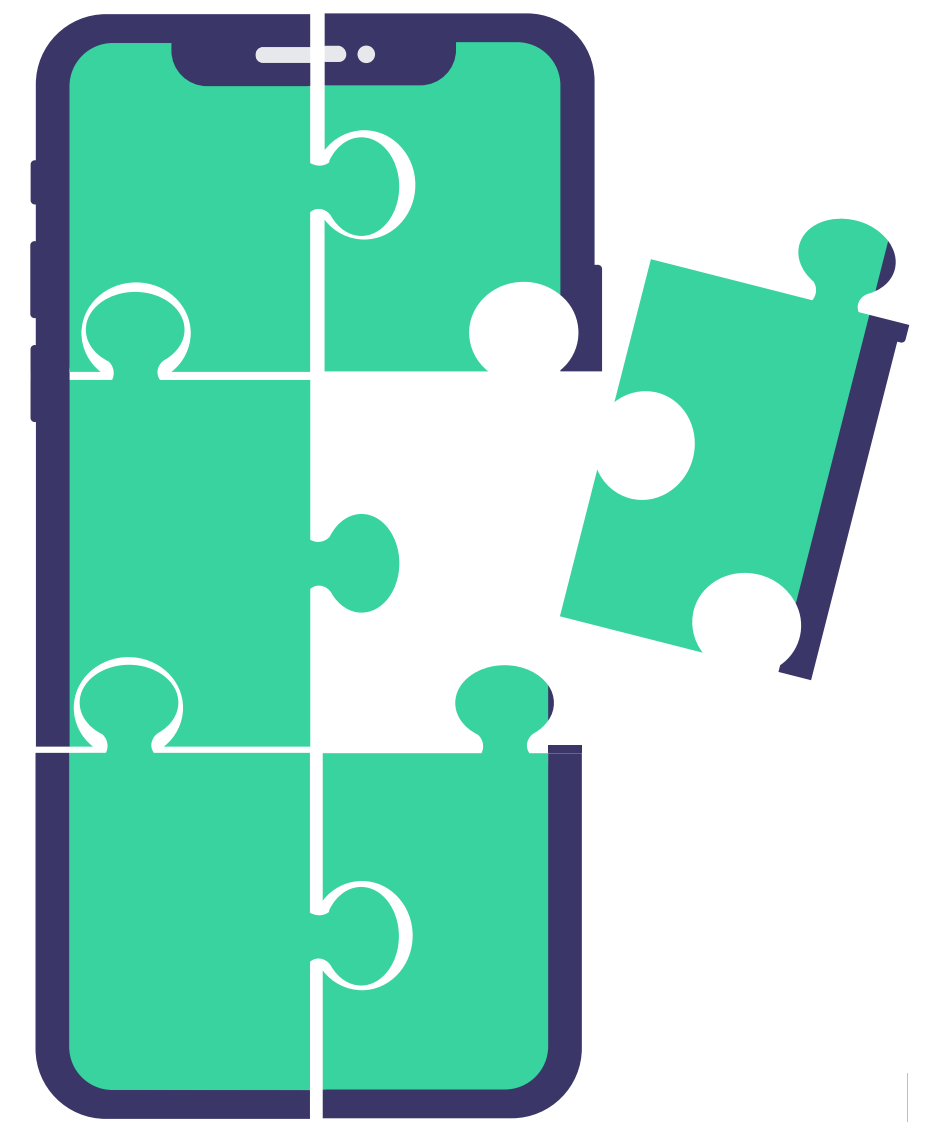


Android App Programming

# And09\_LayoutInflater

레이아웃 붙이기 ( LayoutInflater )

