

Tinker board

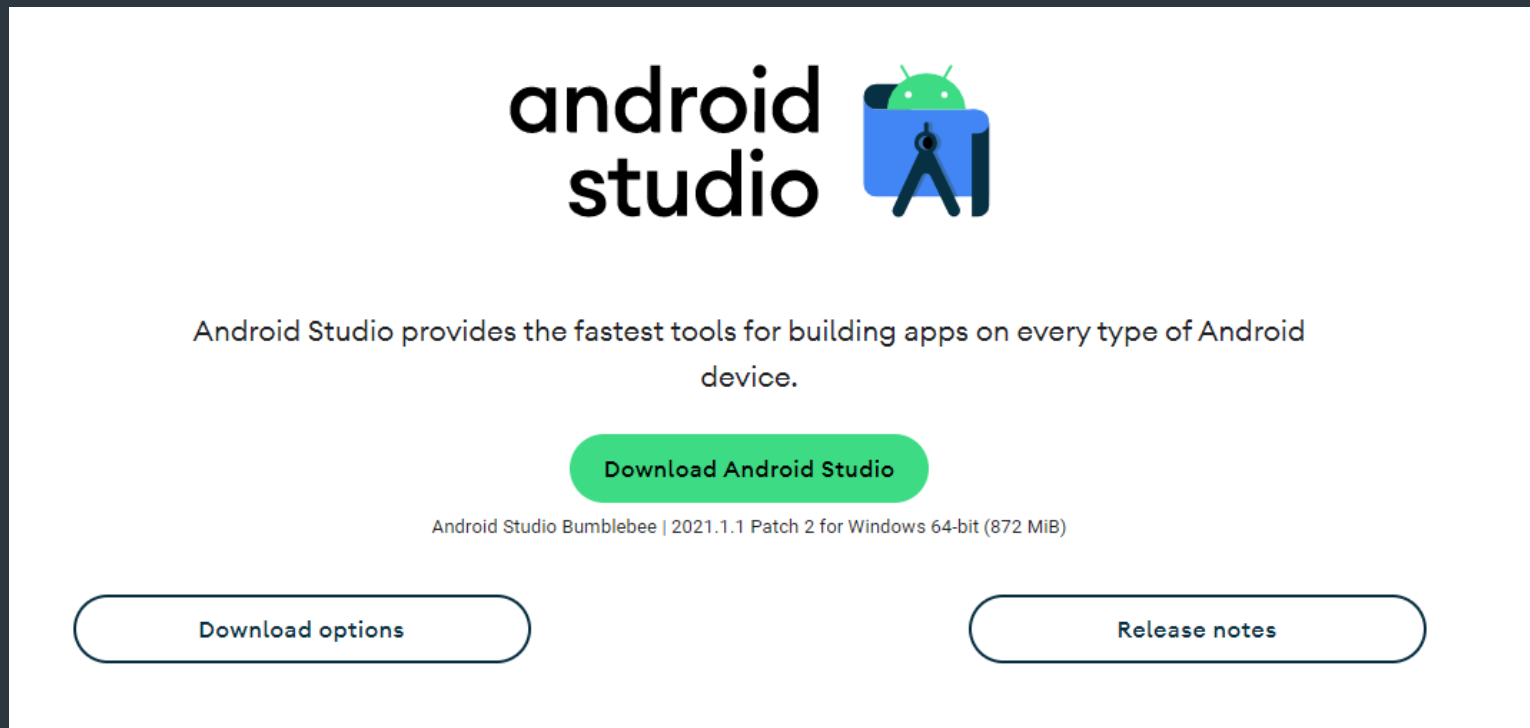
IoT를 위한 최선의 엣지 단말기



