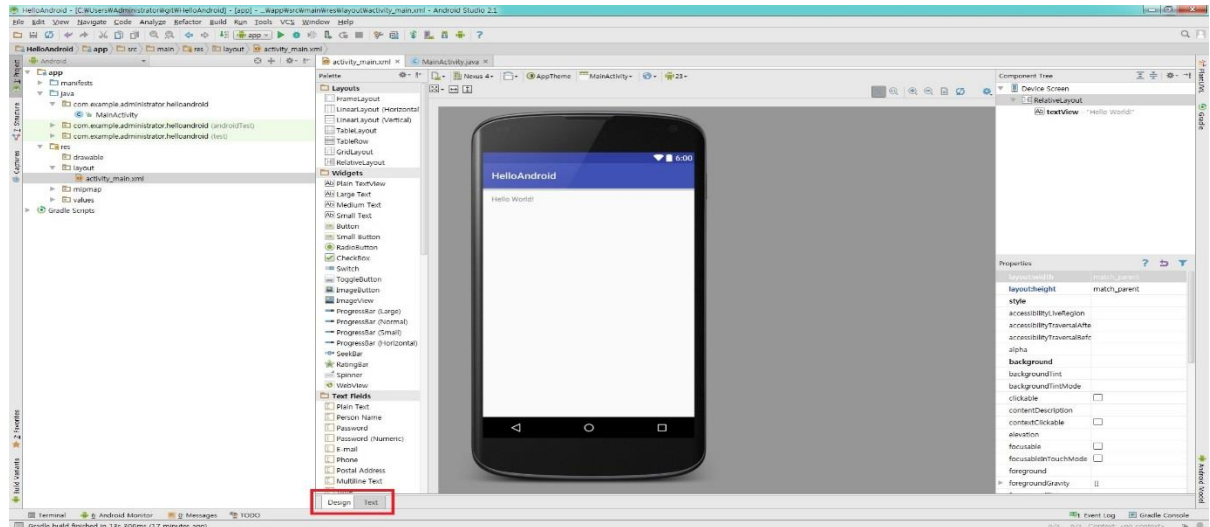


라. 애플리케이션 개발

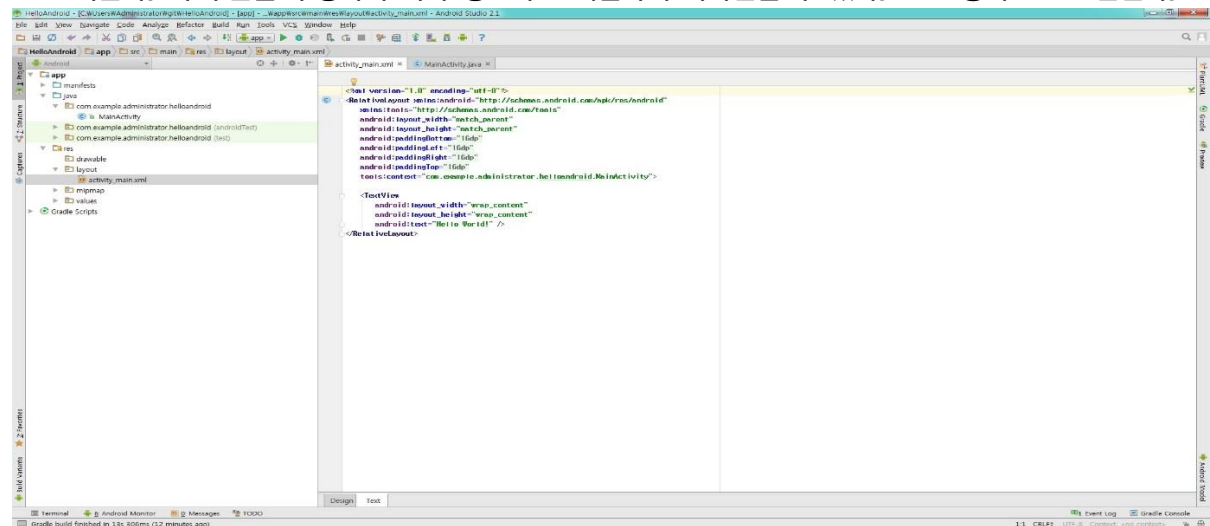
애플리케이션을 개발하기 앞서 기기를 선택해줘야 한다. 기본적으로 Nexus 4가 선택되어 있다. 우리는 AVD로 Nexus 5를 사용하고 있으므로 이걸 선택한다.



