**오픈소스전문프로젝트 Report #01**

**3. FourButton**

**4조: 2018037006 오지현**

**2018037007 김효희**

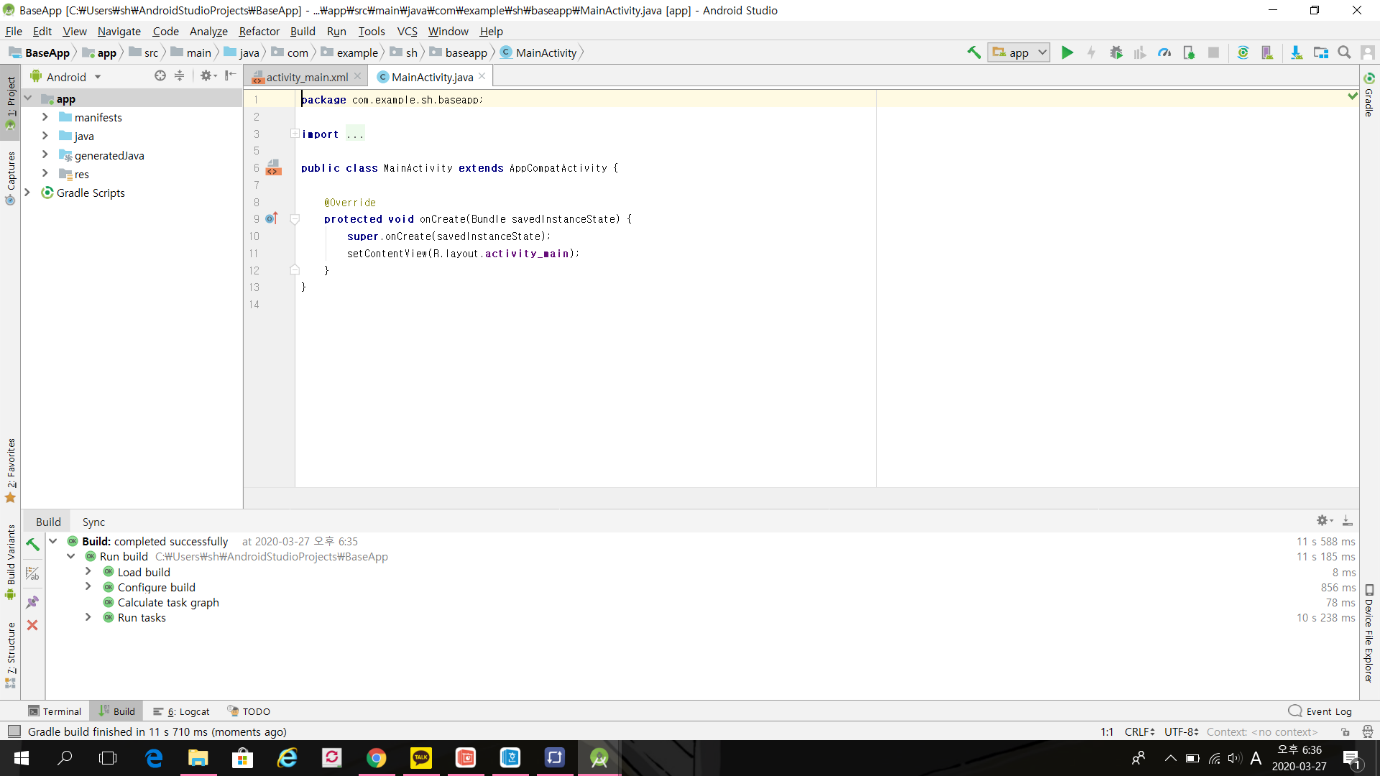
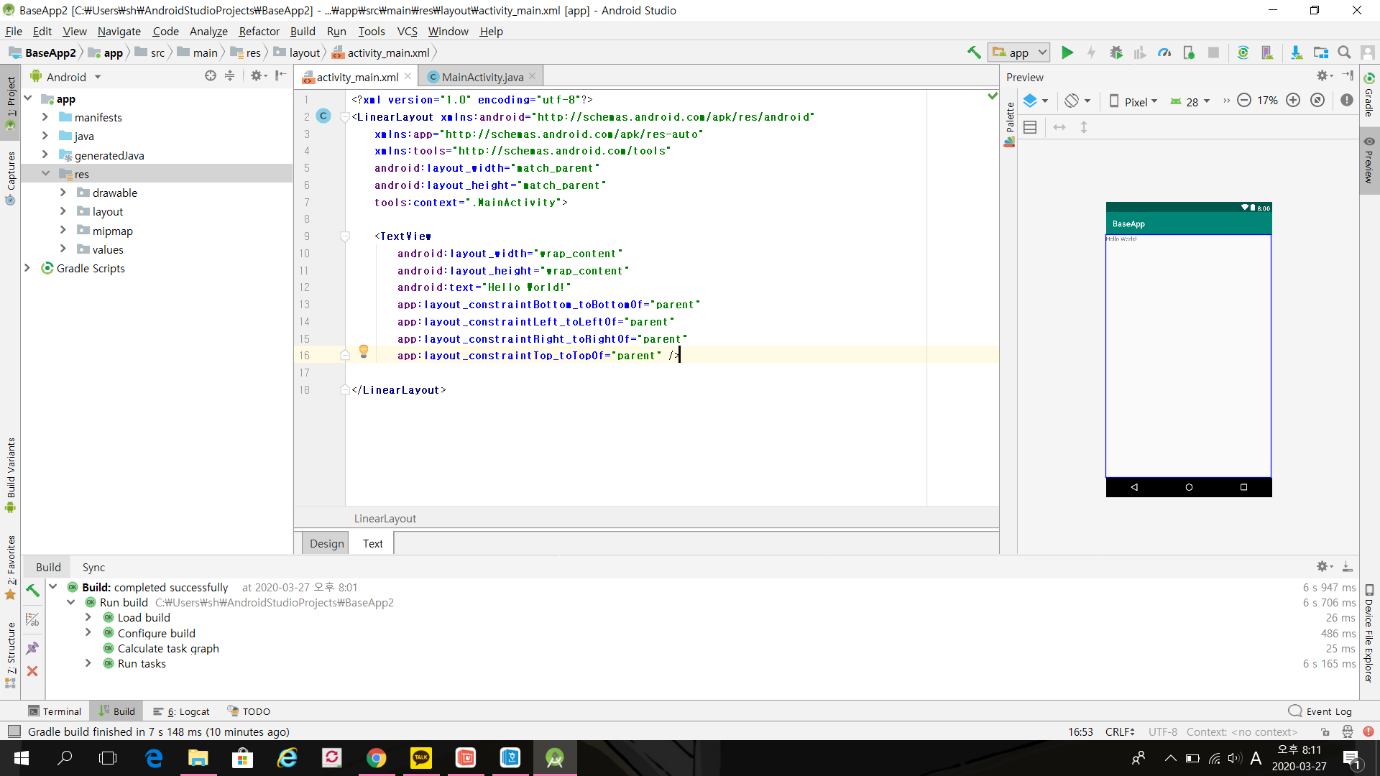
**2018037013 이제은**

**2018037015 김은진**

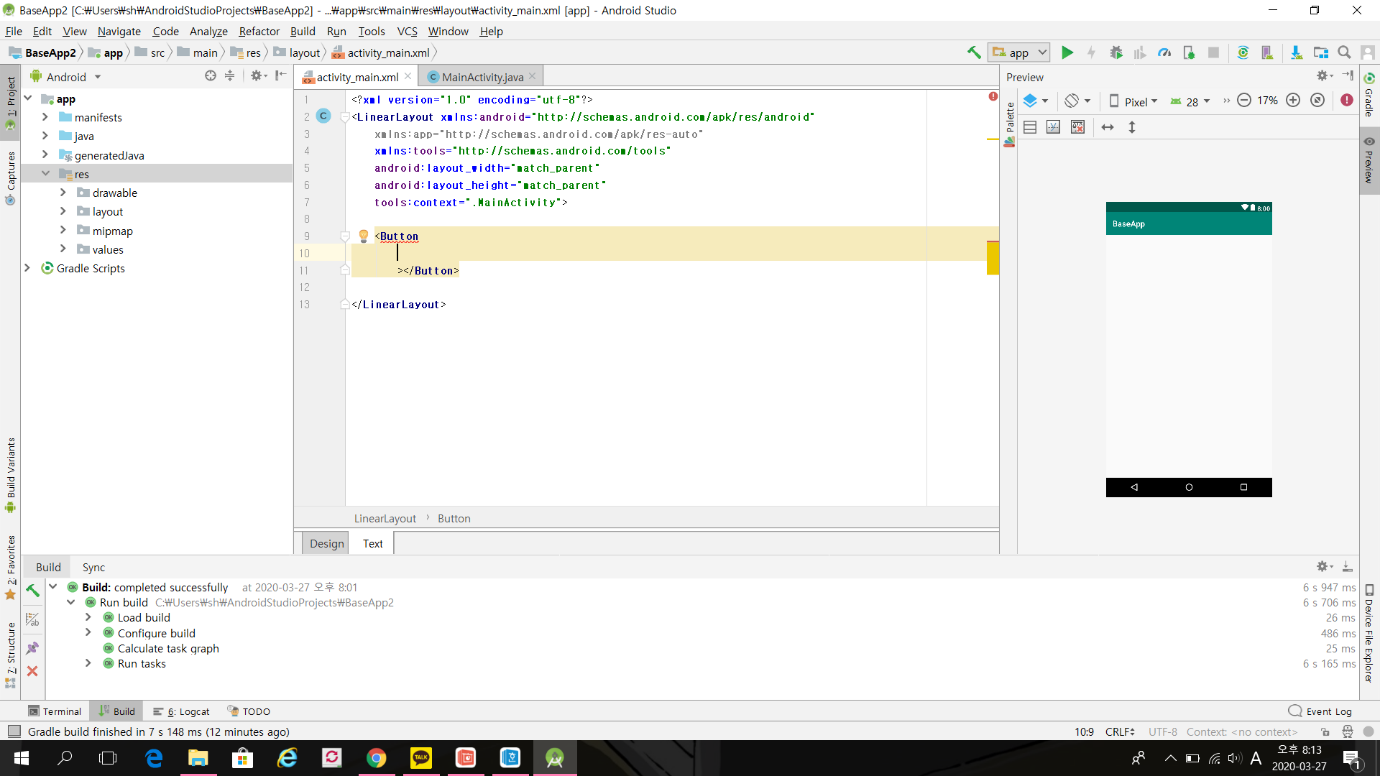
**2018037019 김소현**

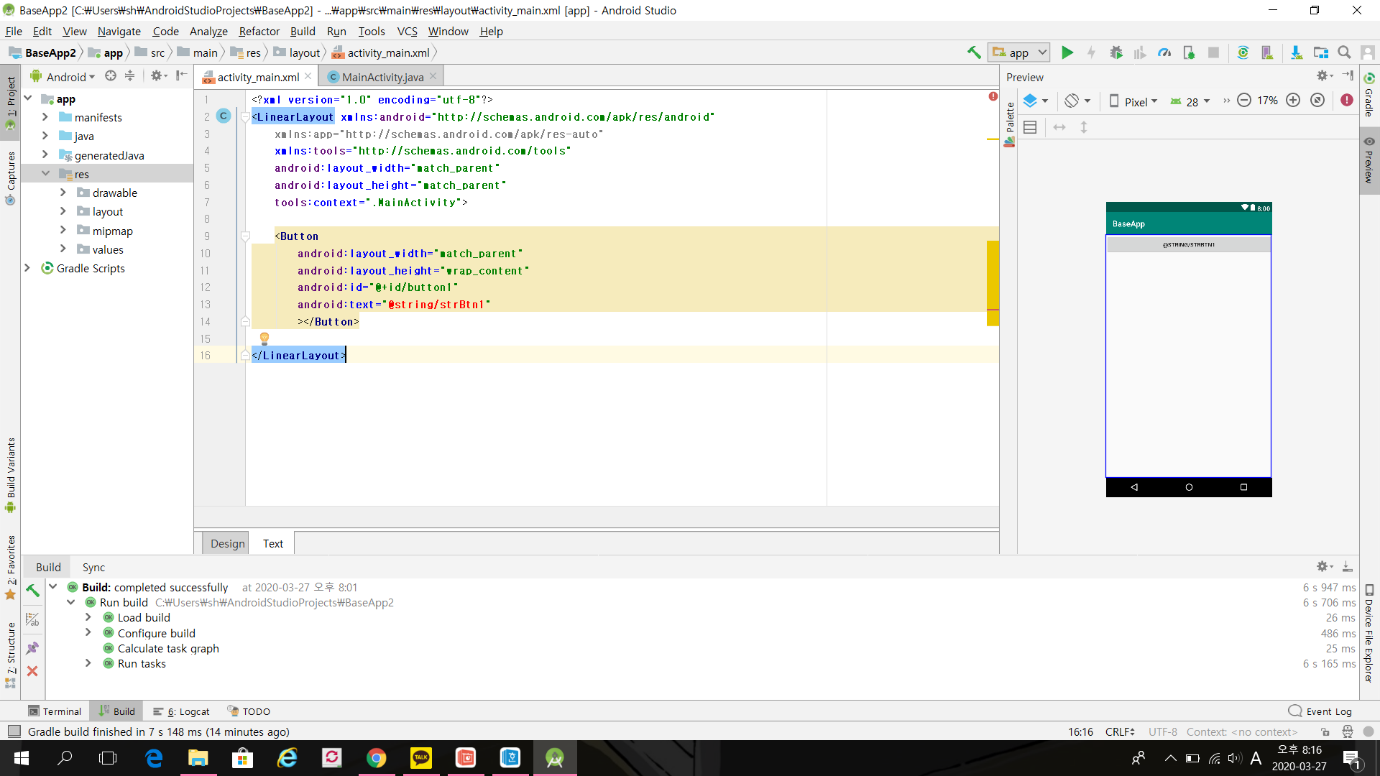
**2018037019 김소현**

**[FourButton 기능 구현을 위한 프로젝트 기본 세팅]**

안드로이드 프로젝트를 만드는 과정을 모두 끝내면 화면 디자인 편집 화면인 java-main activity가 기본적으로 열린다.

