

액티비티와 레이아웃

코틀린 안드로이드 앱 개발

강사: 배정만

Hello 화면 만들기

- 앱이름: Hello
- Company domain: 아무거나
(예: kotlin.study)
- Project location: 프로젝트 폴더 설정
- Include Kotlin 체크
- Minimum SDK: 23 (6.0)
- Empty Activity 선택
- MainActivity / Layout 생성 체크
- Backward Compatibility 체크

주요 생성 파일

- 레이아웃 파일

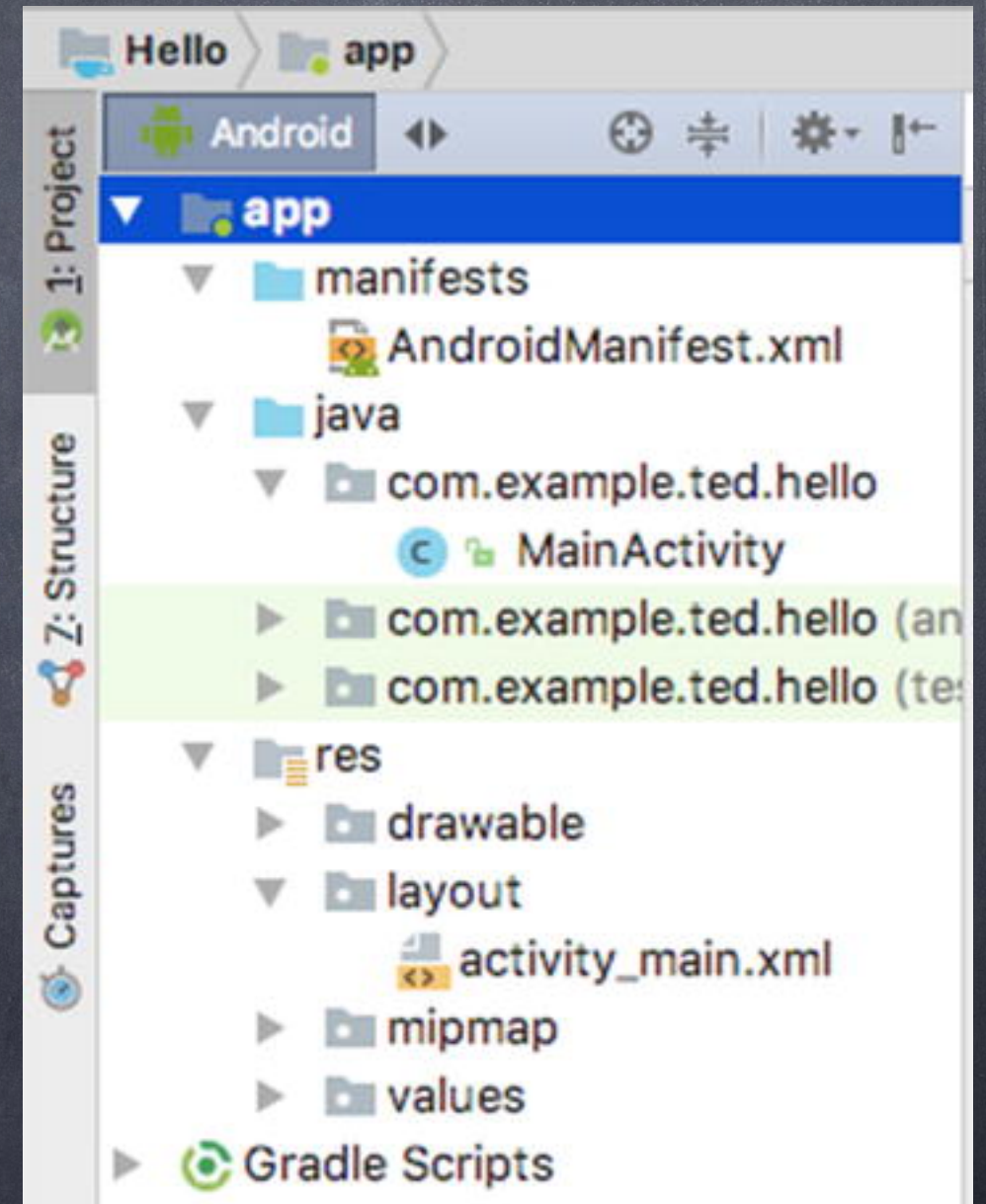
- `app/src/main/res/layout/activity_main.xml`