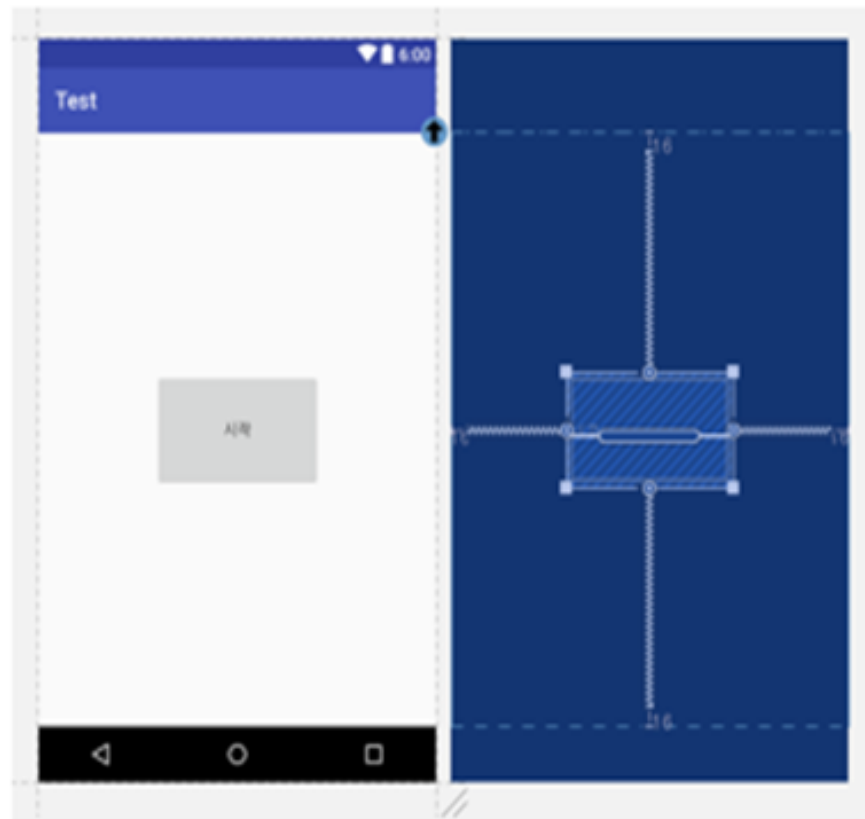


android 

# LayoutInflater

XML 레이아웃 파일과 자바 소스 파일의 매칭

- setContentView 메소드에서 XML 레이아웃 파일 매칭



XML 레이아웃 파일  
activity\_main.xml



자바 소스 코드 파일  
MainActivity.java

실행

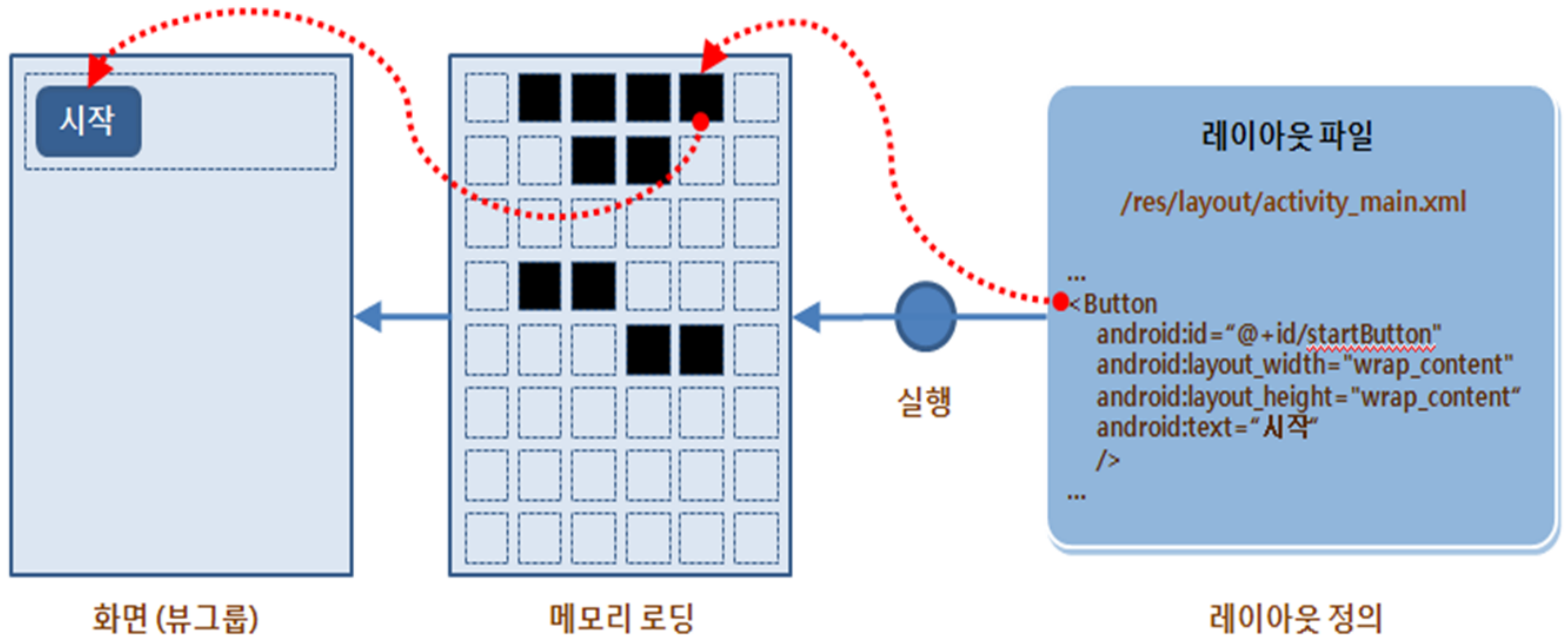


R.layout.레이아웃 파일 이름

# LayoutInflater

## 인플레이션

- 인플레이션 : XML 레이아웃에 정의된 내용이 메모리에 객체화되는 과정



["시작" 버튼의 레이아웃 인플레이션 과정]