LinearLayout으로 바꾸어 주었다. <TextView/>도 있는데 불필요하다고 생각되면 지워야 한다.

Text를 지우고 Button을 만들어 주었고

Button칸에 명령어들을 제대로 입력하면 오른쪽 가상 모바일에 버튼이 생겨난다.

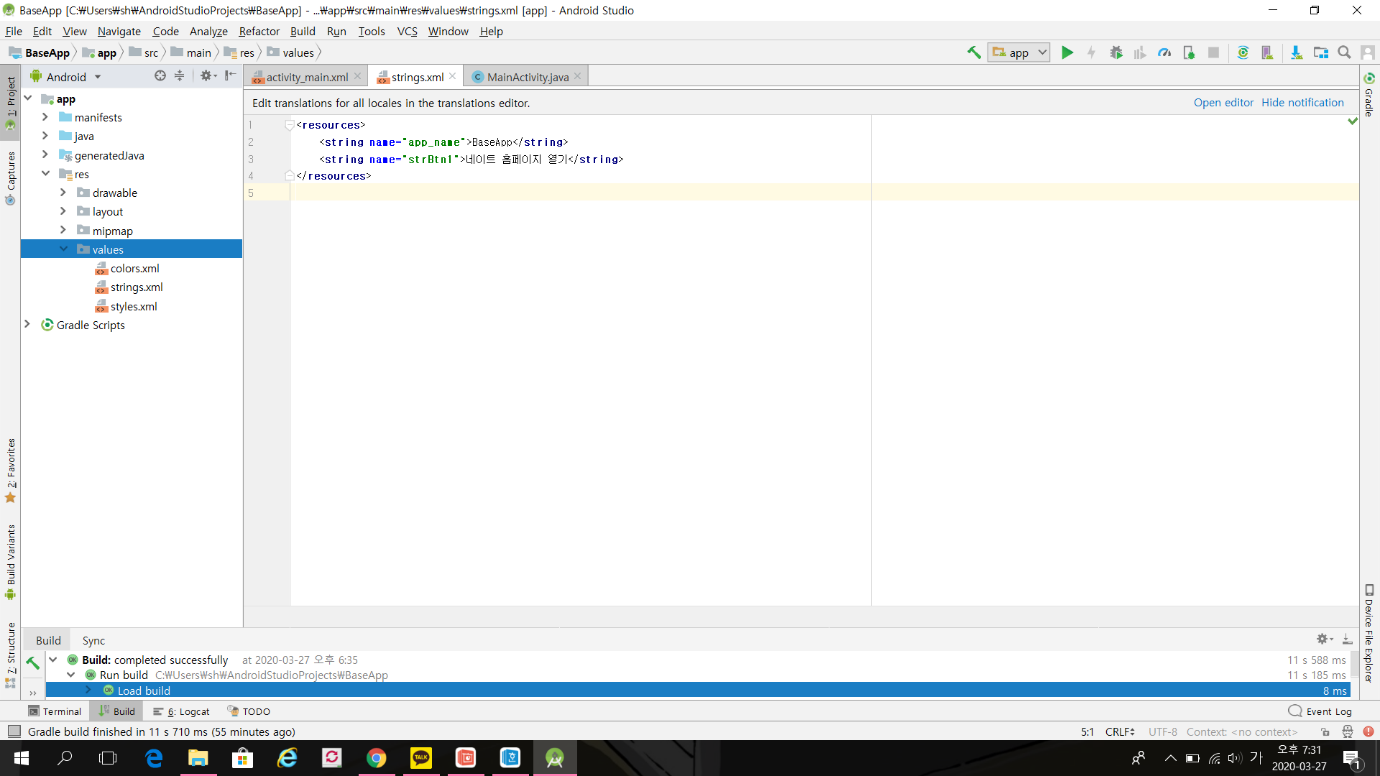
10번째 줄의 코드 의미는 레이아웃의 가로 크기를 설정하는 속성이며 부모 뷰의 내부 여백을 제외한 크기로 설정한다. (match\_parent)

11번째 줄의 코드 의미는 레이아웃의 높이 크기를 설정하는 속성이며 포함된 내용을 보여줄 수 있는 크기로 설정한다. (wrap\_content)

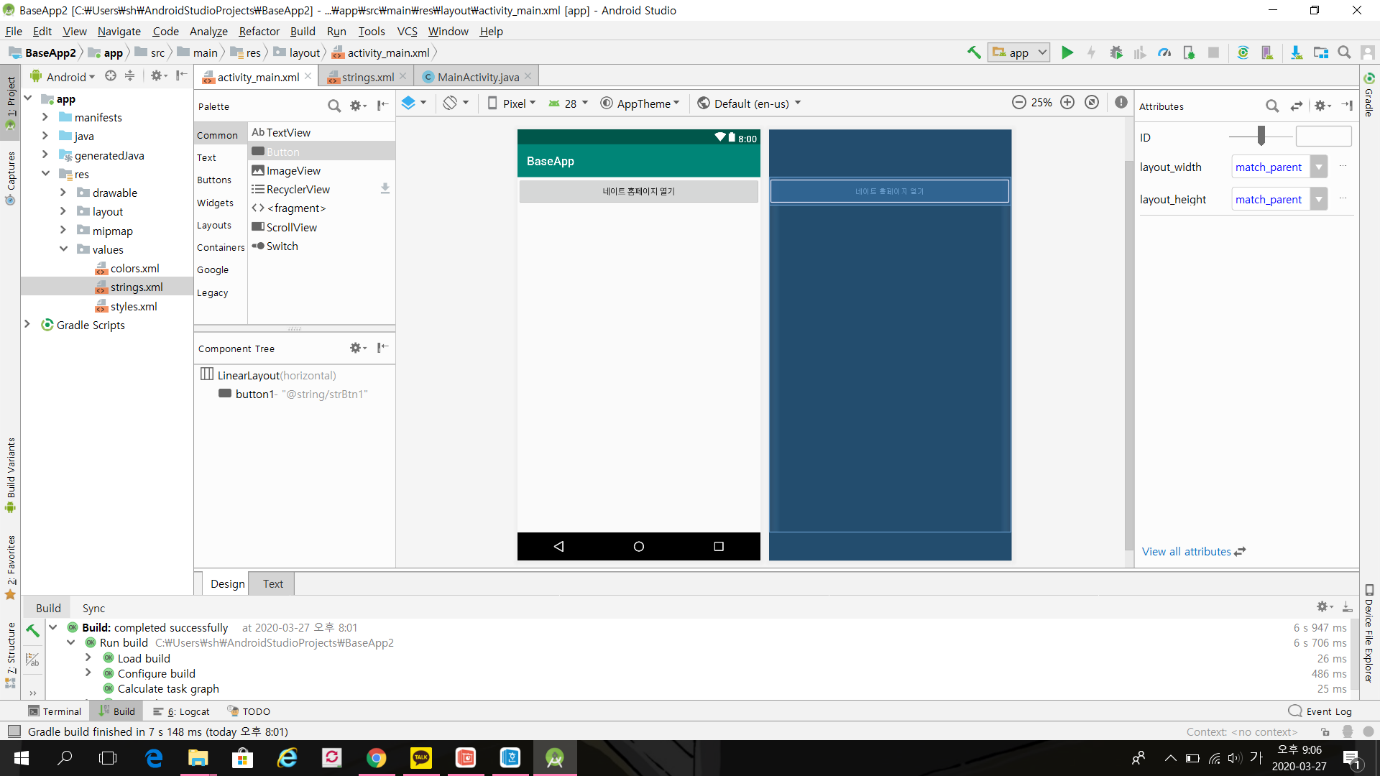
layout\_width, layout\_height는 가로세로의 비율을 맞춘다.

12번째 줄의 코드 의미는 버튼 id를 지정한다. Activity에 Button을 사용하기 위한 첫 번째 단계는 Layout 리소스 XML에 Button을 추가하는 것이다. Activity의 Layout리소스 XML에 Button을 추가할 때 지정한 “id” 속성 값을 사용하여 Button에 대한 참조를 획득할 수 있다. 이렇게 획득한 참조를 가지고 Button의 속성을 변경하거나, 기능을 수행하기 위한 함수를 호출할 수 있다.

13번째 줄의 코드 의미는 strings.xml파일에 해당 문자열 이름으로 문자열이 지정되어 참조하겠다는 의미이다.

string.xml파일에 <string name=”strBtn1”>네이트 홈페이지 열기</string>이라는 코드를 넣었다.

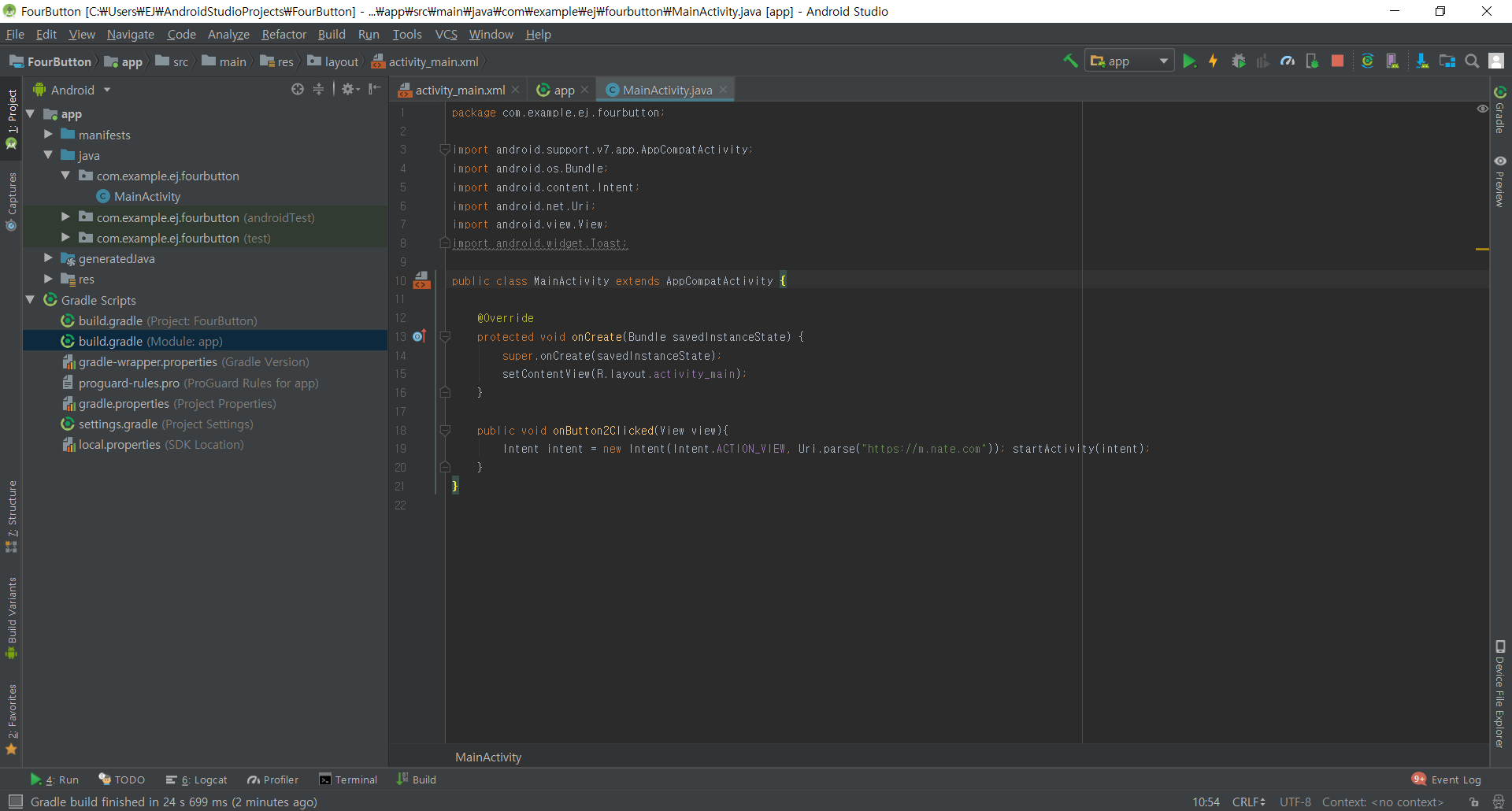
3번째 줄의 의미는 strBtn1이라는 문자열을 추가한 것이다. 문자열을 추가한 후 저장하면 activity\_main.xml에 있는 빨간색으로 표시되는 오류가 사라진다.

activity\_main.xml에서 하단의 Design탭을 클릭하면 그래픽 화면을 확인할 수 있다.

그 다음으로 java파일을 수정할 단계이다.

