

오픈소스 전문 프로젝트 보고서

-Hello Android-

<8조>

담당 교수님 : 박수창 교수님

담당 조교님 : 김소연 조교님

제 출 일 : 2020년 3월 28일

학 과 : 컴퓨터 공학과

조 원 : 2018037021 강주형

2016038003 이찬진

2016038019 정주형

2016038022 최준호

2016038035 채범근

목 차

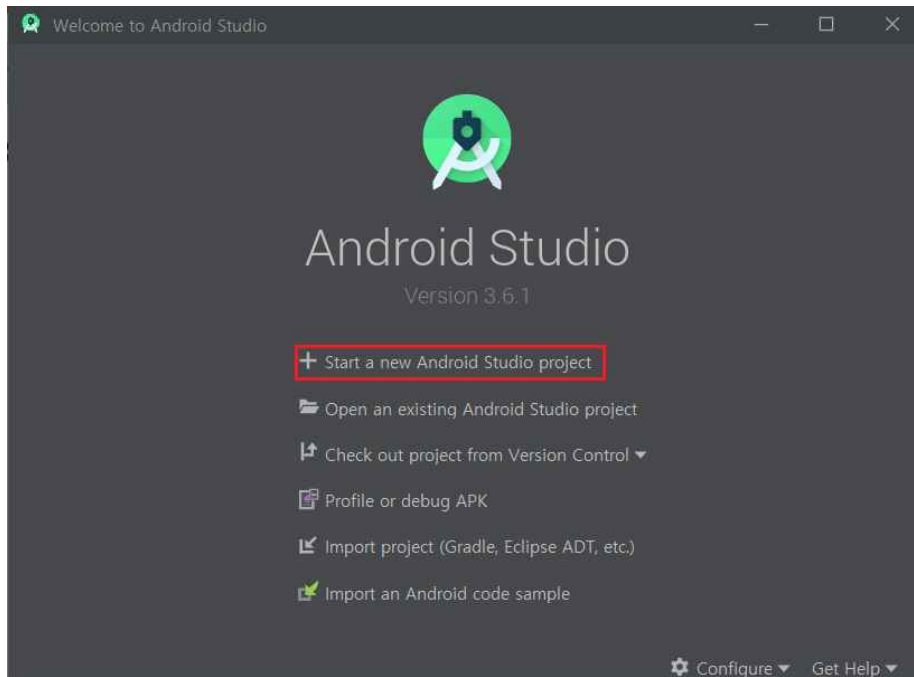
I. 프로젝트 생성	3
II. 화면 디자인 및 편집	5
III. 프로젝트 실행, 확인	8
IV. 직접 풀어보기 2-1	9

I. 프로젝트 생성

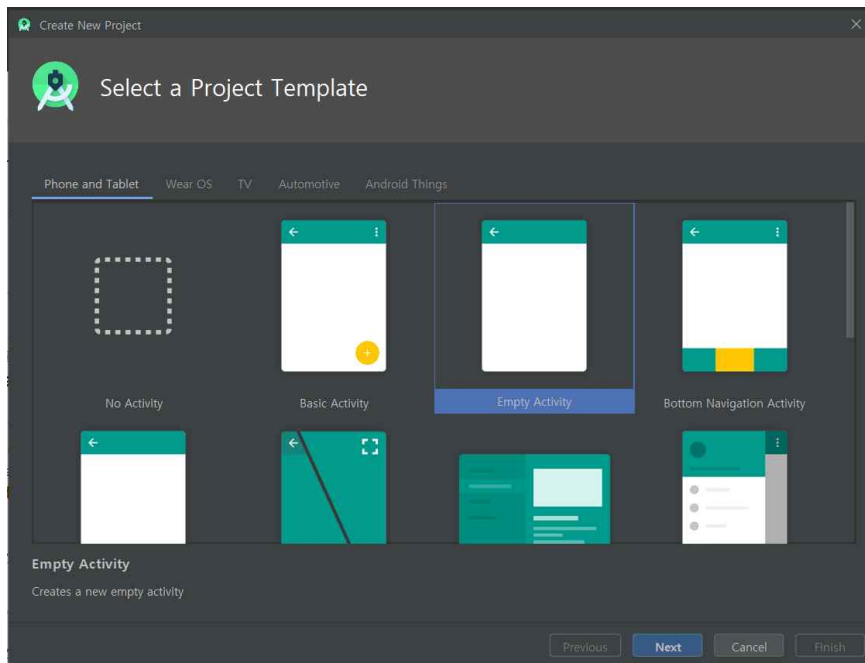
이 섹션은 강주형 조원이 담당하였다.

