**오픈 소스 전문 프로젝트**

**Report #01-2 Hello Android**

|  |  |
| --- | --- |
| **학과** | **컴퓨터공학과** |
| **학번** | **2018037002 손지현**  **2018037004 이진희**  **2018037025 이도희**  **2018037026 윤소영**  **2018037054 신지애** |
| **교수님** | **박수창 교수님** |
| **제출일** | **2020/03/29** |

****

**목 차**

1. **과제 개요**
2. **안드로이드 프로젝트 생성**
3. **화면 디자인 및 편집**
4. **Java 코드 작성 및 수정**
5. **프로젝트 실행 및 결과 확인 (AVD)**
6. **AVD에서 앱 아이콘을 출력하는 방법**

**1. 과제 개요**

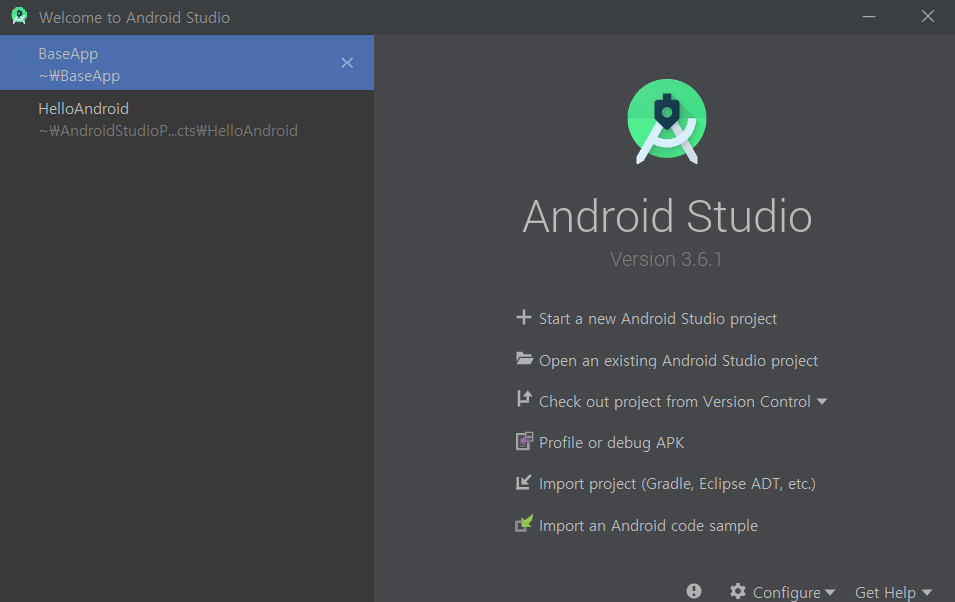
조별로 첨부된 주교재 내용을 따라 Hello Android를 작성하고 코드와 주석, 상세 과정들을 캡쳐하고 단계별 설명하시오. (섹션별 또는 설명 별 보고서 상 담당 조원 이름을 함게 적으시오.)