- 프로그램 파일

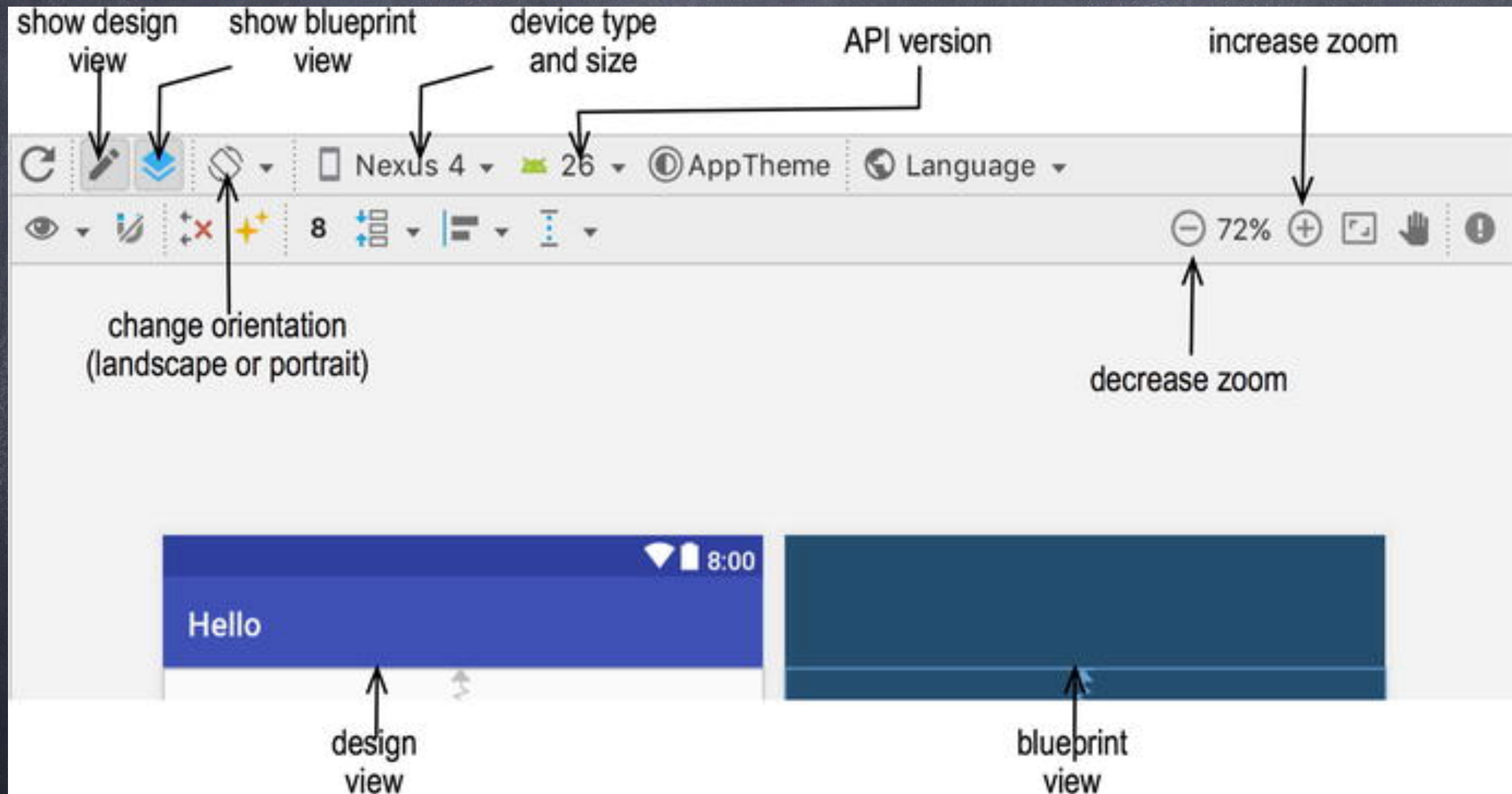
- `app/src/main/java/MainAcitivity.java`

레이아웃 파일

- activity_main 아직 열려 있지 않은 경우
 - 프로젝트 도구 창에서 시작
- 도구창은 프로젝트, 패키지, 프로젝트 파일 등으로 전환 가능
- 레이아웃 파일은 디자인 모드 (wysiwyg) 또는 텍스트 모드로 볼 수 있음