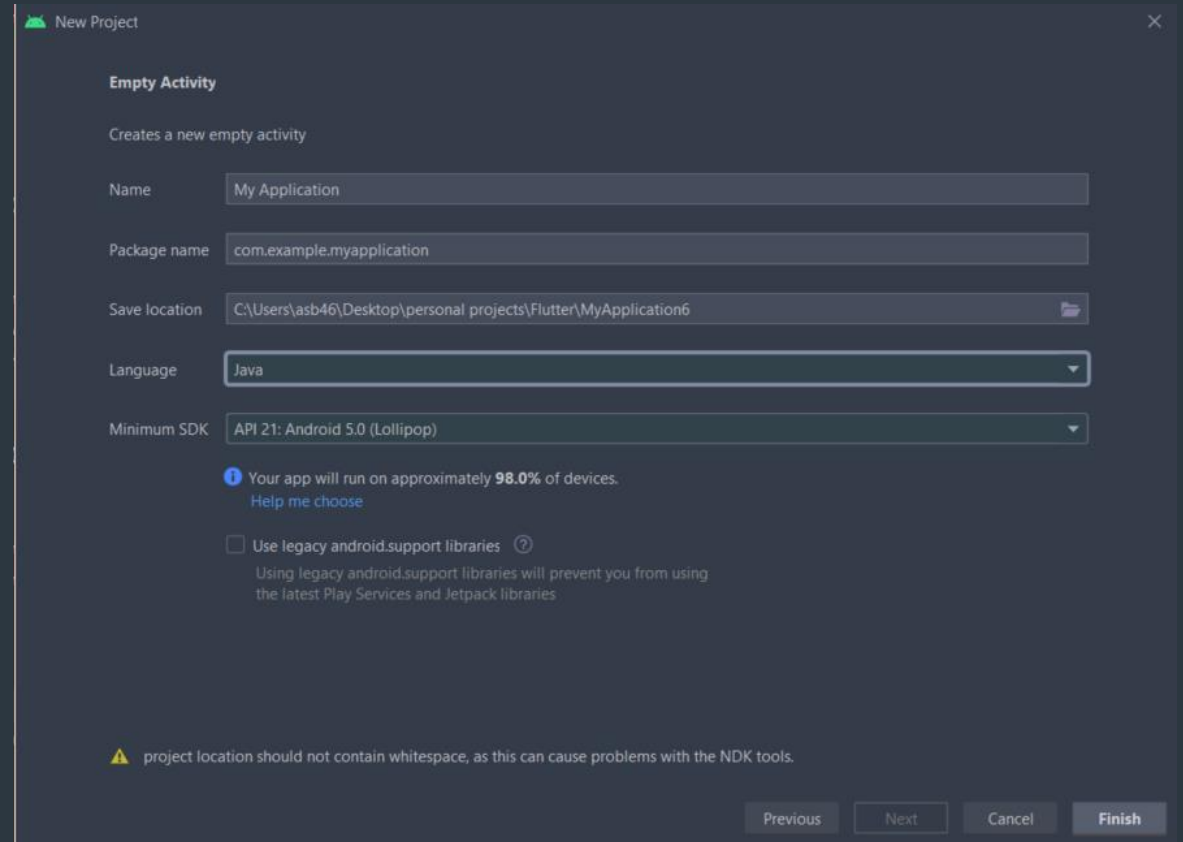
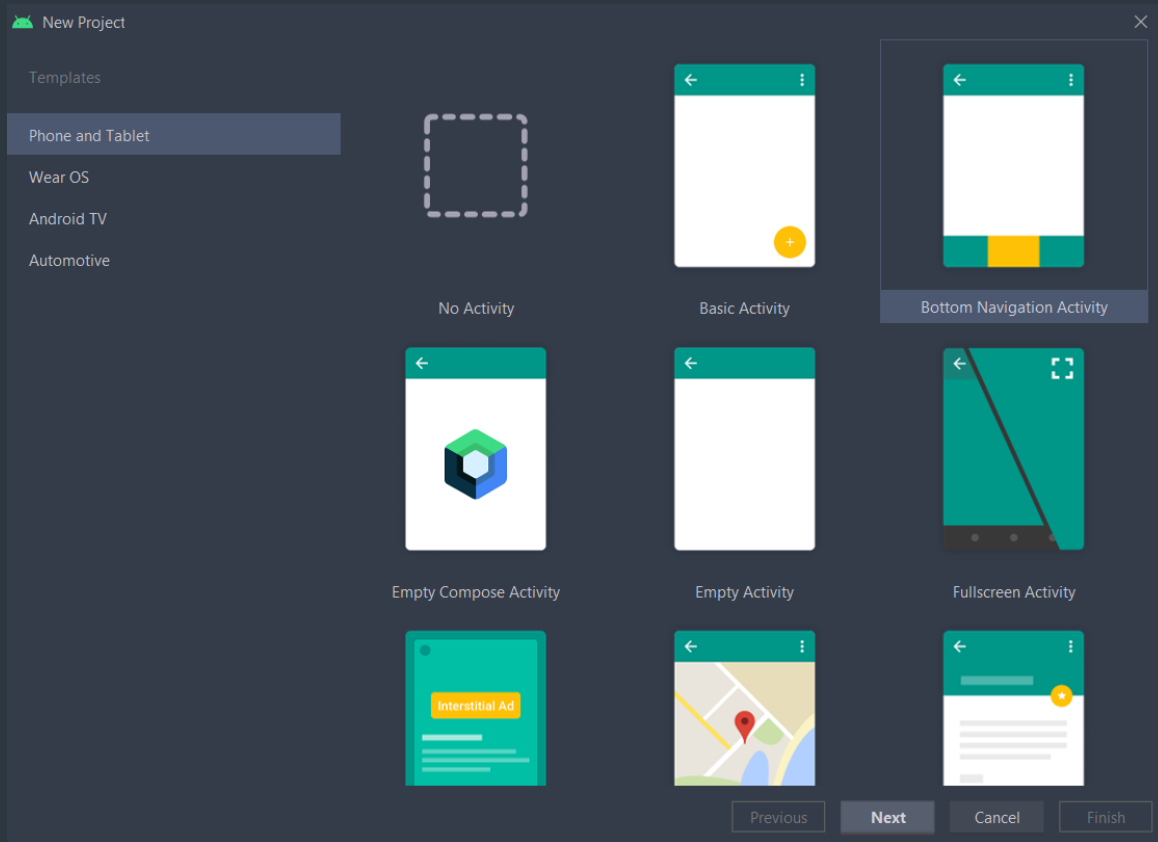
MAKER SPACE
G·CAMP