※ 버전의 차이로 주교재와 차이가 있을 수 있습니다.

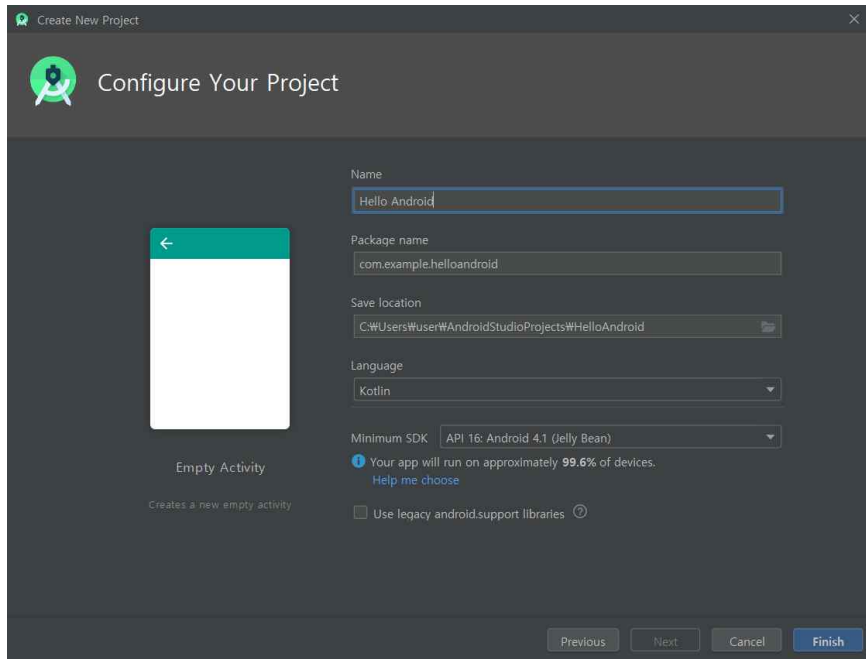
1. Android Studio를 실행한 후 [Start a new Android Studio project]를 클릭했다.



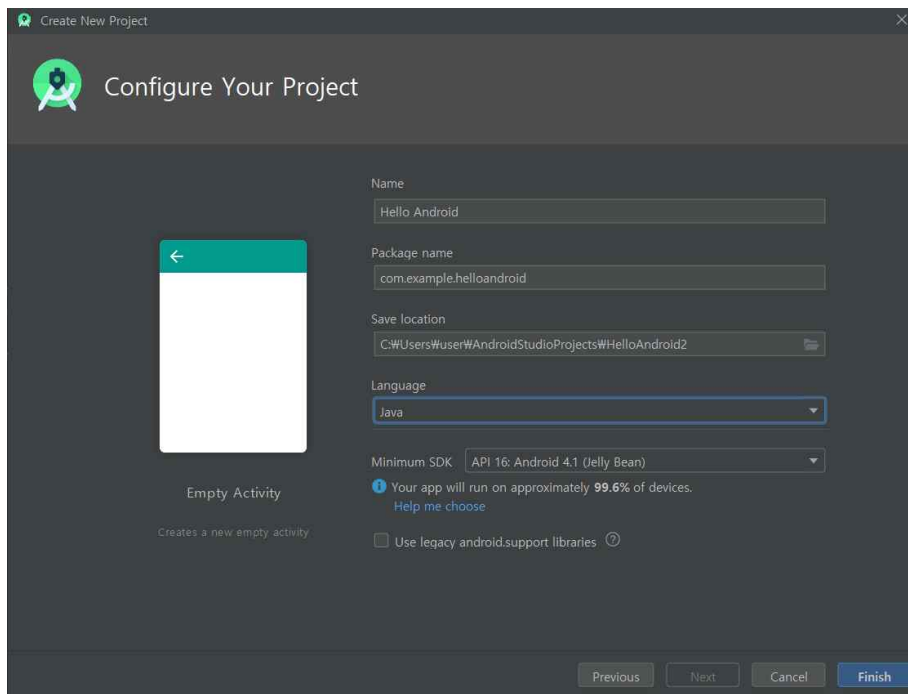
2. [Select a Project Template] 창에서 Empty Activity 항목을 선택했다.