디자인 에디터



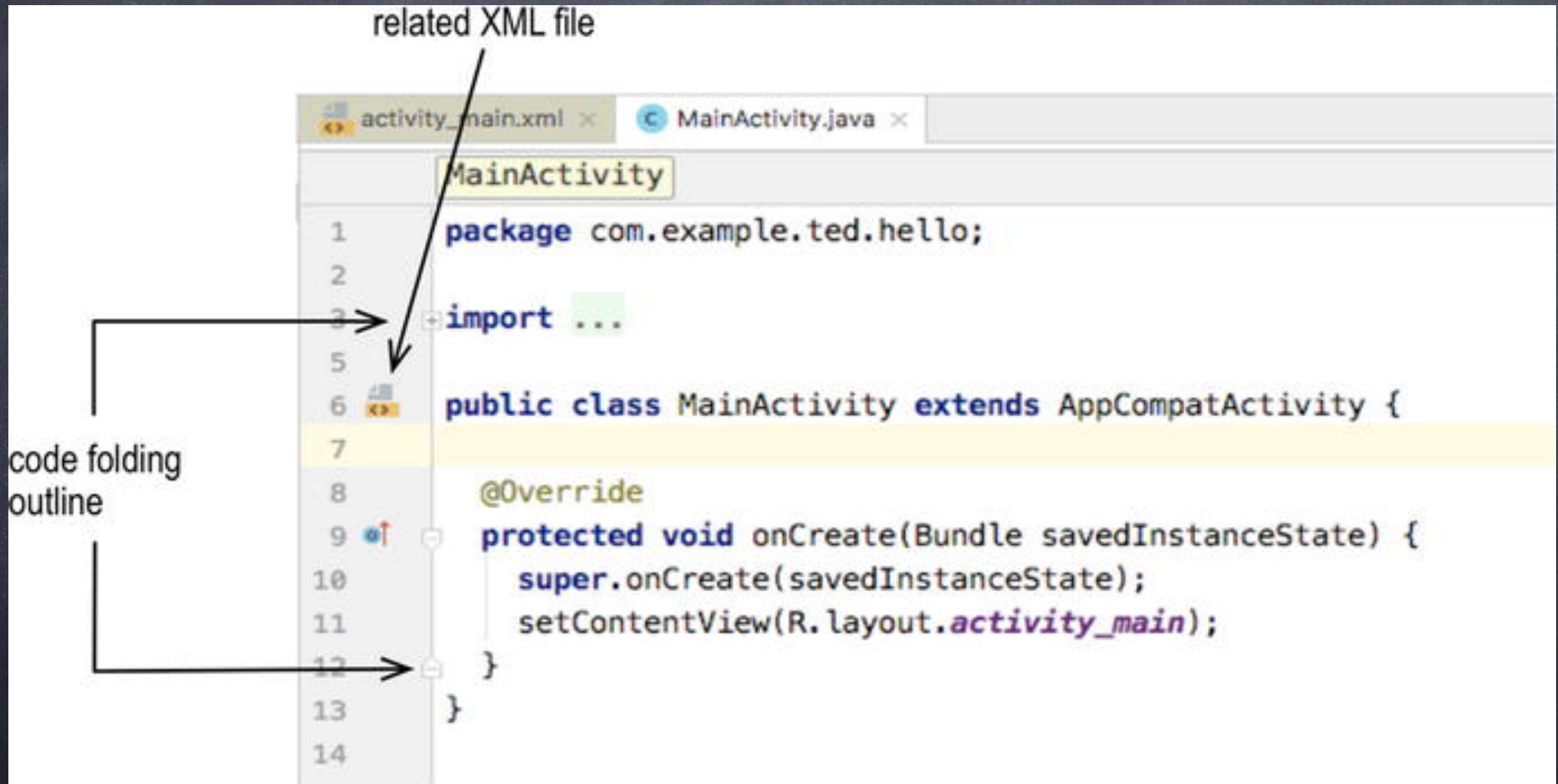
디자인 에디터 설명

show design view	디자인 뷰를 표시하거나 숨김
showblueprint view	블루프린트를 보거나 숨김
change orientation	레이아웃의 방향 전환 // 가로, 세로로 화면을 변경함
device type and size	휴대전화, 태블릿, TV, 웨어 디바이스 등 유형을 선택 // 화면 크기와 밀도를 변경할 수 있음
api version	안드로이드 버전에 따른 미리보기 지원
design view	레이아웃의 색상 미리보기 표시
blueprint view	블루프린트 미리보기 표시
increase zoom	레이아웃의 배율 조정 (크게)
decrease zoom	레이아웃의 배율 조정 (작게)