**2018037015 김은진**

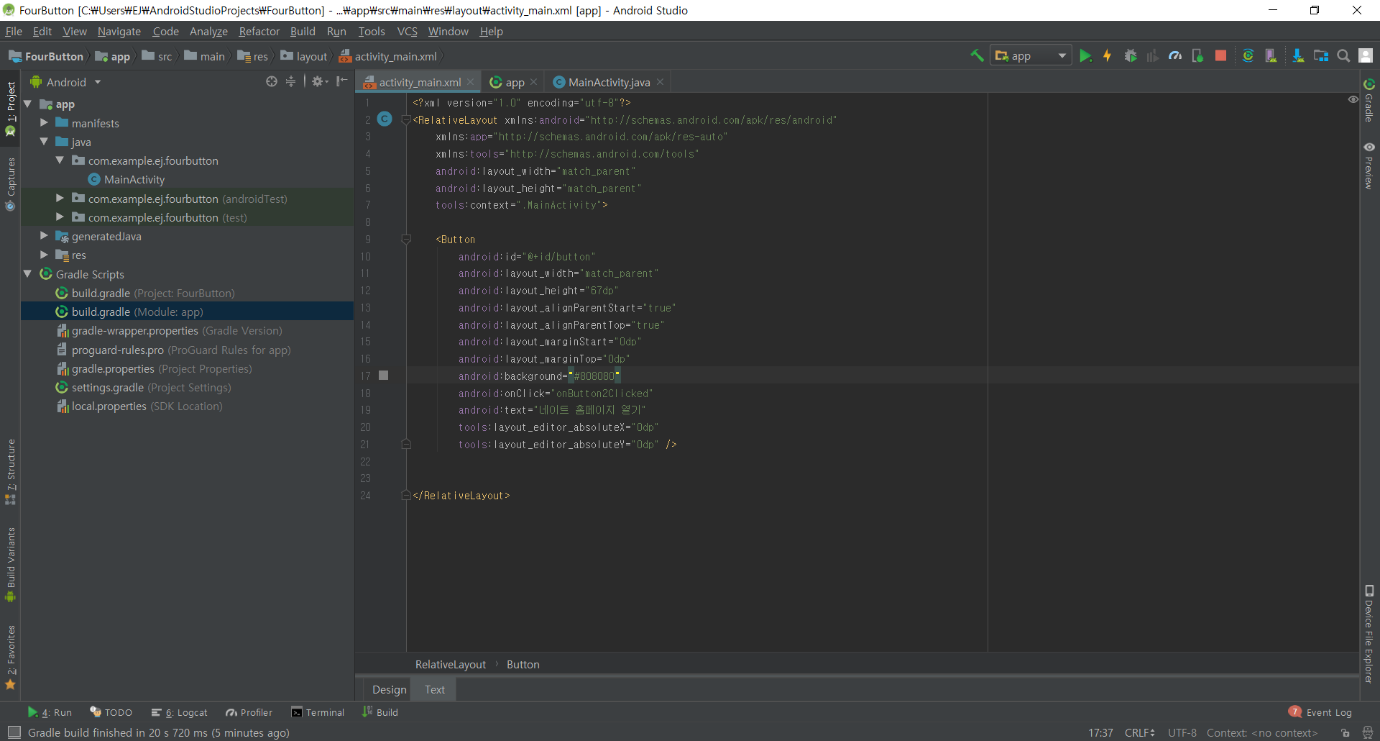
**[첫 번째 버튼\_네이트 홈페이지 열기]**



java 파일에 동작하는 코드를 짰다. 웹페이지를 띄우기 위해 “onButton2clicked”라는 메소드를 정의한다. 버튼을 클릭하면 <https://m.nate.com> URL이 실행된다.

package com.example.ej.fourbutton;  
  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.content.Intent; //intent를 사용하기 위해서 불러온다  
import android.net.Uri; //URL를 위해서 불러온다  
import android.view.View;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 }  
  
 public void onButton2Clicked(View view){ //메소드 정의하기, 아무 이름이나 상관없다  
 Intent intent = new Intent(Intent.*ACTION\_VIEW*, Uri.*parse*("https://m.nate.com")); // 변수 intent를 선언하여 버튼을 클릭하였을 때 실행할 동작을 구현한다.

startActivity(intent); // 실행한다.  
 }  
}

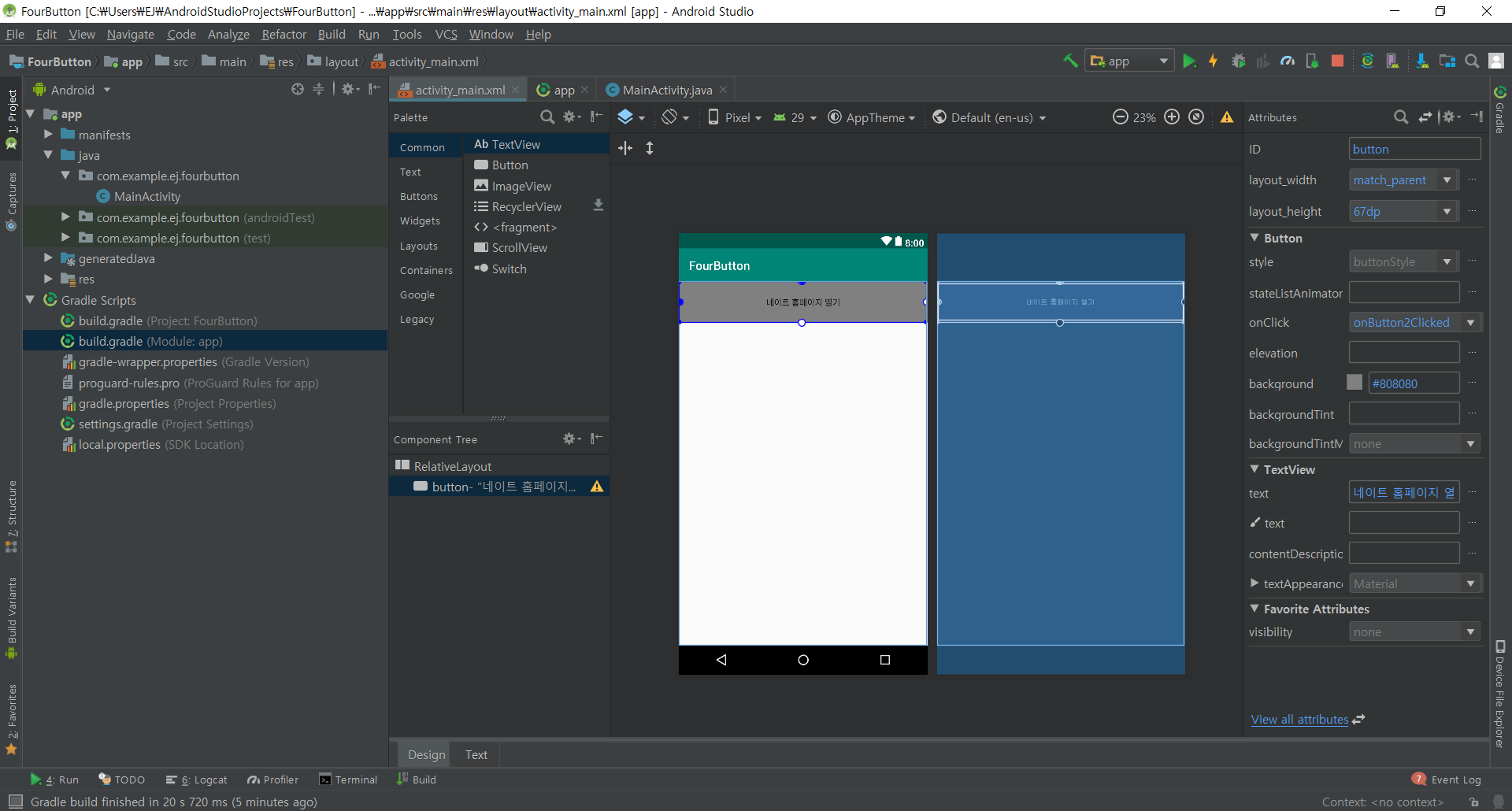


예시와 같은 웹페이지 열기 버튼을 생성하려면 크게 두가지 방법으로 만들 수 있다. 미리보기 화면에서 버튼 아이콘을 드래그해서 만드는 방법과 코드를 추가해서 만드는 방법이 있다. activity\_main.xml에 button을 만들어준다. 그 다음에는 onclick 옵션에 위에서 정의했던onButton2Clicked 라는 이름으로 지정해준다.

Android: ~ 문법에 따라 버튼의 기능을 구현할 수 있다. 그 중에서도 가장 중요한 코드는android:background=”#808080”, android:onclick=”onButton2clicked” , android:text=”네이트온 홈페이지 열기” 이다. background color와 text는 미리보기 화면에서도 변경 가능하다.

예시에 나와 있듯이 진회색의 컬러 코드를 입력하였다.

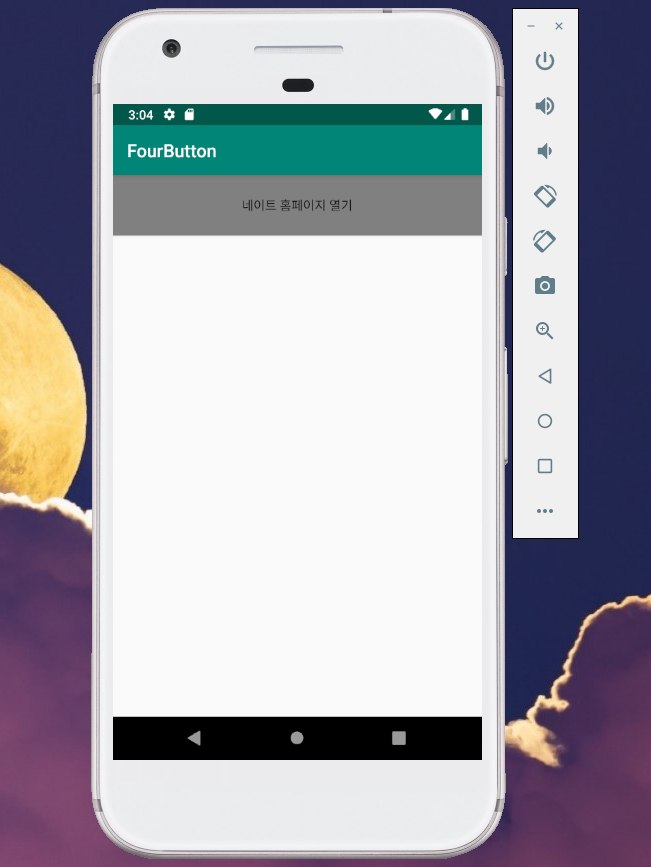
<Button  
 android:id="@+id/button" //id 지정  
 android:layout\_width="match\_parent" //layout의 넓이 지정  
 android:layout\_height="67dp" //높이지정  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"   
 android:layout\_marginStart="0dp"  
 android:layout\_marginTop="0dp" //margintop 을 0으로 지정, 위에 margin이 없어서 최상 단 노출  
 android:background="#808080" //button의 background color 지정  
 android:onClick="onButton2Clicked" //메소드 이름 지정  
 android:text="네이트 홈페이지 열기"//버튼에 들어갈 text 입력  
 tools:layout\_editor\_absoluteX="0dp"  
 tools:layout\_editor\_absoluteY="0dp" />



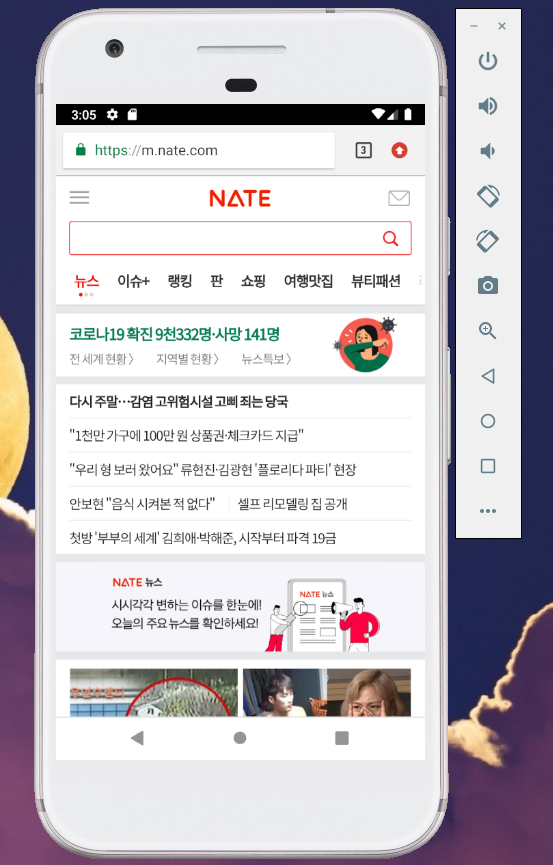
위에 코드대로 입력을 한 후 Design 탭을 누르면 내가 설정한 값 대로 미리보기 화면이 뜬다. “네이트 홈페이지 열기” 버튼이 생성된 것을 볼 수 있다. 버튼을 누르면 오른쪽에 설정 값들이 뜬다. 버튼의 background-color나, text등의 내용들을 오른쪽 상세보기에서 바꿀 수 있다.



Run “app”으로 실행하면 앱이 만들어진 것을 볼 수 있다. 내가 설정한 프로젝트 이름으로 앱이 만들어졌다.



