



<1장> 안드로이드 개요

학습 목표

- 안드로이드에 대한 개념과 주요 특징을 이해한다.
- 안드로이드 애플리케이션을 구성하는 기본 요소를 이해한다.
- 안드로이드 프로그래밍에 필요한 개발 환경을 구축한다.
- 안드로이드 프로젝트를 생성하고 실행하는 방법을 익힌다.
- 안드로이드 애플리케이션의 프로젝트 구조를 이해한다.

목차

01 안드로이드 이해하기

02 안드로이드 개발 환경 구축하기

03 안드로이드 프로젝트 시작하기

04 안드로이드 프로젝트 구조 살펴보기

[도서 쇼핑몰] 프로젝트 시작하기

01

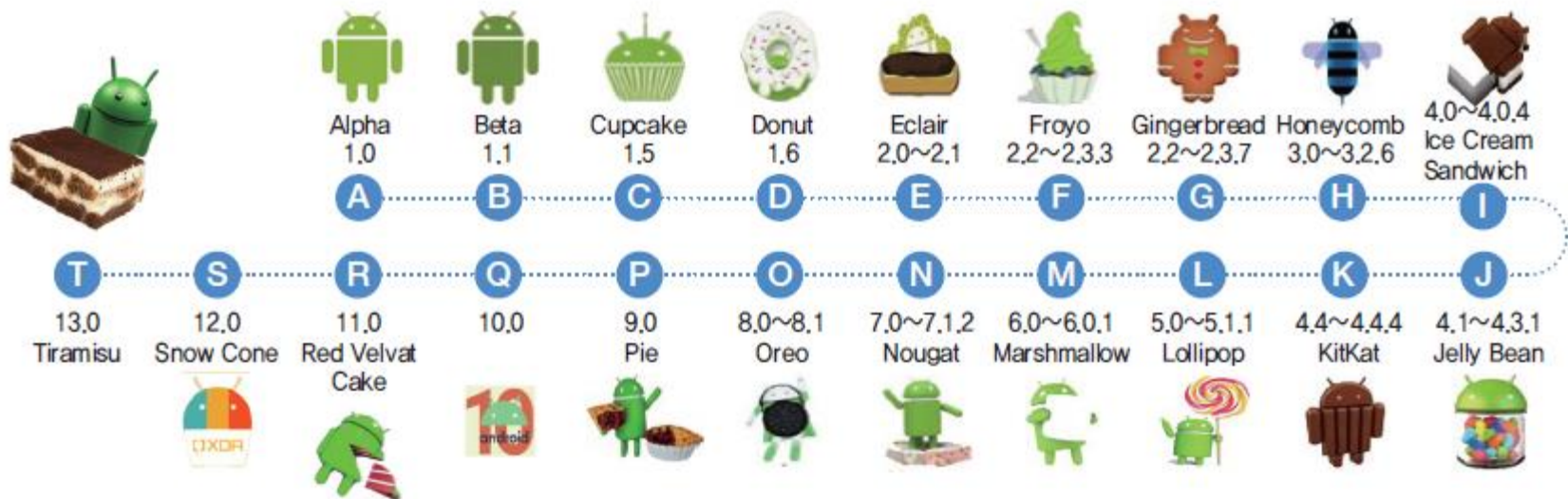
안드로이드 이해하기

1. 안드로이드 이해하기

■ 안드로이드란

- 모바일 장치(스마트폰, 태블릿 컴퓨터 등)에서 사용되는 운영체제와 앱 개발을 위한 플랫폼
- iOS와 함께 스마트폰 운영체제의 주요 플랫폼으로 인기
- 구글 플레이를 통해 다양한 앱과 콘텐츠 제공
- 안드로이드 애플리케이션 개발을 위한 도구는 소프트웨어 개발 키트(SDK, Software Development Kit)로 제공

그림 1-1 안드로이드 운영체제 버전



1. 안드로이드 이해하기

■ 안드로이드의 특징

- 자바 언어 사용
- 리눅스 기반의 오픈 소스
- 높은 시장 점유율
- 유연한 통합 환경
- 개발 시간과 비용 절감
- 풍부한 개발 환경 제공

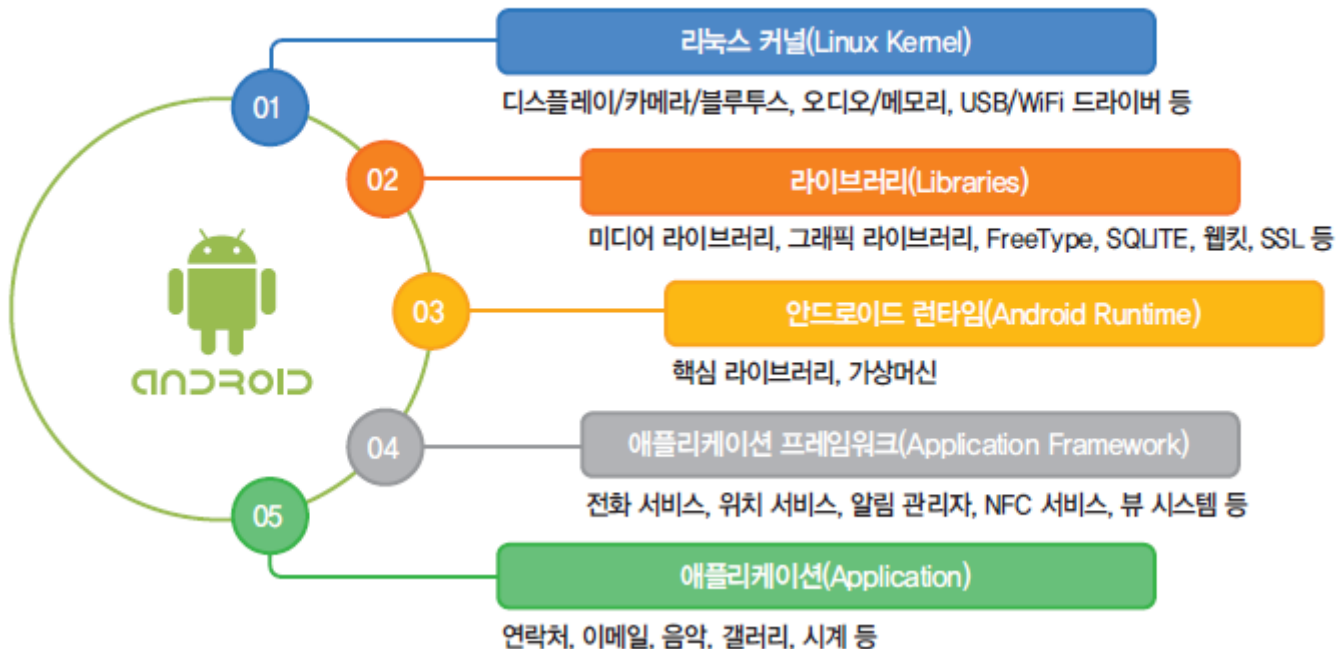
그림 1-2 안드로이드 특징



1. 안드로이드 이해하기

■ 안드로이드의 구조

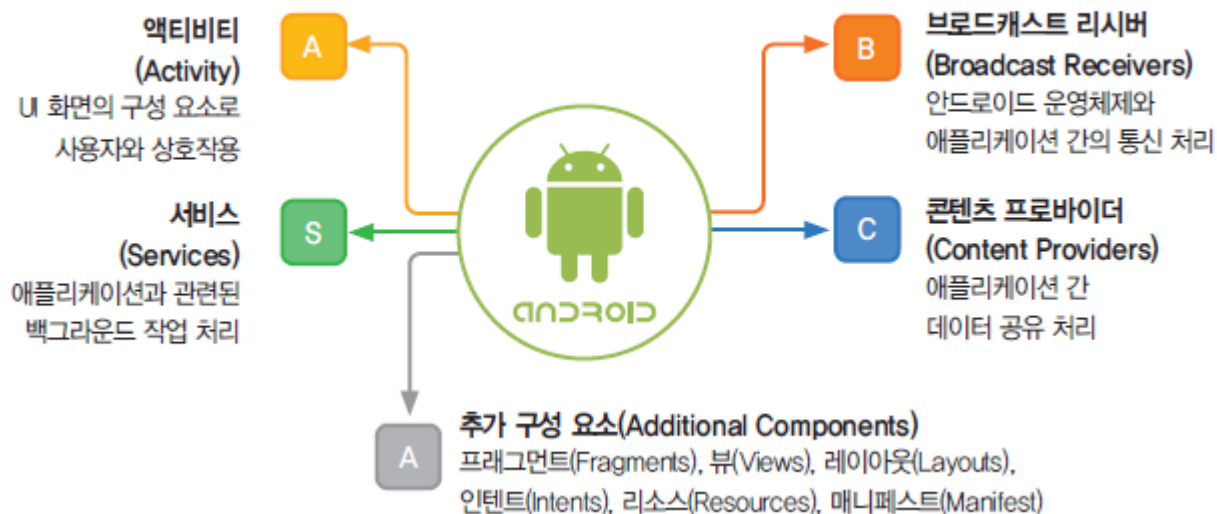
그림 1-3 안드로이드의 구조



1. 안드로이드 이해하기

■ 안드로이드 애플리케이션의 구성 요소

그림 1-4 안드로이드 애플리케이션 구성 요소



02

안드로이드 개발 환경 구축하기

2. 안드로이드 개발 환경 구축하기

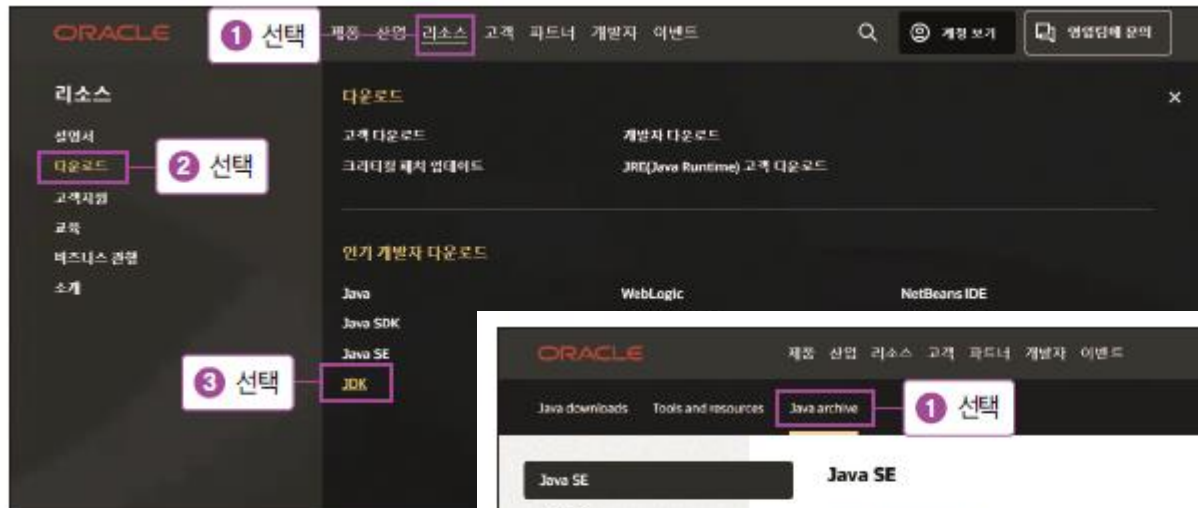
표 1-1 안드로이드 개발 환경 도구

요소	도구	설명
자바 개발 환경	JDK(Java Development Kit)	안드로이드는 기본적으로 자바를 필요로 하므로 자바 개발 도구인 JDK를 반드시 설치해야 합니다.
통합 개발 환경	안드로이드 스튜디오 (Android Studio)	안드로이드 앱을 개발하기 위해 코드를 작성하고, 이를 컴파일하여 오류를 검사하고 실행 결과를 확인할 수 있는 통합 개발 환경 (IDE, Integrated Development Environment)으로는 안드로이드 스튜디오를 사용합니다.