**2. 안드로이드 프로젝트 생성**

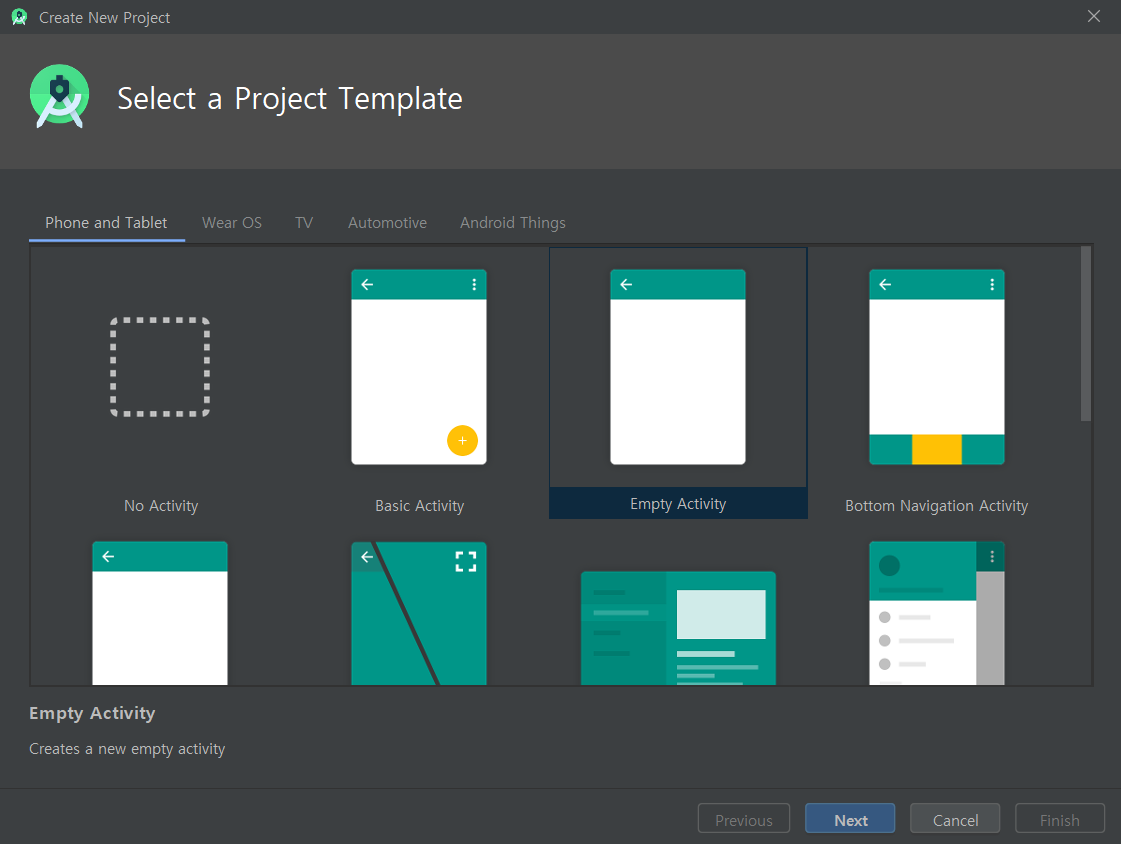
**[담당: 2018037026\_윤소영]**

***(version 3.6.1 사용)***

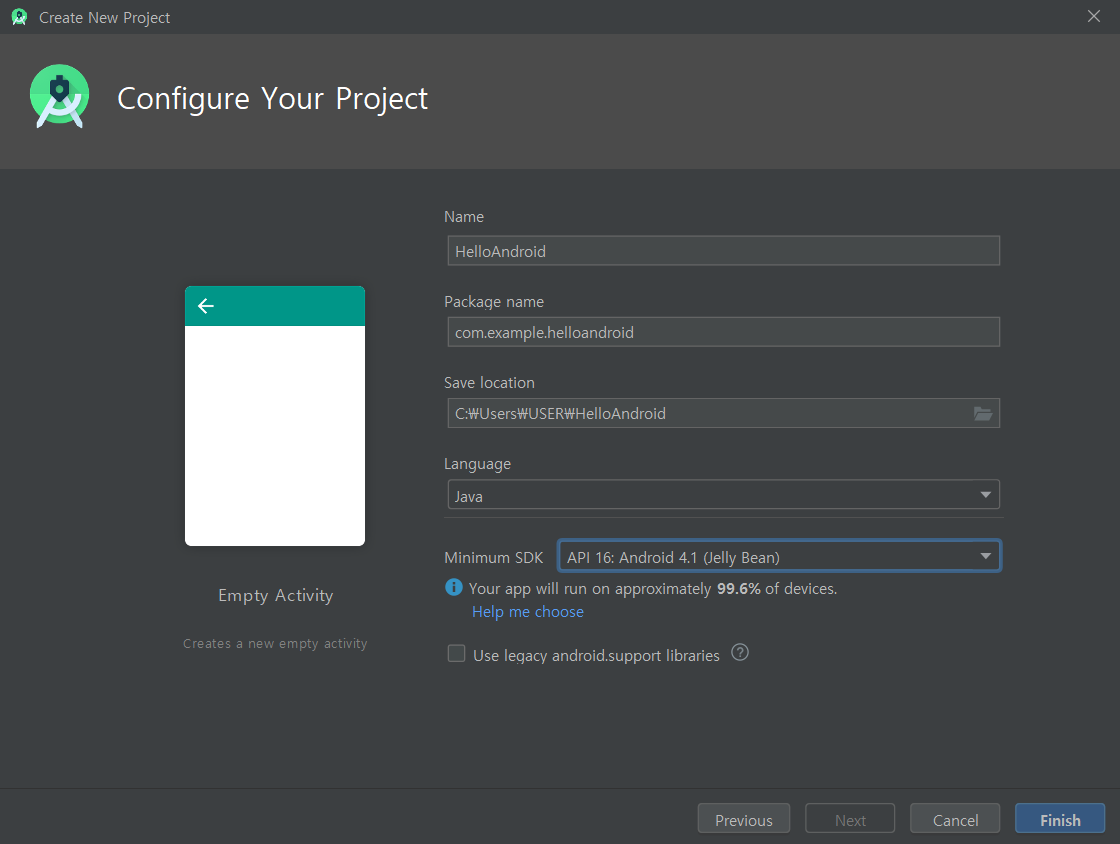
* + - * 1. **Android Studio를 실행한 후 [Start a new Android Studio project] 클릭**



* + - * 1. **Select a Project Template 에서 [Empty Activity] 클릭**



* + - * 1. **[Configure Your Project]에서 프로젝트 정보 입력**



# Name : 프로젝트 이름은 자동으로 애플리케이션 이름과 동일하게 설정 (HelloAndroid)

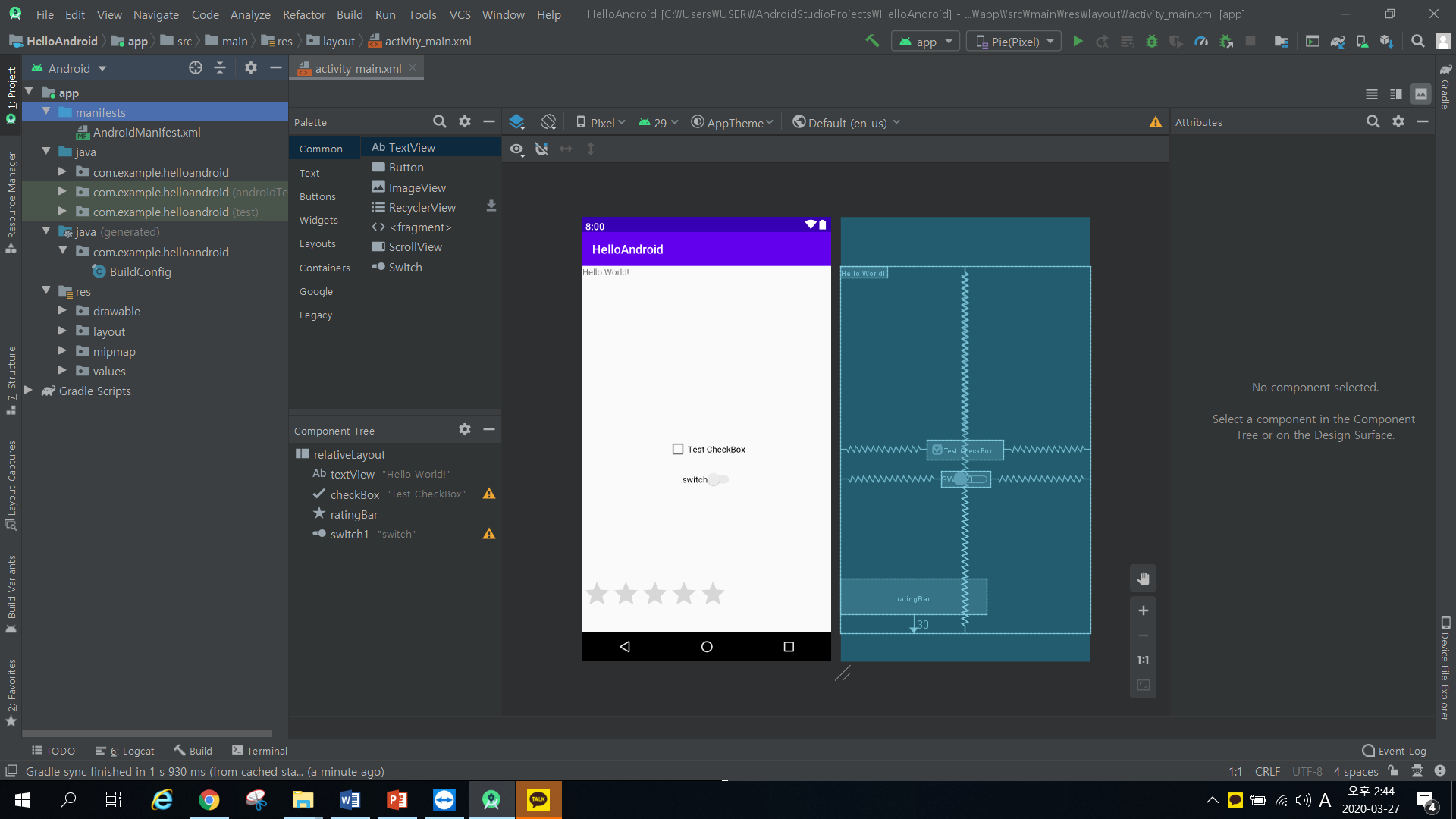
# Package name : 자동 생성 되며 도메인 이름과 애플리케이션 이름이 반대로 이어짐.