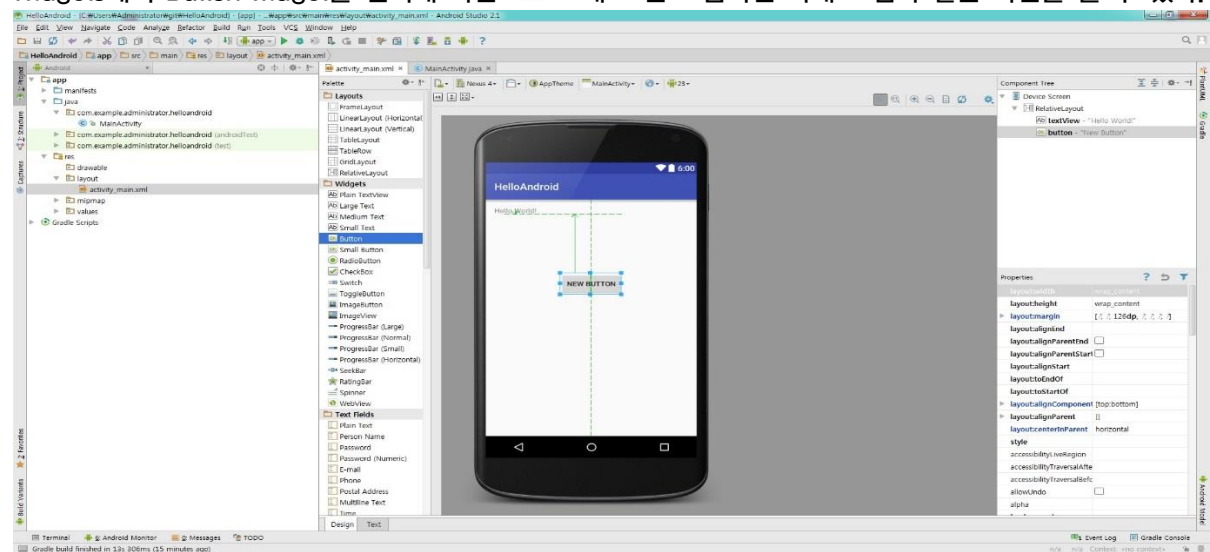
레이아웃 창에는 두 가지 모드를 지원한다. 첫 번째는 Design 모드이다. 여기서 Widgets, Text Fields 등 필요한 기능을 마우스로 화면에 드래그 앤 드롭하여 레이아웃을 디자인 할 수 있다.