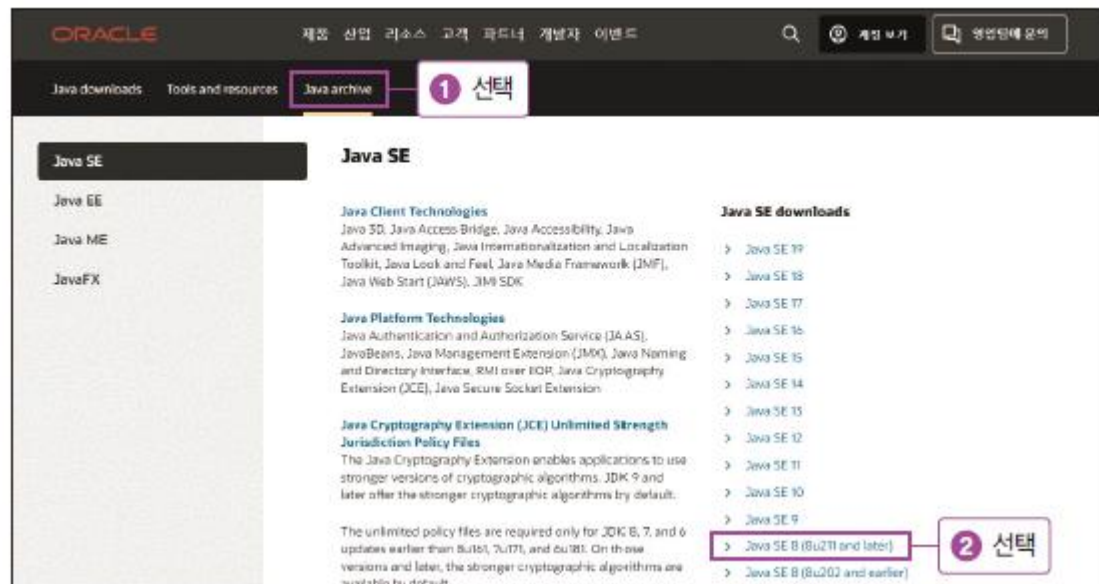
2. 안드로이드 개발 환경 구축하기

■ 자바 설치

■ (1) 오라클 사이트 접속하기



■ (2) JDK 선택하기



2. 안드로이드 개발 환경 구축하기

■ 자바 설치

- (3) 운영체제 버전에 맞는 설치 파일 선택 및 라이선스 동의하기

The screenshot shows the Oracle Java SE Development Kit 8u341 download page. The page lists various operating systems and architectures with their corresponding file sizes and download links. Three steps are annotated with red boxes and numbers:

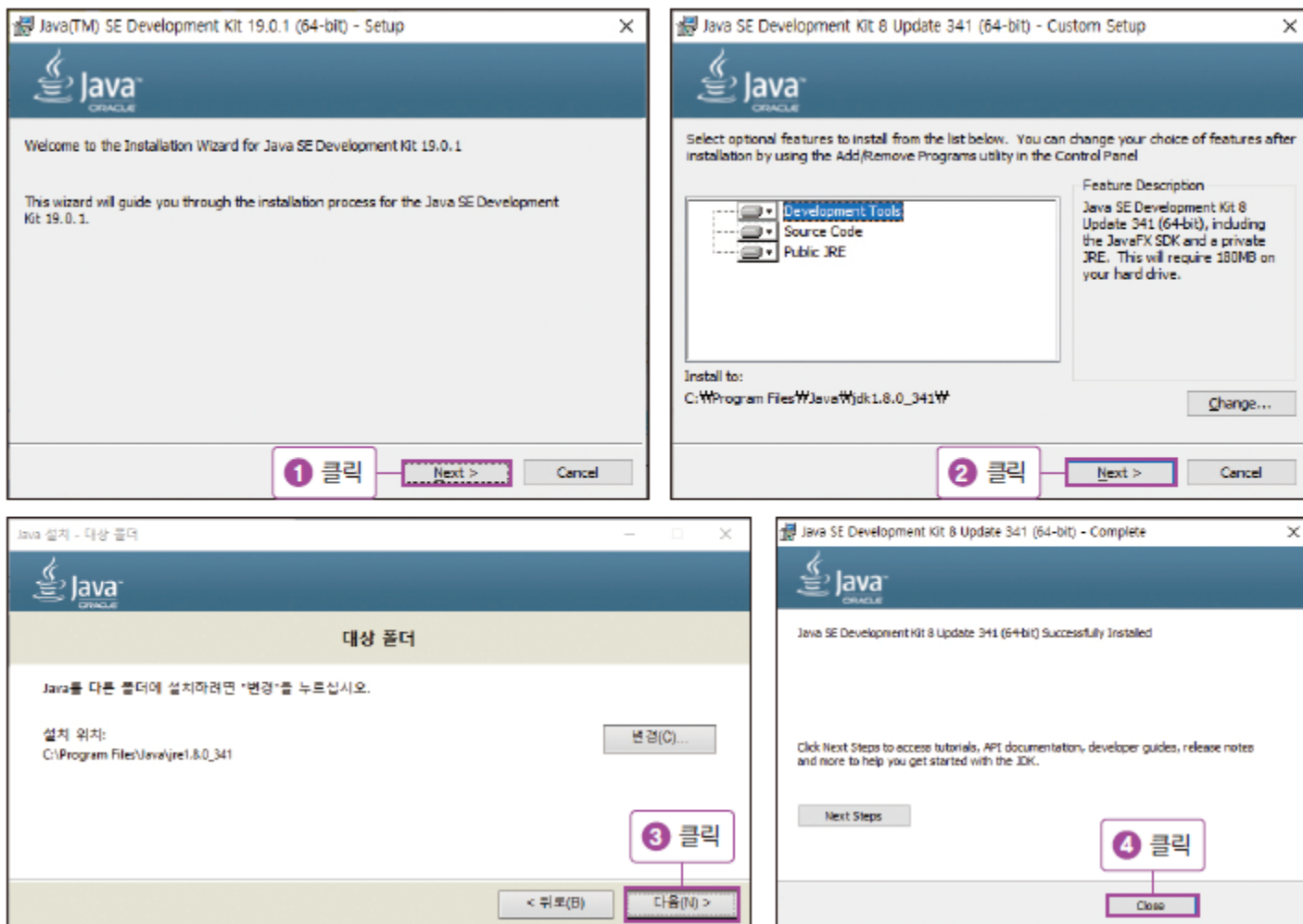
- 1 선택**: Selecting the appropriate download link for the operating system (e.g., Windows x64 Installer).
- 2 체크 표시**: Checking the "I reviewed and accept the Oracle Technology Network License Agreement for Oracle Java SE" checkbox.
- 3 클릭**: Clicking the "Download jdk-8u341-windows-x64.exe" button.

Product / File Description	File Size	Download
Linux ARM 64 RPM Package	59.29 MB	jdk-8u341-linux-arm64.rpm
Linux ARM 64 Compressed Archive	7104 MB	jdk-8u341-linux-arm64.tar.gz
Linux ARM 32 Hard Float ABI	73.75 MB	jdk-8u341-linux-arm32-vfp-hflt.tar.gz
Linux x86 RPM Package	111.11 MB	jdk-8u341-linux-i586.rpm
Linux x86 Compressed Archive	140.83 MB	jdk-8u341-linux-i586.tar.gz
Linux x64 RPM Package		jdk-8u341-linux-x64.rpm
Linux x64 Compressed Archive		jdk-8u341-linux-x64.tar.gz
macOS x64 DMG Installer		jdk-8u341-macosx-x64.dmg
Solaris SPARC 64-bit (SVR64)		jdk-8u341-solaris-sparcv9.tar.gz
Solaris SPARC 64-bit Compressed Archive	94.92 MB	jdk-8u341-solaris-sparcv9.tar.gz
Solaris x64 (SVR64 package)	154.89 MB	jdk-8u341-solaris-x64.tar.Z
Solaris x64 Compressed Archive	92.84 MB	jdk-8u341-solaris-x64.tar.gz
Windows x86 Installer	159.66 MB	jdk-8u341-windows-i586.exe
Windows x64 Installer	173.16 MB	jdk-8u341-windows-x64.exe