# LayoutInflater

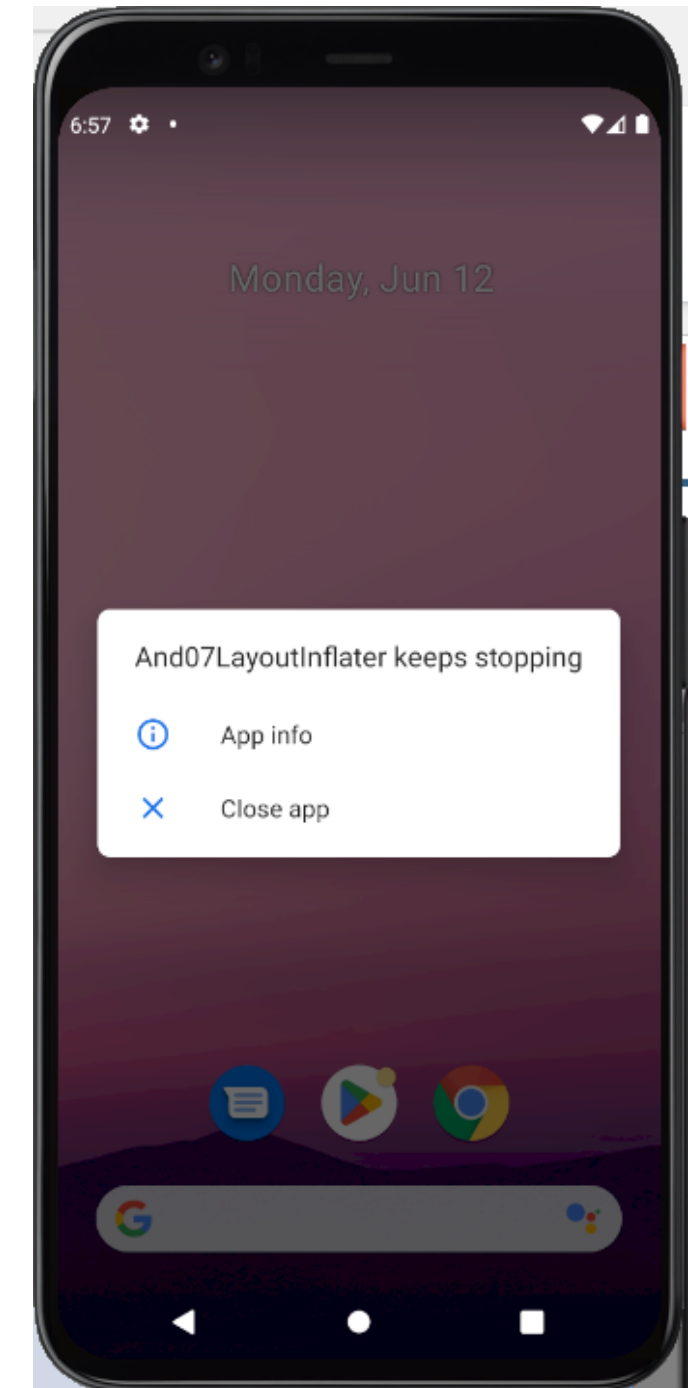
SetContentView 내부적으로 LayoutInflater라는 객체가 레이아웃 인플레이션 과정의 작업을 함.

```
2 usages
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        Button btn = findViewById(R.id.btn);

        setContentView(R.layout.activity_main);

        btn.setText("글자 바꾸기");
    }
}
```