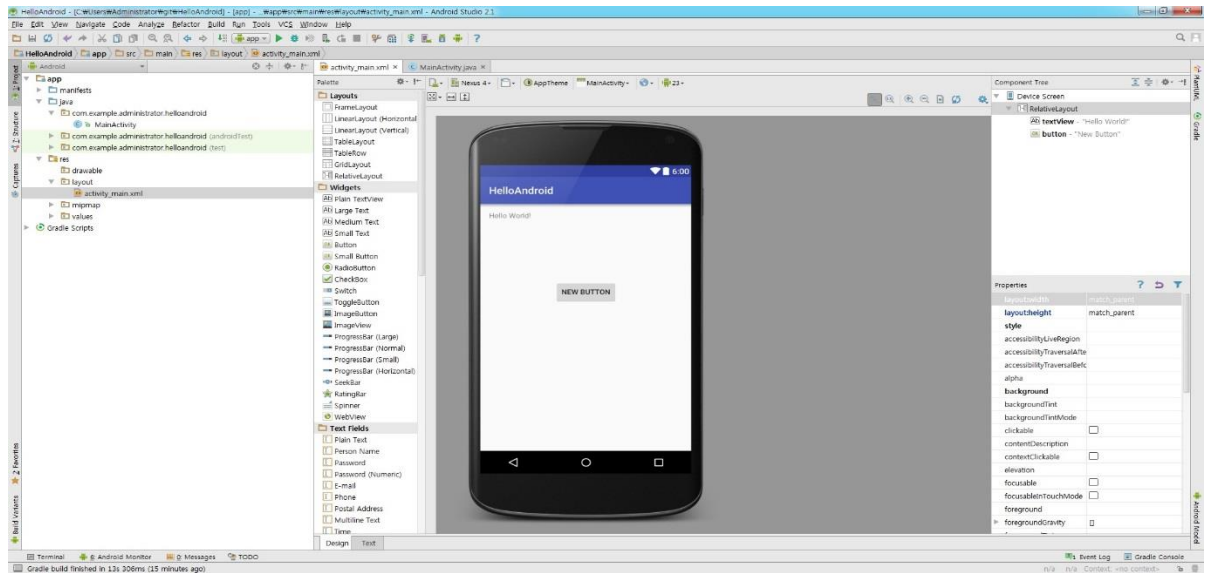


두 번째는 Text 모드이다. Text 모드에는 Design에서 꾸민 Widgets, Text Fields 같은 것들이 소스 코드로 보여준다. 이 부분을 수정하여 더욱 정교하고 세밀하게 디자인할 수 있다. XML 형식으로 표현된다.

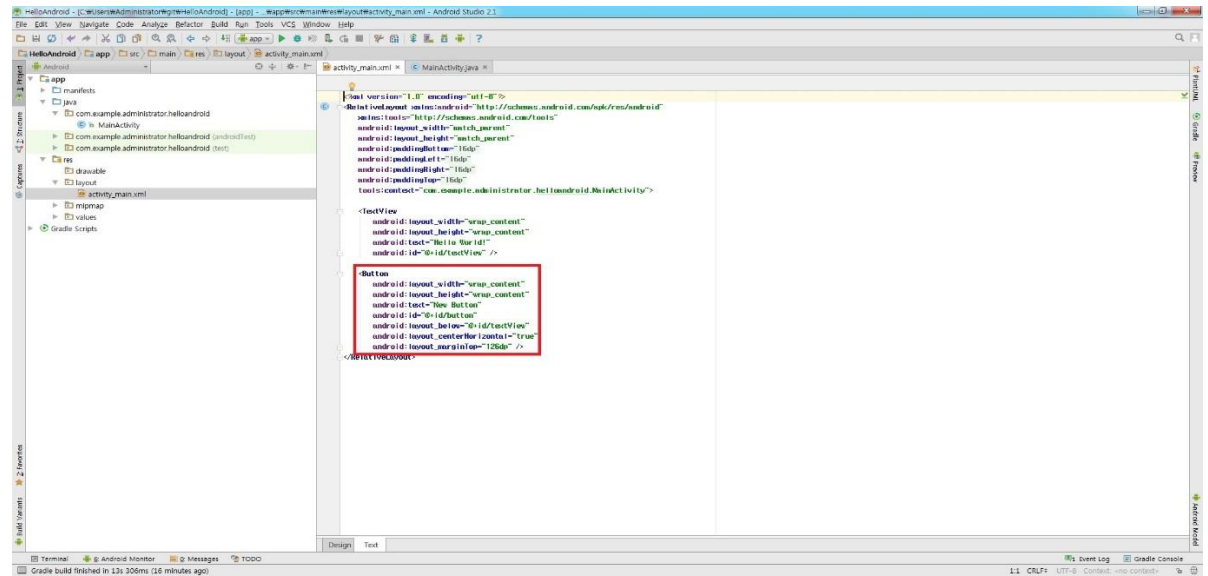


Widgets에서 Button widget을 선택해 화면으로 드래그 앤 드롭하면 아래 그림과 같은 화면을 볼 수 있다.