안드로이드 스튜디오 설치

<https://developer.android.com/studio>



Project 생성



API ↑: able to use more functionalities

API ↓: covers lower Android versions

MainActivity 구성

```
MainActivity.java x activity_main.xml x
1 package com.example.myapplication;
2
3 import androidx.appcompat.app.AppCompatActivity;
4 import android.os.Bundle;
5
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     }
```

MainActivity 파일이 위치한 디렉토리 (프로젝트 package 이름)

안드로이드 하위버전(API)을 지원하기 위한 액티비티

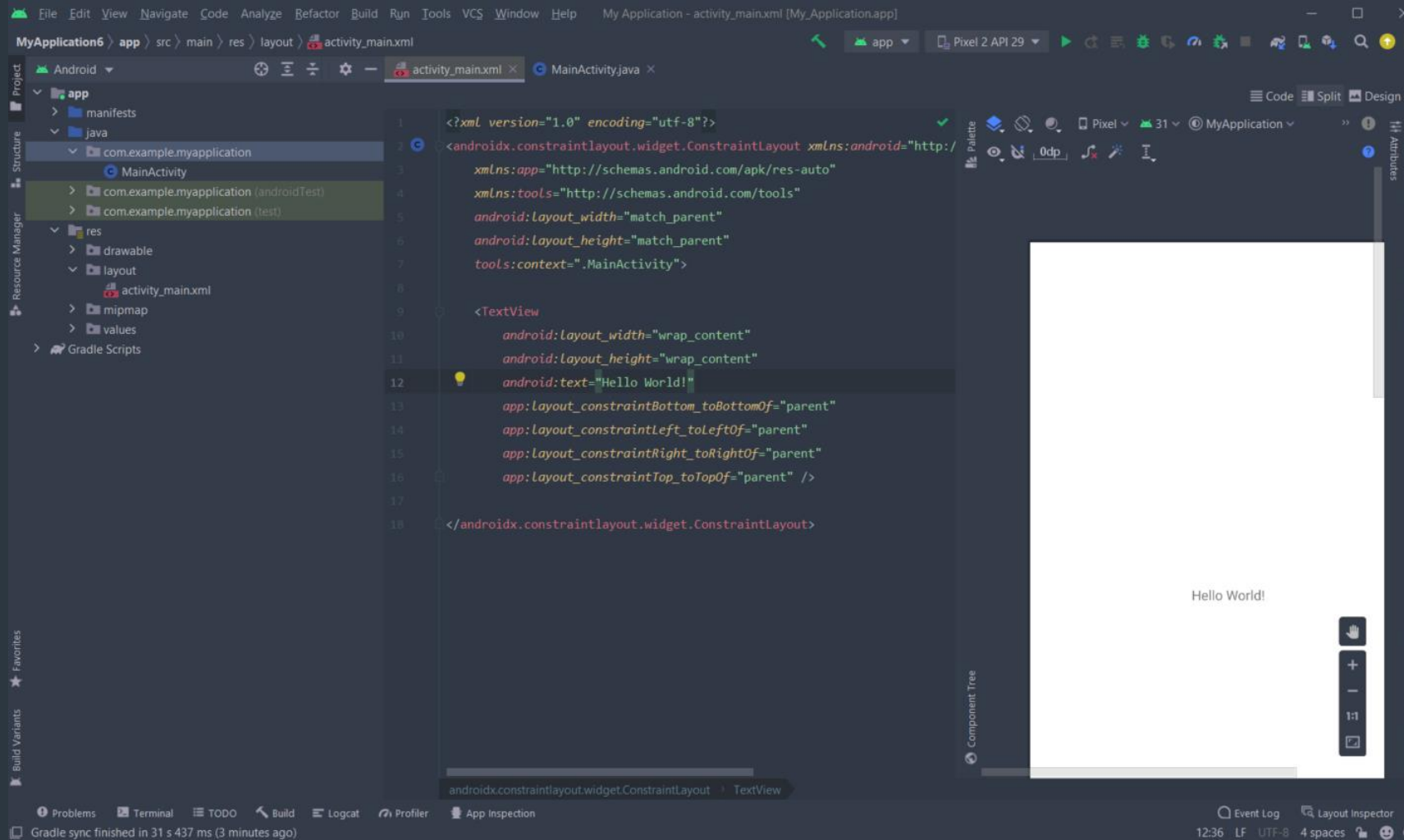
AppCompatActivity 상속

AppCompatActivity의 onCreate 함수 재정의

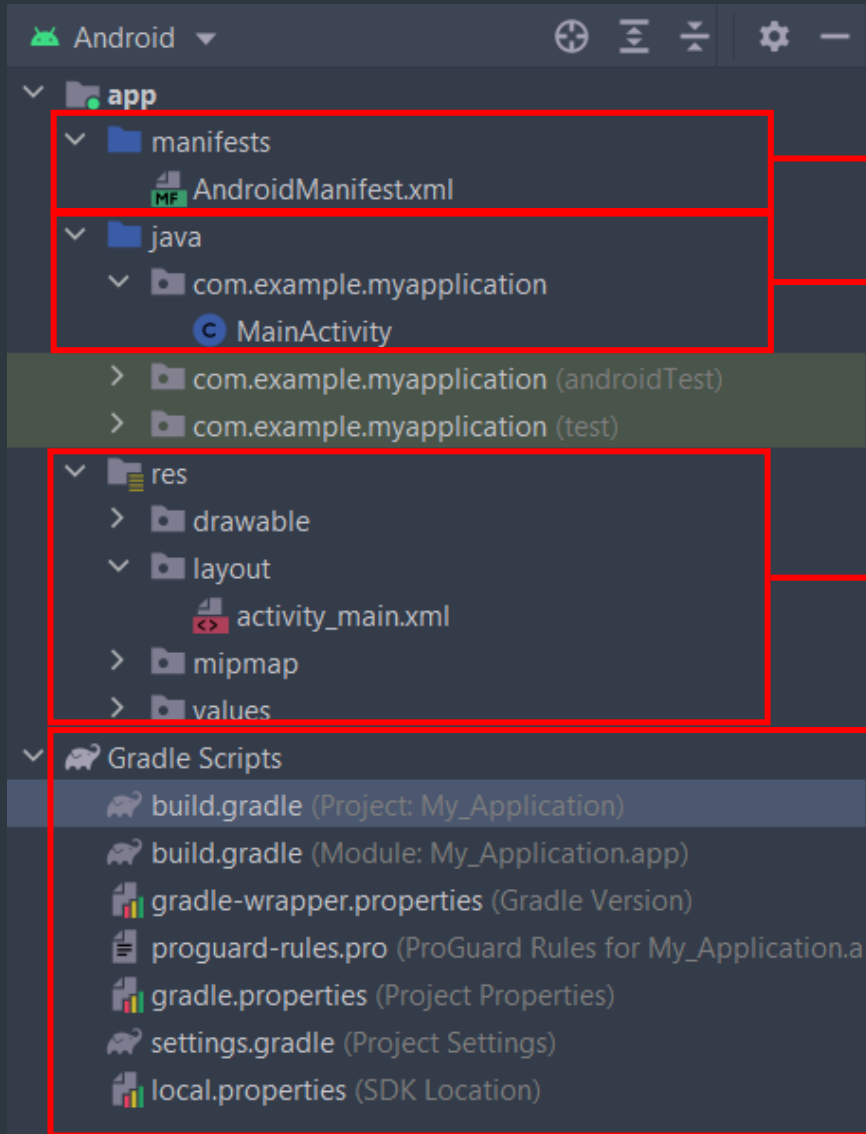
AppCompatActivity의 onCreate 함수 호출

화면에 무엇을 보여줄 지를 결정하는 함수