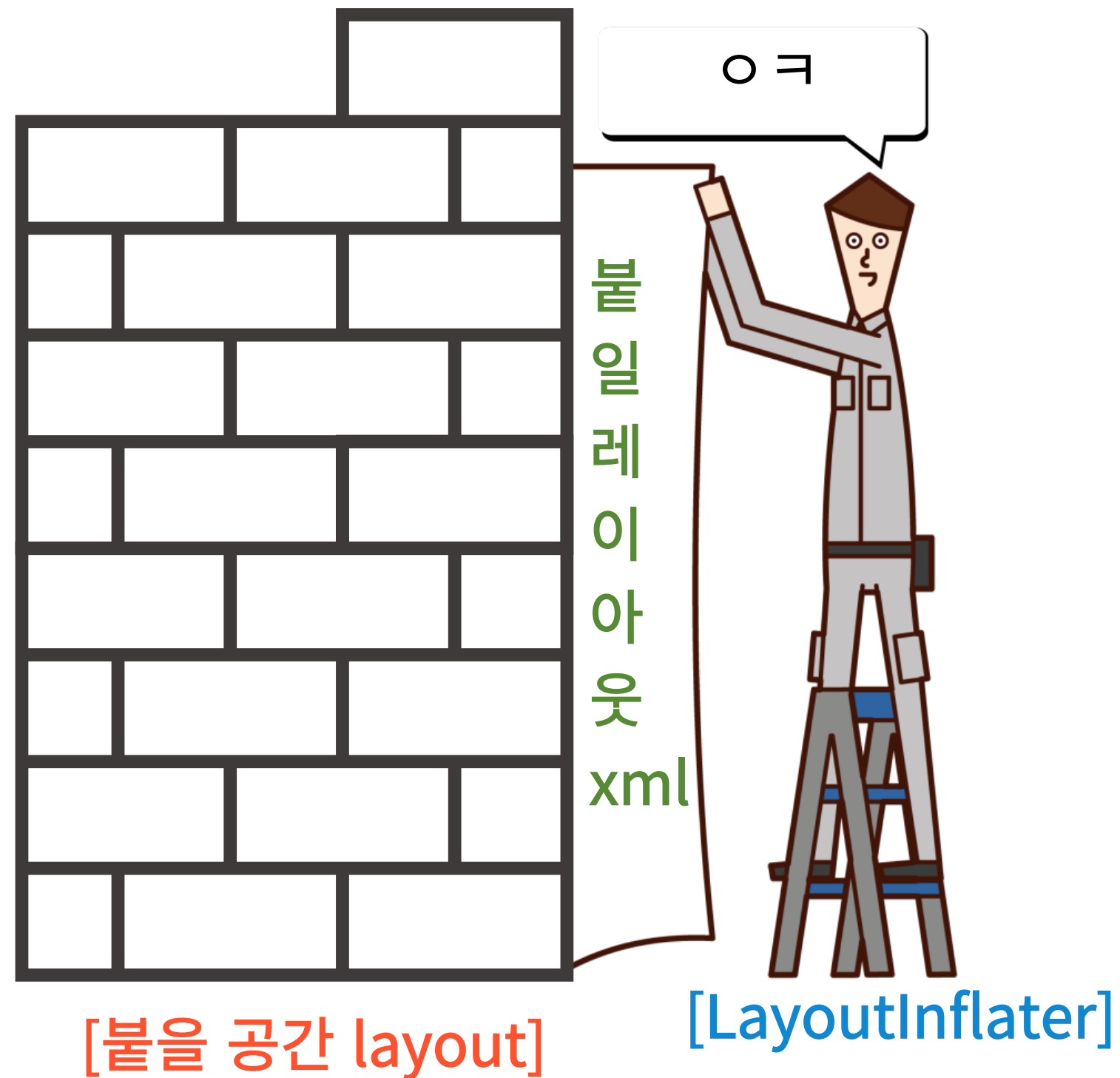


[setContentView() 코드와 findViewById() 메소드의 호출 순서를 바꾼 경우  
메모리에서 btn이라는 버튼을 찾을 수 없기 때문에 NullPointerException이 발생하고 앱이 종료됨]

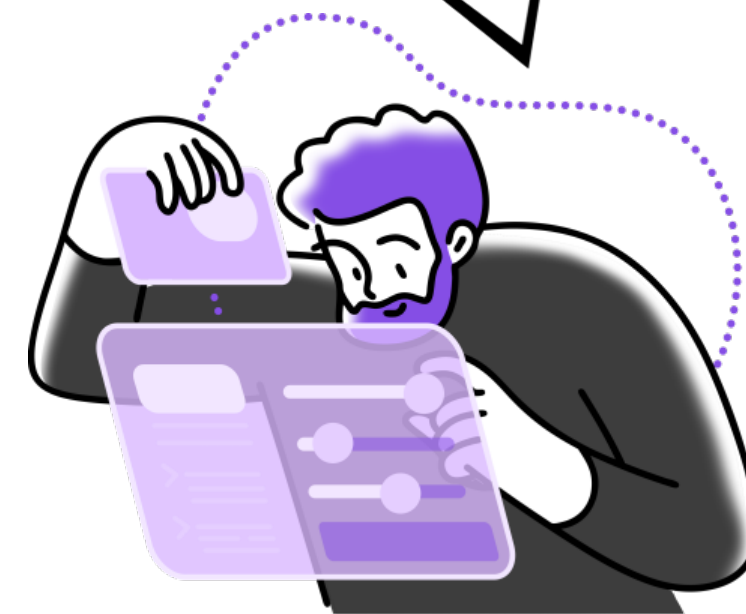


# LayoutInflater

LayoutInflater은 어떤 공간(벽==Layout)에 벽지 ( xml로 되어있는 Layout)을 붙이는 역할



LayoutInflater야 ,  
LinearLayout에  
sub\_layout1을 붙여라



[코드로 작성]