2. 안드로이드 개발 환경 구축하기

■ 자바 설치

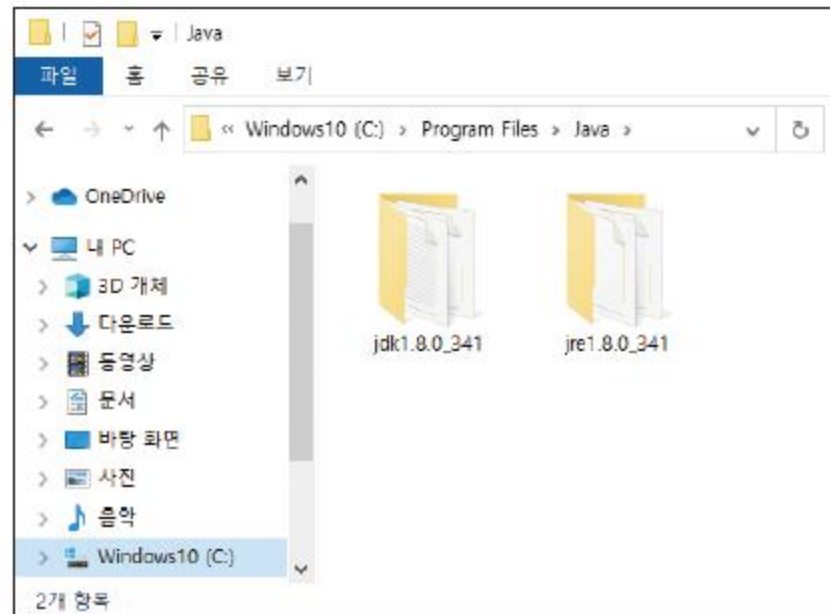
■ (4) 자바 설치하기



2. 안드로이드 개발 환경 구축하기

■ 자바 설치

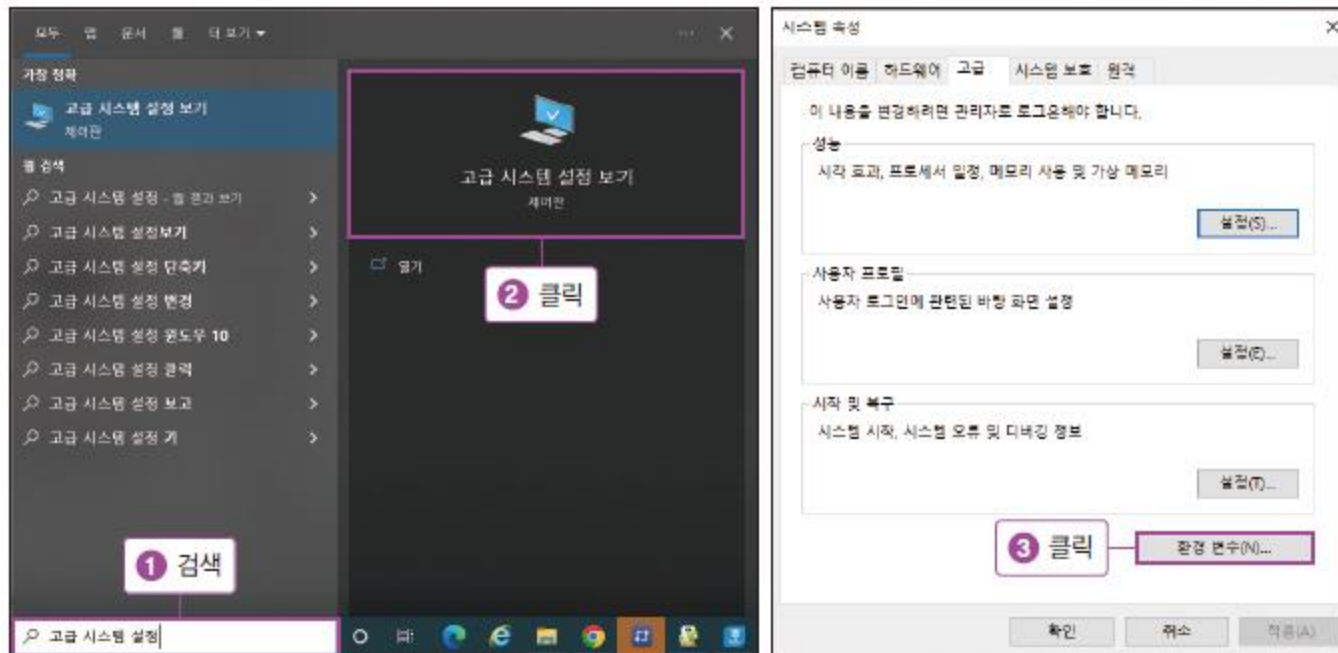
- (5) 자바 설치 위치 확인하기



2. 안드로이드 개발 환경 구축하기

■ 환경 변수 설정

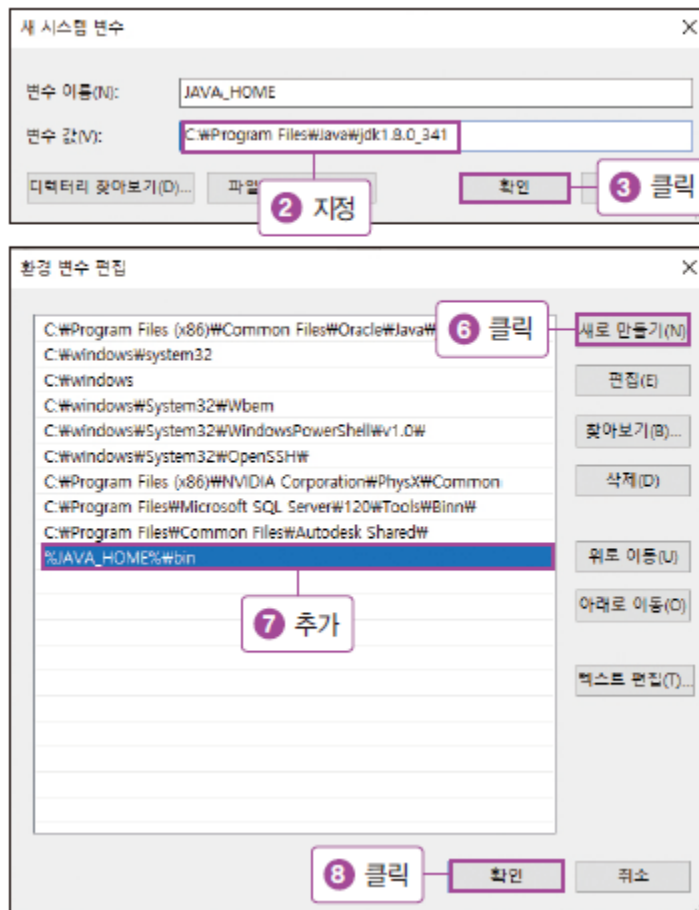
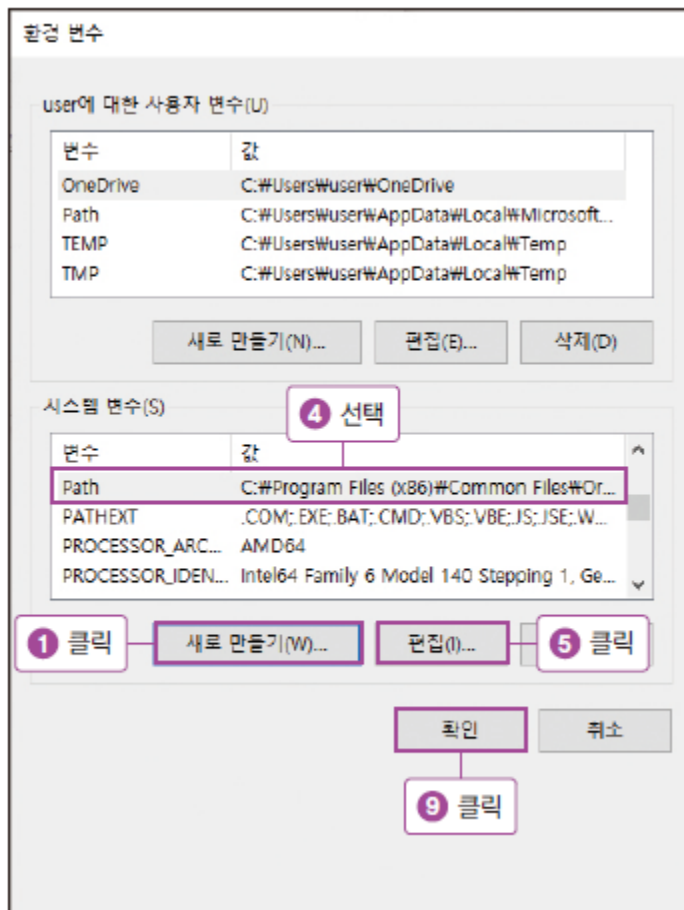
- (1) [환경 변수] 대화상자 열기



2. 안드로이드 개발 환경 구축하기

■ 환경 변수 설정


■ (2) 자바 환경 변수 설정하기



2. 안드로이드 개발 환경 구축하기

■ 환경 변수 설정

■ (3) 설정한 환경 변수 확인하기



A screenshot of a Windows Command Prompt window titled '명령 프롬프트'. The window shows the output of the 'java -version' command. The command is entered at the prompt 'C:\Users\Muser>'. The output displays the Java version as '1.8.0_341' and provides details about the Java(TM) SE Runtime Environment and the Java HotSpot(TM) 64-Bit Server VM. A pink box highlights the command 'java -version', and a pink arrow points from it to a pink box containing the Korean word '입력' (Input).

```
Microsoft Windows [Version 10.0.19044.2251]
(c) Microsoft Corporation. All rights reserved.

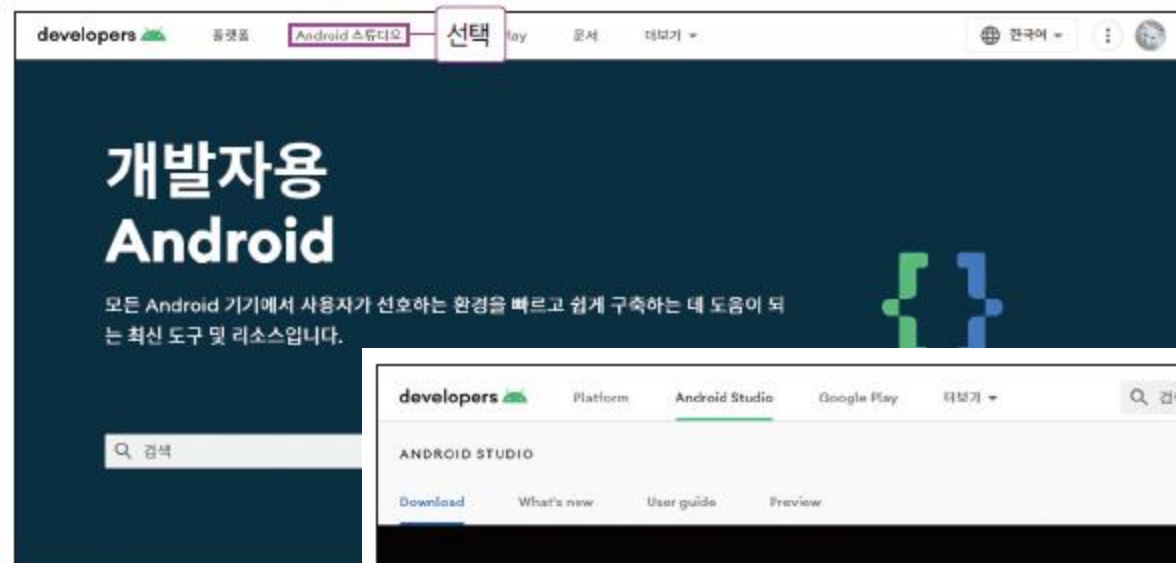
C:\Users\Muser>java -version
java version "1.8.0_341"
Java(TM) SE Runtime Environment (build 1.8.0_341-b10)
Java HotSpot(TM) 64-Bit Server VM (build 25.341-b10, mixed mode)

C:\Users\Muser>
```