# Save location : 저장할 폴더 이름 지정

# Language : Java / Kotlin (수업은 Java로 진행함.)

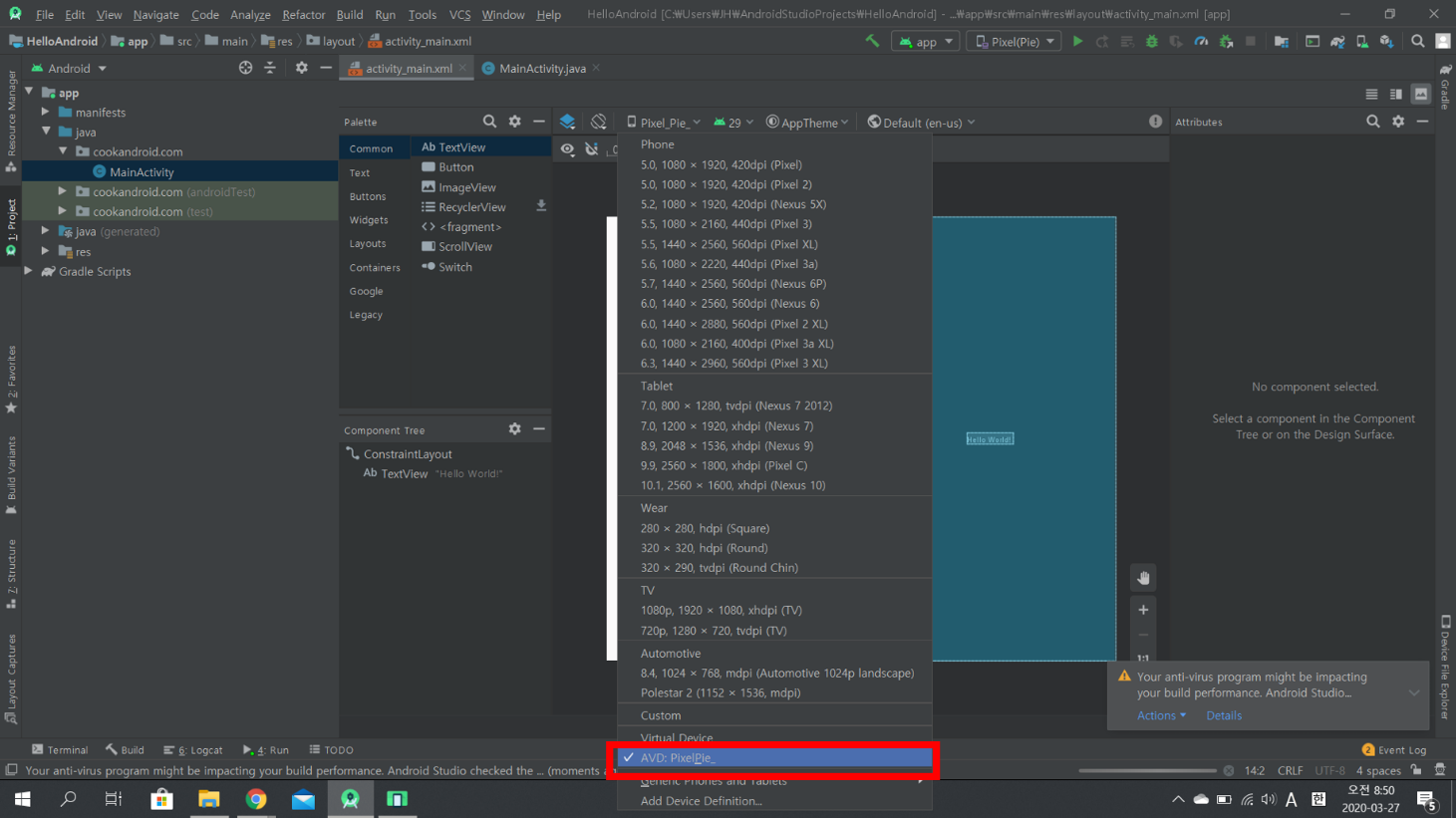
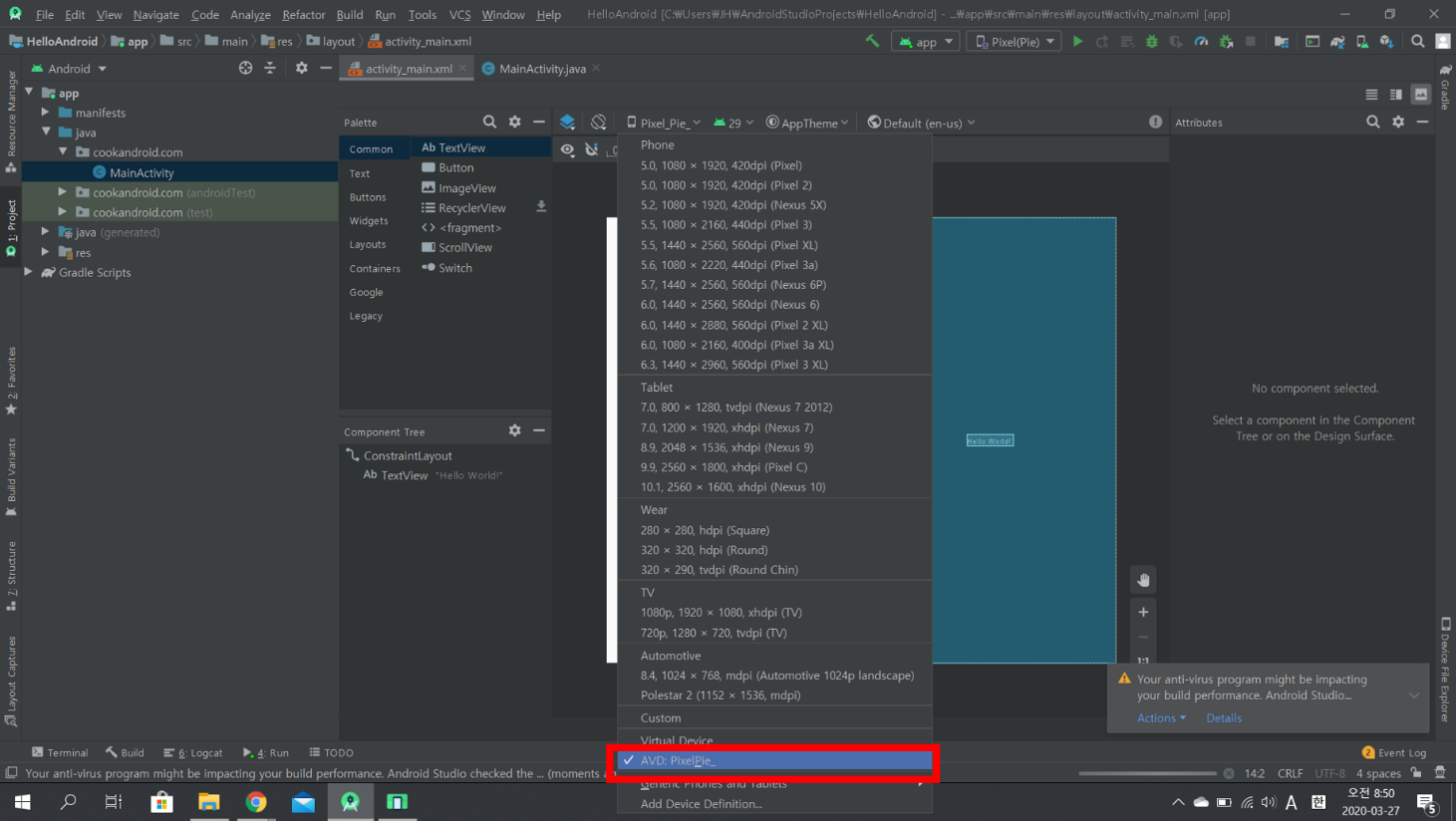
# Minimum SDK : API 16 : Android 4.1 (Jelly Bean)으로 설정