Layout XML 구성



안드로이드 프로젝트 파일시스템 구성



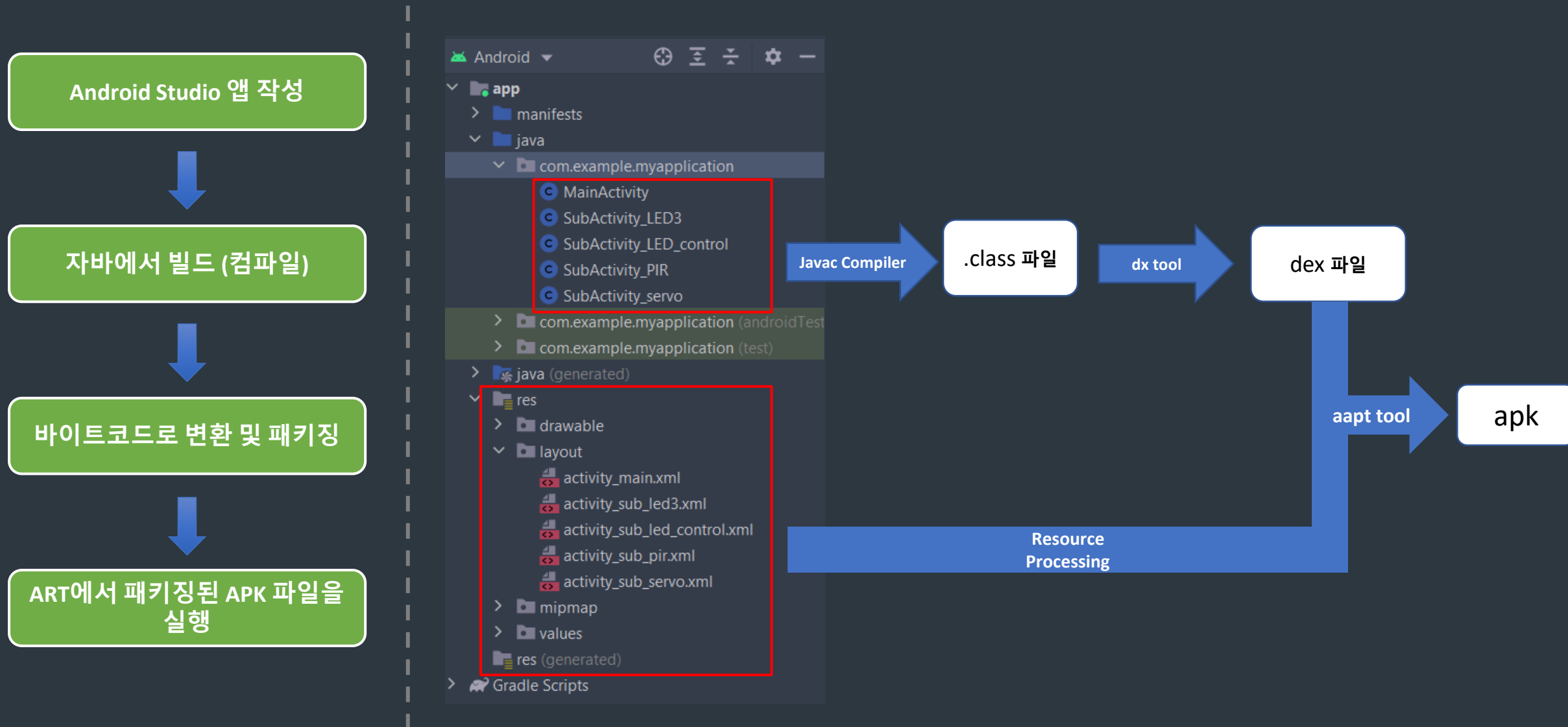
앱의 구성 요소나 권한 등의 정보를 담고 있는 파일

앱의 기능을 담당하는 소스 파일이 있는 폴더

XML 레이아웃, img 파일등의 리소스를 담고 있는 폴더

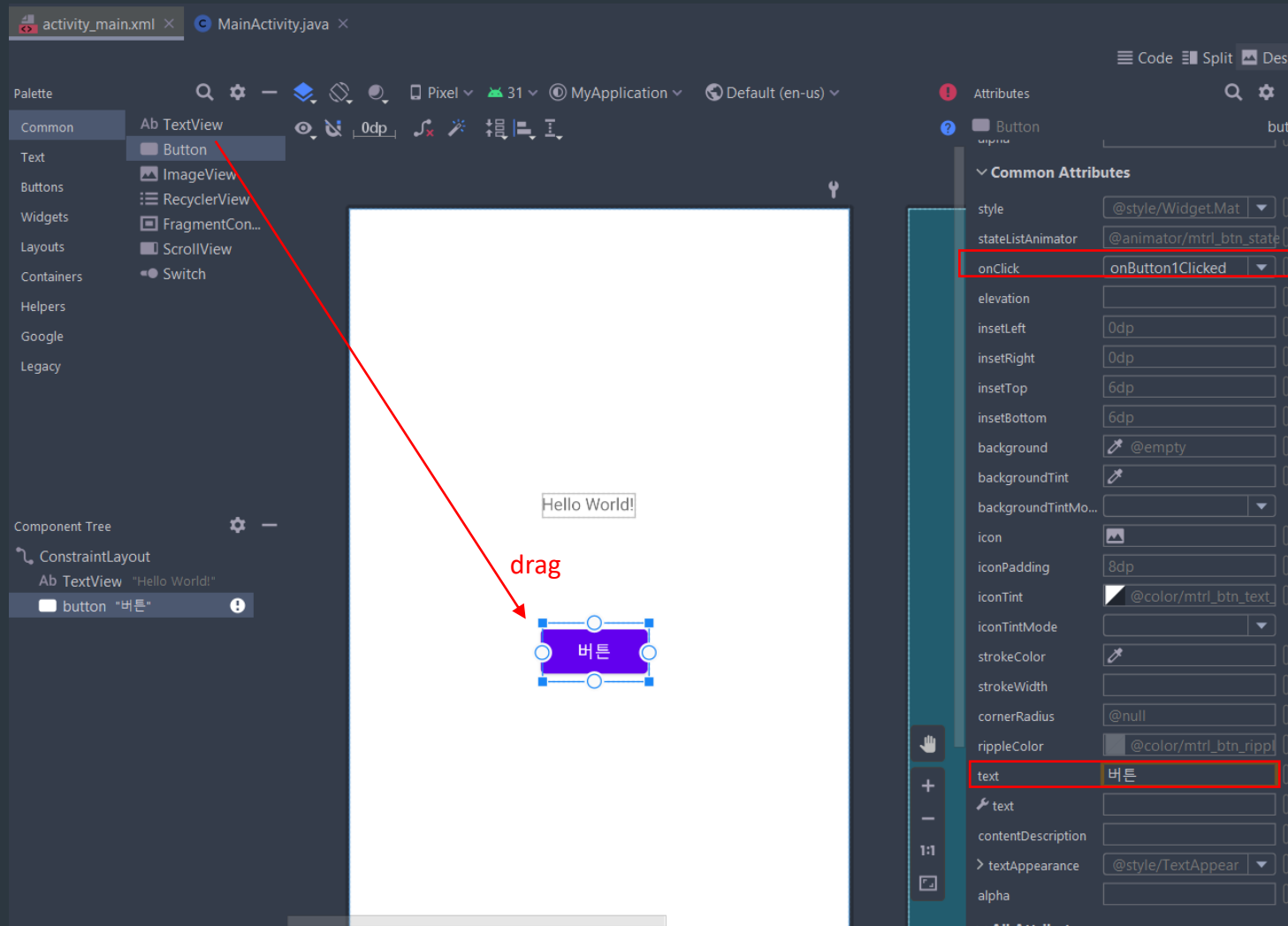
소스파일과 리소스파일을 빌드하기 위한 script파일들

소스 파일과 리소스가 apk로 빌드되는 과정



버튼 구현 예제: layout에서 버튼 배치하고 함수 연동

Tinker Board 연결



