



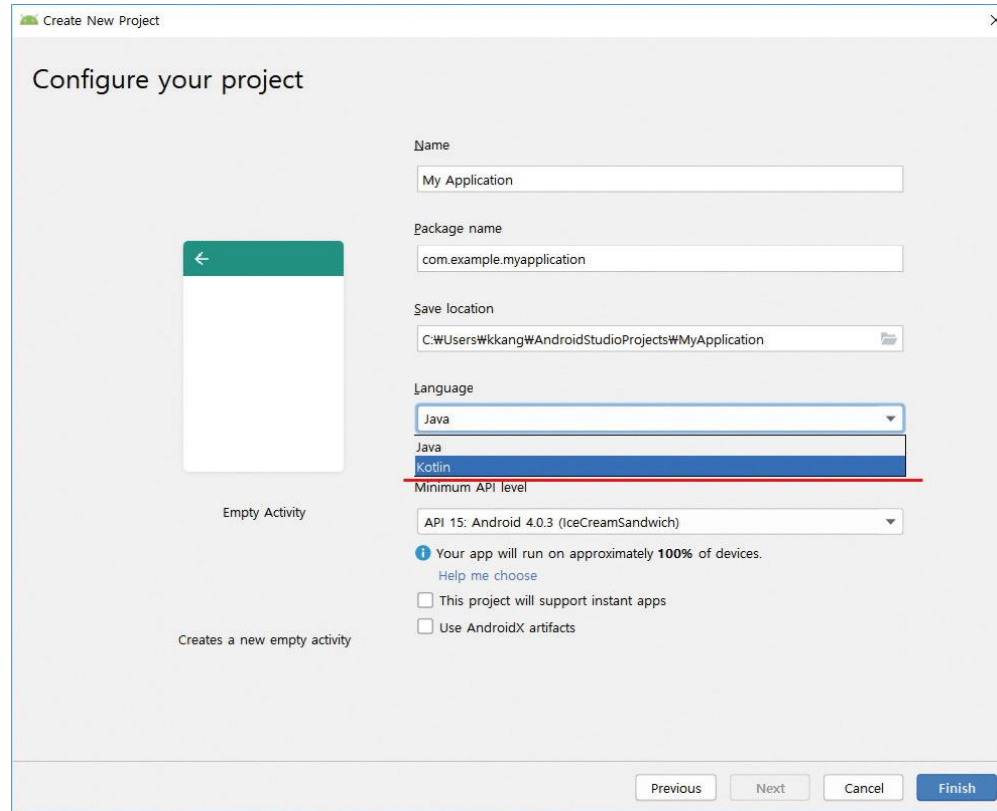
31장. 안드로이드 스튜디오에서 코틀린으로 개발하기

31.1 코틀린 소개

- 코틀린은 IntelliJ IDE로 유명한 JetBrains (<https://www.jetbrains.com/idea/>)에서 오픈소스 그룹을 만들어 개발한 소프트웨어 개발 언어
- 100% interoperable with Java™ and Android™
- Java, Android, Browser, Native Application 개발
- IntelliJ, Android Studio, Eclipse, CLI 등을 통해 개발 가능
- 함수형 언어, Lambdas, Extension, Null Safety 등 최신 언어의 트렌드 지원
- 코틀린으로 개발된 코드가 자바 클래스로 빌드되어 JVM에서 동작
- 안드로이드뿐만 아니라 자바나 자바스크립트로 개발되던 웹 애플리케이션 분야에도 사용할 수 있는 언어를 목표

31.2 안드로이드 스튜디오에서 코틀린 개발환경

31.2.1. 코틀린 프로젝트 생성



31.2 안드로이드 스튜디오에서 코틀린 개발환경

- 프로젝트 수준의 그레이들 파일

```
buildscript {  
    ext.kotlin_version = '1.3.21'  
    //...  
    dependencies {  
        classpath 'com.android.tools.build:gradle:3.3.2'  
        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version"  
    }  
}
```

- 모듈 수준의 그레이들 파일

```
apply plugin: 'com.android.application'  
apply plugin: 'kotlin-android'  
apply plugin: 'kotlin-android-extensions'  
...  
dependencies {  
    ...  
    implementation "org.jetbrains.kotlin:kotlin-stdlib-jdk7:$kotlin_version"  
}
```

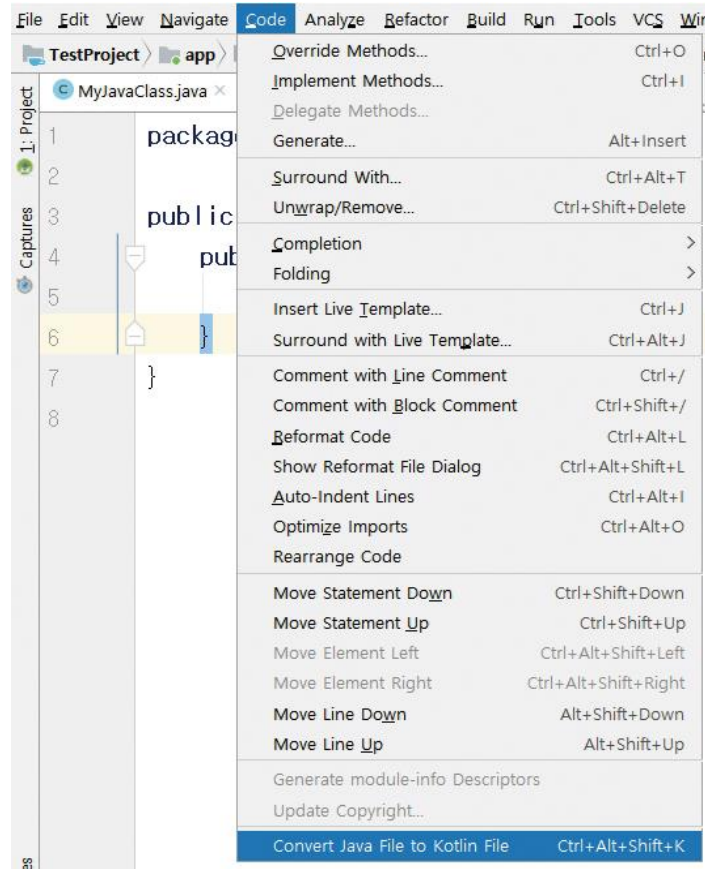
31.2 안드로이드 스튜디오에서 코틀린 개발환경

31.2.2. 코틀린 클래스 추가



31.2 안드로이드 스튜디오 3에서 코틀린 개발환경

31.2.3. 자바 코드를 코틀린 코드로 변환



31.3 간단한 코드 분석으로 코틀린 이해하기

```
import android.support.v7.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.Toast
import kotlinx.android.synthetic.main.activity_main.*
class MainActivity : AppCompatActivity(), View.OnClickListener{
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        testBtn.setOnClickListener(this)
    }
    override fun onClick(p0: View?) {
        Toast.makeText(this, "button click..", Toast.LENGTH_SHORT).show()
    }
}
```