3. [Configure Your Project]에서 프로젝트의 제목을 설정했다.



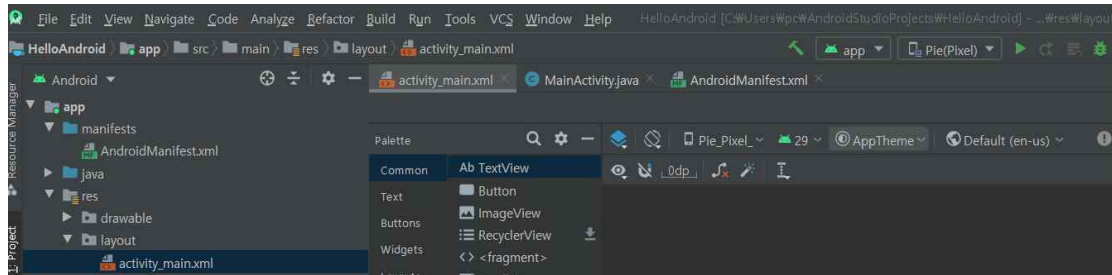
4. Language 항목을 JAVA로 바꿔준다고 Finish를 눌러 프로젝트 생성을 마친다.



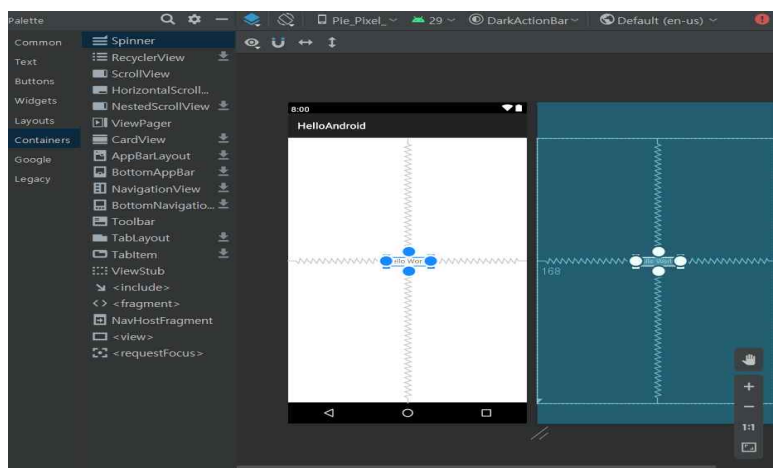
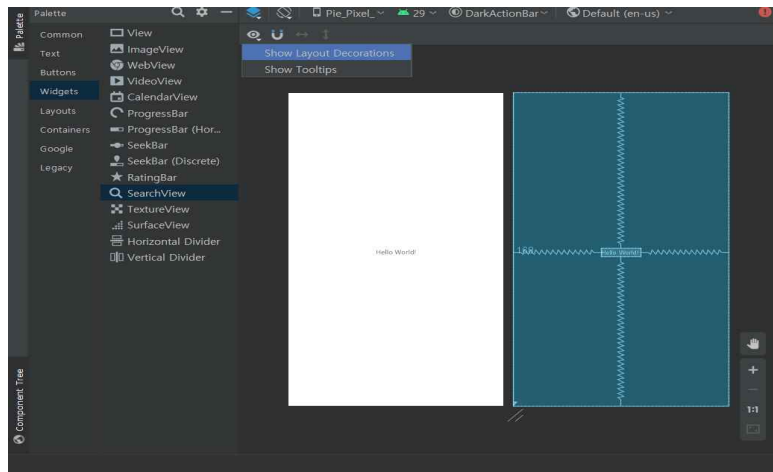
II. 화면 디자인 및 편집

이 섹션은 이찬진 조원, 정주형 조원이 담당하였다.