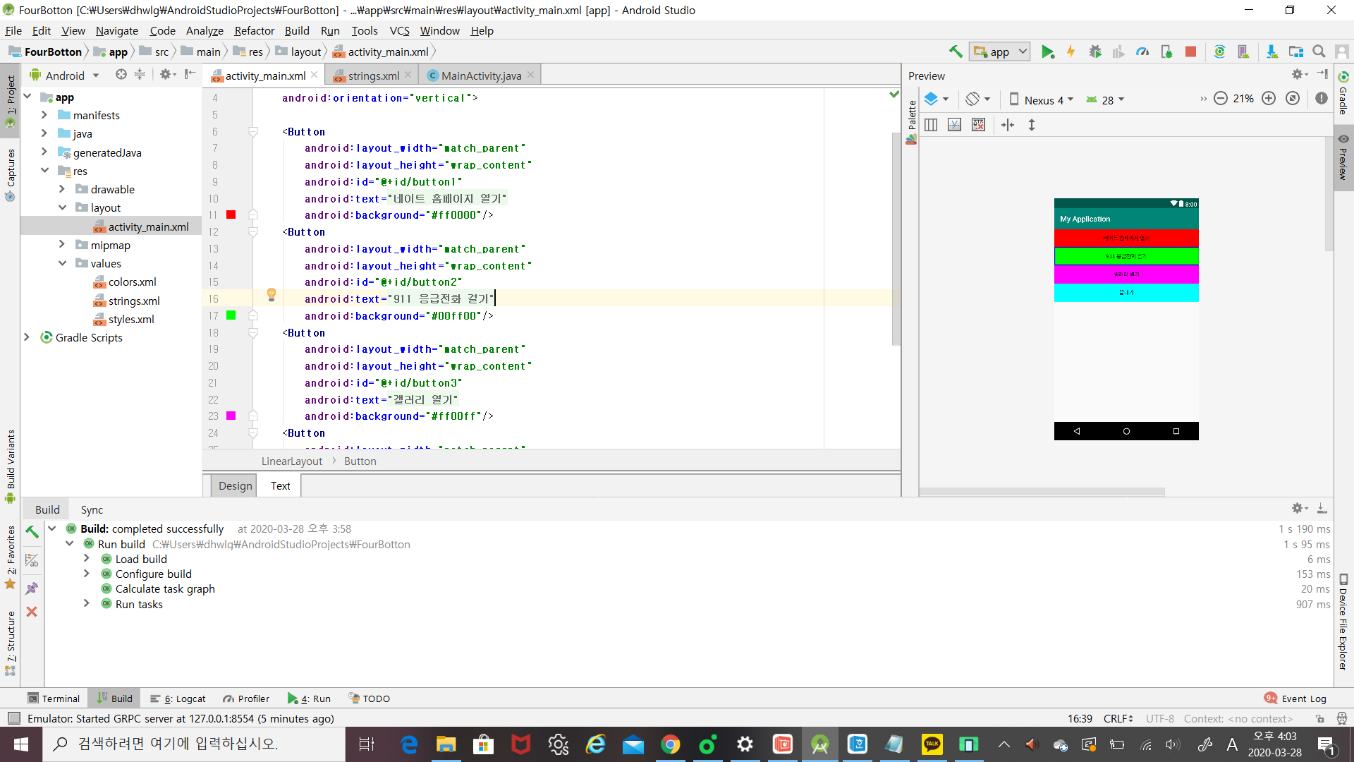
FourButton 앱을 클릭하면 미리보기에서 보았던 화면이 나온다. 버튼의 색깔이 내가 설정한 색깔임을 확인할 수 있다. 또한 버튼의 이름도 바뀌어서 정상적으로 적용되어 있는 모습을 확인할 수 있다.



버튼을 클릭하면 내가 설정한 URL로 연결되는 홈페이지가 나온다. 모바일 네이트 홈페이지가 켜진 모습을 확인할 수 있다.

**2018037006 오지현**

**[두 번째 버튼\_911 전화 걸기]**

**activity\_main.xml )**

<**LinearLayout xmlns:android=**[**http://schemas.android.com/apk/res/android**](http://schemas.android.com/apk/res/android)

**android:layout\_width="match\_parent"**

**android:layout\_height="match\_parent"**

**android:orientation="vertical"**>

<**Button**

**android:layout\_width="match\_parent"**

**android:layout\_height="wrap\_content"**

**android:id="@+id/button1"**

**android:text="@string/strBtn1"**

**android:background="#ff0000"**/>

<**Button**

**android:layout\_width="match\_parent"**

**android:layout\_height="wrap\_content“ //911버튼을 실행하기위한부분**

**android:id="@+id/button2"**

**android:text="@string/strBtn2"**

**android:background="#00ff00"**/>

<**Button**

**android:layout\_width="match\_parent"**

**android:layout\_height="wrap\_content"**

**android:id="@+id/button3"**

**android:text="@string/strBtn3"**

**android:background="#ff00ff"**/>

<**Button**

**android:layout\_width="match\_parent"**

**android:layout\_height="wrap\_content"**

**android:id="@+id/button4"**

**android:text="@string/strBtn4"**

**android:background="#00ffff"**/>

</**LinearLayout**>

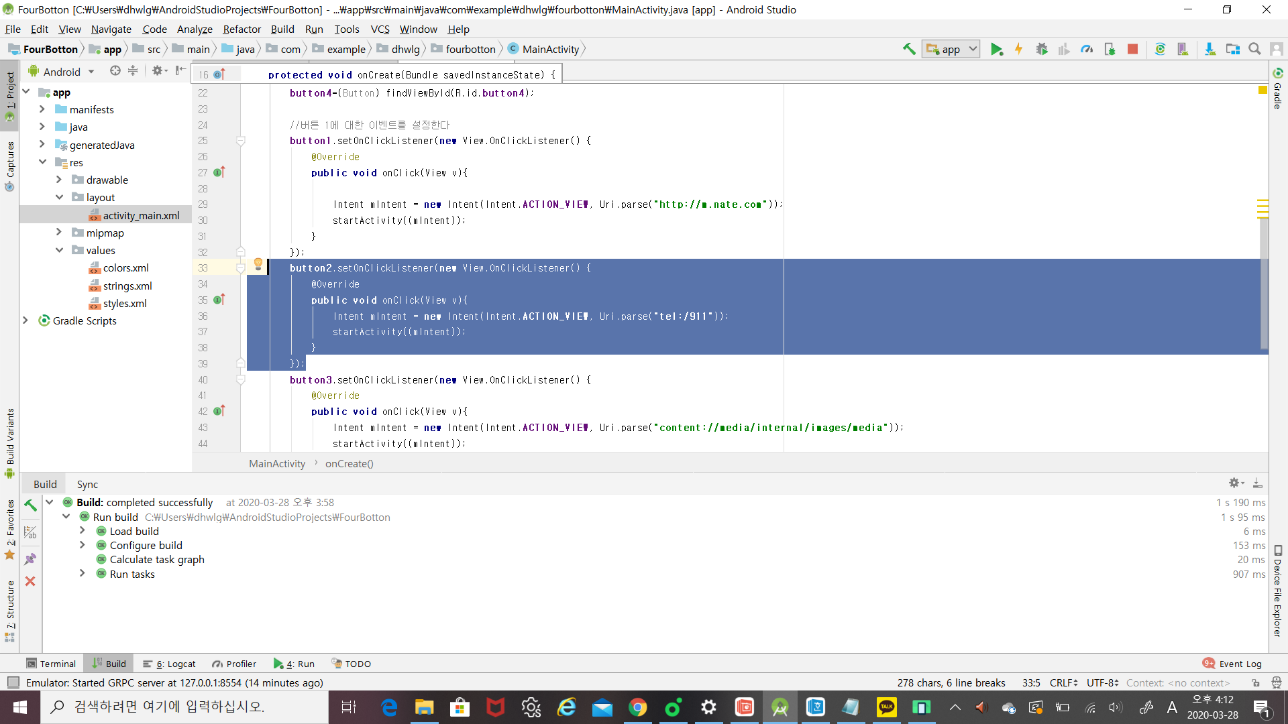
**strings.xml )**

<**resources**>

<**string name="app\_name"**>My Application</**string**>

<**string name="strBtn2"**>911 응급전화 걸기</**string**> // 버튼2를 위한 소스

**<resources**>

**MainActivity.java )**

**button2**.setOnClickListener(**new** View.OnClickListener() {

@Override**public void** onClick(View v){

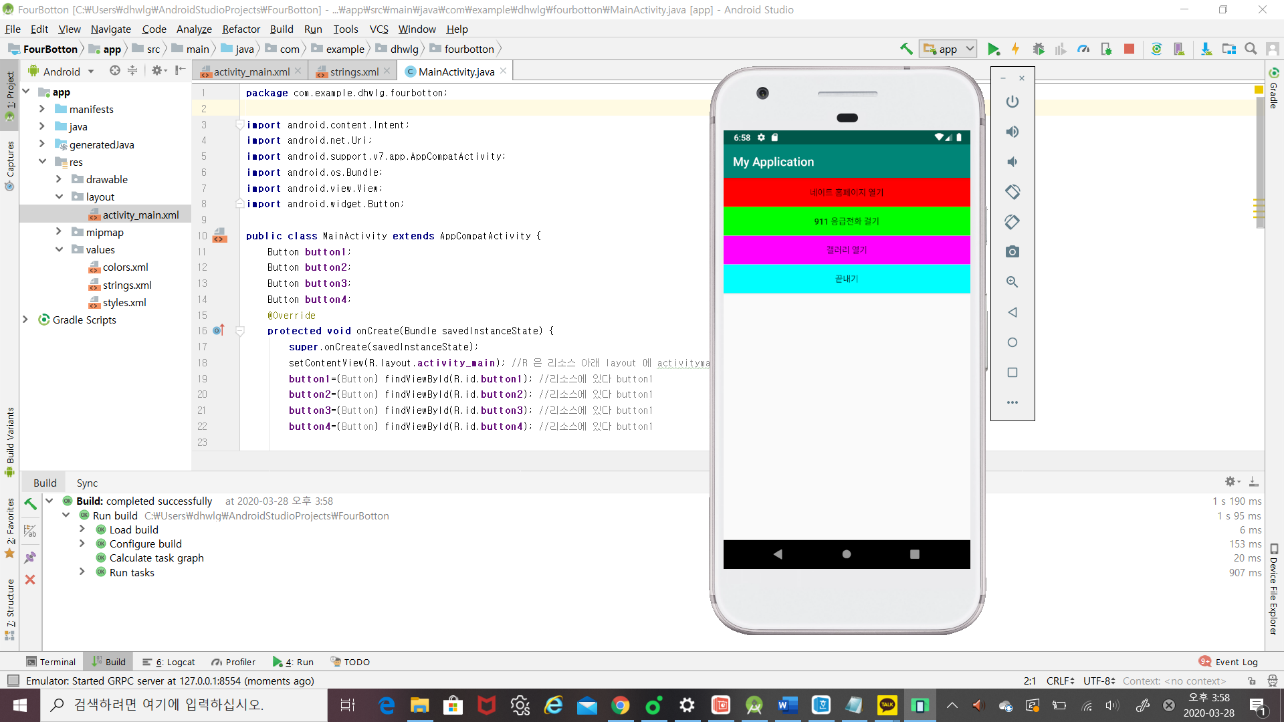
Intent mIntent = **new** Intent(Intent.***ACTION\_VIEW***, Uri.*parse*(**"tel:/911"**));

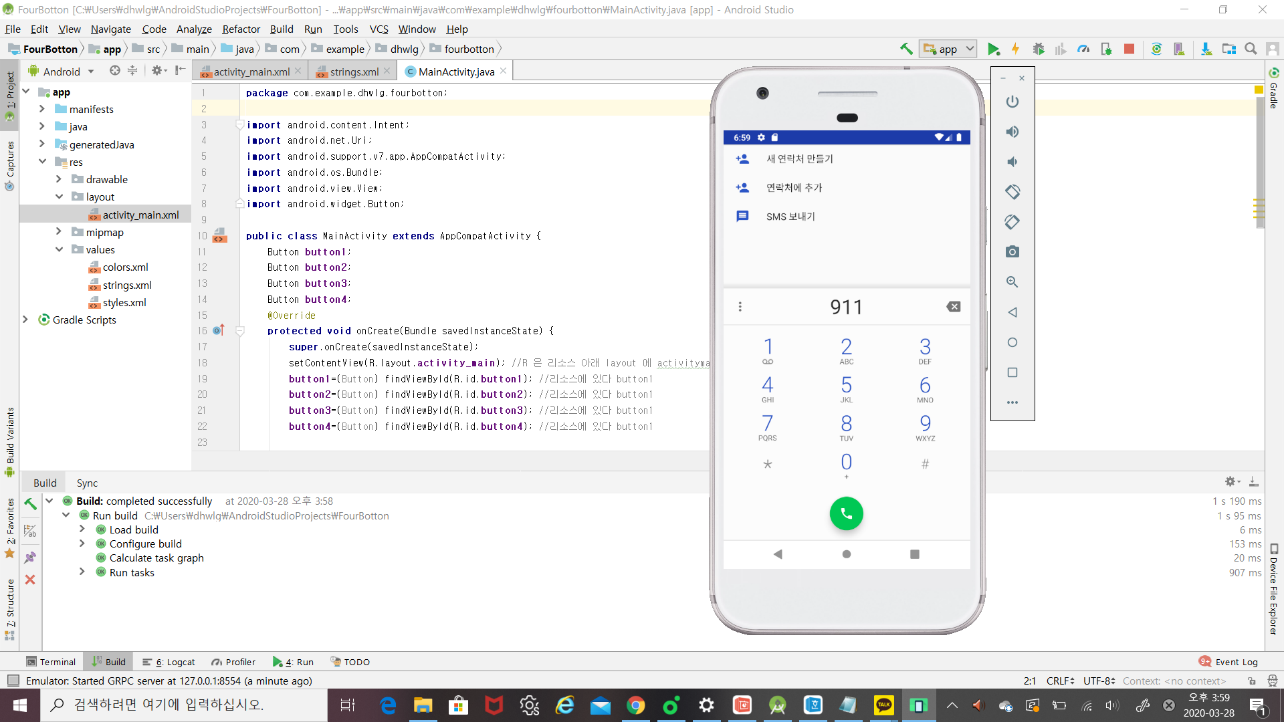
startActivity((mIntent));

}

});

//tel : 은 911번호 전화번호를 세팅하여 통화버튼을 누르면 바로 통화가 가능하다.

