

Android application 만들기

# 6. View Binding

Binding Class

# Widget을 코드에 연결할 때 불편한 점

xml 파일에 선언된 Widget을 setContentView를 이용해 객체를 만든 다음 findViewById 함수를 통해 변수와 연결하는 일련의 과정.

1. 코드에서 제어 할 widget 수 만큼 findViewById를 호출해야 하며
2. 두 개 이상의 레이아웃 파일이 프로젝트에 존재할 경우 사용하지 않는 다른 파일의 위젯 아이디를 넣어 앱 실행 후 Null Exception이 발생 - 컴파일 단계에서는 에러가 발생하지 않음

이를 해결하기 위해 다양한 외부 라이브러리들이 개발되었으며

이 문제를 해결하기 위해 Android Gradle Plugin 3.6.0 부터 View Binding을 제공함.

# View Binding

## 1. Android Gradle Plugin의 버전 확인

상단 File 메뉴 > Project Structure : Android Gradle Plugin Versino 이 3.6.0 이상인지 확인



## 2. build.gradle (Module:app) 파일을 열어 다음과 같이 viewBinding 코드 추가

## 3. 파일 우측 상단의 sync now 클릭

```
apply plugin: 'com.android.application'
```

```
android {  
    compileSdkVersion 29  
    buildToolsVersion "29.0.2"  
    viewBinding.enabled=true  
}
```

# Activity 에서 사용하기

ViewBinding 클래스는 레이아웃 이름을 이용하여 자동으로 만들어짐.

activity\_main.xml -> ActivityMainBinding

MainActivity 의 onCreate를 다음과 같이 수정하면 바인딩을 사용할 수 있음.

```
public class MainActivity extends AppCompatActivity {  
  
    private ActivityMainBinding binding;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        binding=ActivityMainBinding.inflate(getLayoutInflater());  
        setContentView(binding.getRoot());  
    }  
}
```

# Activity 에서 사용하기

onCreate 함수 수정후에는 다음과 같이 코드를 사용할 수 있다.

binding 하지 않은 코드

```
TextView textView=findViewById(R.id.textView);  
textView.setText("Hello");
```

binding 한 경우

```
binding.textView.setText("Hello");
```

따로 findViewById 를 호출할 필요가 없고

해당 레이아웃의 id를 가진 위젯만 자식으로 등록되기 때문에 null 상황이 발생하지 않는다.

# 실습: Widget 프로젝트에 View Binding 적용하기

build.gradle(Module:app) 파일에 다음 config 추가

```
android {  
    compileSdkVersion 29  
    buildToolsVersion "29.0.2"  
    viewBinding{  
        enabled = true  
    }
```

# 실습: Widget 프로젝트에 View Binding 적용하기

MainActivity.java 수정

```
public class MainActivity extends AppCompatActivity
    implements TextWatcher, View.OnClickListener, CompoundButton.OnCheckedChangeListener {
    private ActivityMainBinding binding;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding=ActivityMainBinding.inflate(getLayoutInflater());
        setContentView(binding.getRoot());
        initWidgets();
    }
}
```

이후 initWidgets 함수 및 나머지 함수를 binding 기준으로 변경

# 실습: Widget 프로젝트에 View Binding 적용하기

MainActivity.java 수정

```
private void initWidgets(){  
    binding.editTextName.addTextChangedListener(this);  
    binding.editTextPhone.addTextChangedListener(this);  
    binding.radioButtonAdult.setOnClickListener(this);  
    binding.radioButtonStudent.setOnClickListener(this);  
    binding.checkBoxTerms.setOnCheckedChangeListener(this);  
}
```

나머지 함수에서도 다 수정