1. 화면을 가상 AVD와 동일하게 설정

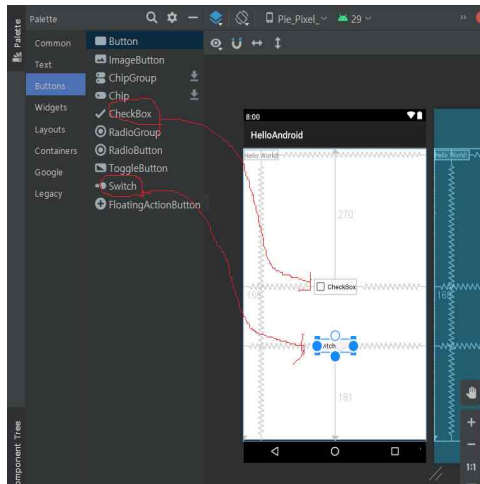


*참고 : activity_main.xml을 봤을 때 레이아웃이 보이지 않는 경우 아래 사진과 같이 show Layout Decorations를 클릭해준다.

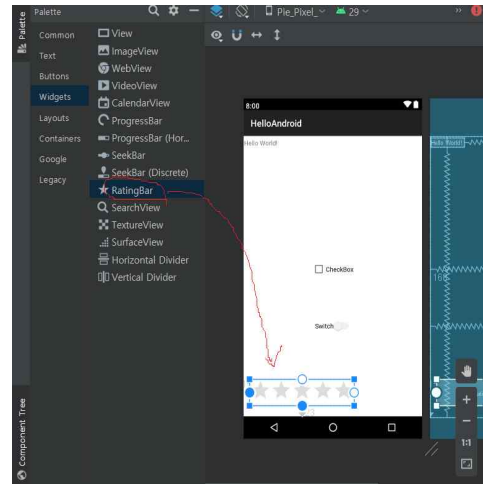


2. Buttons에서 CheckBox, Switch 그리고 Widgets에서 RatingBar를 핸드폰으로 끌어온다.

〈CheckBox, Switch〉



〈RatingBar〉



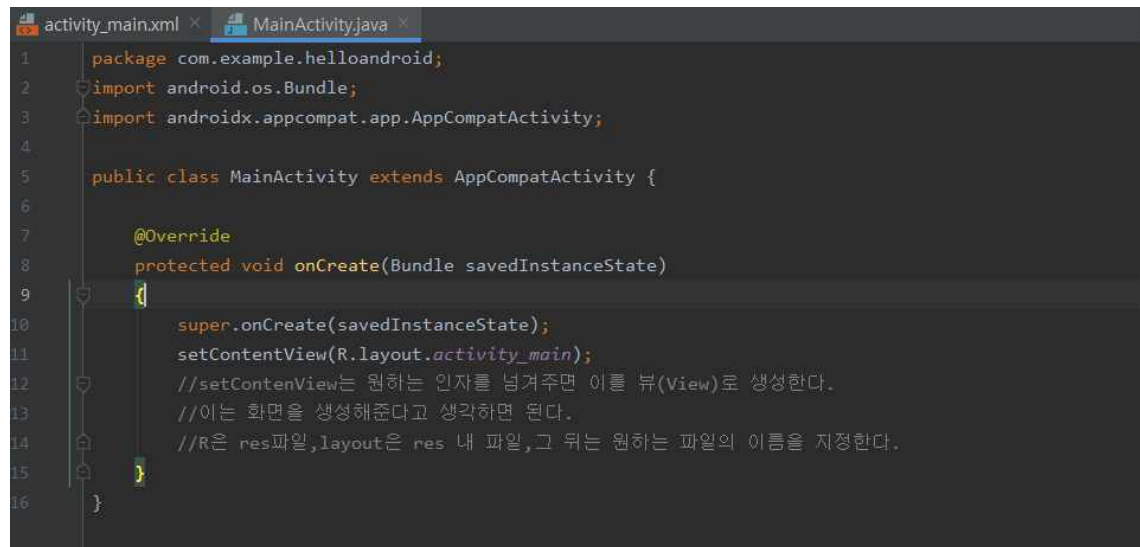
3. activity_main.xml을 Code를 보는 방식으로 변경 후 android.support.constraint.ConstraintLayout을 RelativeLayout으로 변경하고 위에 설치한 위젯들의 위치를 설정한다.

```

activity_main.xml MainActivity.java
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:layout_width="match_parent"
5      android:layout_height="match_parent"
6      tools:context=".MainActivity">
7
8      <!-- RelativeLayout의 '상대적인'이란 뜻으로, 다른 위젯의 상대적인 방향에 배치를 뜻함-->
9      <!-- 위젯을 설정하지 않으면 기본적으로 parent를 기준으로 한다-->
10     <!-- parent는 RelativeLayout을 의미하는데, 이는 화면 정도로 생각하면 된다-->
11     <!-- 앞으로 나올 Top, Bottom은 화면의 상단, 하단으로 이해하면 된다-->
12
13     <TextView
14         android:id="@+id/textView"
15         android:layout_width="wrap_content"
16         android:layout_height="wrap_content"
17         android:layout_centerInParent="true"
18         android:text="Hello World!"/>
19     <!-- 핵심인 문장이므로 centerInParent로 중심에 배치하게 함-->
20     <!-- id는 위젯의 이름 정도로 생각할 수 있다-->
21
22     <Switch
23         android:id="@+id/switch1"
24         android:layout_width="wrap_content"
25         android:layout_height="wrap_content"
26         android:layout_centerHorizontal="true"
27         android:layout_marginTop="288dp"
28         android:text="Switch"/>
29     <!-- centerHorizontal은 수평선의 중심에 둔다는 뜻-->