* + - * 1. **[finish] 클릭하면 프로젝트 생성**



**3. 화면 디자인 및 편집 [담당:2018037004 이진희]**

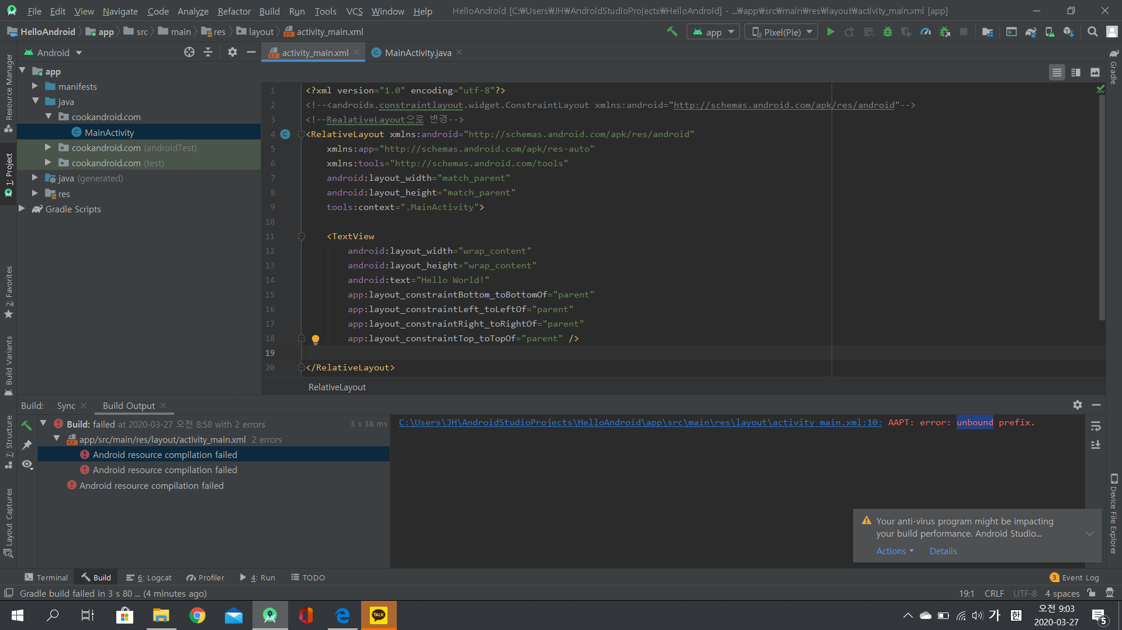
**(1) 화면 상단에 Device for preview 선택창에서 AVD Pixel(Pie)를 선택하여 첫째주에 만들어 놓은 AVD(Pixel(Pie))와 같은 환경을 만들어준다**



Device for preview

**…**

**(2) 원래 android.support.constraint.ConstraintLayout으로 되어있던 코드를 Relative layout으로 바꿔 레이아웃을 바꾼다.**



(오른쪽 상단 Code/Split/Design 모드가 있는데 Code모드에서 코드를 확인 할 수 있다)