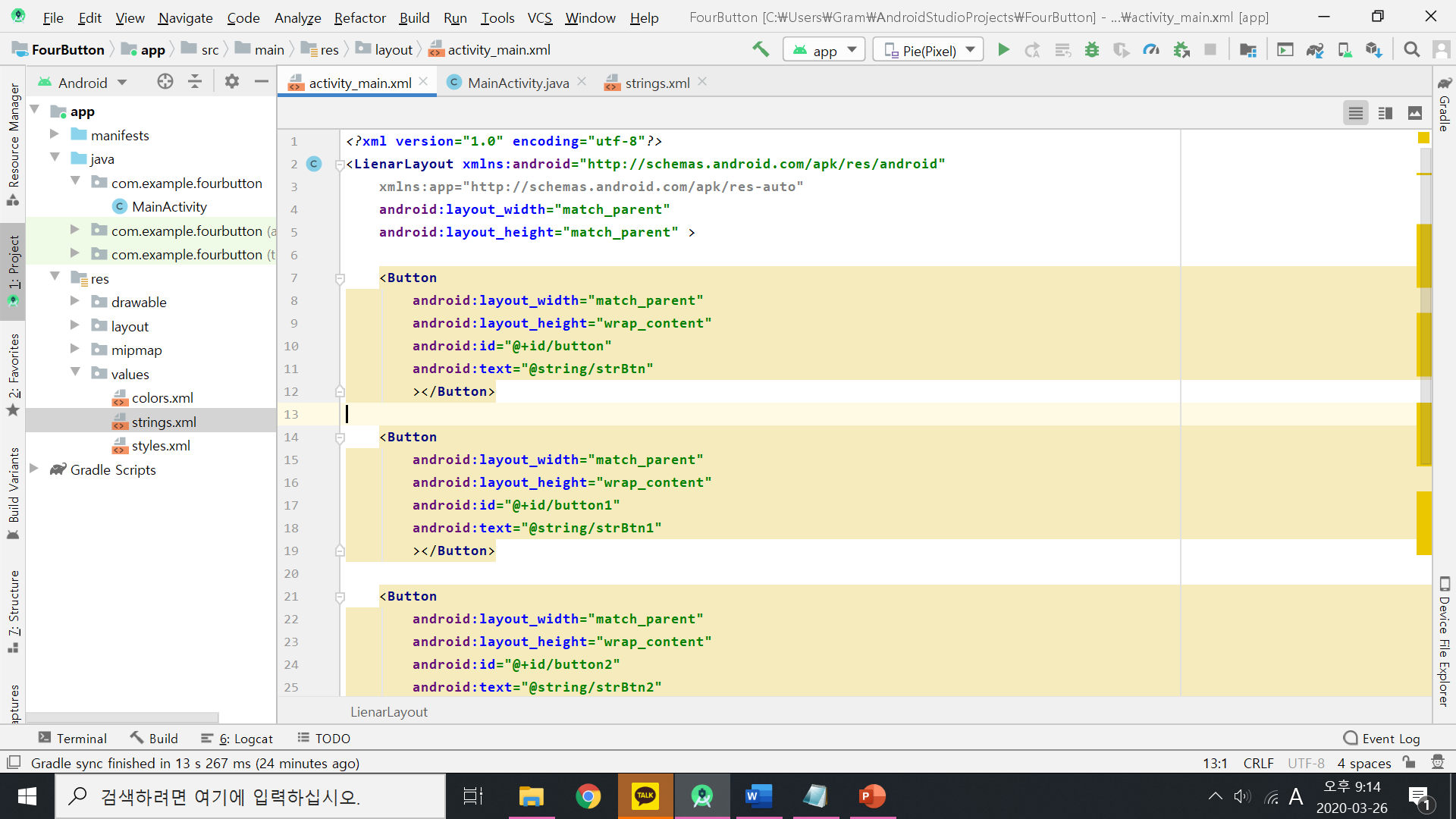
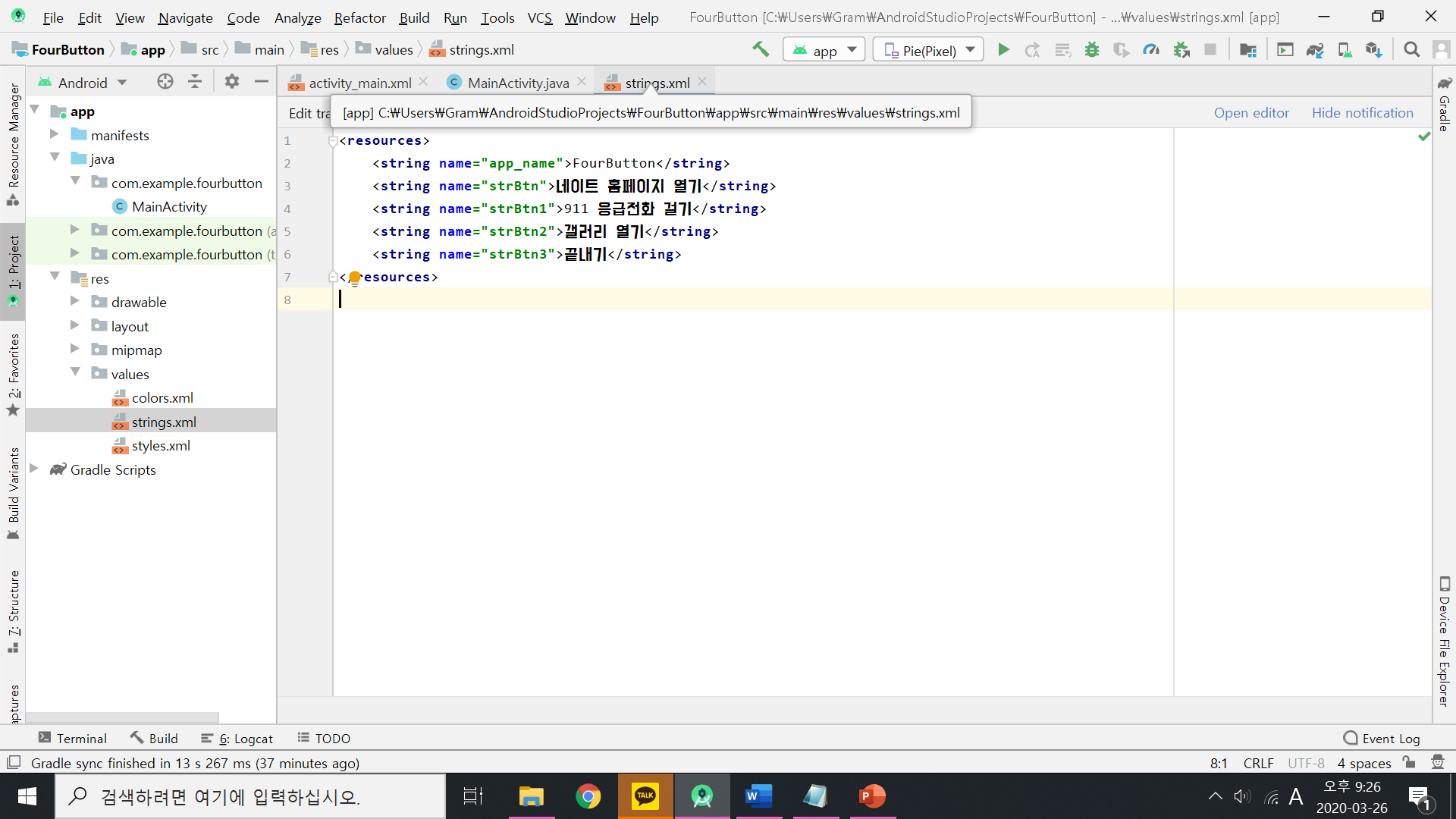


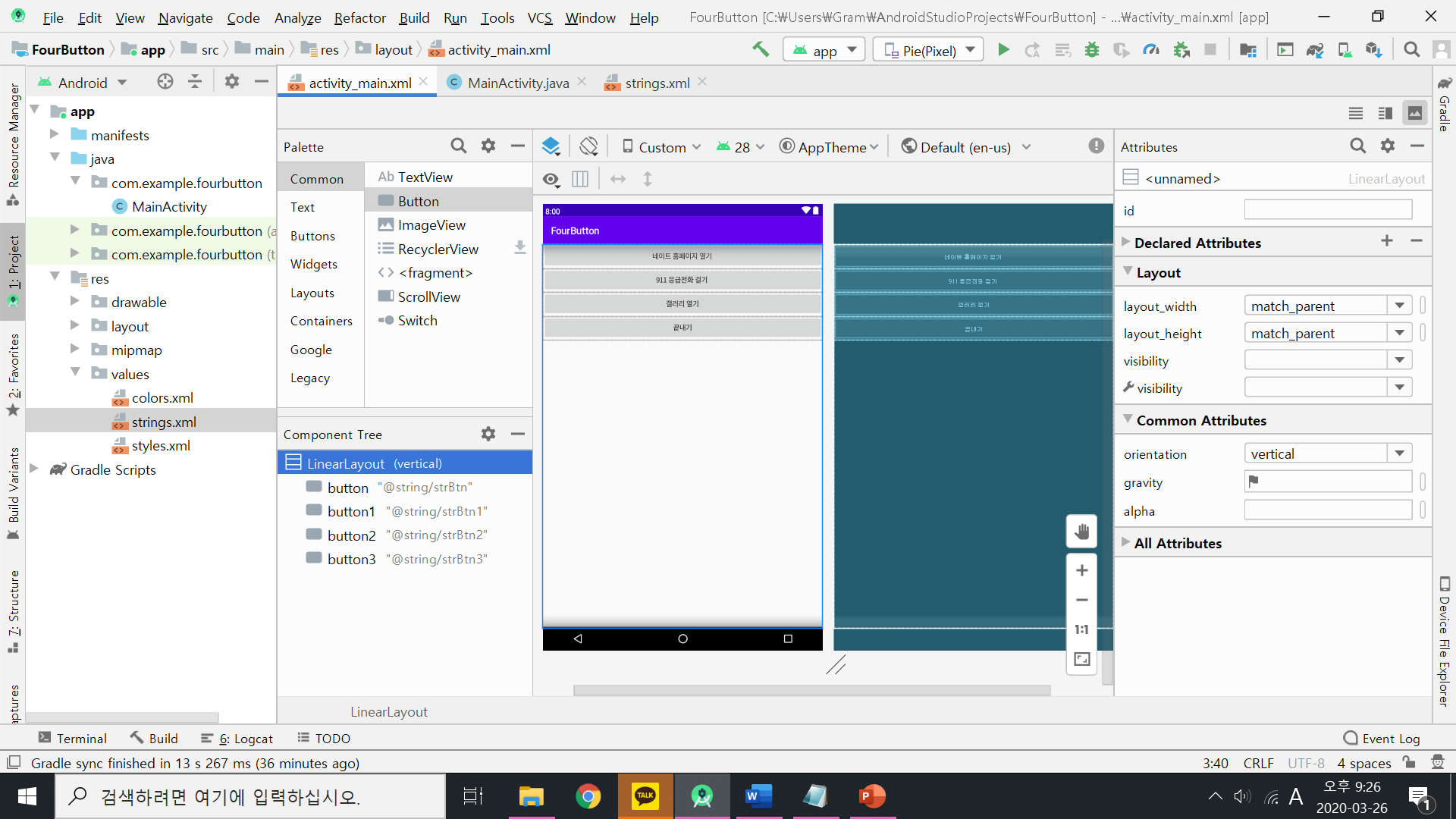
Run하면 위처럼 4개의 버튼이 나온다.

그 중 2번째 버튼인 911 응급전화 걸기를 누르면 위처럼 실행된다.