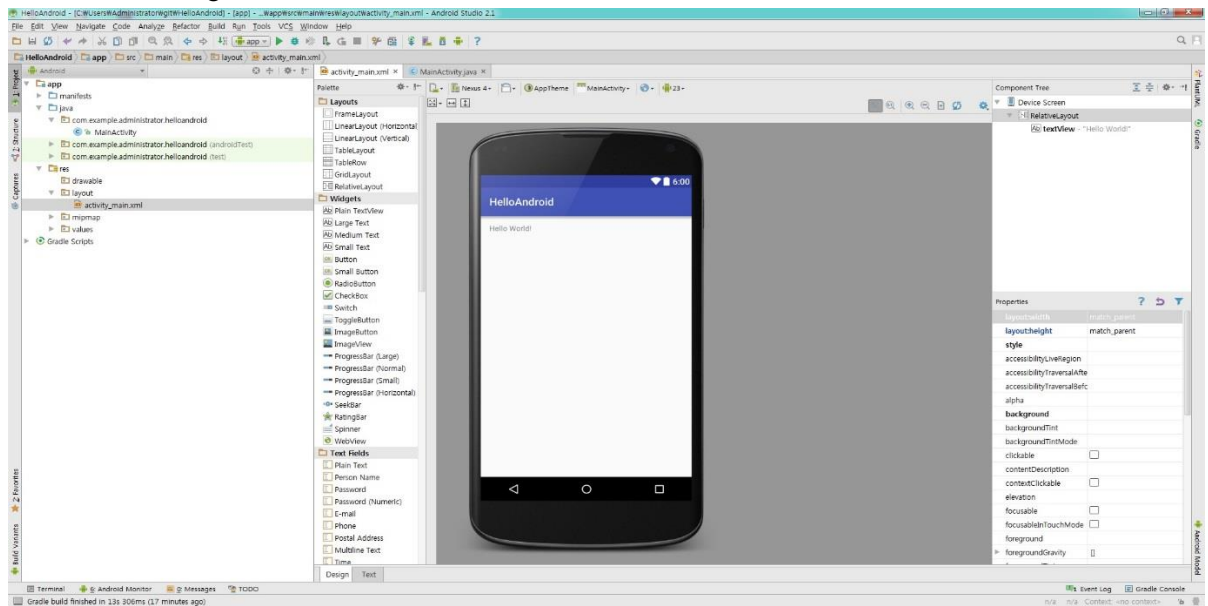




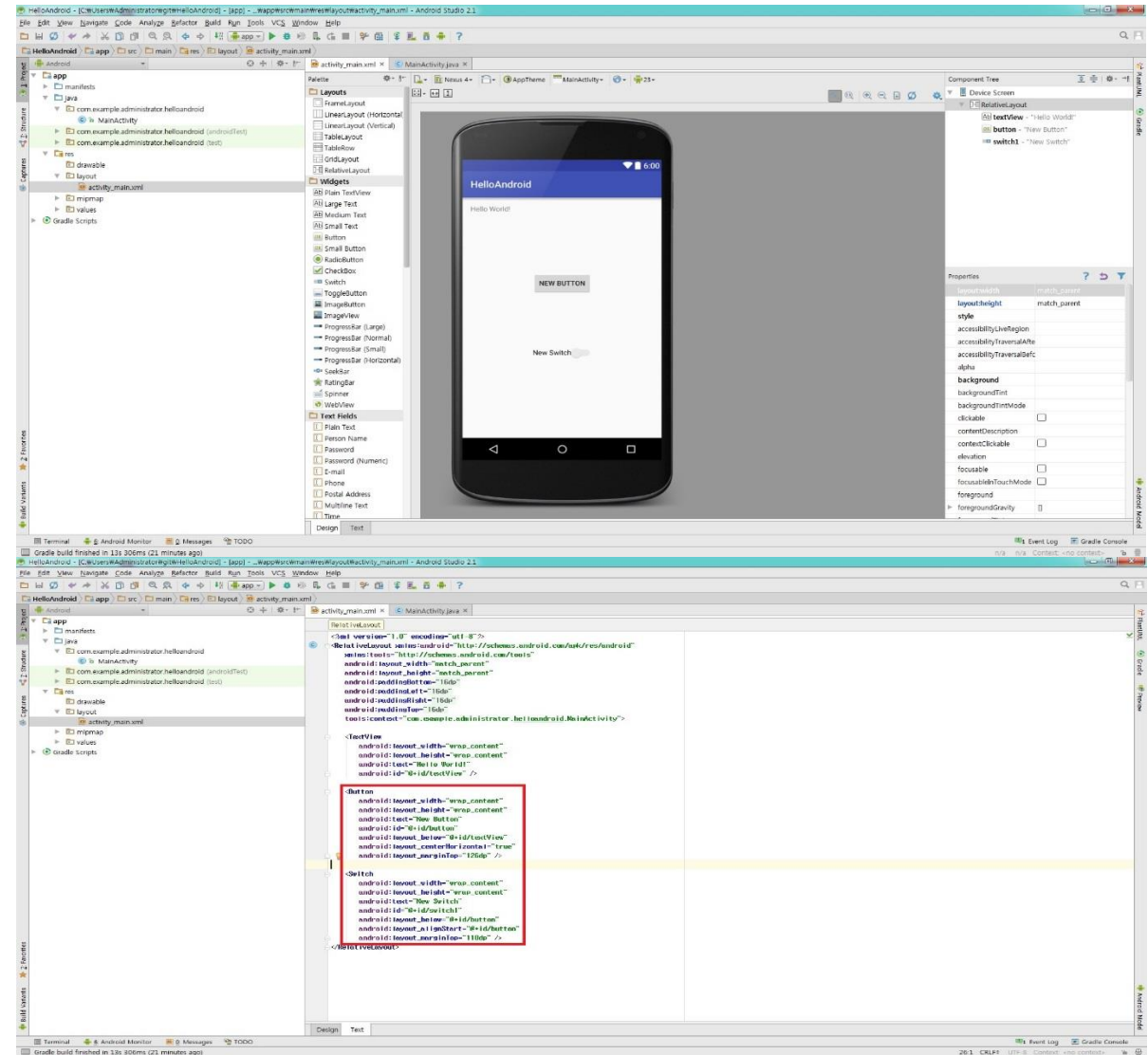
Design 모드에서 추가한 버튼을 Text 모드에서 확인해 볼 수 있다. 그 후, 이 부분을 삭제하면,