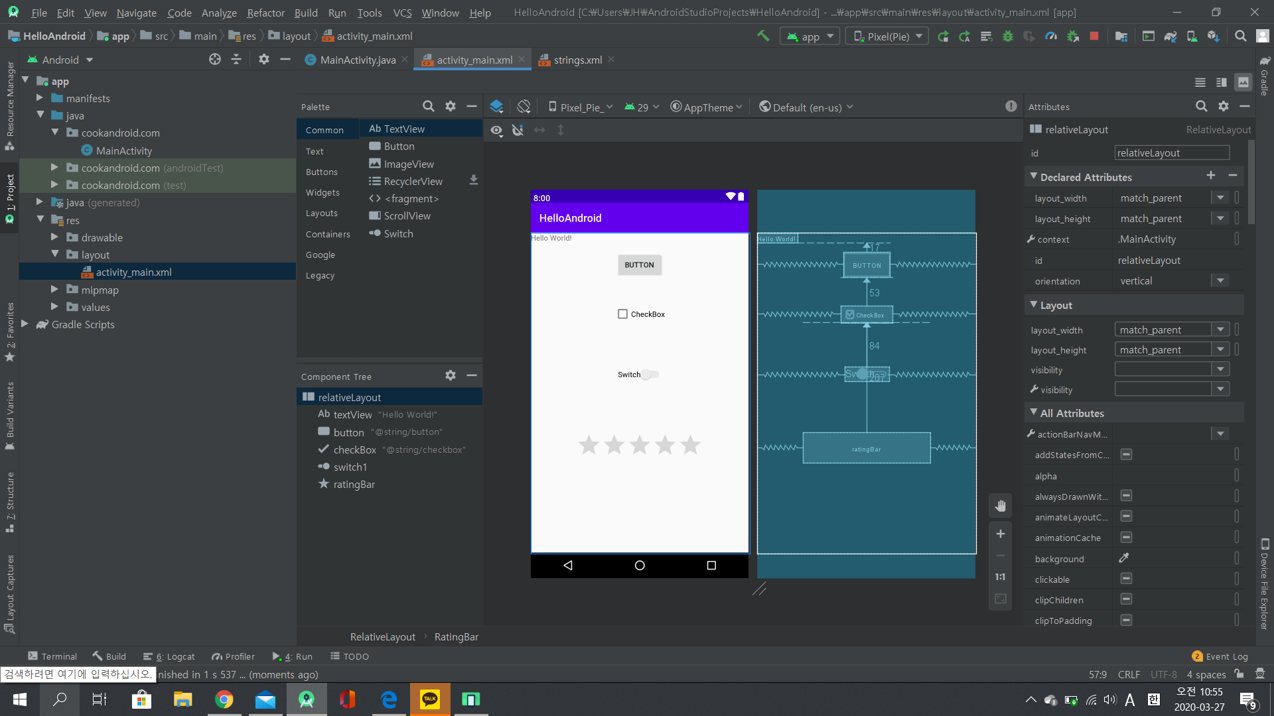
**Constraintlayout vs relative layout**

Relative layout : 레이아웃에 뷰를 배치할 때 뷰 위젯들의 상대적 위치 관계에 따라 뷰가 표시될 영역을 지정하는 방법.(자식 뷰와 부모 뷰간의 관계에 따라 배치)

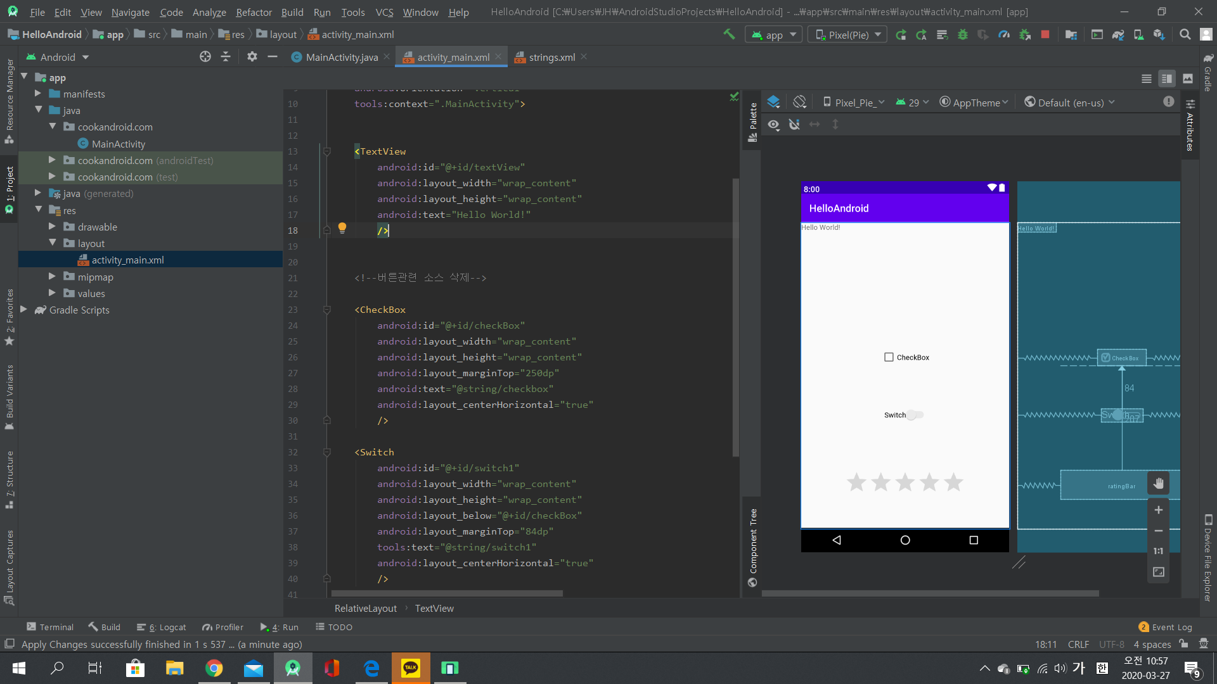
Constraintlayout : 뷰 위젯의 위치와 크기를 유연하게 조절할 수 있게 만들어주는 레이아웃