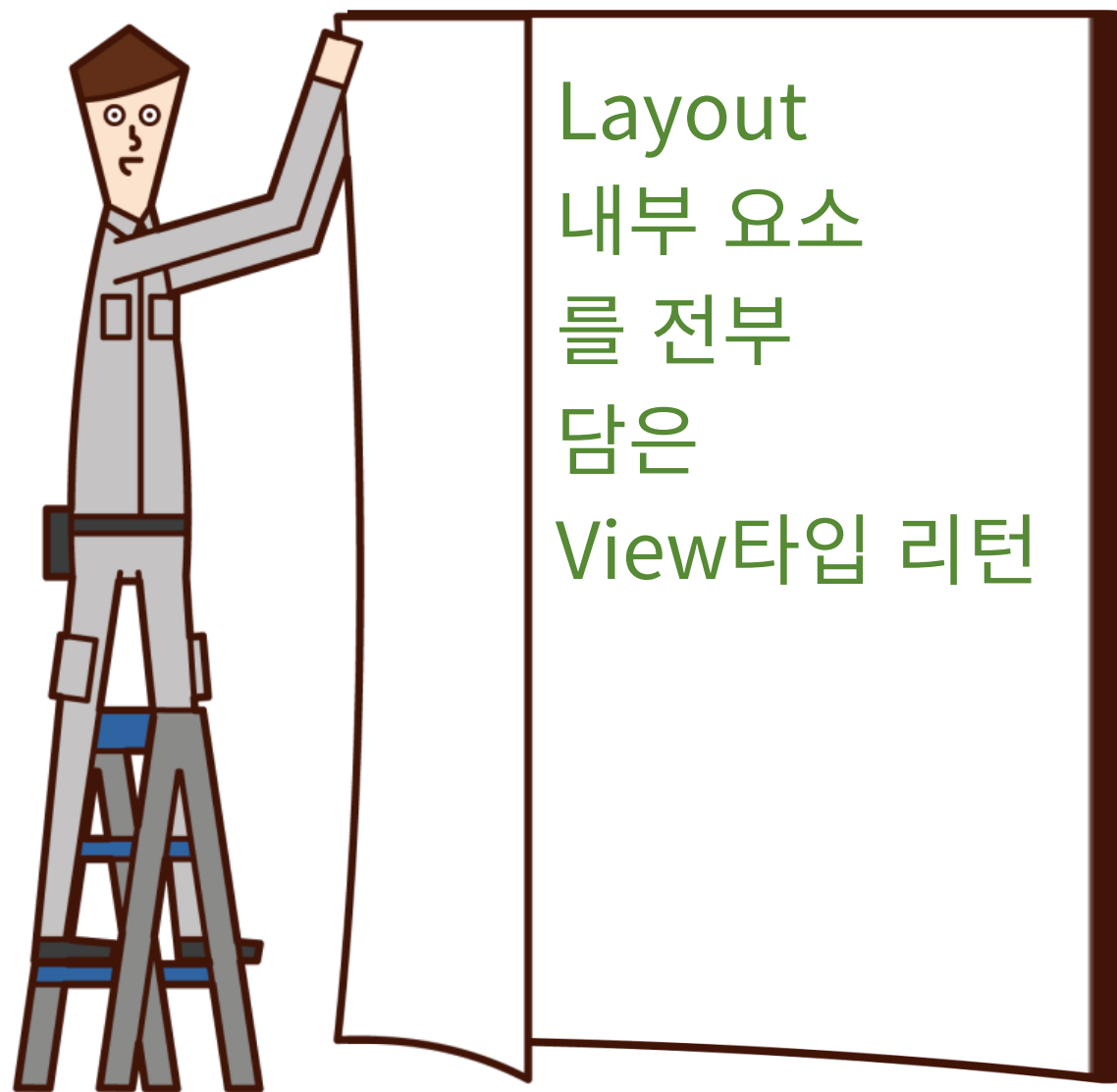




# LayoutInflater

정상적으로 레이아웃을 붙이는 처리를 하고나서는 결과를 View타입으로 묶어서 return해줌

다붙임 View타입으로 묶어서  
다시 드림 find등 사용가능



[LayoutInflater]