버튼 구현 예제: MainActivity 정의

```
package com.example.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

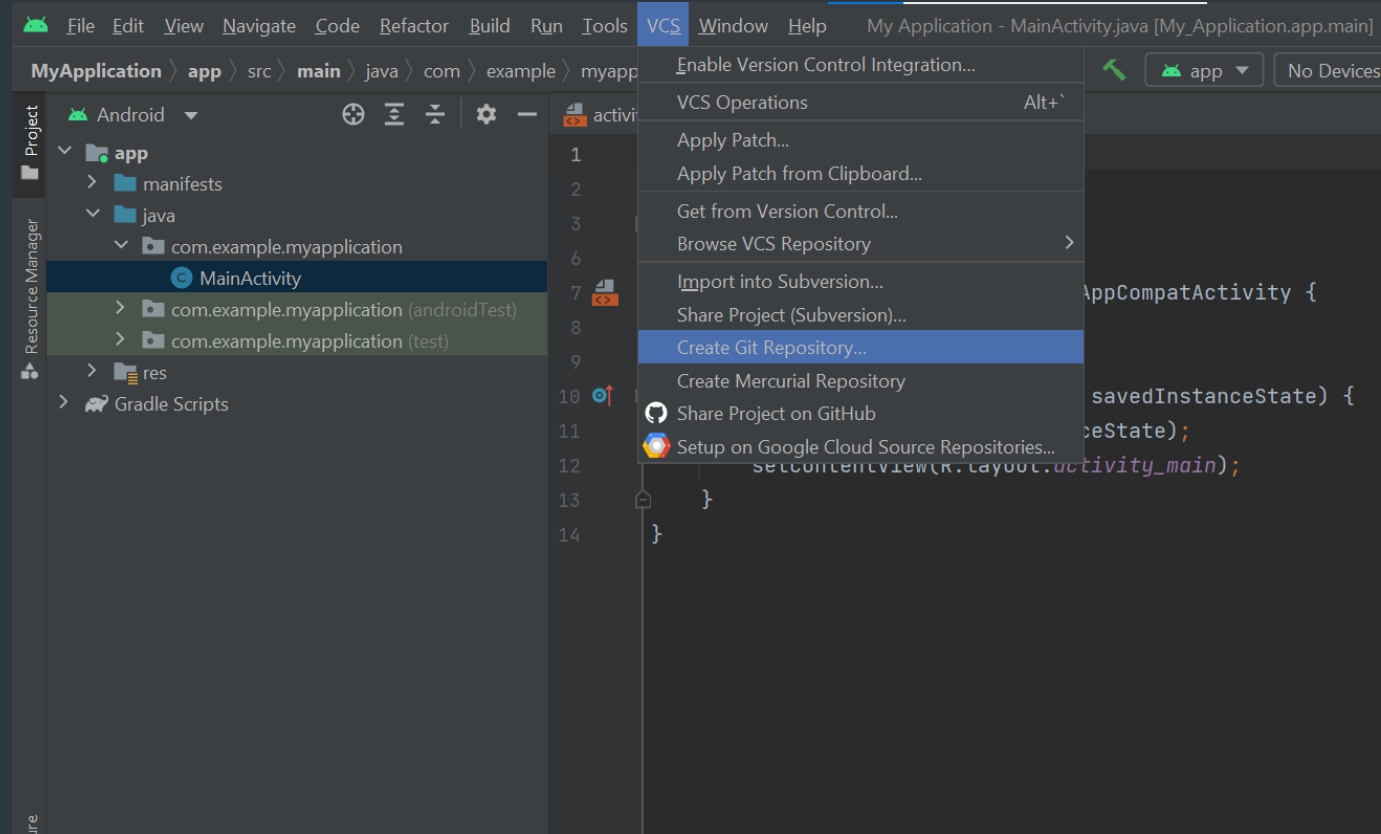
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void onClicked(View v) {
        Toast.makeText(context: this, text: "버튼이 눌리졌습니다", Toast.LENGTH_LONG).show();
    }
}
```