**2018037007 김효희**

**[세 번째 버튼\_갤러리 열기]**

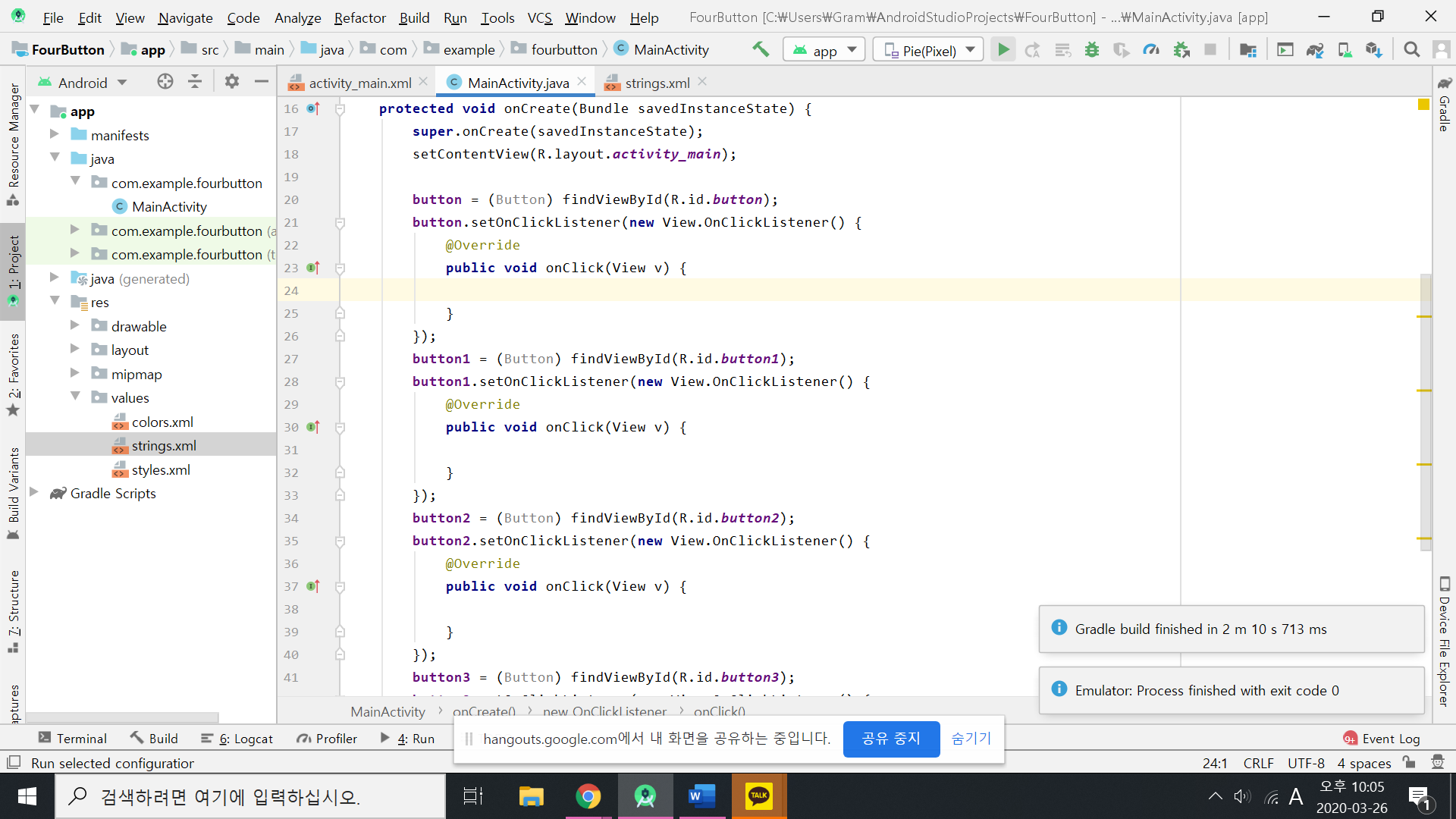
 



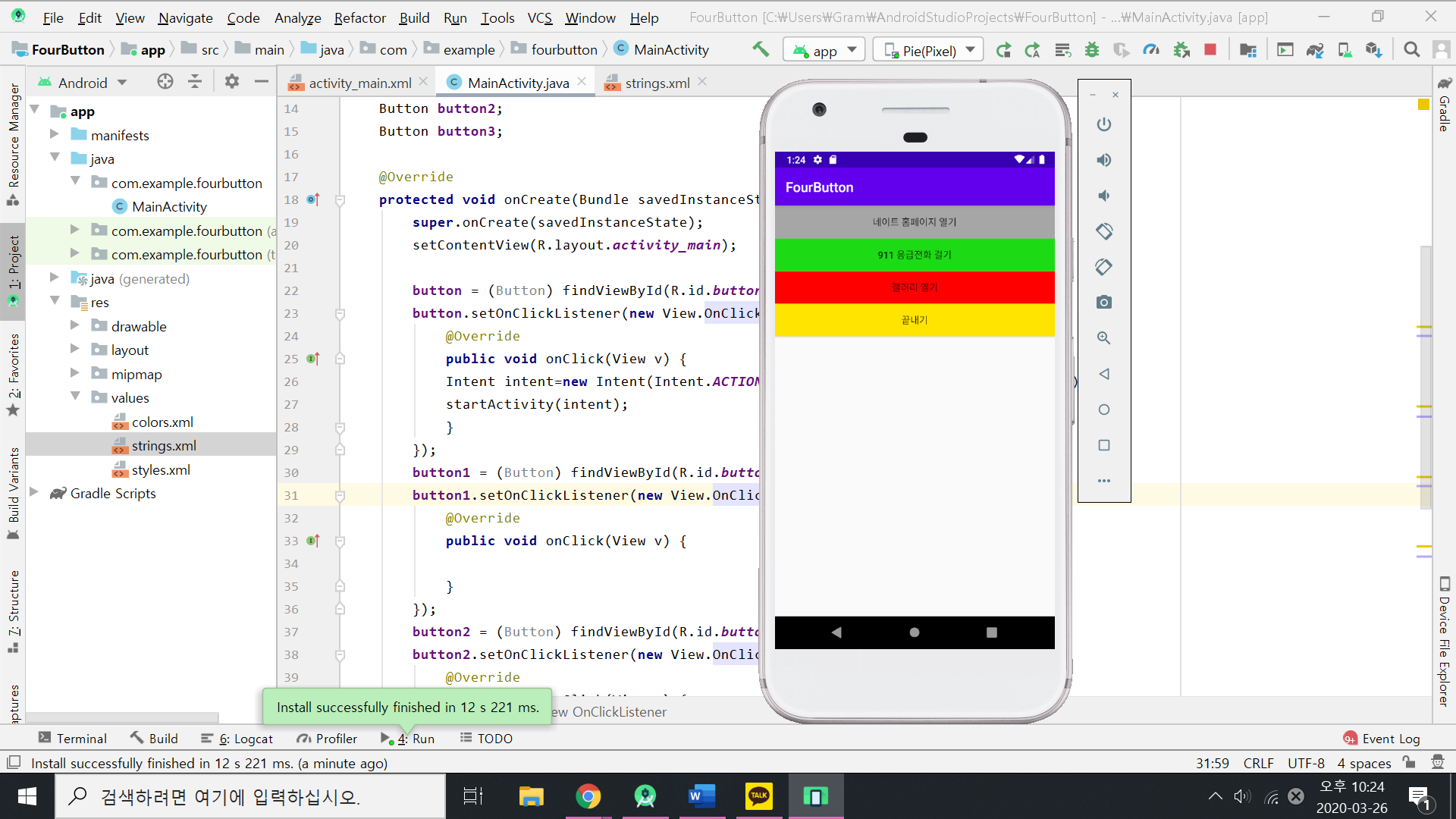
버튼의 기능을 구현하기 전에 버튼을 먼저 만들어준다. 위의 코드는 기본 버튼을 만들기 위한 것이다. 코드를 저장하고 strings.xml에 버튼의 이름을 써준다. (갤러리 열기) 디자인 화면에서 버튼의 이름이 제대로 생성된 것을 볼 수 있다.

<**Button  
 android:id="@+id/button2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="#FF0000"  
 android:text="@string/strBtn2"**></**Button**>  
‘갤러리 열기’ 버튼의 색을 빨간색으로 지정해주기 위해서

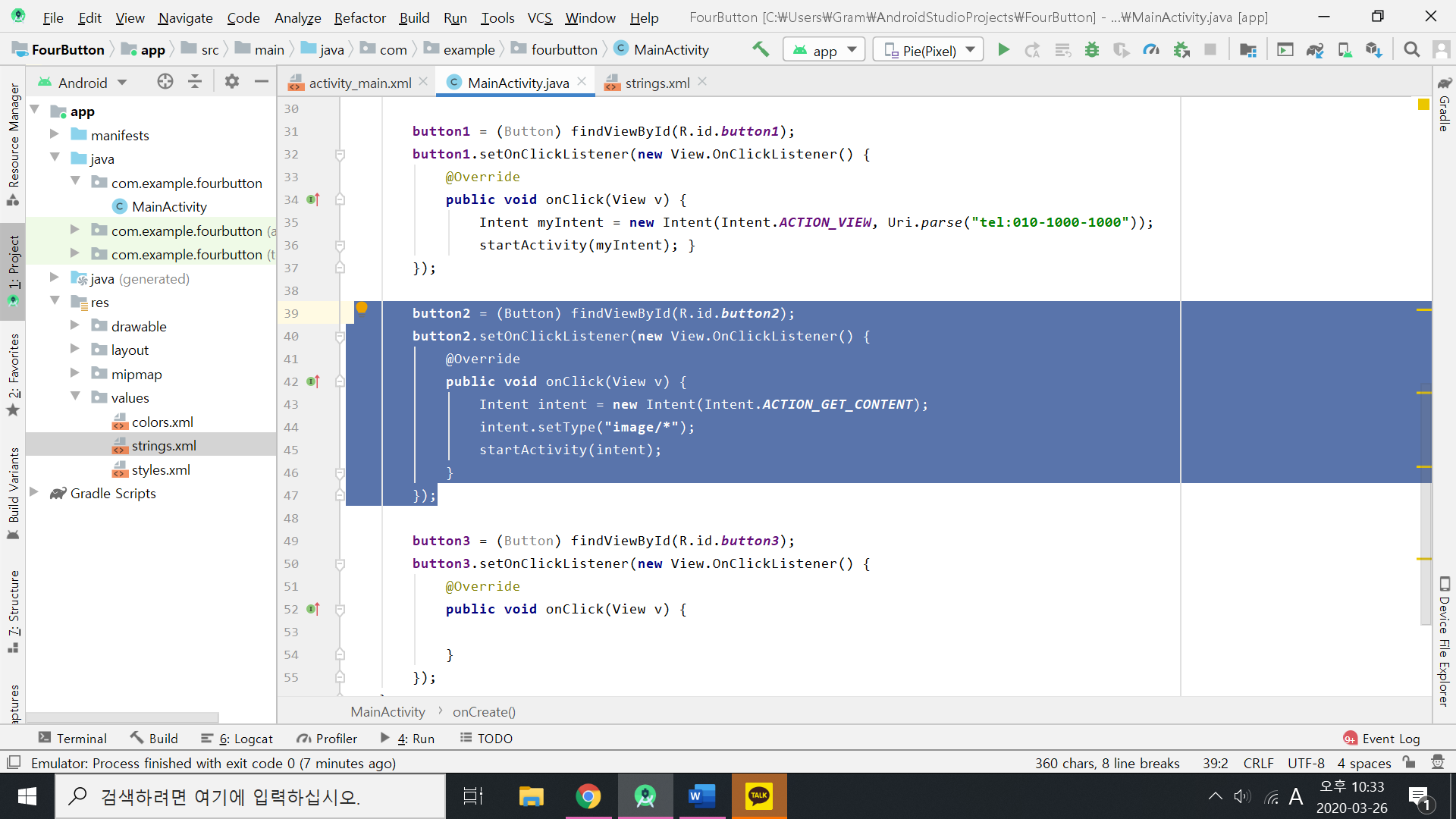
activity\_main.xml에서 android:background를 색상코드 “FF0000”으로 한다.



화면 상단의 Run ‘app’ 초록색 버튼을 눌러 디바이스를 실행시킨다.



실행시킨 후에 Mainactivity.java창에 들어간다.



세 번째 버튼의 ‘갤러리 열기’를 수행하기 위해서 아래 코드를 추가해주었다.

세 번째 버튼의 이름은 button2 이므로 public void onClick(View v) {…} 사이에 아래의 기능 수행을 위한 코드를 추가한다.

Button2를 클릭하면 새로운 창이 열리게 되고, 파일 형식의 시작이 image/…인 것들만, 즉 갤러리를 불러오게 된다.