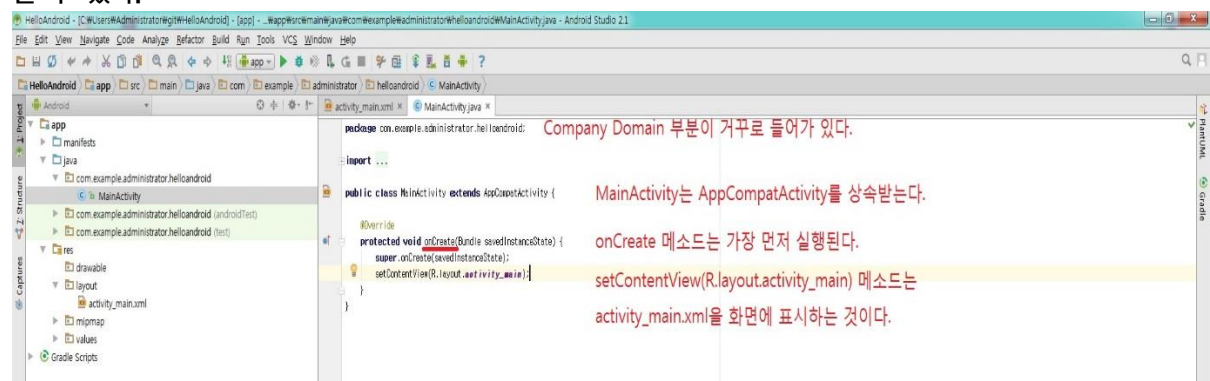
아래와 같이 Design 모드에서 버튼이 사라진 것을 볼 수 있다.