기본 레이아웃 XML

```
<? xml version = "1.0"encoding = "utf-8"?>
<android.support.constraint.ConstraintLayout ❶
    xmlns : android = "http://schemas.android.com/apk/res/android"
    xmlns : app = "http://schemas.android.com/apk/res-auto"
    xmlns : tools = "http://schemas.android.com/tools"
    android : layout_width = "match_parent"
    android : layout_height = "match_parent"
    tools : context = "com.example.ted.hello.MainActivity">
    <TextView ❷
        android : layout_width = "wrap_content"
        android : layout_height = "wrap_content"
        android : text = "Hello World!"
        app : layout_constraintBottom_toBottomOf = "parent"
        app : layout_constraintLeft_toLeftOf = "parent"
        app : layout_constraintRight_toRightOf = "parent"
        app : layout_constraintTop_toTopOf = "parent"/>
</android.support.constraint.ConstraintLayout>
```


액티비티 컴포넌트 Java 파일



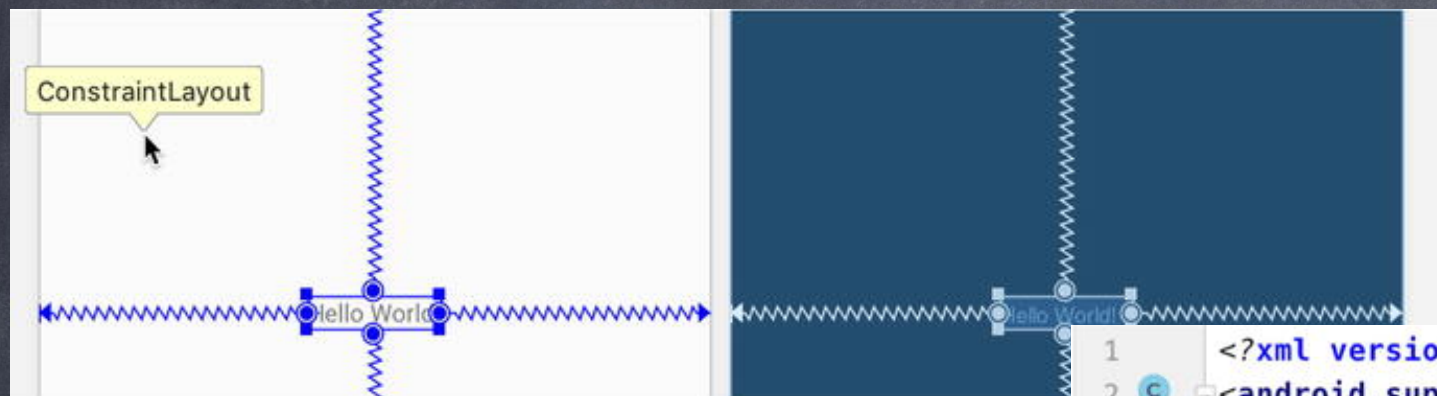
단축키

Task	Windows or Linux	macOS
설정 또는 환경 설정 대화 상자 열기	Ctrl + Alt + S	⌘ + ',' (comma)
코드 블록 확장	Ctrl + Shift + '+' (plus)	⌘ + '+' (plus)
접기 코드 블록	Ctrl + Shift + '-' (minus)	⌘ + '-' (minus)

MainActivity.java

```
package com.example.ted.hello; ❶
import android.support.v7.app.AppCompatActivity; ❷
import android.os.Bundle; ❸
public class MainActivity extends AppCompatActivity { ❹
    @Override ❺
    protected void onCreate(Bundle savedInstanceState) { ❻
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main); ❼
    }
}
```


뷰, 레이아웃



```
1  <?xml version="1.0" encoding="utf-8"?>
2  <android.support.constraint.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context="com.example.ted.hello.MainActivity"
9      tools:layout_editor_absoluteY="81dp"
10     tools:layout_editor_absoluteX="0dp">
11
12     <TextView
13         android:layout_width="wrap_content"
14         android:layout_height="wrap_content"
15         android:text="Hello World!"
16         tools:layout_editor_absoluteY="247dp"
17         tools:layout_editor_absoluteX="154dp"
18         android:id="@+id/textView"
19         app:layout_constraintTop_toTopOf="parent"
20         app:layout_constraintStart_toStartOf="parent"
21         app:layout_constraintBottom_toBottomOf="parent"
22         app:layout_constraintEnd_toEndOf="parent"/>
23
24 </android.support.constraint.ConstraintLayout>
```


실습 1

1. 레이아웃XML에서 기본 텍스트뷰 삭제
2. (디자인모드에서)
editText, Button, TextView 컴포넌트 추가
3. ConstraintLayout 속성을 이용해 컴포넌트들을 화면에 배치
4. 에뮬레이터에서 실행

실습 1 화면