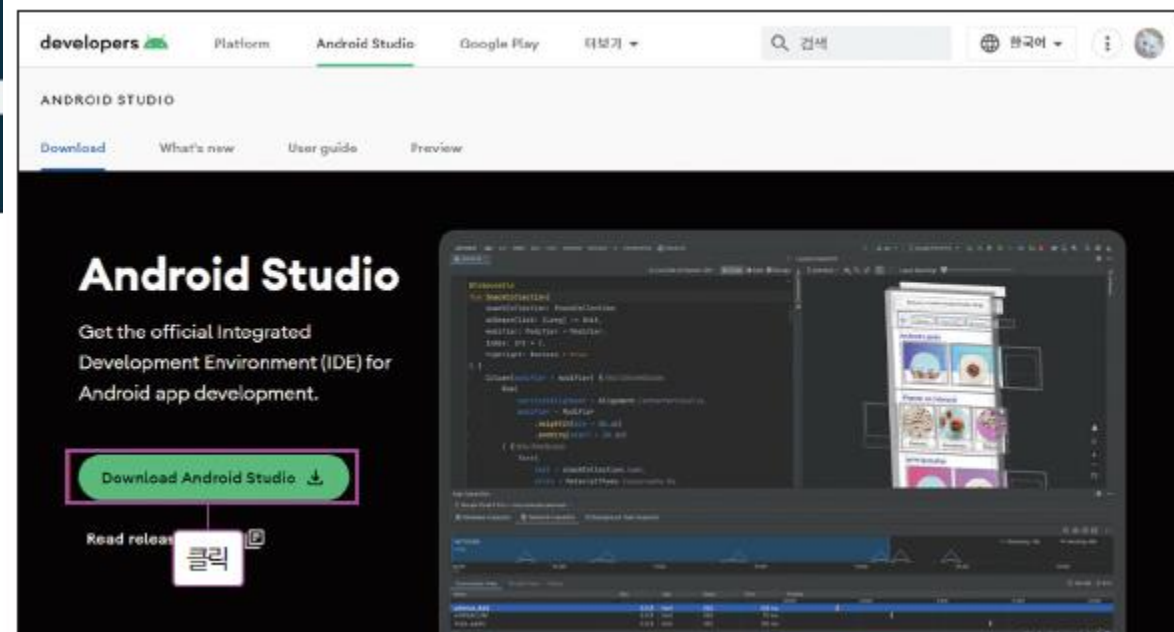
2. 안드로이드 개발 환경 구축하기

■ 안드로이드 스튜디오 설치

- (1) 안드로이드 개발자 사이트 접속하기



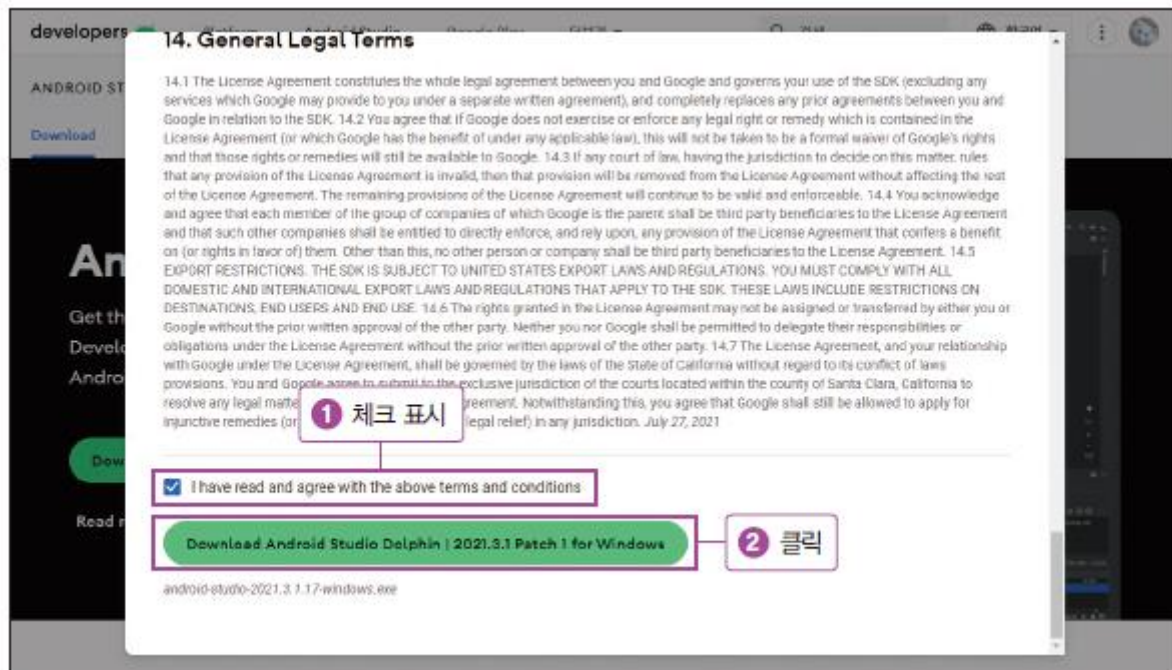
- (2) 안드로이드 스튜디오 설치 파일 선택하기



2. 안드로이드 개발 환경 구축하기

■ 안드로이드 스튜디오 설치

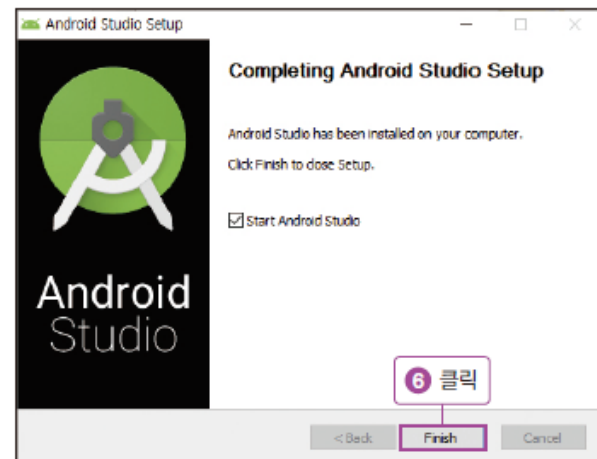
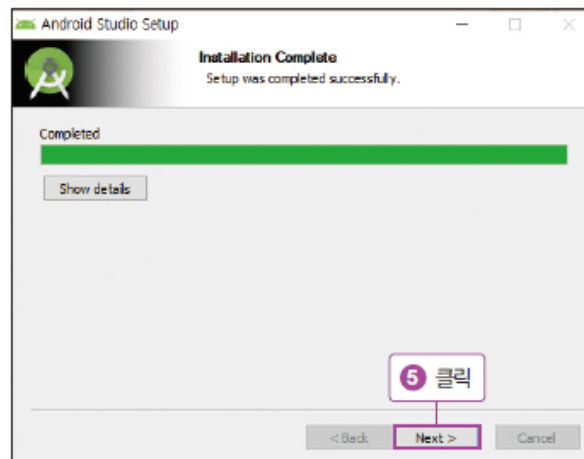
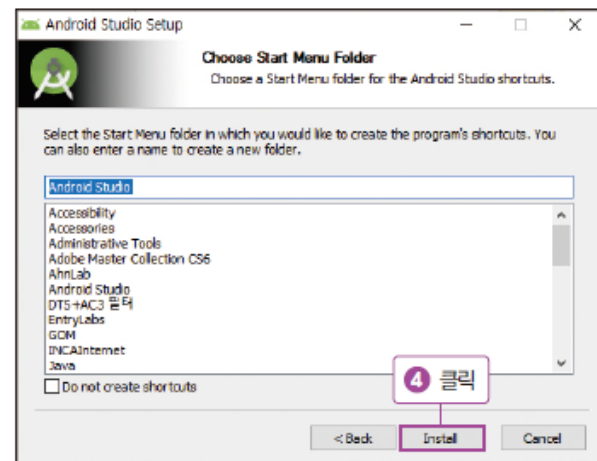
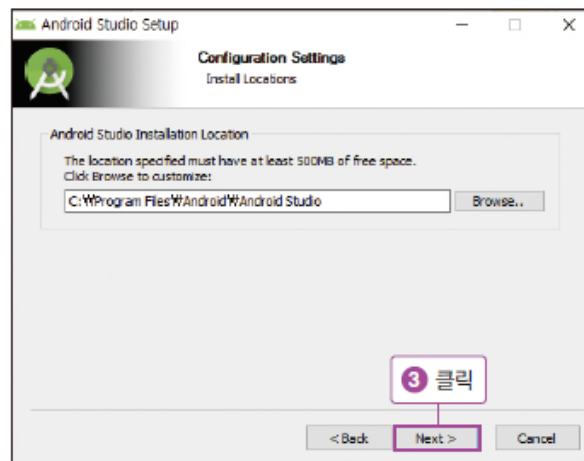
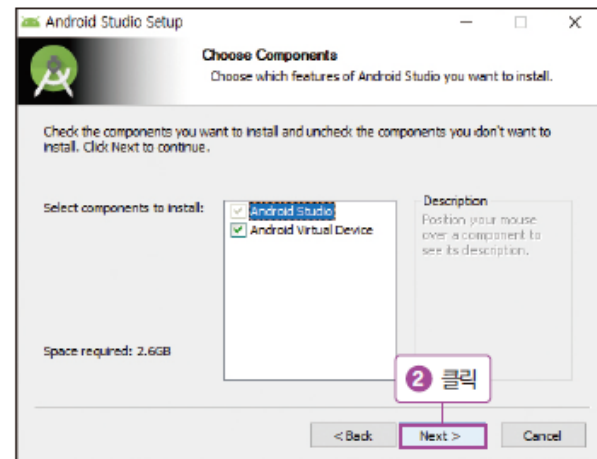
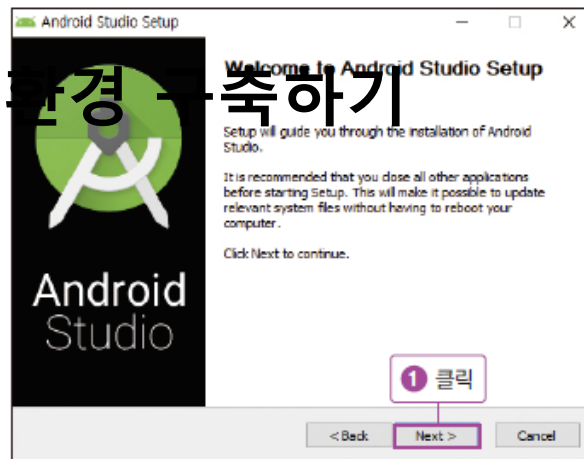
- (3) 사용 약관 동의 및 설치 파일 다운로드하기