**(3) 왼쪽 상단의 palette에서 위젯을 끌어와 화면에 놓는다.**

**끌어 놓은 위젯은 component Tree에서 확인할 수 있다.**



**(4) Split모드를 켜고 xml 소스에서 <button … />을 삭제하면 화면에서도 사라지는 것을 오른쪽 창에서 확인할 수 있다.(3.6.1 기준)**



삭제된 코드

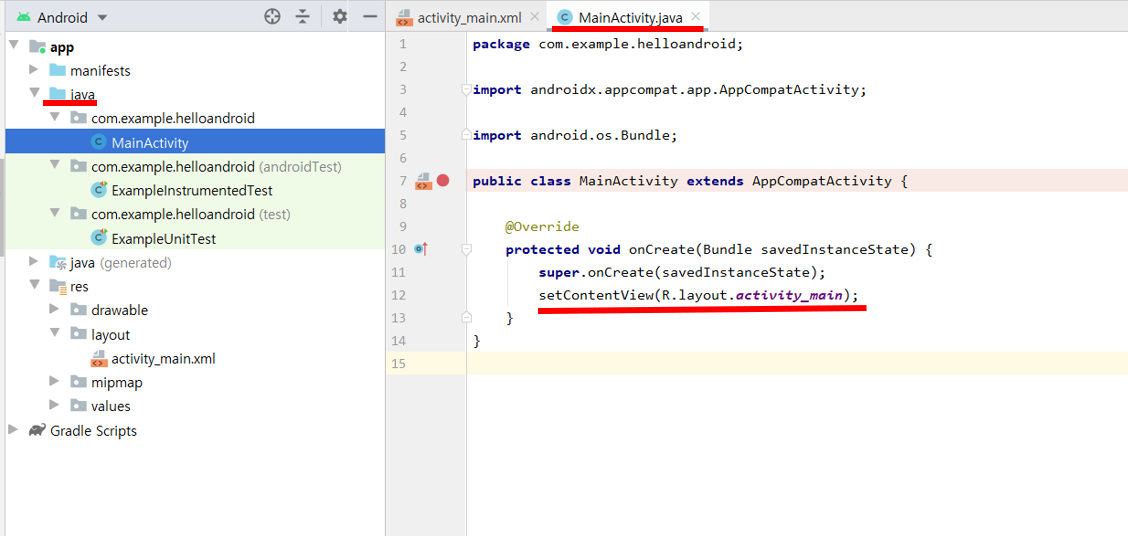
<Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/button" … />

버튼 사라짐

버튼 없어짐

**(5) (4)번 내용을 3.6.1 이하 버전에서 확인하려면 text모드에서 <button />소스를 삭제하고 design모드로 들어가서 확인해야 한다.(text모드와 design모드 밖에 없다.)**

**4. Java 코드 작성 및 수정 [담당: 2018037002 손지현]**



**MainActivity.java ->액티비티를 제어하는 java코드가 담긴 파일, 위 화면의 코드는 미리 작성 되어있다.**

**onCreate() 함수의 setContentView(R.layout.*activity\_main*);**

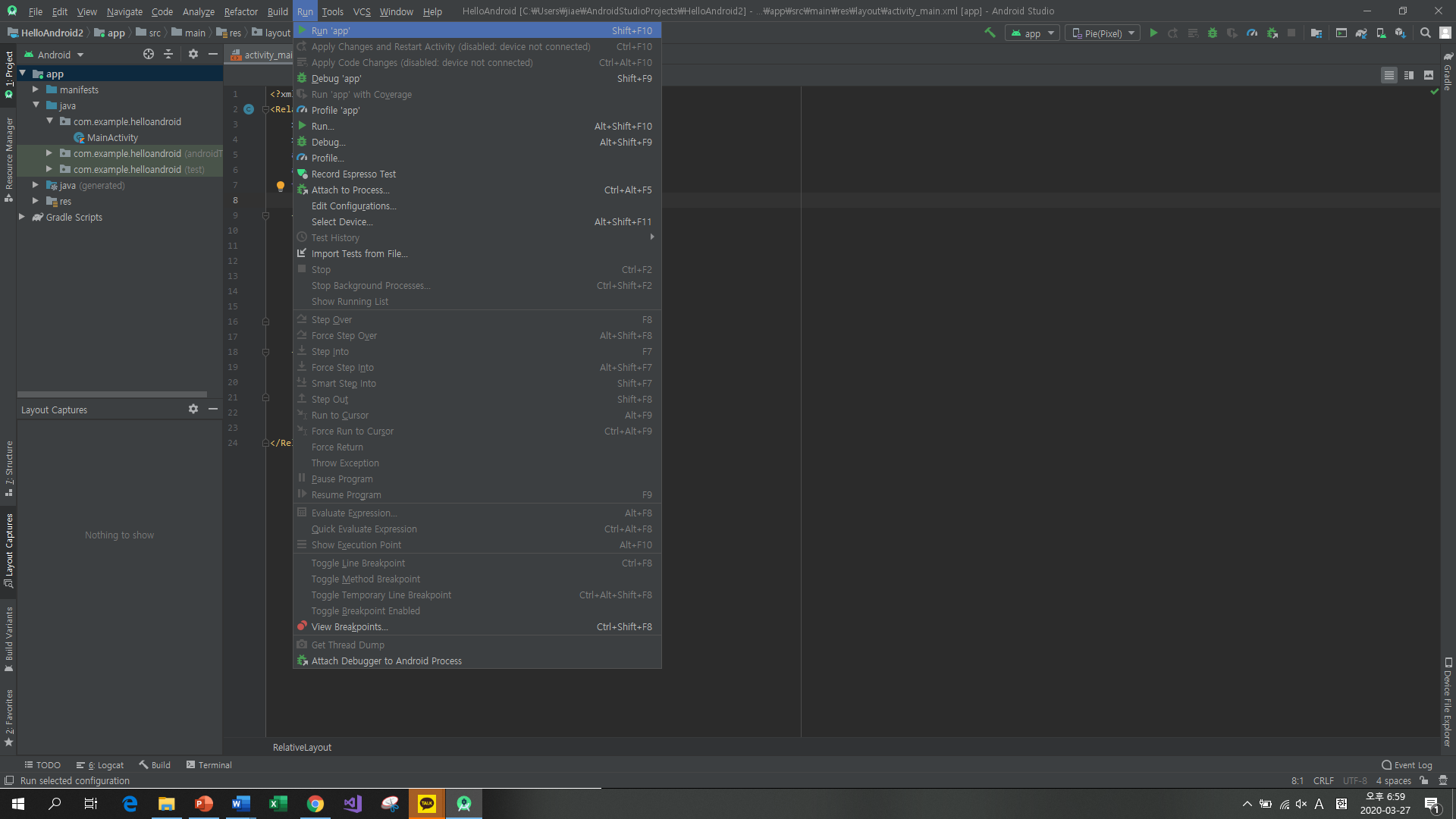
**setContentView 메서드 : 액티비티에 사용할 레이아웃을 지정한다.**