**activity\_main.xml )**

|  |
| --- |
| <Button |
|  | android:id="@+id/button2" |
|  | android:layout\_width="match\_parent" |
|  | android:layout\_height="wrap\_content" |
|  | android:background="#FF0000" |
|  | android:text="@string/strBtn2"></Button> |
|  |  |

**Mainactivity.java )**

**button2** = (Button) findViewById(R.id.***button2***);  
**button2**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 Intent intent = **new** Intent(Intent.***ACTION\_GET\_CONTENT***);  
 intent.setType(**"image/\*"**);  
 startActivity(intent);  
 }  
});

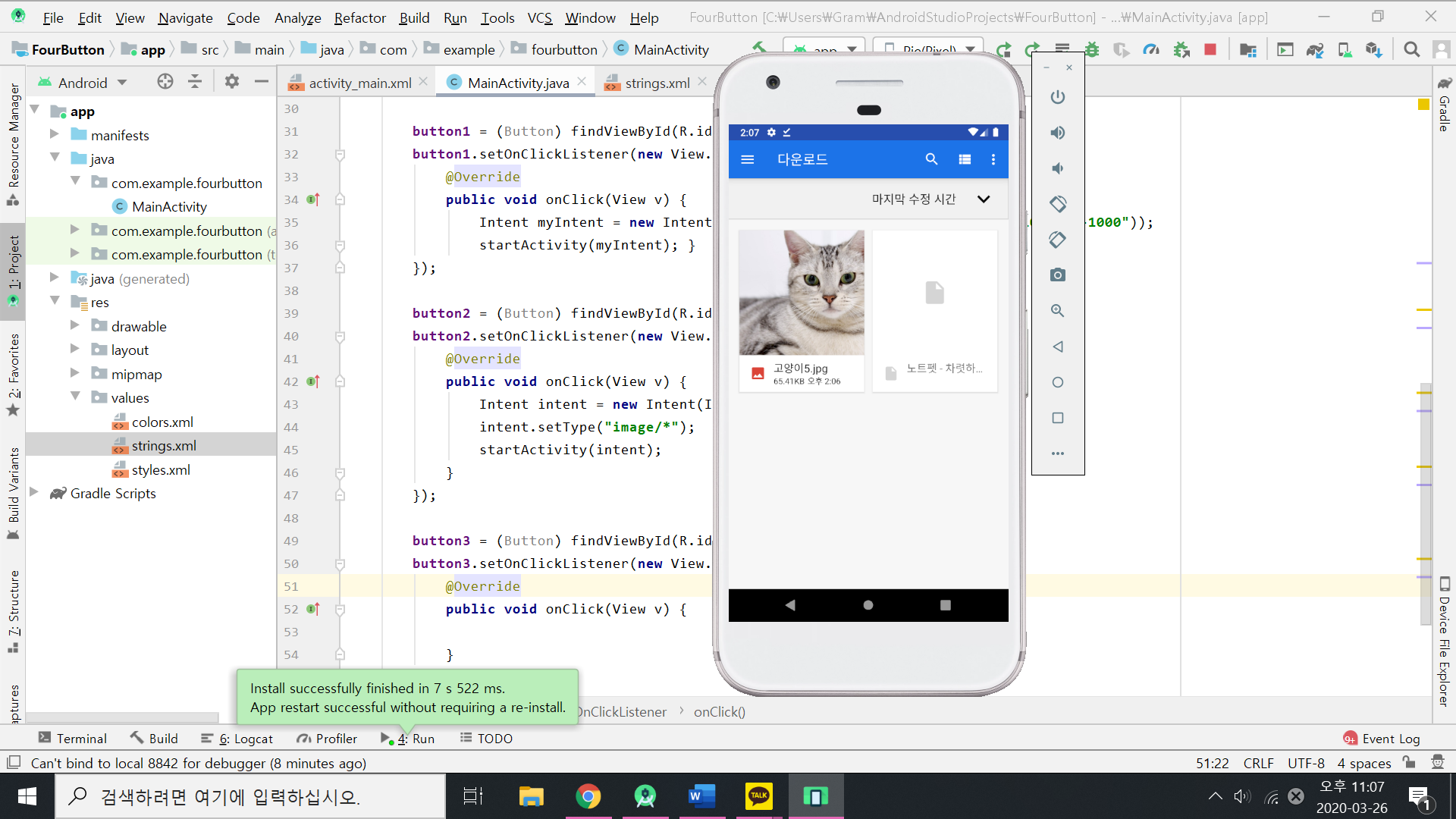
// **Mainactivity.java**의 코드에서 Intent를 통하여 안드로이드에서 원래 열려 있던 화면을 다른 화면으로 전환할 수 있다. (Intent 이용하여 activity전환) Intent는 activity와 같은 전환이 일어날 때 메시지를 전달한다.

// startActivity(…)는 액티비티를 띄우기 위해 사용하는 메서드이다.

**strings.xml )**

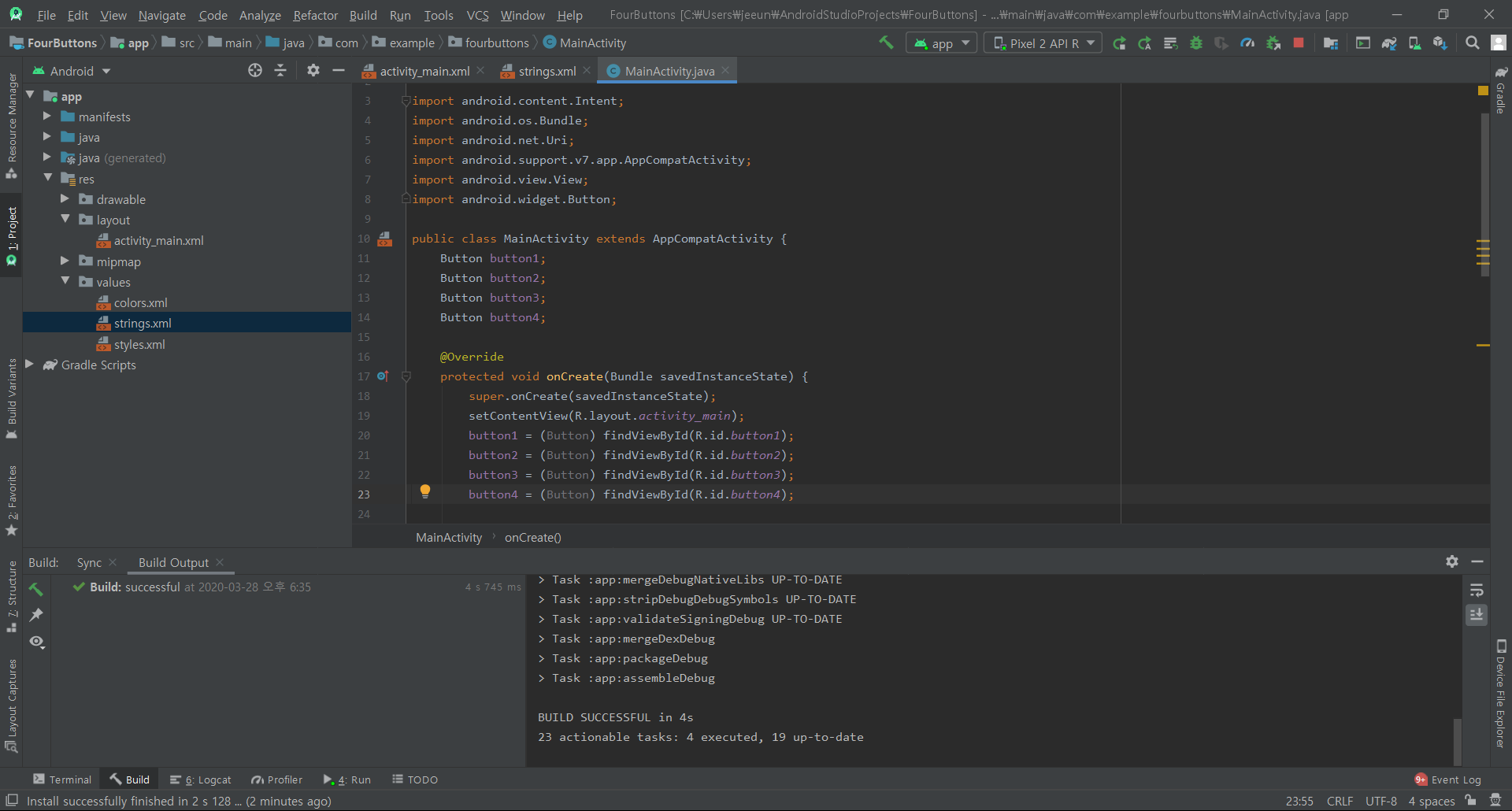
<string name="strBtn2">갤러리 열기</string>

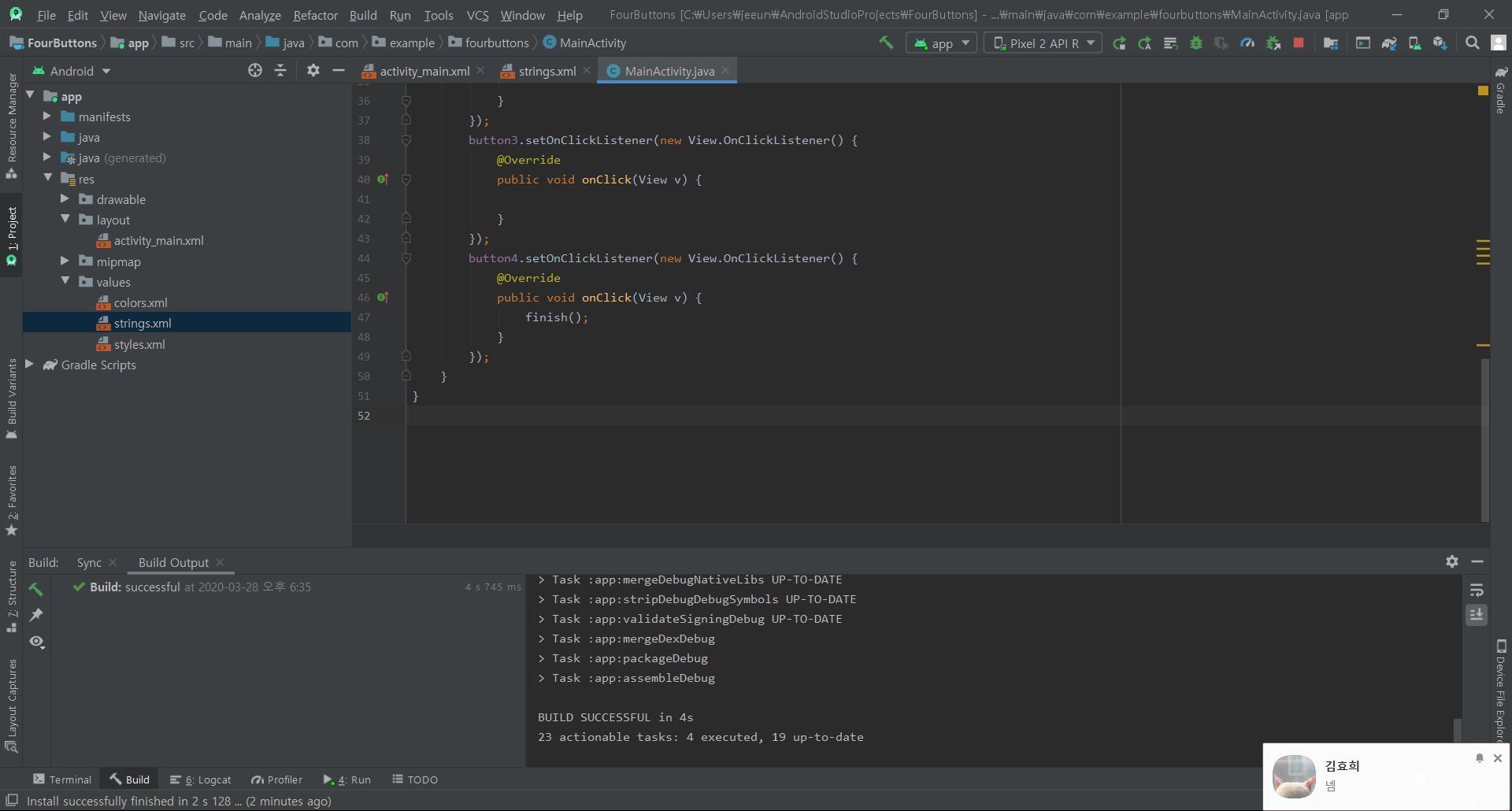
//strings.xml의 코드에 위의 한 줄을 작성하면 버튼에 대한 에러 메시지가 사라진다.

 갤러리가 열리고나서 미리 다운로드 받아 놓은 이미지를 불러온 모습을 볼 수 있다.

**2018037013 이제은**

**[네 번째 버튼\_끝내기]**





**MainActivity.java )**

button4.setOnClickListener(new View.OnClickListener() {

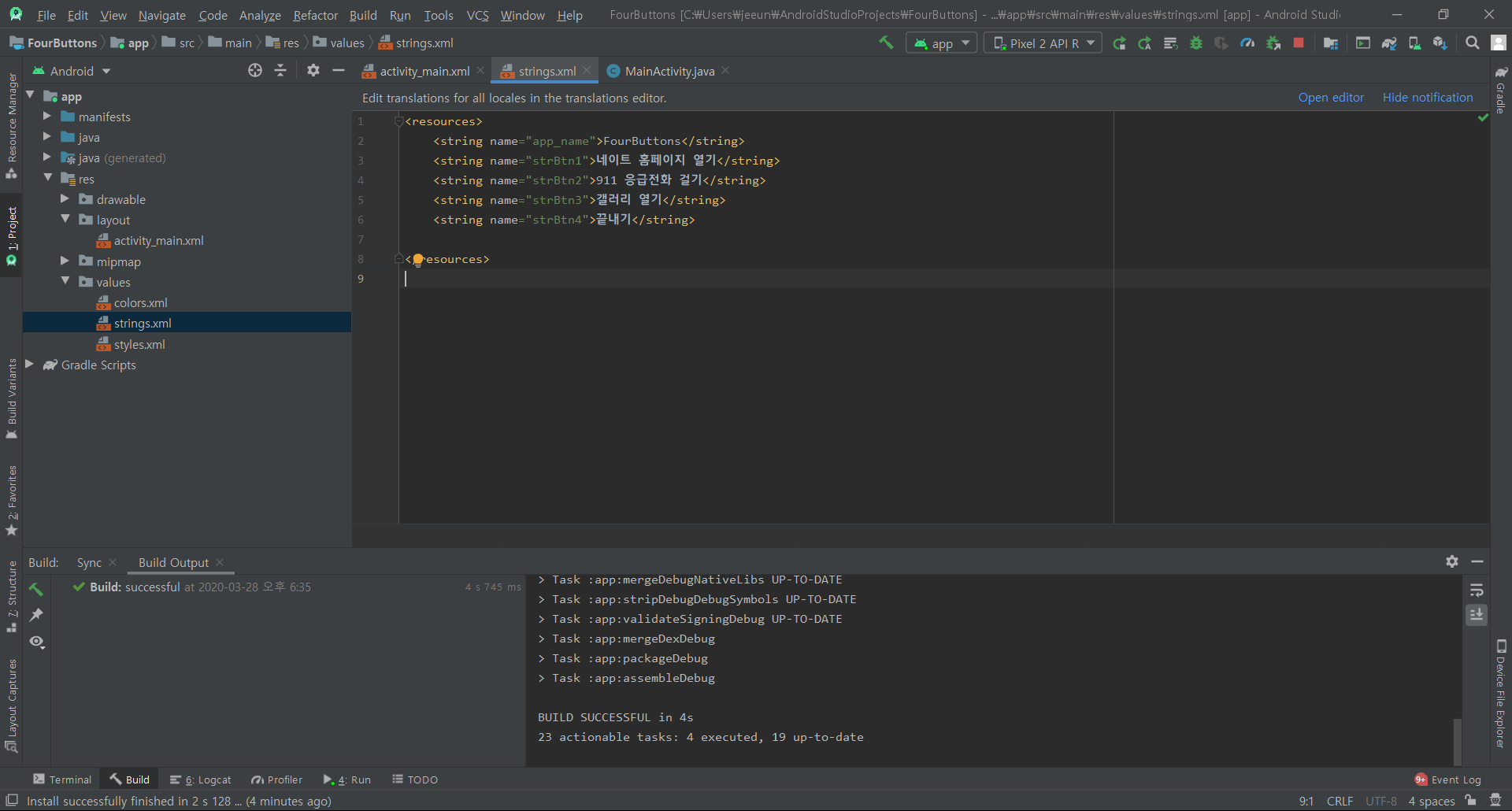
@Override

public void onClick(View v) {

finish();