30     <!-- marginTop은 위에서 공백이니, 화면 위로부터 얼마나 거리를 줄지 의미-->
31
32     <RatingBar
33         android:id="@+id/ratingBar"
34         android:layout_width="wrap_content"
35         android:layout_height="wrap_content"
36         android:layout_centerHorizontal="true"
37         android:layout_alignParentBottom="true"
38         android:layout_marginBottom="70dp"/>
39     <!-- 아래에서부터 거리를 두고 싶다면 ParentBottom을 사용한다-->
40
41     <CheckBox
42         android:id="@+id/checkBox"
43         android:layout_width="wrap_content"
44         android:layout_height="wrap_content"
45         android:layout_alignTop="@+id/textView"
46         android:layout_centerHorizontal="true"
47         android:layout_marginTop="100dp"
48         android:text="CheckBox"/>
49     <!-- 굳이 상대적거리를 parent에서 풀 필요 없이, 다른 위젯으로부터 풀 수도 있다-->
50     <!-- CheckBox가 상단거리를 풀 대상을 textView로 두었다-->
51     <!-- 그 외 거리는 parent를 상대적 거리로 풀 때와 같다-->
52 </RelativeLayout>

```

4. java코드 확인을 위해 Project Tree의
java-com.cookandroid.helloandroid – MainActivity를 클릭하여 확인한다.

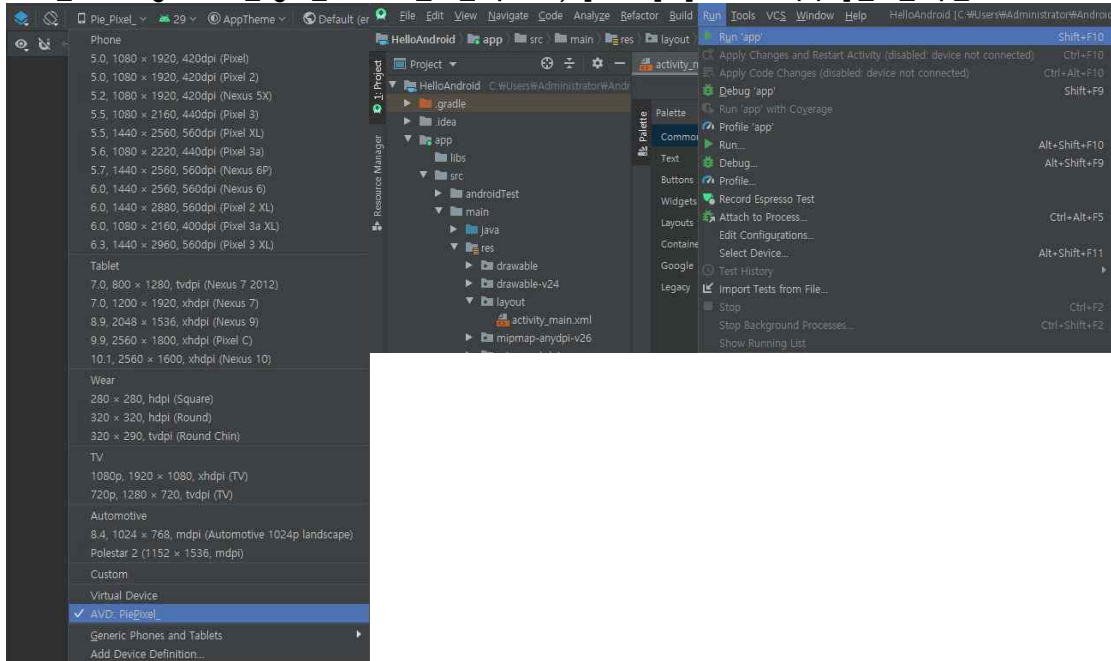


```
1 package com.example.helloandroid;
2 import android.os.Bundle;
3 import androidx.appcompat.app.AppCompatActivity;
4
5 public class MainActivity extends AppCompatActivity {
6
7     @Override
8     protected void onCreate(Bundle savedInstanceState)
9     {
10         super.onCreate(savedInstanceState);
11         setContentView(R.layout.activity_main);
12         //setContentView는 원하는 인자를 넘겨주면 이를 뷰(View)로 생성한다.
13         //이는 화면을 생성해준다고 생각하면 된다.
14         //R은 res파일,layout은 res 내 파일,그 뒤는 원하는 파일의 이름을 지정한다.
15     }
16 }
```