2. 안드로이드 개발 환경 구축하기

■ 안드로이드 스튜디오 설치

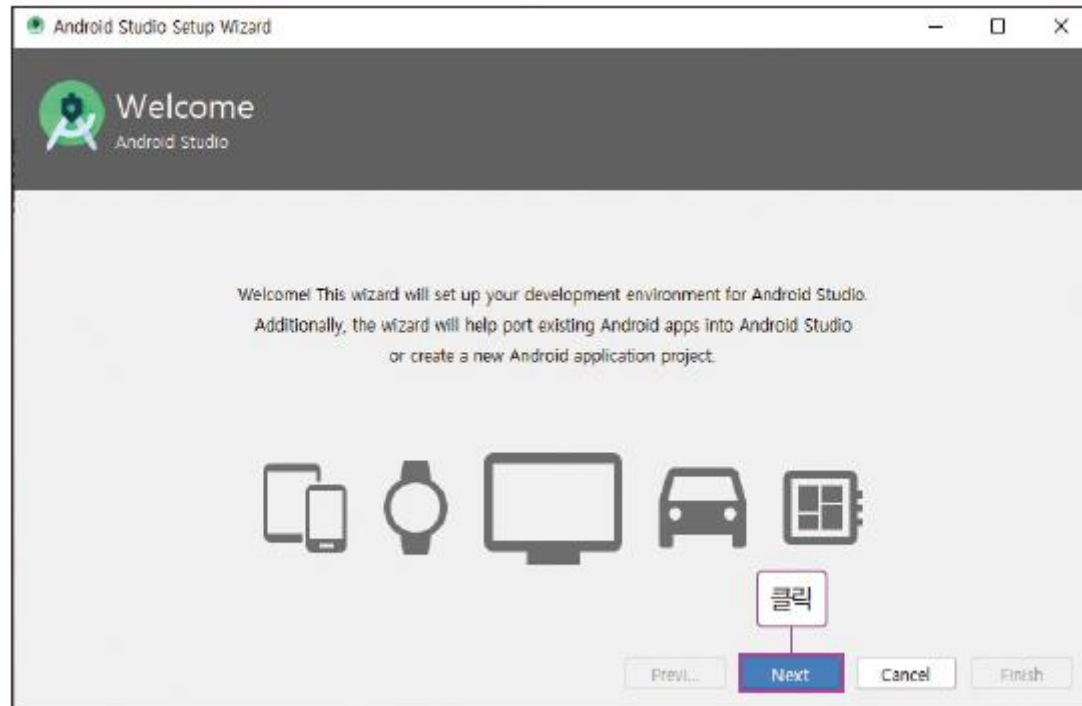
- (4) 안드로이드 설치 완료하기



2. 안드로이드 개발 환경 구축하기

■ 안드로이드 스튜디오 환경 설정

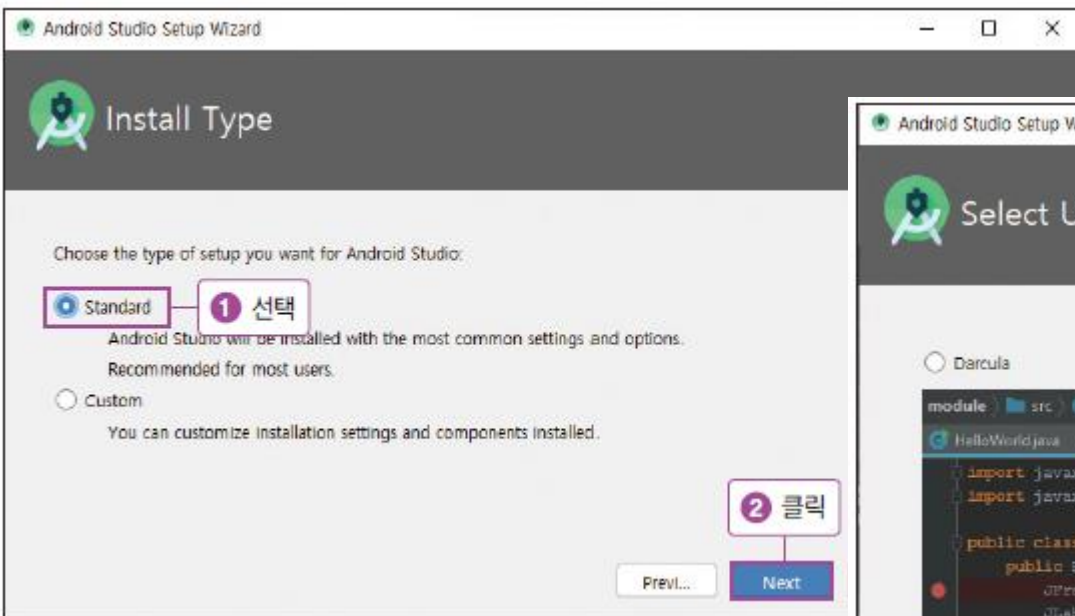
- (1) 안드로이드 설정 읽어오고 개발 환경 설정 마법사 실행하기



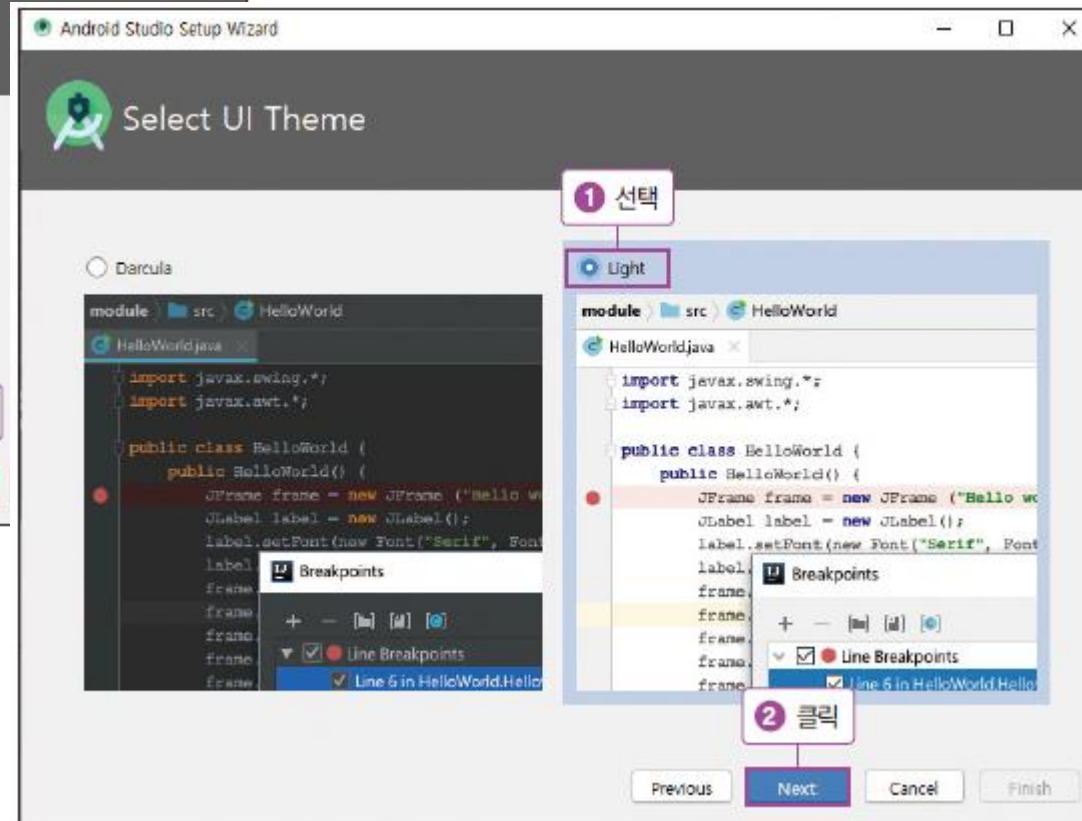
2. 안드로이드 개발 환경 구축하기

■ 안드로이드 스튜디오 환경 설정

- ## ■ (2) 개발 환경 설정 방식 결정하기



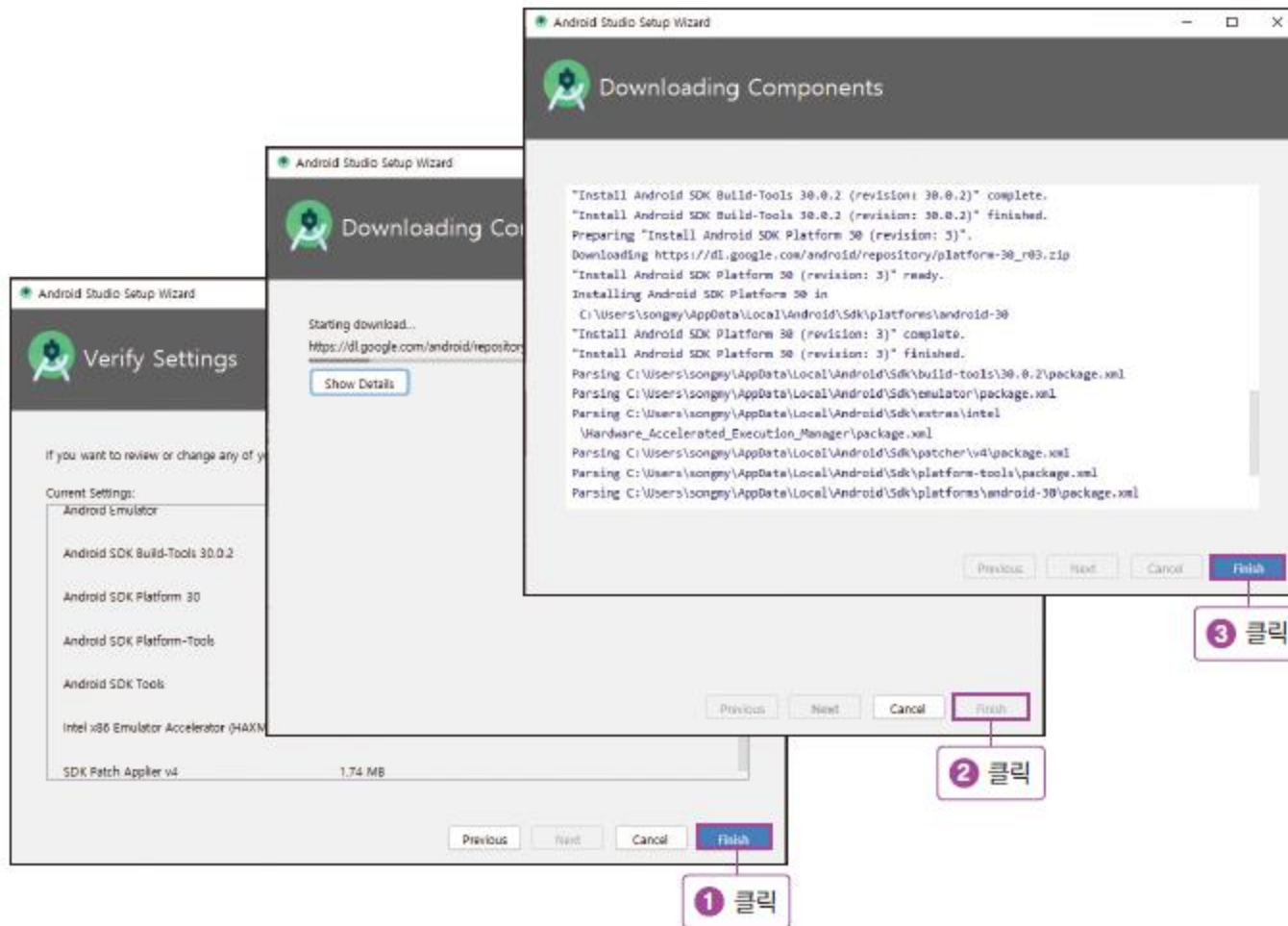
- ### ■ (3) UI 테마 설정하기



2. 안드로이드 개발 환경 구축하기

■ 안드로이드 스튜디오 환경 설정

- (4) 선택 사항 확인 및 구성 요소 설치 마치기



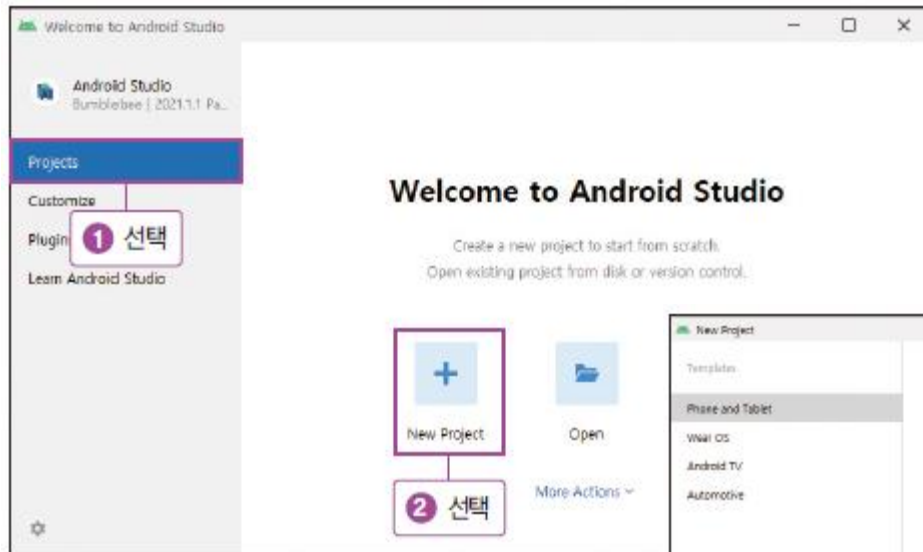
03

안드로이드 프로젝트 시작하기

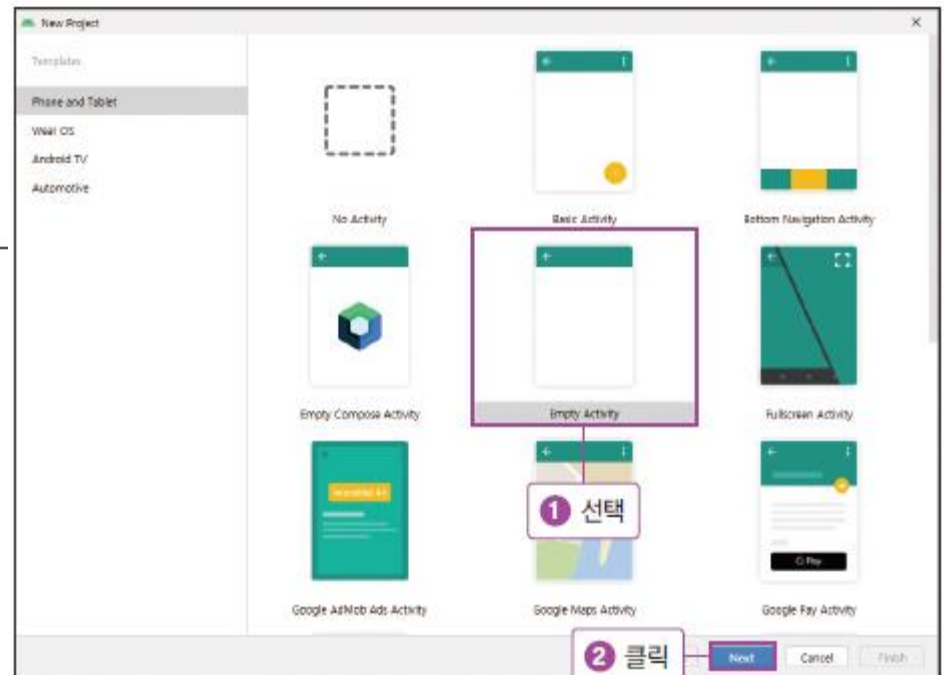
3. 안드로이드 프로젝트 시작하기

■ 안드로이드 프로젝트 생성

- (1) 안드로이드 스튜디오 프로젝트 생성하기



- (2) 프로젝트 선택하기



3. 안드로이드 프로젝트 시작하기

■ 안드로이드 프로젝트 생성

- (3) 안드로이드 스튜디오 프로젝트 생성하기

New Project

Empty Activity

Creates a new empty activity

Name: HelloWorld **1 입력**

Package name: com.androidbook.helloworld **2 입력**

Save location: C:\Users\user\AndroidStudioProjects\HelloWorld **3 지정**

Language: Java **4 선택**

Minimum SDK: API 21: Android 5.0 (Lollipop) **5 선택**

i Your app will run on approximately 98.8% of devices.
[Help me choose](#)