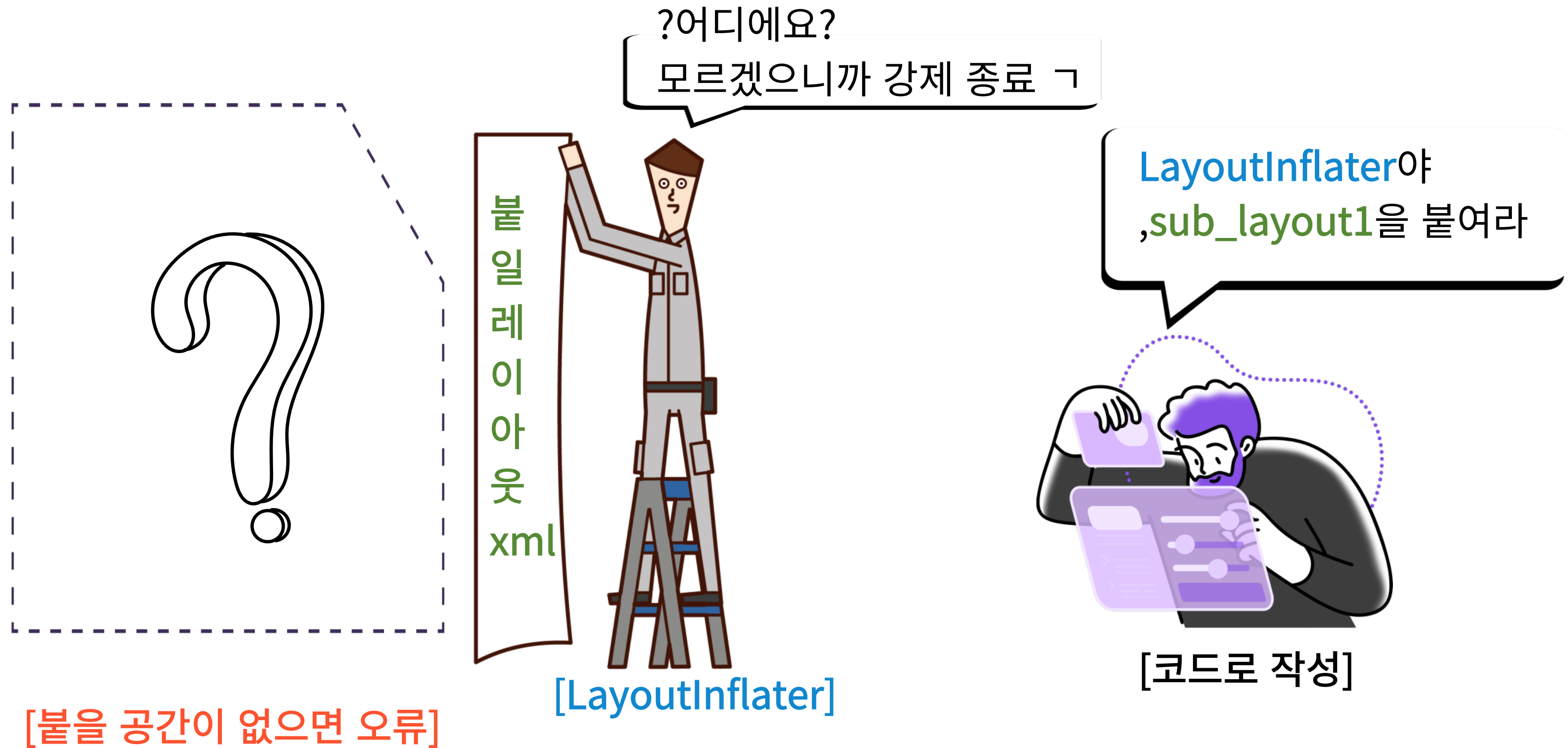


[코드로 작성]



# LayoutInflater

오류가 발생하는 경우는 없는 공간(layout를 지정하는 경우가 빈번함)





# LayoutInflater

## LayoutInflater의 초기화 식

1. Activity에서 사용하는 경우 AppCompatActivity의 getLayoutInflater사용.
2. getSystemService(Context.LAYOUT\_INFLATER\_SERVICE)

## LayoutInflater의 기능 사용 ( Inflate )

inflater.inflate( **붙일 레이아웃 리소스 (R.layout.name)** , **붙을 공간 ( ViewGroup )** , **true** );

**붙일 레이아웃 리소스** : R.layout. 하위폴더에 별도로 만들어둔 Layout리소스 파일

**붙을 공간** : Activity또는 현재 화면에서 ViewGroup(Layout종류,자식 객체를 가질수있는..)종류

attachToRoot : true & false 지금 내가 바로 붙일것인지에 대한여부.

(추후 많이 사용되는 Fragment등에서는 false를 사용, 지금 바로 붙이는게 아니라 초기화가 끝나고 알아서 붙음. 지금은 true로 사용하자)





# 실습 화면 구현 해보기

LayoutInflater

1. 붙을 공간을 Activity\_main  
에 만들기

2. 붙일 레이아웃 리소스 파일  
만들기.