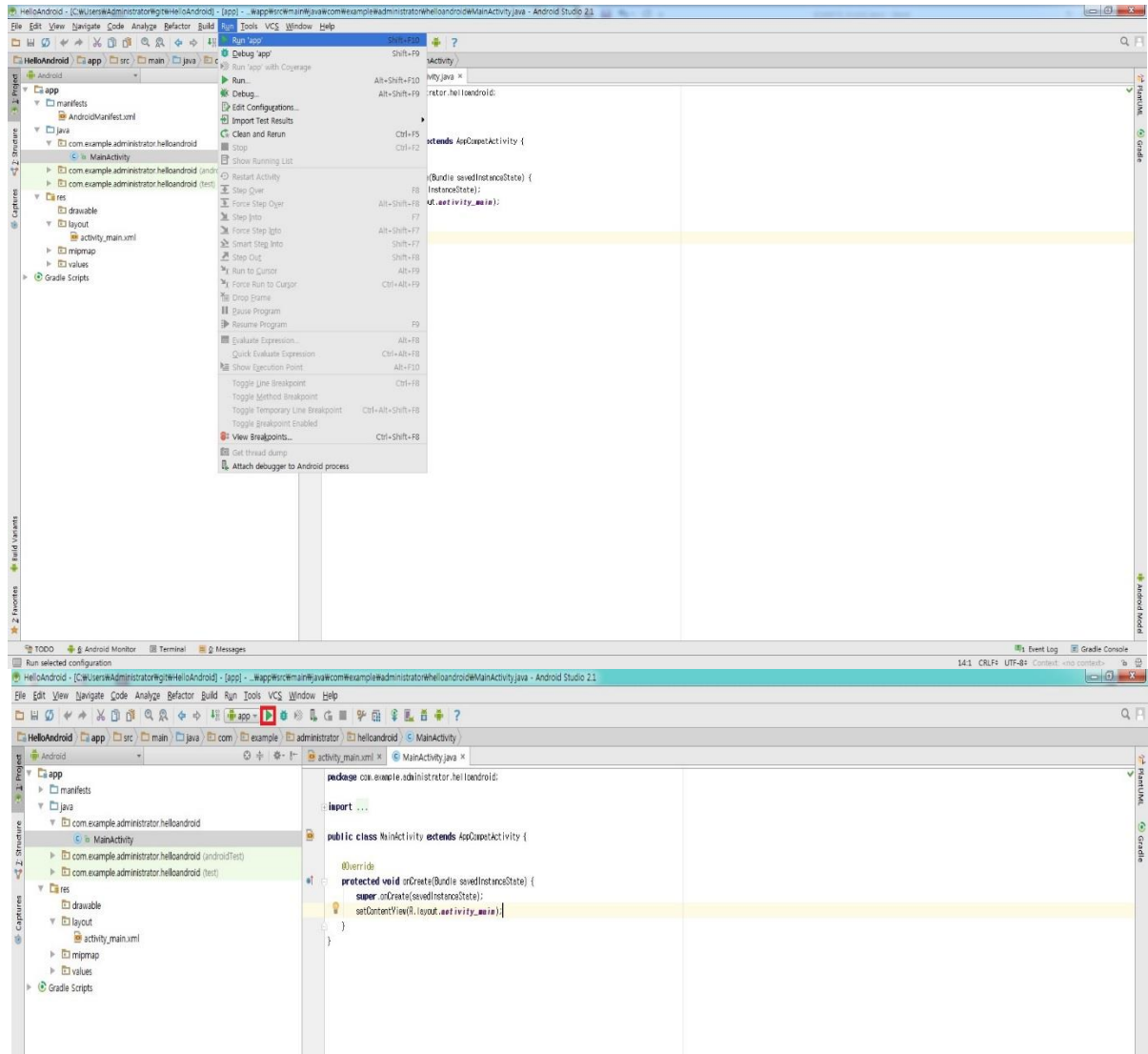
그렇다면, 버튼과 스위치 Widget을 드래그 앤 드롭 해보자. 그 후, 소스 코드를 확인해보면, 아래와 같이 추가된 모습을 확인할 수 있다.