VCS on Android Studio

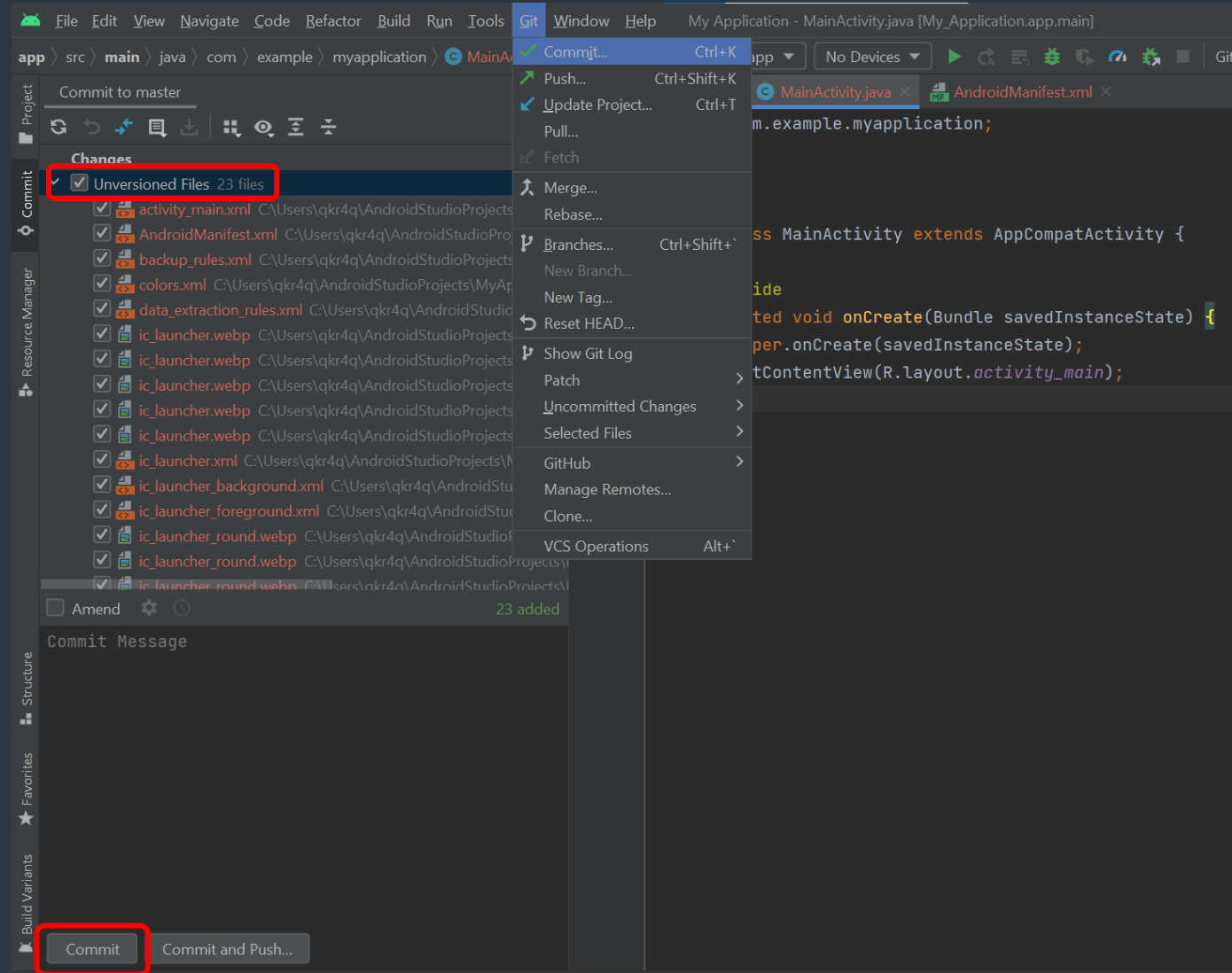
VCS를 통해 Git Repository 생성

1. 상단 바의 VCS 클릭
2. Create Git Repository 클릭
3. git repository 위치를 Project/app/src/main으로 설정



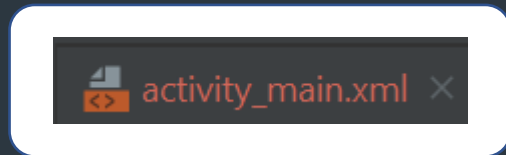
git add & commit

1. 상단 바의 git 클릭
2. Commit 클릭
3. 왼쪽에 열린 Changes의 Unversioned Files 확인
4. 모든 파일 체크
5. commit 메시지 작성
6. 왼쪽 하단의 commit 버튼 클릭
7. checks failed 메시지가 뜨면 한번더 commit 클릭
8. 빨간 색으로 표시된 파일들이 흰색으로 표시됨