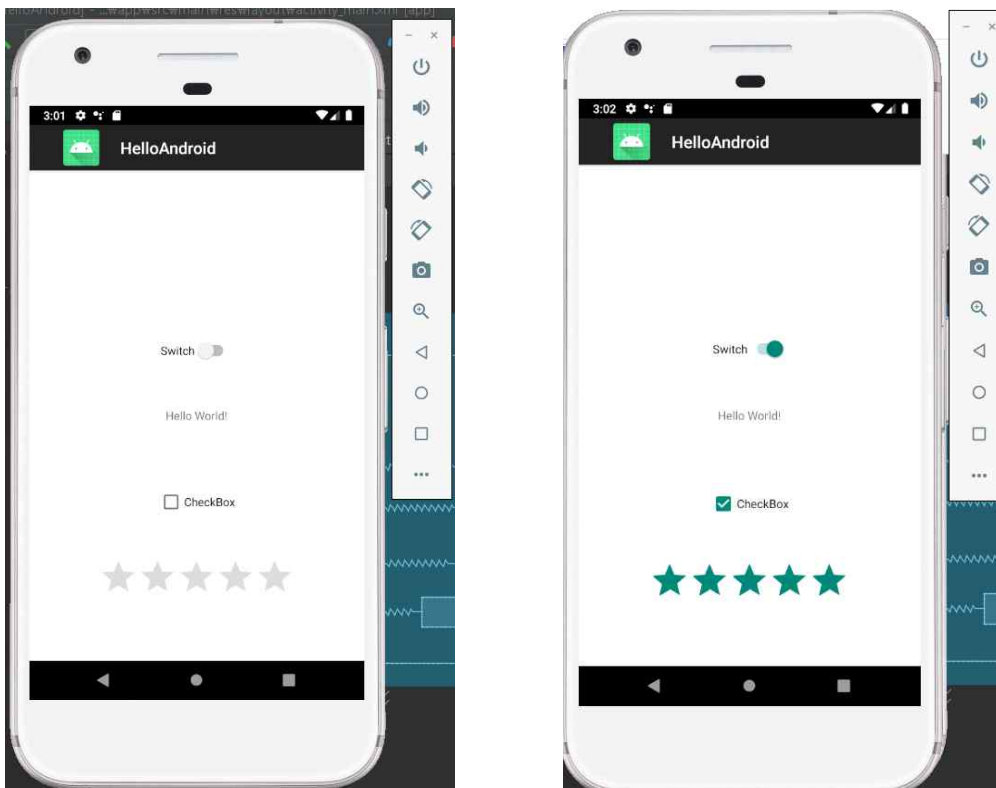
III. 프로젝트 실행, 확인

이 섹션은 채범근 조원이 담당하였다.