[붙을 공간 각각  
layout(ViewGroup)]

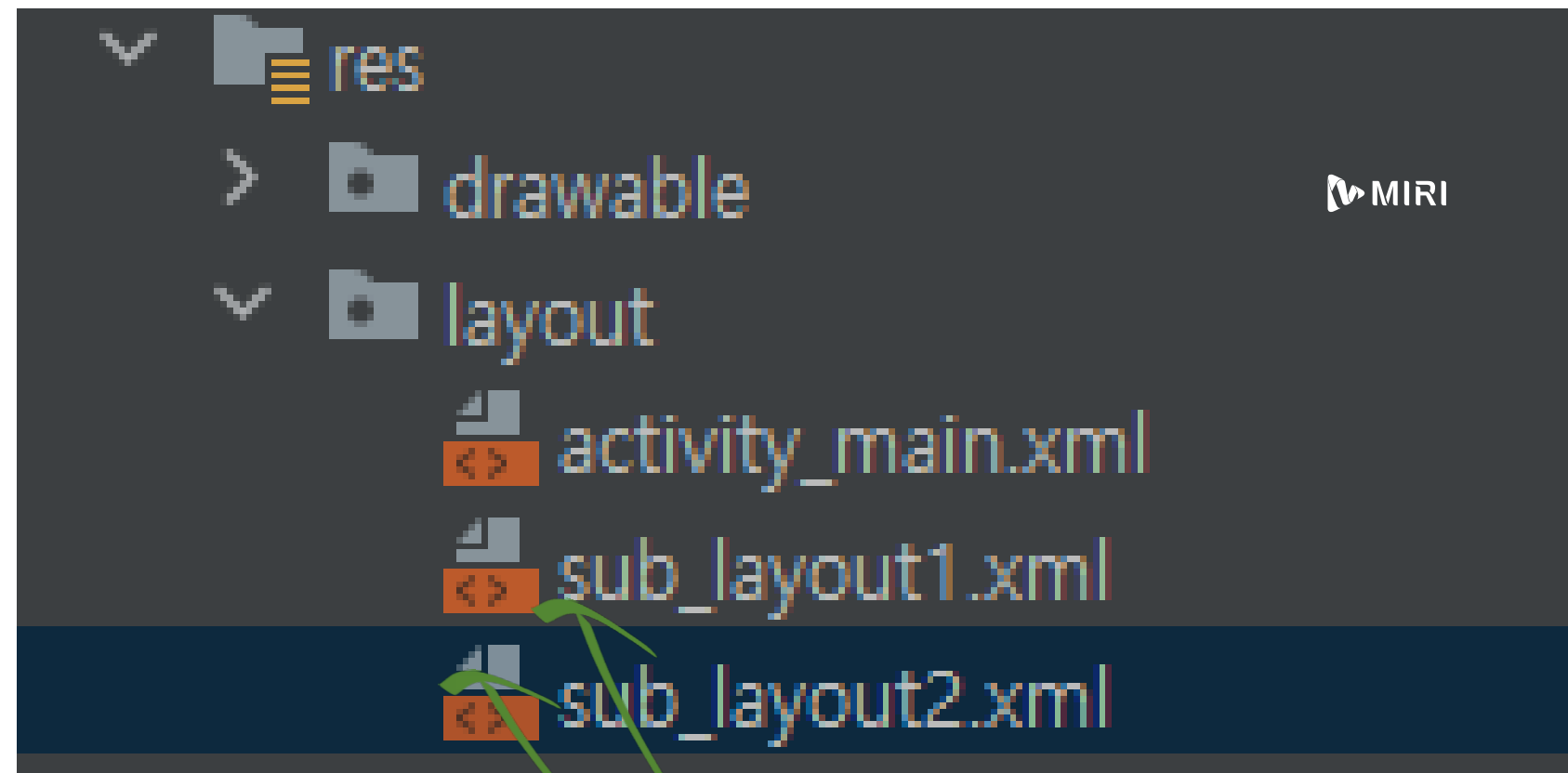
아무 레이아웃 종류를 2개넣어서 화면을 1/2로 나누기.  
해당하는 레이아웃을 식별할수있는 id를 부여하기.