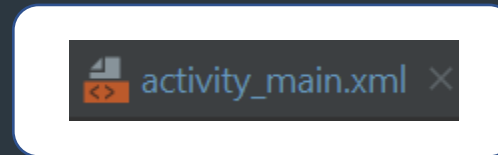


git status에 따른 색상 변화

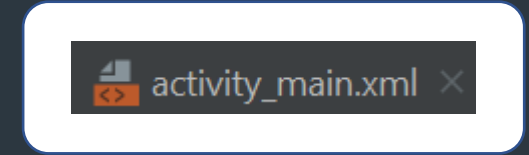
not added



edited

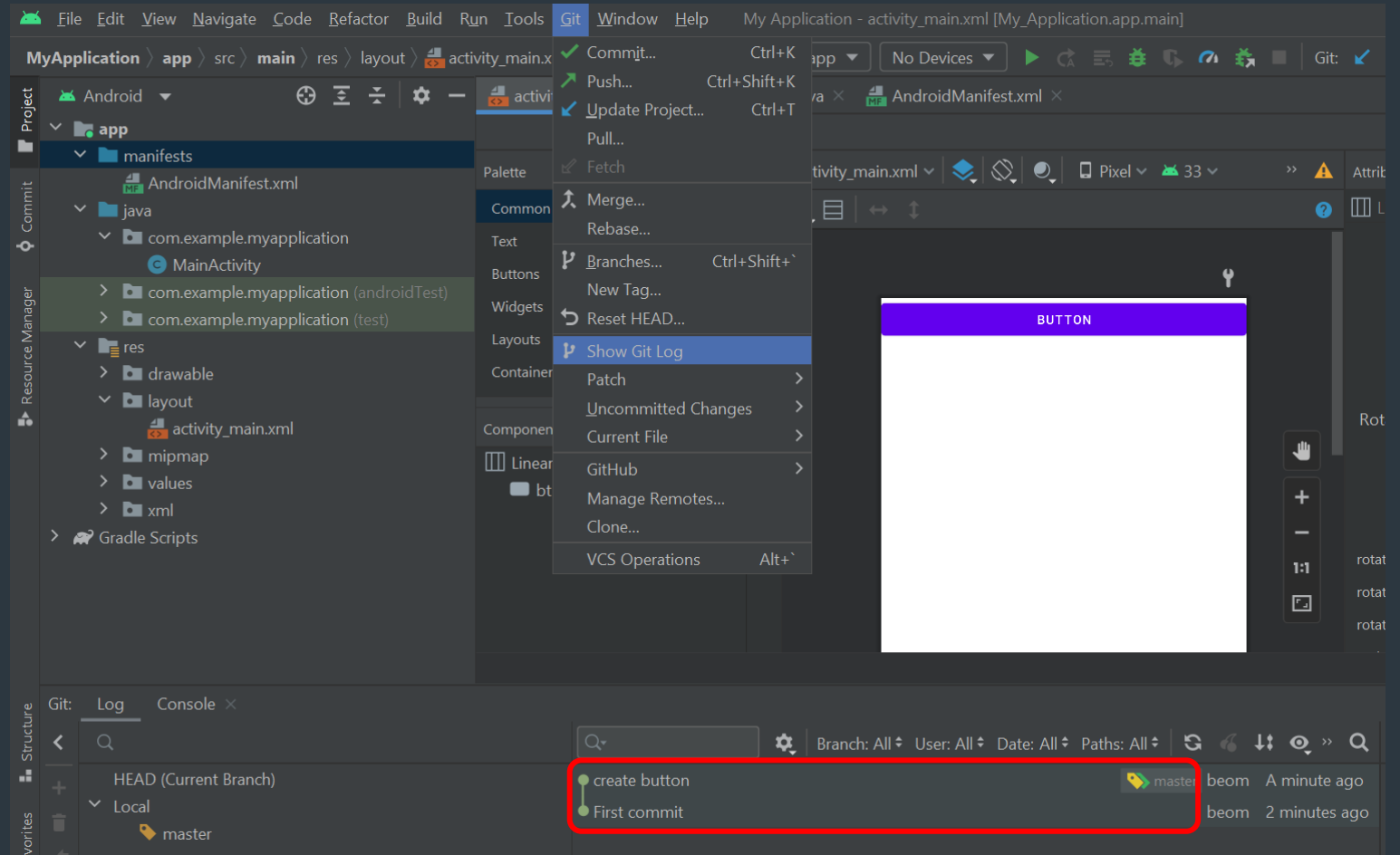


committed



Git Log 확인 및 체크아웃

1. 상단 바의 git 클릭
2. Show Git Log 클릭
3. 하단 부에서 나타나는 git log 확인
4. checkout 하고자 하는 commit 우클릭
5. Checkout Revision 'commit hash' 클릭



git branch 생성

1. 상단 바의 Git 클릭
2. New Branch 클릭
3. branch name 설정 후 create
4. 오른쪽 하단의 git branch icon 클릭
5. 원하는 branch 선택 후 checkout 클릭