1. 팔레트 창에서 실행 할 AVD를 선택하고, [Run]-[Run 'app']을 클릭한다.



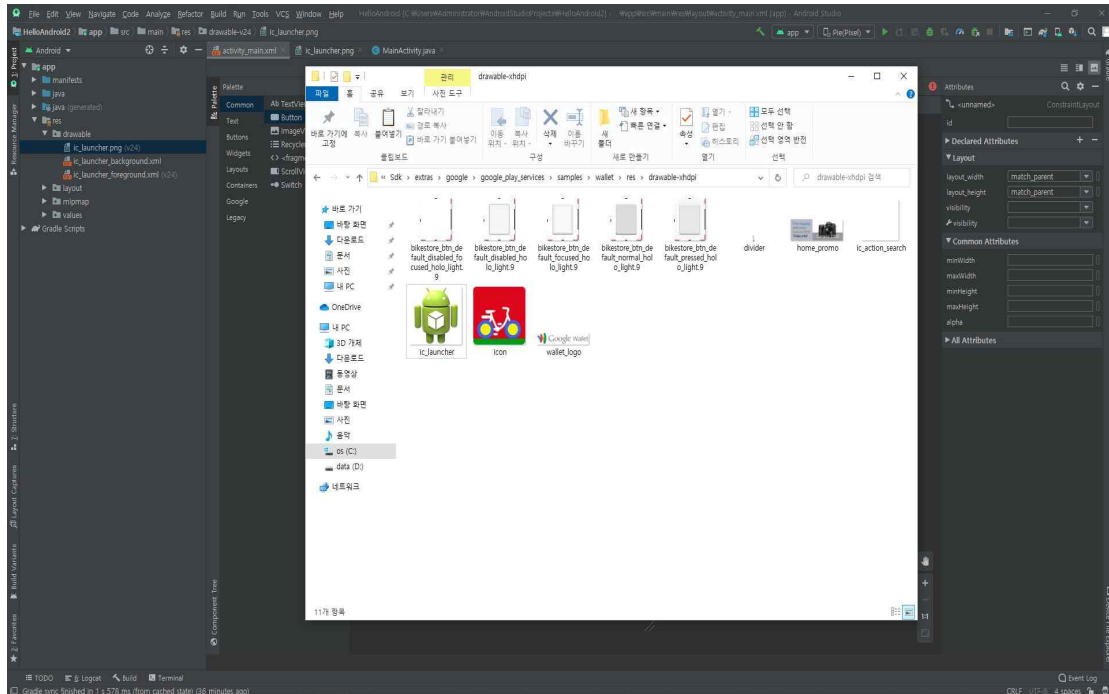
2. AVD화면에 위젯들이 잘 보이고 동작이 되는지 확인한다.