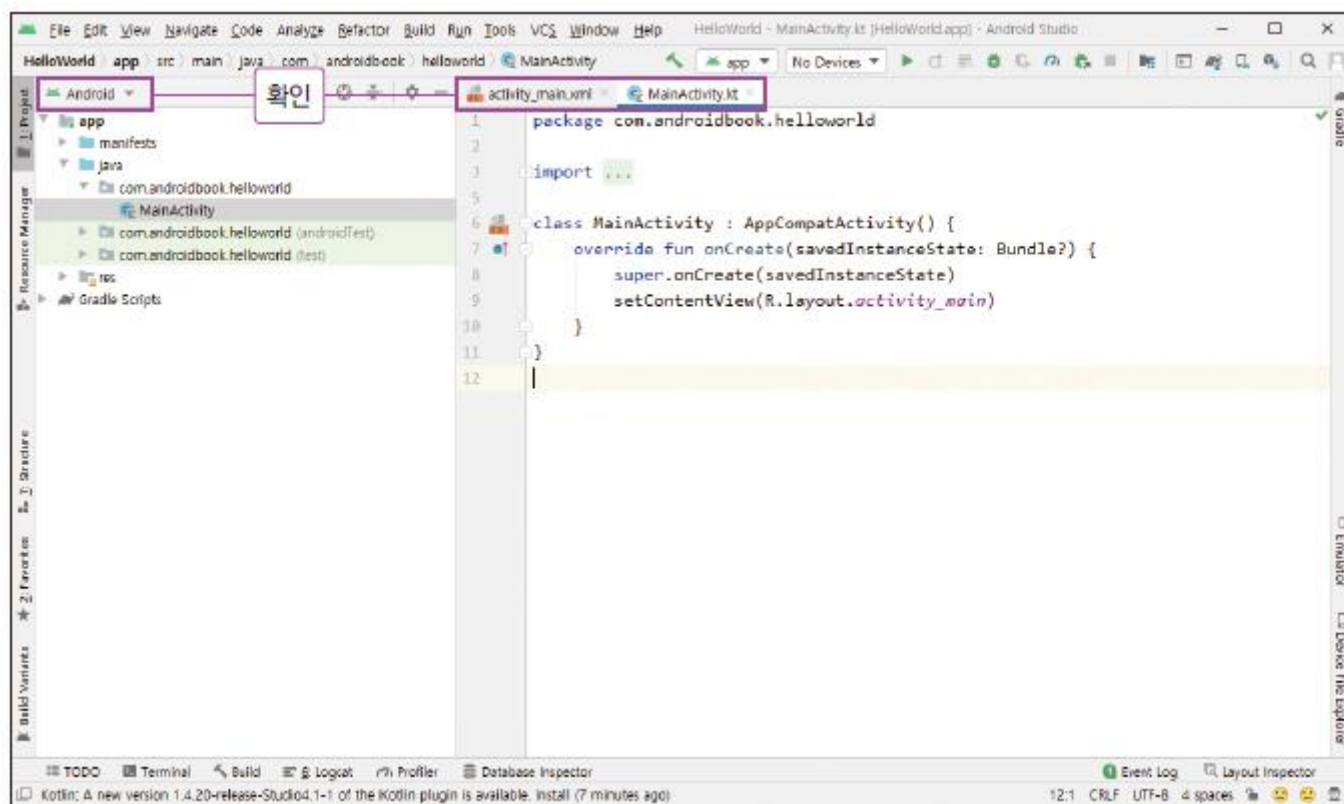
☐ Use legacy android.support libraries **?**
Using legacy android.support libraries will prevent you from using the latest Play Services and Jetpack libraries