**위 코드는 -layout 패키지 아래에 있는 activity\_main.xml 이라는 파일을 View로 연결한다는 뜻이다.**

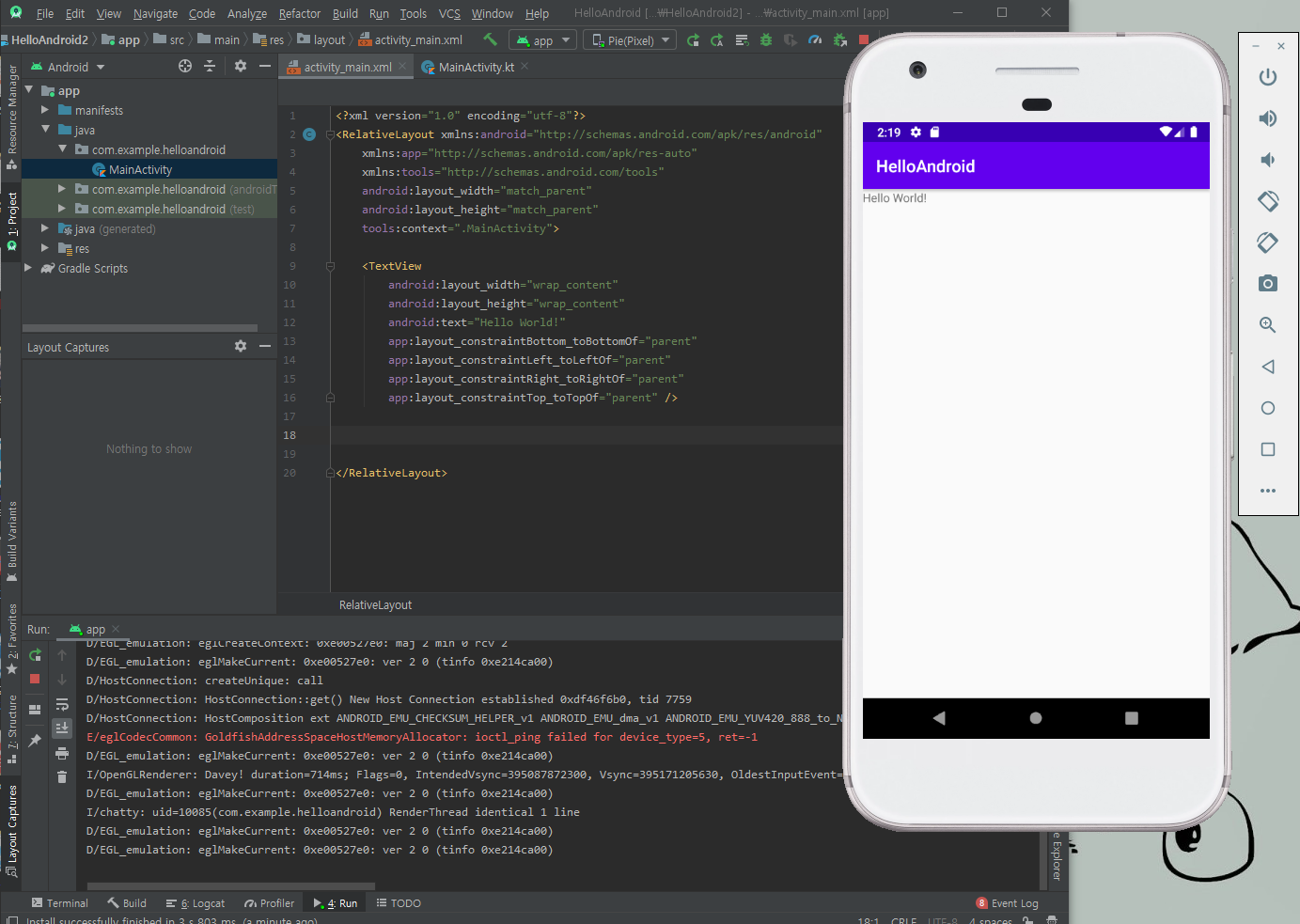
**5. 프로젝트 실행 및 결과 확인 (AVD)**

**[담당: 2018037054 신지애]**

**(1) 메뉴의 [Run As]-[Run ‘app’]을 선택하거나 [Run ‘app’] 아이콘을 클릭**

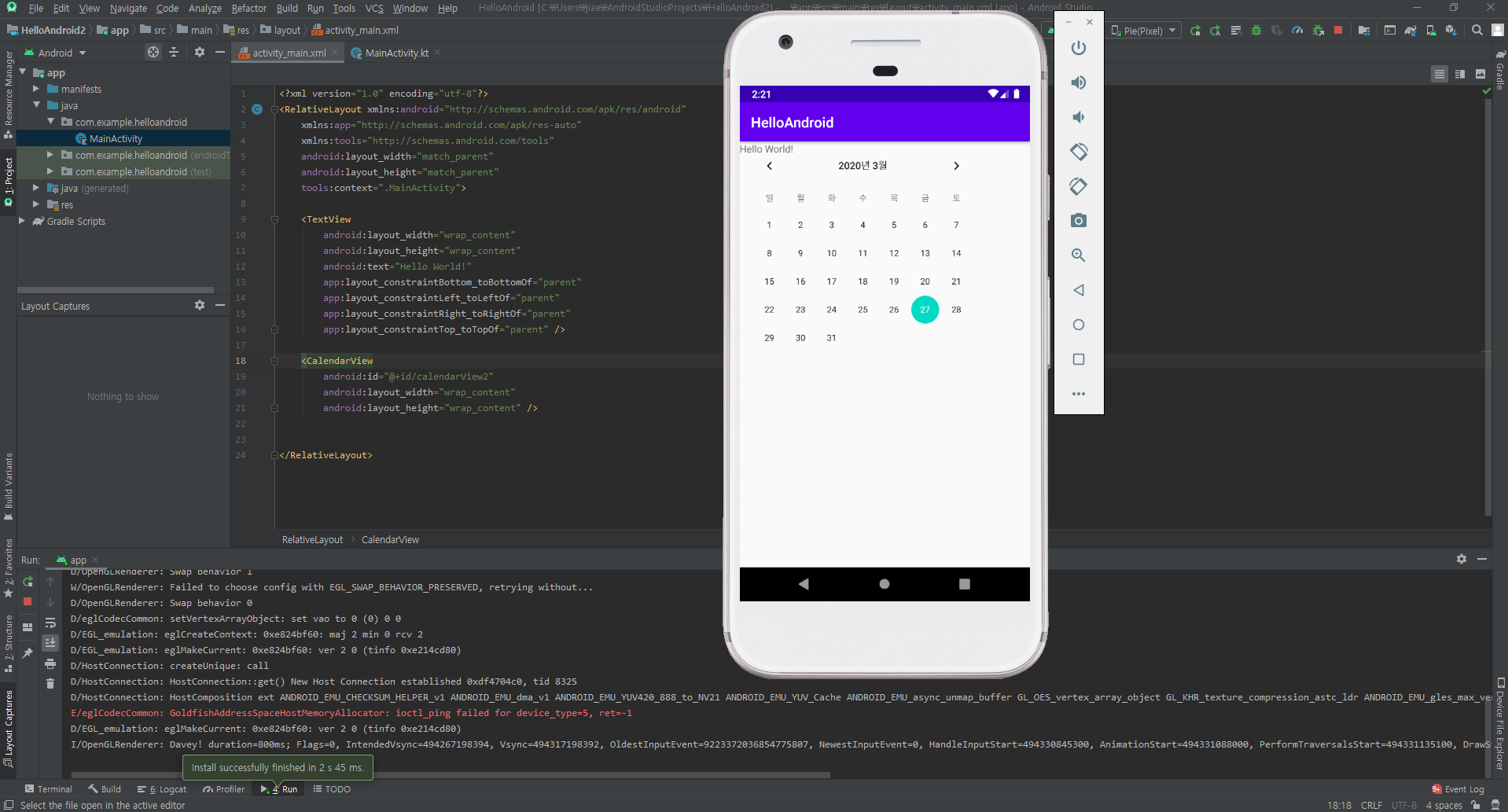


**(2)AVD가 부팅된 후 실행 결과 화면**



**화면에 아무것도 추가하지 않고 실행한 모습**

**<TextView …… /> : 실행 화면에 Hello World! 출력**



**화면에 달력을 추가하고 실행한 모습**

**<CalendarView … /> : 실행화면에 달력 출력**

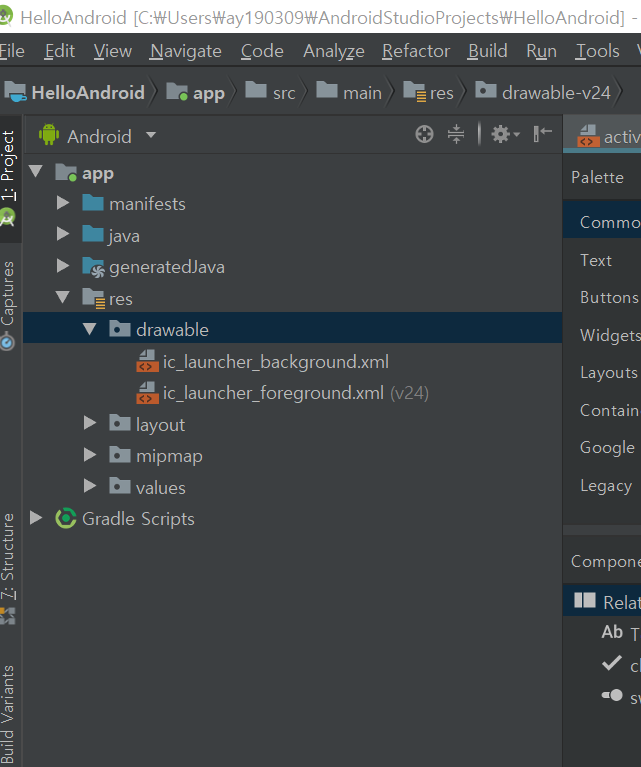
**6. AVD에서 앱 아이콘을 출력하는 방법**

**[담당: 2018037025 이도희]**

**아이콘으로 사용할 파일 다운로드**

**(※파일 제목에 특수 기호 있을 시 오류발생)**

**[Icon\_bear.png]**





**▶**

**[res]-[drawable] 폴더 안에 아이콘 파일을 넣어준다.