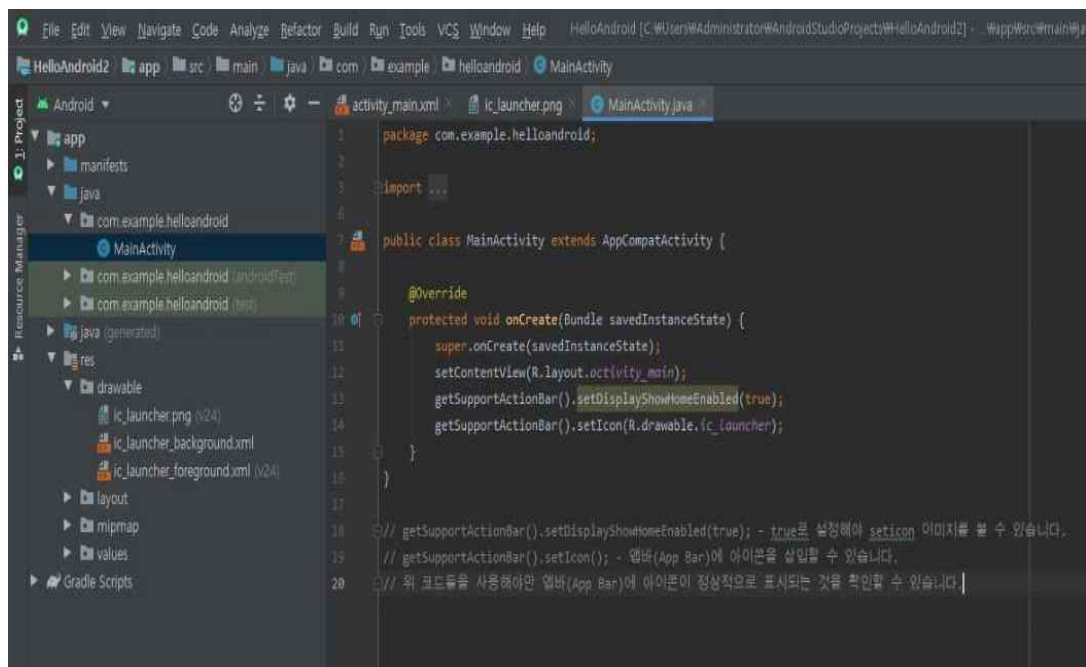
IV. 직접 풀어보기 2-1

이 섹션은 최준호 조원이 담당하였다.

1. 앱 아이콘으로 사용 할 이미지를 Project Tree의 [app]-[res]-[drawable]에 복사했다.

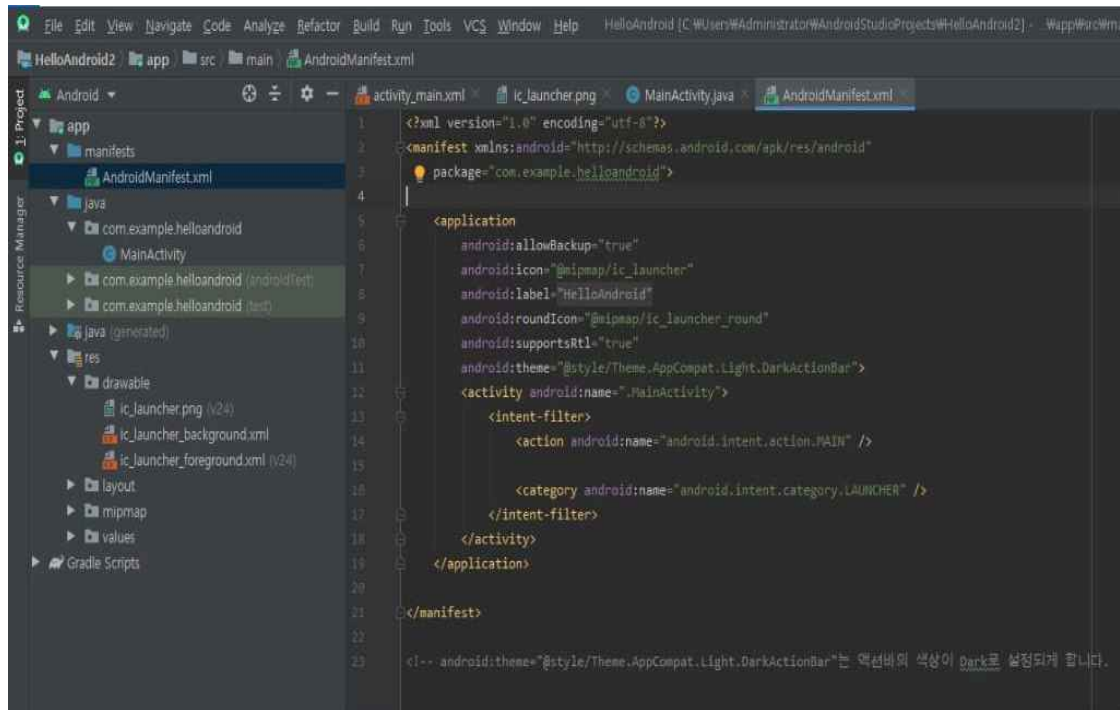


2. Project Tree의 [app]-[java]-[com.example.helloandroid]-[MainActivity]에
`getSupportActionBar().setDisplayHomeAsUpEnabled(true);`
`getSupportActionBar().setIcon(R.drawable.ic_launcher);` 두 메서드를 추가했다.



3. Project Tree에 [app]-[manifests]-[AndroidManifest.xml]에서

android:theme="@style/AppTheme"를
android:theme="@style/Theme.AppCompat.Light.DarkActionBar"로 수정한다.



4. 프로젝트를 실행해 AVD의 액션바를 확인한다. (이미지, 색상 확인)