Previous Next Cancel Finish

3. 안드로이드 프로젝트 시작하기

■ 안드로이드 프로젝트 생성

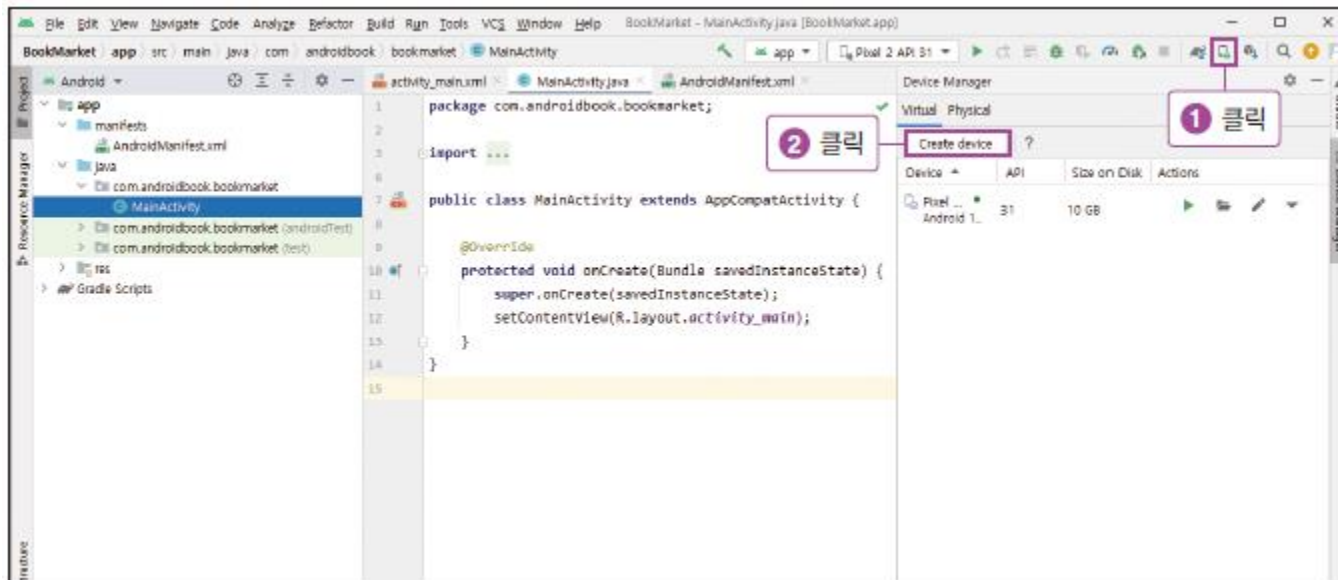
- (4) 프로젝트 생성 결과 확인하기



3. 안드로이드 프로젝트 시작하기

■ 에뮬레이터 생성

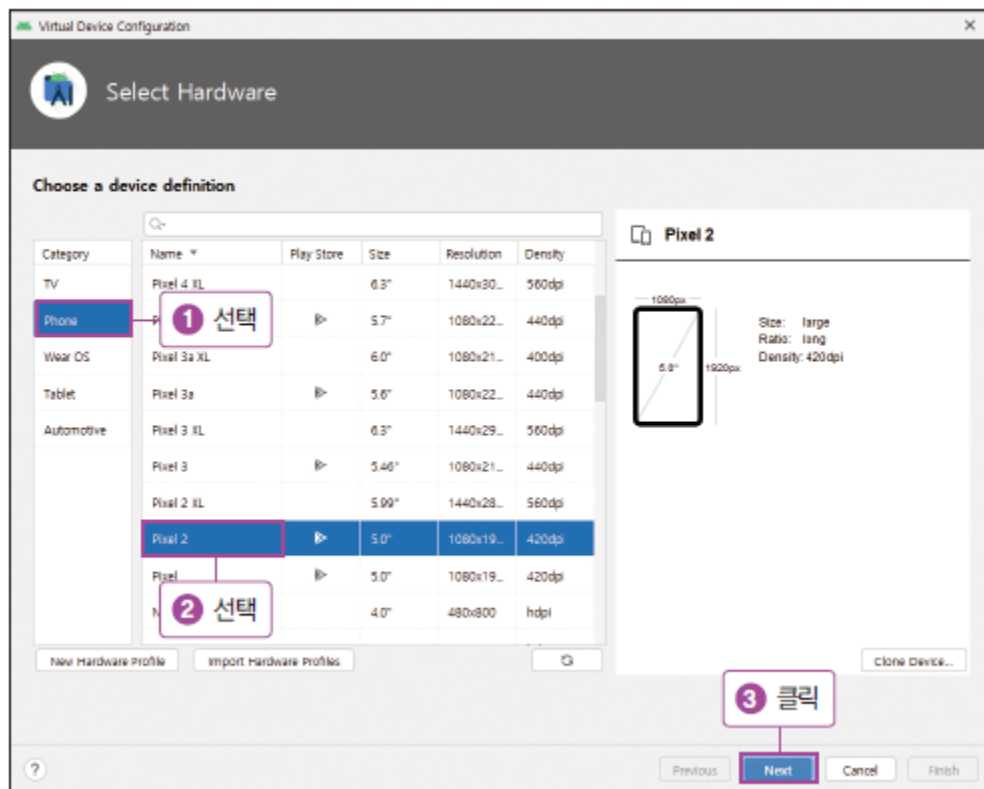
- (1) 애플레이터 생성하기



3. 안드로이드 프로젝트 시작하기

■ 에뮬레이터 생성

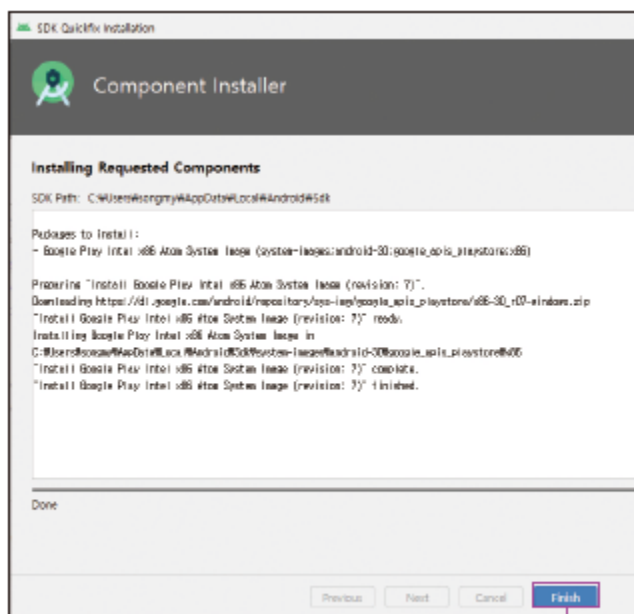
- (2) 하드웨어 장치 선택하기



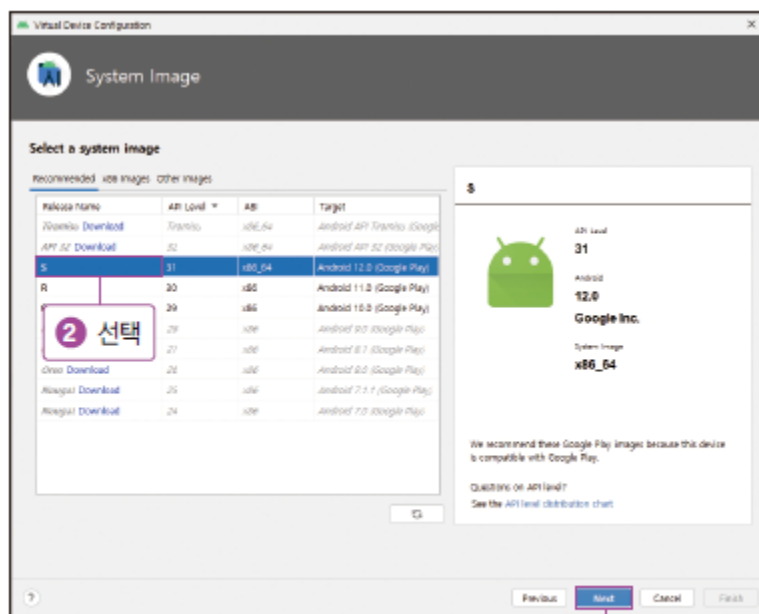
3. 안드로이드 프로젝트 시작하기

■ 에뮬레이터 생성

- ### ■ (3) 시스템 이미지 선택하기



1. 목적

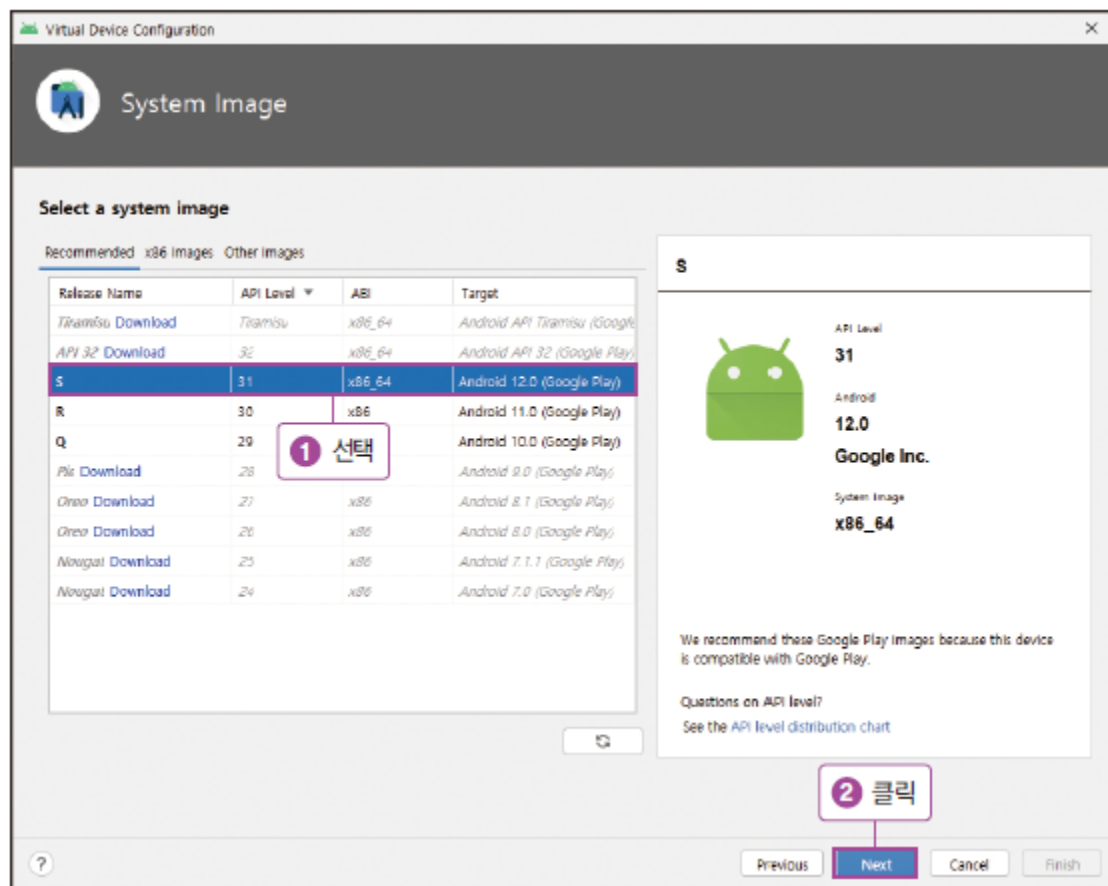


3. 불리

3. 안드로이드 프로젝트 시작하기

■ 에뮬레이터 생성

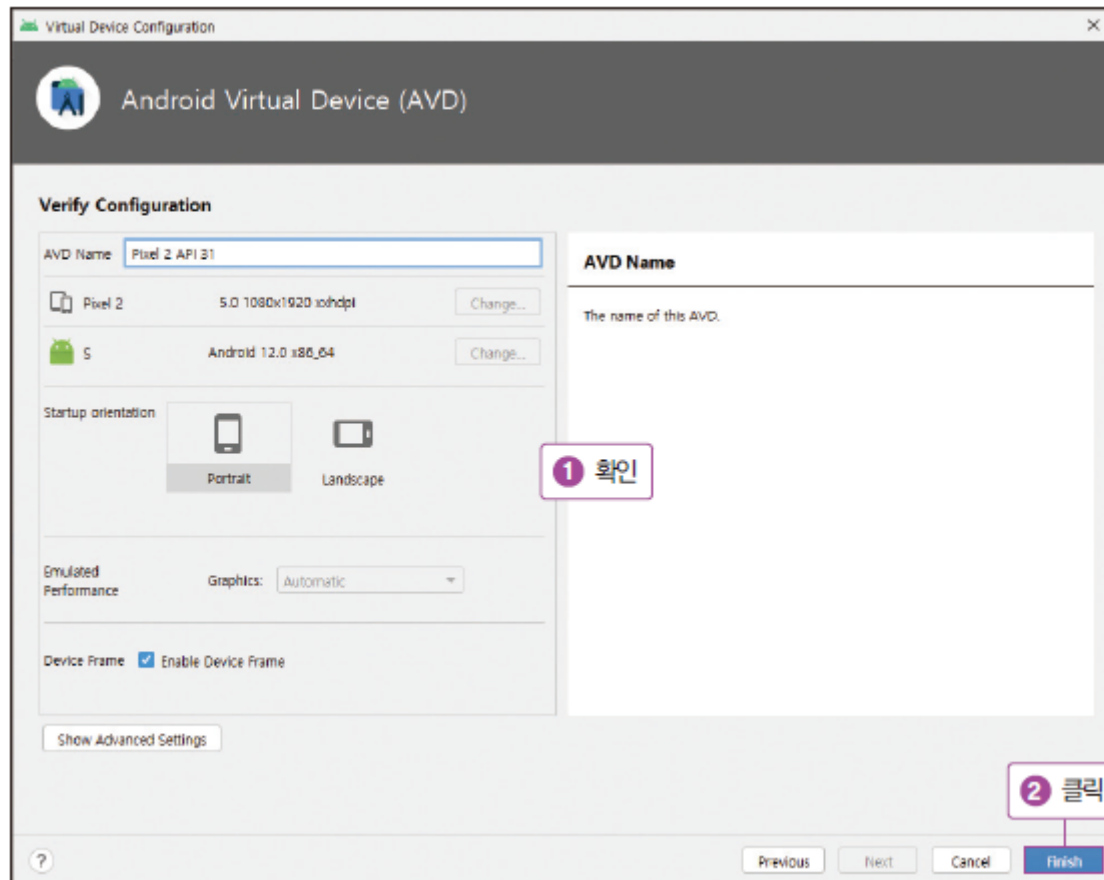
- (4) 설치할 시스템 이미지 선택하기



3. 안드로이드 프로젝트 시작하기

■ 에뮬레이터 생성

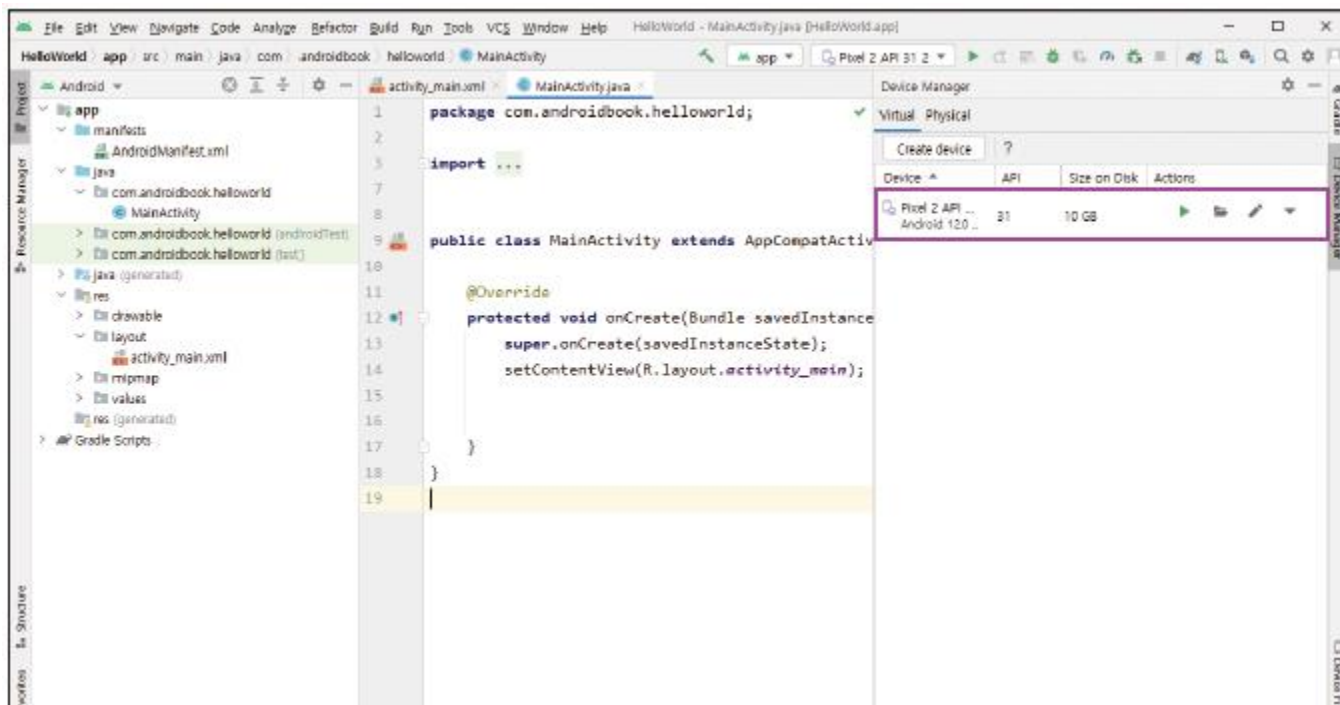
- (5) AVD 설정 확인하기



3. 안드로이드 프로젝트 시작하기

■ 에뮬레이터 생성

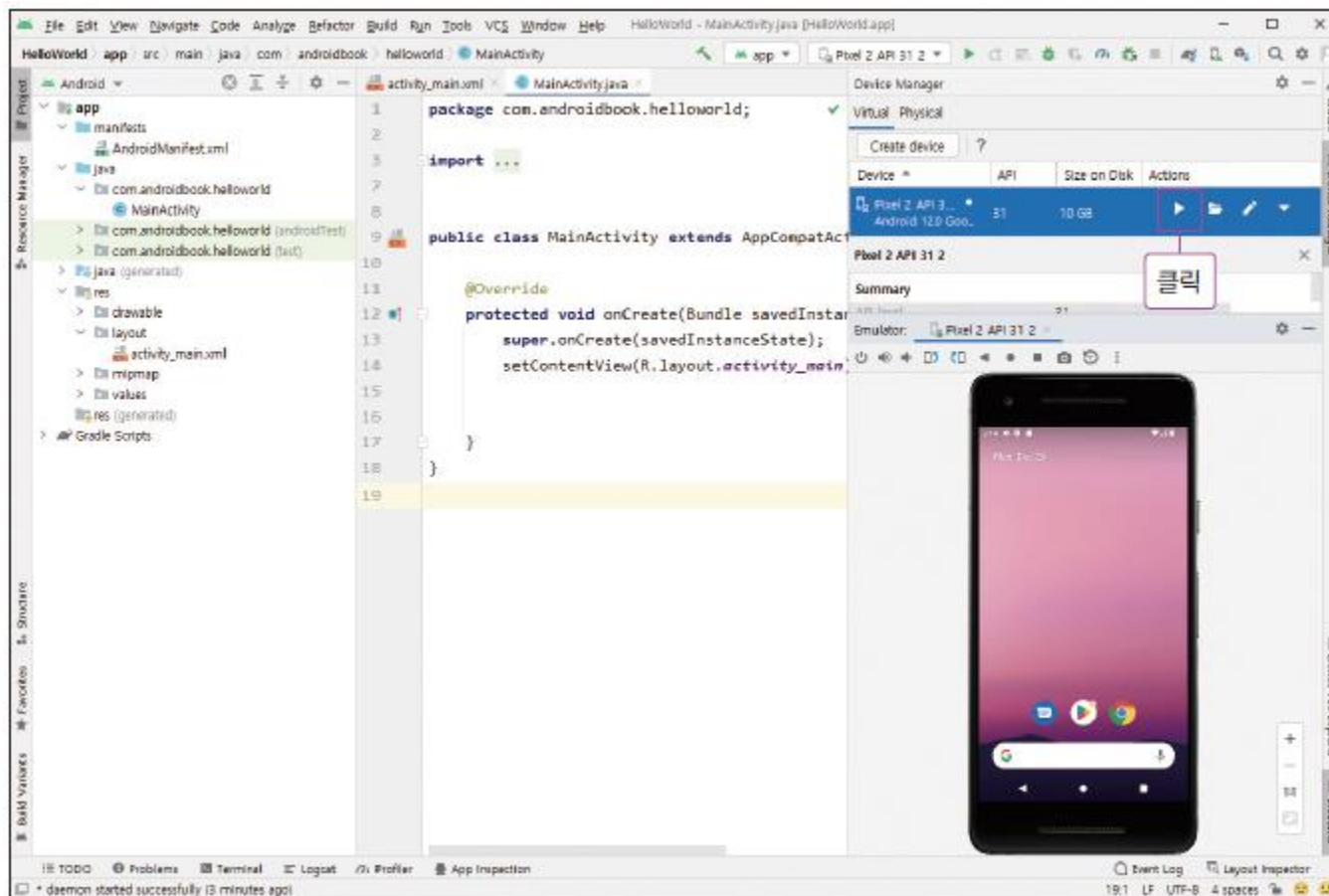
- (6) AVD 생성 확인하기



3. 안드로이드 프로젝트 시작하기

■ 안드로이드 프로젝트 실행

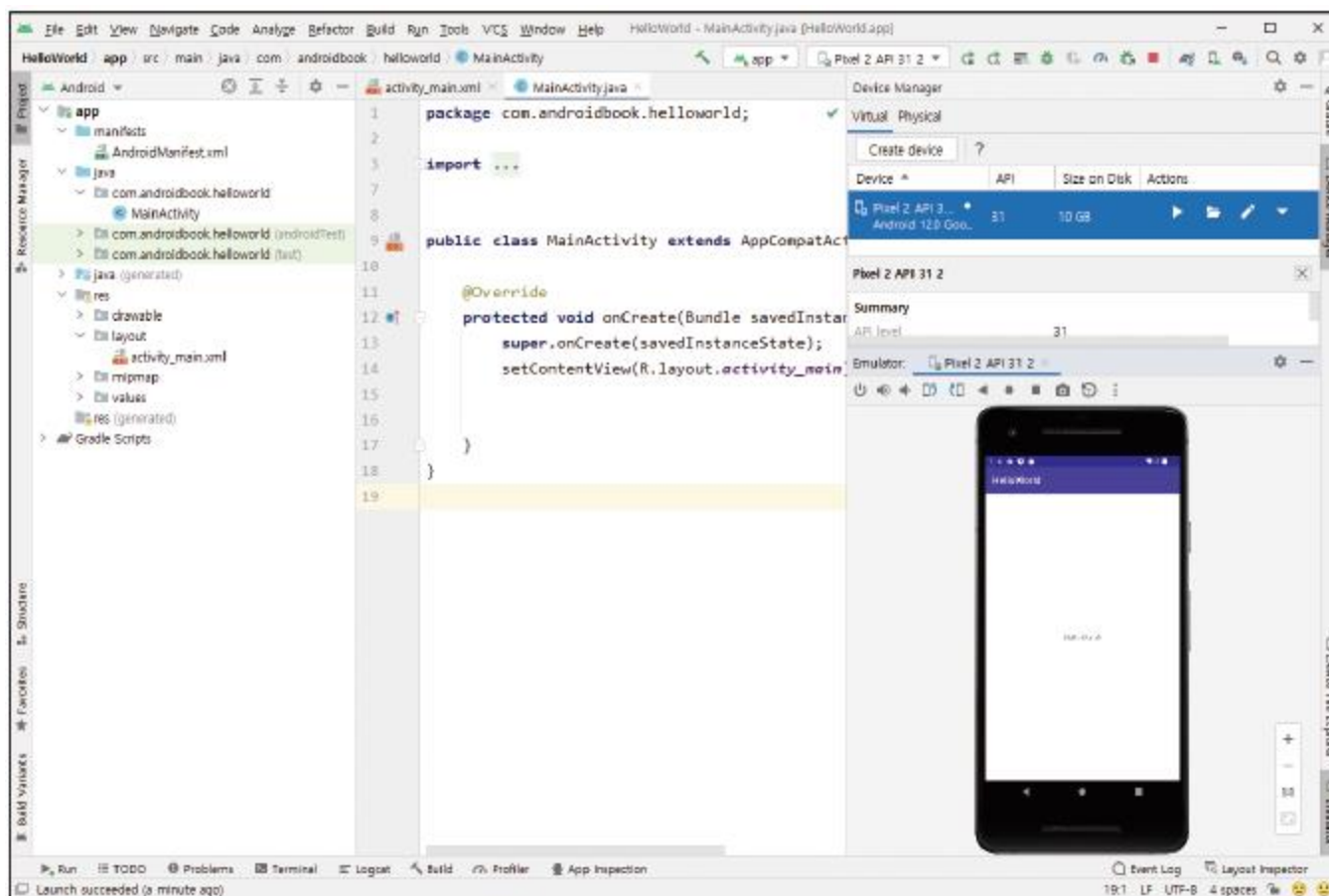
■ (1) 에뮬레이터 실행하기



3. 안드로이드 프로젝트 시작하기

■ 안드로이드 프로젝트 실행

■ (2) 프로젝트 실행 결과 확인하기



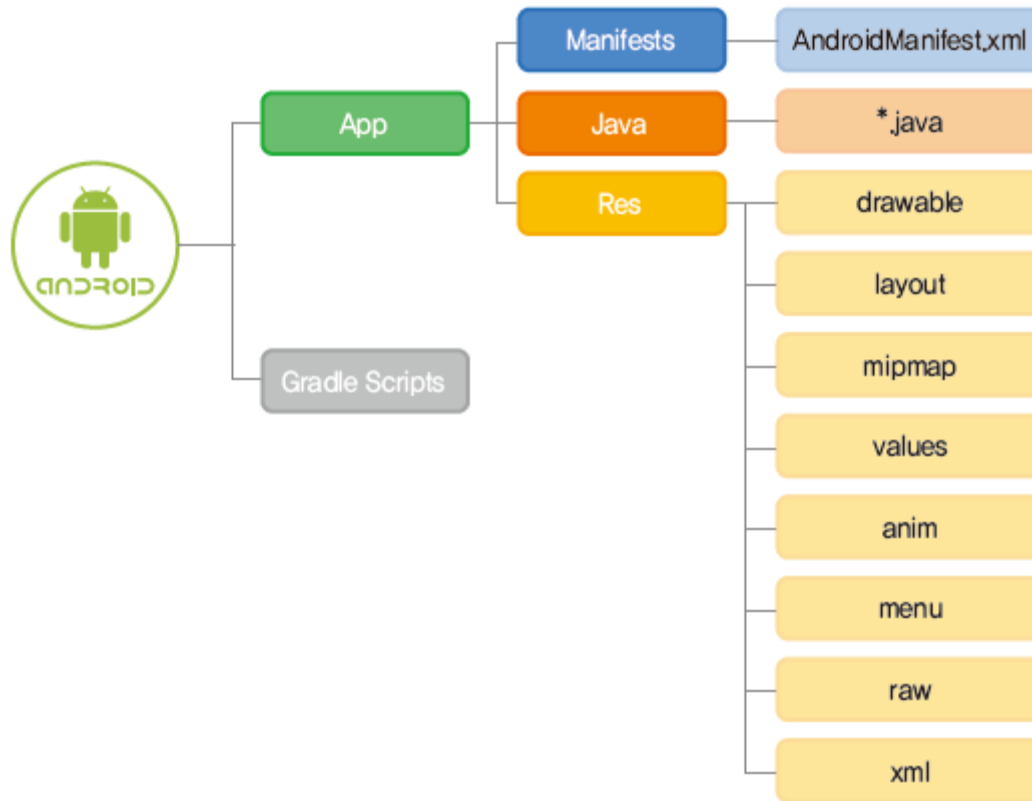
04

안드로이드 프로젝트

구조 살펴보기

4. 안드로이드 프로젝트 구조 살펴보기

■ 안드로이드 프로젝트 구성



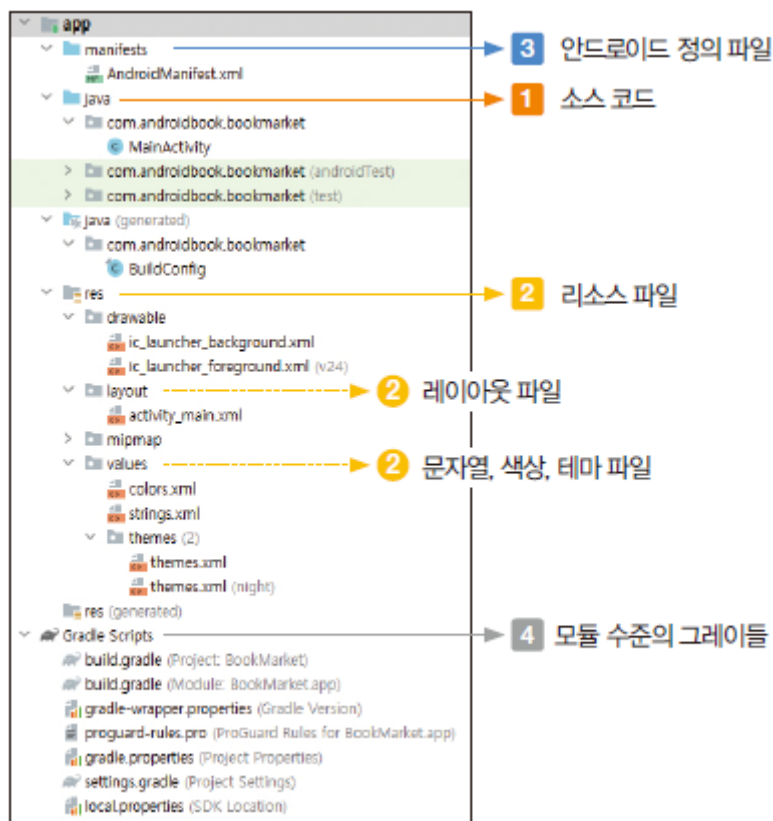