# 실습 화면 구현 해보기

## LayoutInflater

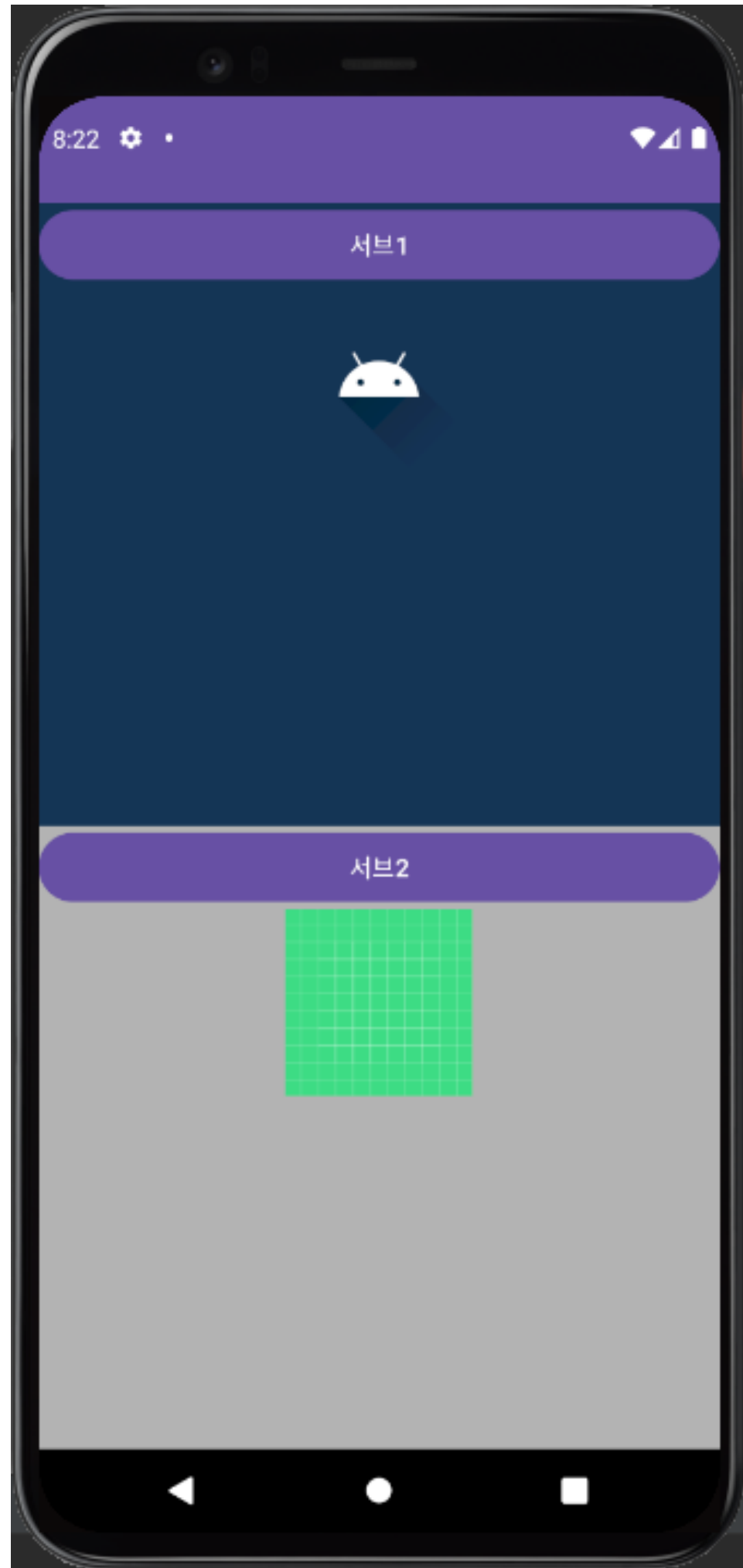
1. 붙을 공간을 Activity\_main에 만들기

2. 붙일 레이아웃 리소스 파일 만들기.



붙일 레이아웃 파일(xml)

두 레이아웃을 구분할수있게 만들고  
버튼은 무조건 하나 넣기. 그 외엔 자유



# 실습 화면 구현 해보기

LayoutInflater

1. 붙을 공간을 Activity\_main에 만들기
2. 붙일 레이아웃 리소스 파일 만들기.
3. Java코드로 붙이기

LayoutInflater의 inflate메소드를 이용하여  
두개의 레이아웃을 각각 붙이기.



# 정리

## LayoutInflater

레이아웃을 붙이는 객체로 inflate메소드를 이용하여  
레이아웃을 동적으로 붙이고 레이아웃 내부의 위젯들을  
View타입으로 묶어 사용자에게 리턴해줌.

## setContentView

Activity가 어떤 레이아웃과 연결 되는지가 정해지는 메소드  
로 내부적으로는 LayoutInflation과정이 진행된다.

## Container

다른 레이아웃이 붙는 ViewGroup(Layout)의 아이디를  
Container라는 단어를 이용해서 만들어준다.  
관례는 안지켜도 사용에 불편함은 없지만 개발자끼리 소스  
를 빨리 파악할때 불편함이 있다.