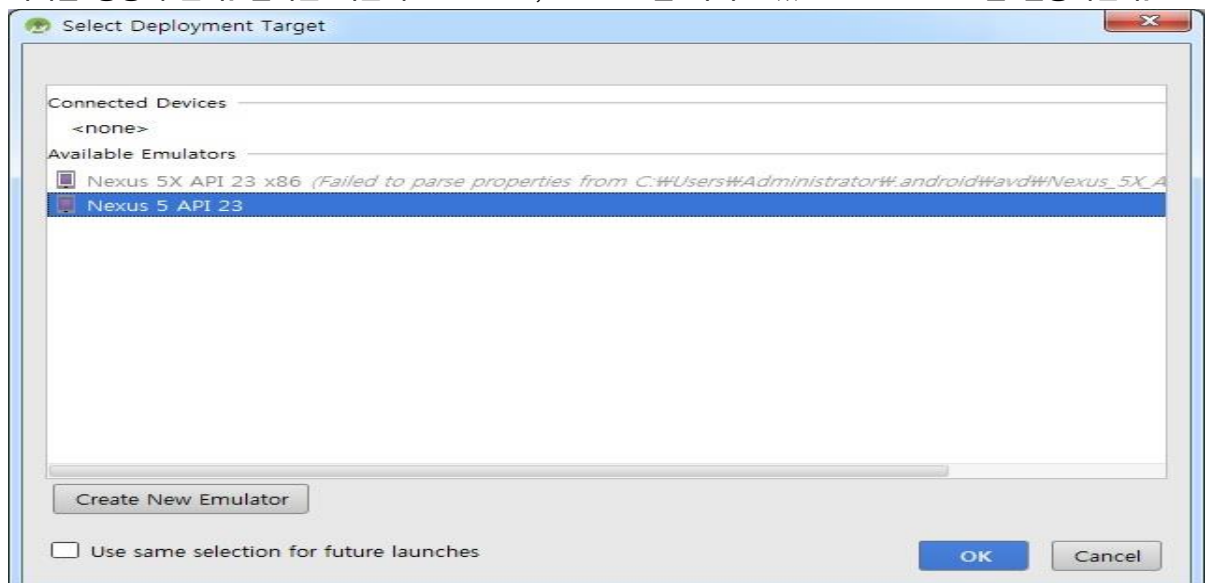
Java → com.example.administrator.helloandroid → MainActivity 에 들어가 보면, 애플리케이션에 관한 소스 코드를 볼 수 있다. 첫 번째 줄 부분의 패키지에서 패키지는 간단히 말하면, 관련된 클래스 또는 인터페이스들을 한 곳에 모아놓은 꾸러미 같은 것이다. Import 부분은 구현에 관련된 API 호출 시 선언해 줘야 하는 부분이다. MainActivity는 AppCompatActivity를 상속받으며, onCreate() 메소드를 갖는다. onCreate() 메소드는 가장 먼저 실행되며, 내부적으로 setContentView() 메소드를 사용해 activity_main.xml을 화면에 표시한다. 이러한 메소드는 안드로이드 공식 문서를 참고하면 더욱 자세하게 알 수 있다.



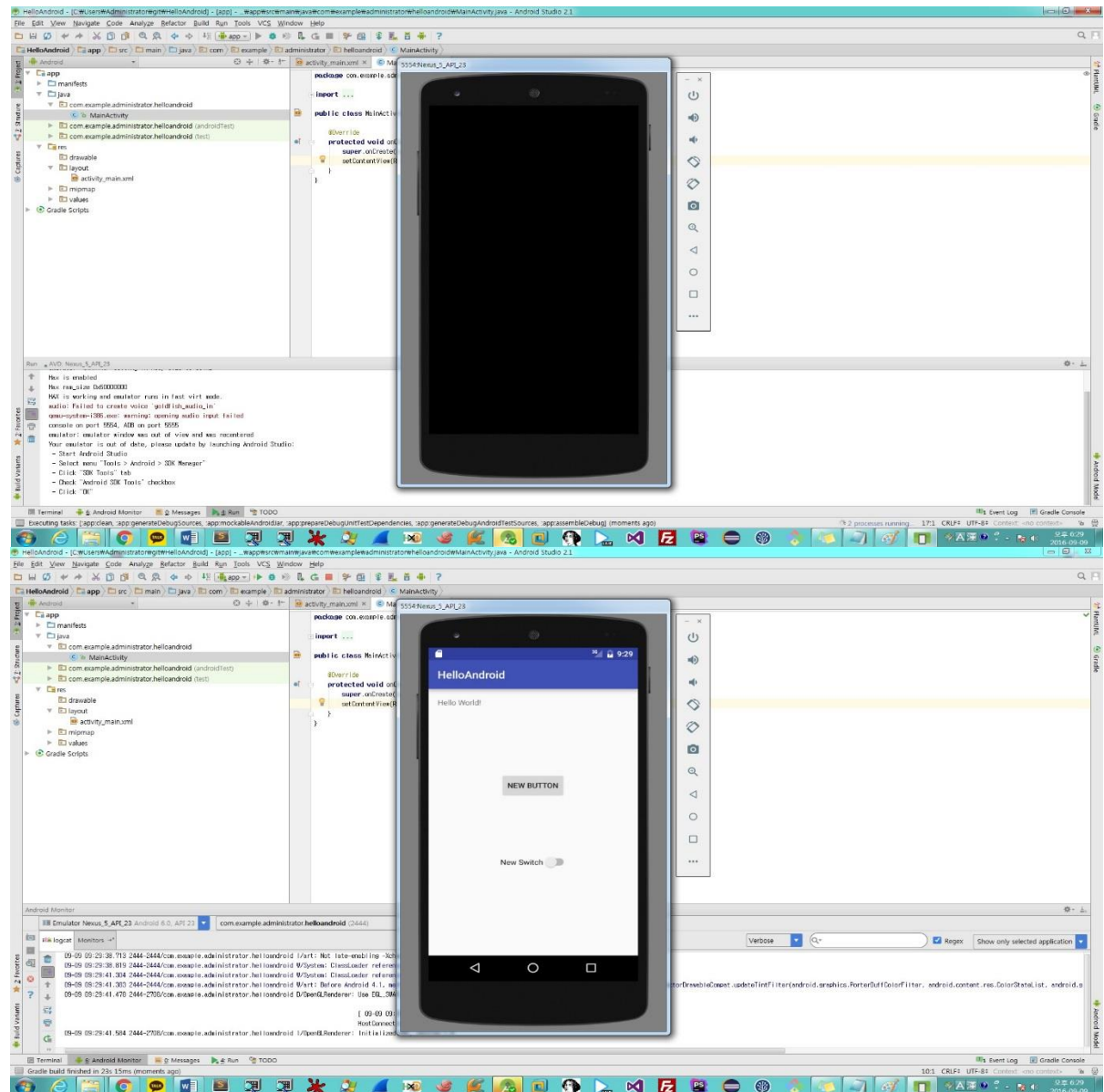
그럼, 애플리케이션을 실행시켜보자. 상단 탭 Run → Run app, 화살표 또는 단축키 Shift + F10을 통해 AVD를 선택하고 애플리케이션을 실행한다.