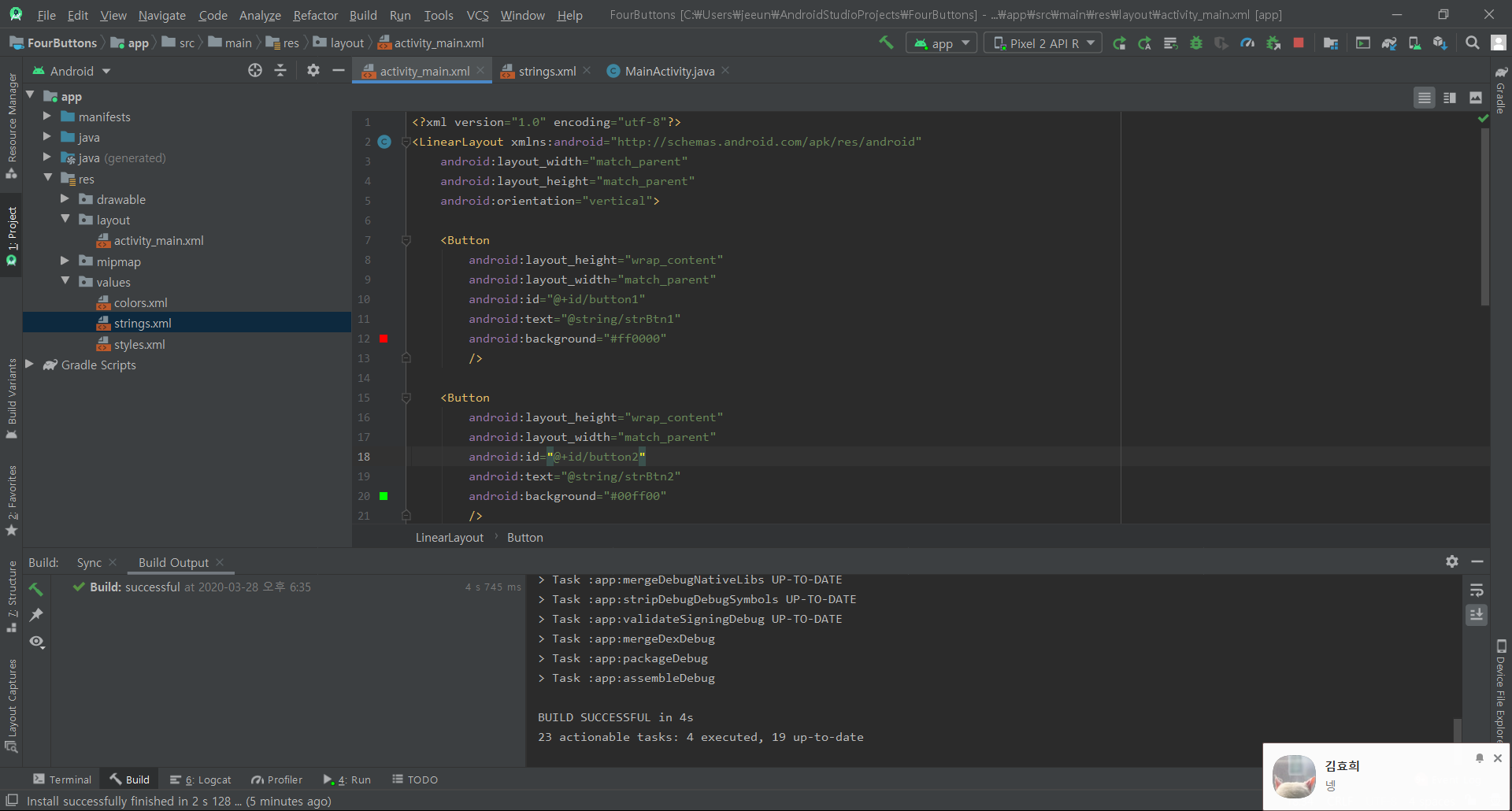
}

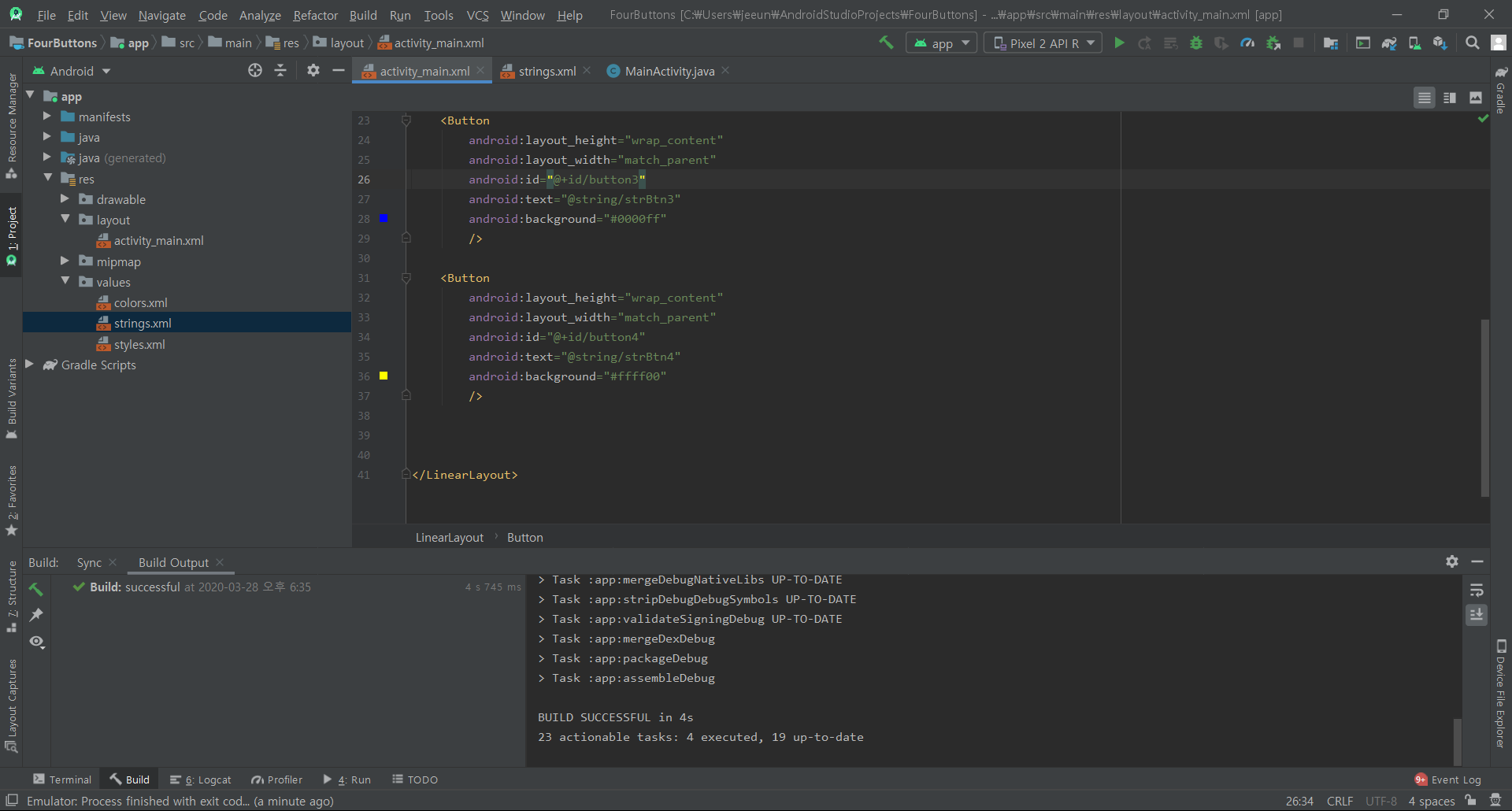
}); // 끝내기 버튼에 대한 코드를 추가 finish()라는 함수 이용



**stirngs.xml )**

각 버튼 별 이름을 지정해주고 string으로 변수 선언





**activity\_main.xml )**

각 변수별로 id, 버튼 명, 배경 색상을 코드로 지정해주었다.

<Button

android:layout\_height="wrap\_content"

android:layout\_width="match\_parent"

android:id="@+id/button4"

android:text="@string/strBtn4"

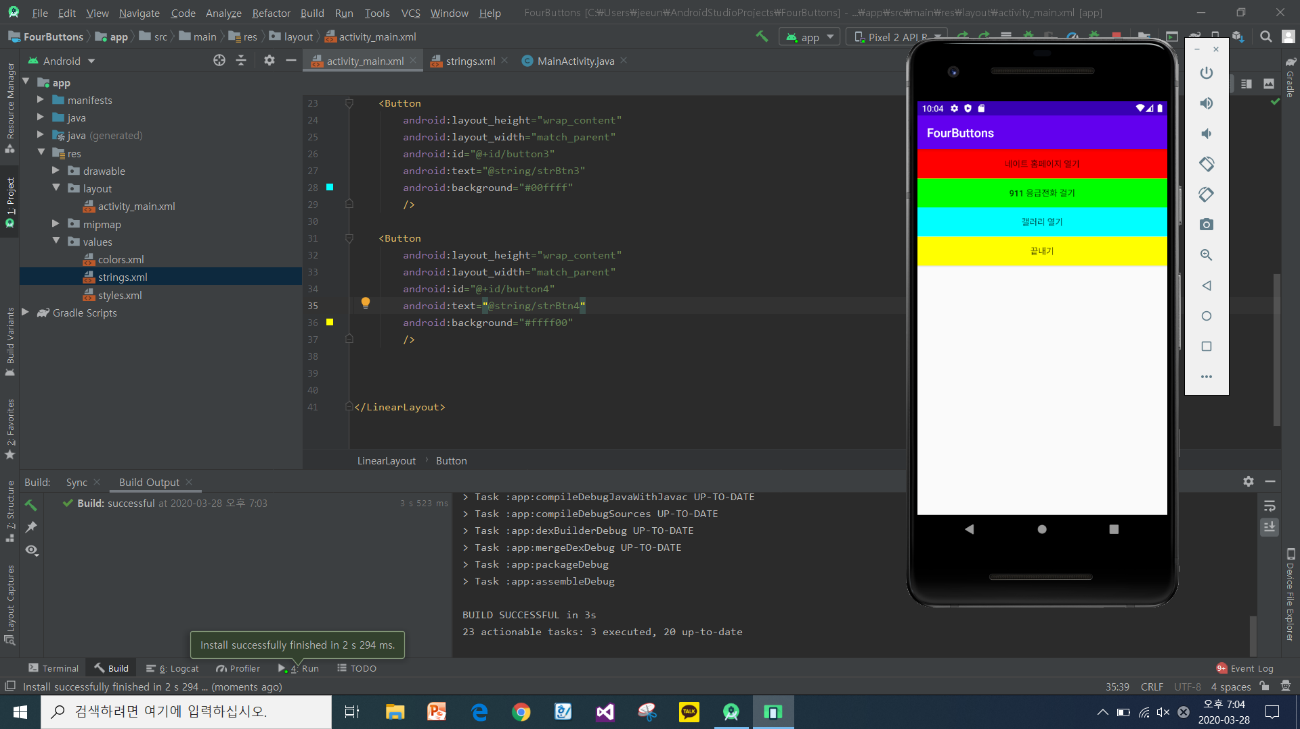
android:background="#ffff00"

/>

4번 버튼 코드 내용이다.

id, text는 string에 선언한 strBtn4를 변수로 넣어 설정해준다.

**실행 화면**



Run을 눌러 애뮬레이터로 실행시켰을 때 4번째 버튼 끝내기를 누를 경우 홈화면으로 나와지게 된다.

처음 실행 때 finish() 함수를 이용하지 않고 다른 방법으로 진행을 해보려고 했으나 코드에 오류가 발생하여 여러 차례 고친 후에 finish() 함수를 알게 되었다.

이를 이용해 어플리케이션을 종료시켰다.