**



**MainActivity에 다음 두 행을 코딩한다.**

**getSupportActionBar().setDisplayShowHomeEnabled(true);**

//setIcon 이미지를 볼 수 있도록 setDisplayShowHomeEnabled(true)로 설정해놔야함

**getSupportActionBar().setIcon(R.drawable.아이콘 파일 이름);**

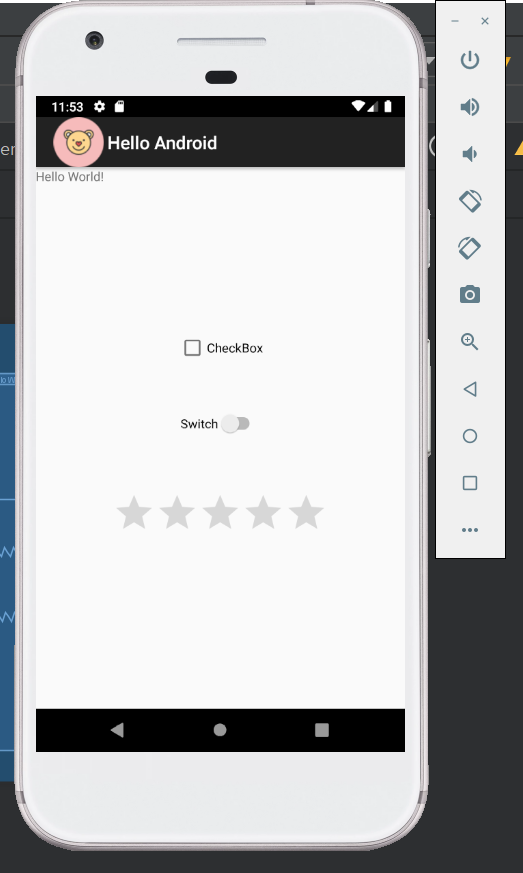
//액션바에 아이콘 이미지를 삽입

**[app]-[manifests]-[AndroidManifest.xml]에 android:theme를 다음과 같이 수정한다. (생략가능)**

**android:theme = “@style/AppTheme”**

* **android:theme = “style/Theme.AppCompat.Light.DarkActionBar”**

//액션바는 Dark로 View는 Light로 설정



**다음과 같이 화면의 왼쪽 상단에 앱 아이콘이 출력 된다.**