처음 실행을 할 경우, 사용 가능한 에뮬레이터가 없기 때문에 Create New Emulator 버튼을 눌러 에뮬레이터를 생성해 준다. 필자는 기존에 Nexus 5X, Nexus 5를 가지고 있으므로 Nexus 5를 실행시킨다.



아래 그림과 같이 첫 번째 애플리케이션이 실행되는 것을 확인할 수 있다.



4. 결과 및 분석

데모 영상 : <https://www.youtube.com/watch?v=C6U2kFHBr4g>



맨 왼쪽 첫 번째 사진은 우리가 제작한 애플리케이션을 실행한 모습이다. activity_main.xml에서 본 모습과 동일하다. 가운데 사진은 안드로이드 에뮬레이터 배경화면 모습이다. 맨 오른쪽 첫 번째 사진은 안드로이드에 설치된 애플리케이션 목록을 나타낸 사진이다. 그 중에 우리가 제작한 애플리케이션 모습이 보인다.

이번 과제는 모바일 안드로이드 프로그래밍이 어떻게 이루어지는지 체험하는데 의의가 있다. 이에 따라, 특별한 기능들을(로그인 기능 등) 구현하지 않는다. 단순히 버튼, 스위치 widgets을 선택해 드래그 앤 드롭을 해 봄으로써 안드로이드 애플리케이션의 디자인을 꾸미는 방식에 대해서 학습을 하였다. 게다가, Design, Text 모드를 통해 디자인을 정밀하고 세세하게 수정할 수 있음을 알았다. 더 나아가 버튼의 내용을 바꾸고, 스위치의 모양을 바꾸는 등 조금 더 아름답게 만드는 과정을 학습하면 좋을 것 같다.

5. 참고자료

예제 : Android Studio를 활용한 안드로이드 프로그래밍(우재남, 박길식 지음) p56.

핸드폰에 APK 설치하는 방법 : <http://ggari.tistory.com/186>

Mac 버전 확인 : <http://support.hp.com/kr-ko/document/c03330670>

안드로이드 액티비티 : http://www.tipssoft.com/bulletin/board.php?bo_table=FAQ&wr_id=944

Gradle 관련 내용

- 블로그 : <http://uroa.tistory.com/64>

- Gradle github : <https://github.com/gradle/gradle>

- 위키피디아 : <https://en.wikipedia.org/wiki/Gradle>

안드로이드 API 공식 문서에서 setContentView 부분

[https://developer.android.com/reference/android/app/Activity.html#setContentView\(android.view.View\)](https://developer.android.com/reference/android/app/Activity.html#setContentView(android.view.View))