4. 안드로이드 프로젝트 구조 살펴보기

표 1-2 안드로이드 프로젝트 폴더 구성

폴더명	설명
manifests	안드로이드 정의 파일 AndroidManifest.xml이 있습니다. 이 파일에는 최소 안드로이드 버전, 인터넷 액세스 권한, 전화 권한 등 안드로이드 장치 기능에 대한 액세스 권한과 같은 안드로이드 애플리케이션 정보가 포함되어 있습니다.
java	소스 코드를 나타내며, 자바 소스 파일을 포함합니다. 기본으로 MainActivity.java 소스 파일이 포함되어 있습니다. 리소스 파일을 나타내며 그림, XML 파일 등과 같은 리소스 파일을 저장할 수 있습니다. 기본으로 Drawable, Layout 및 Values와 같은 폴더가 포함되어 있습니다.
res	drawable 폴더 다양한 그래픽 파일을 저장하는 데 사용됩니다.
	layout 폴더 텍스트 상자, 버튼 등과 같은 다양한 안드로이드 객체가 화면에서 구성되는 방식을 정의하는 XML 레이아웃 파일을 배치하는 데 사용됩니다.
	values 폴더 제목, 레이블 등과 같은 다양한 문자열 값을 저장하는 XML 파일에 사용됩니다.
	anim 폴더 애니메이션 객체로 컴파일된 XML 파일에 사용됩니다.
	menu 폴더 애플리케이션 메뉴에서 XML 파일을 정의하는 데 사용됩니다.
	raw 폴더 MP3 또는 Ogg 파일 등과 같은 미디어 파일을 저장하는 데 사용됩니다.
	xml 폴더 애플리케이션 구성 요소를 구성하는 데 사용됩니다.
Gradle Scripts	애플리케이션 빌드에 사용하는 Gradle 툴을 편리하게 사용할 수 있도록 도와주는 스크립트 컨테이너입니다. Gradle Wrapper는 적절한 Gradle 바이너리를 자동으로 다운로드해주므로, 별도로 Gradle을 설치하지 않아도 빌드를 진행할 수 있도록 도와줍니다.

4. 안드로이드 프로젝트 구조 살펴보기

■ 생성한 HelloWorld 프로젝트의 주요 구조



4. 안드로이드 프로젝트 구조 살펴보기

■ 소스 코드 - MainActivity.java

```
package com.androidbook.helloworld;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

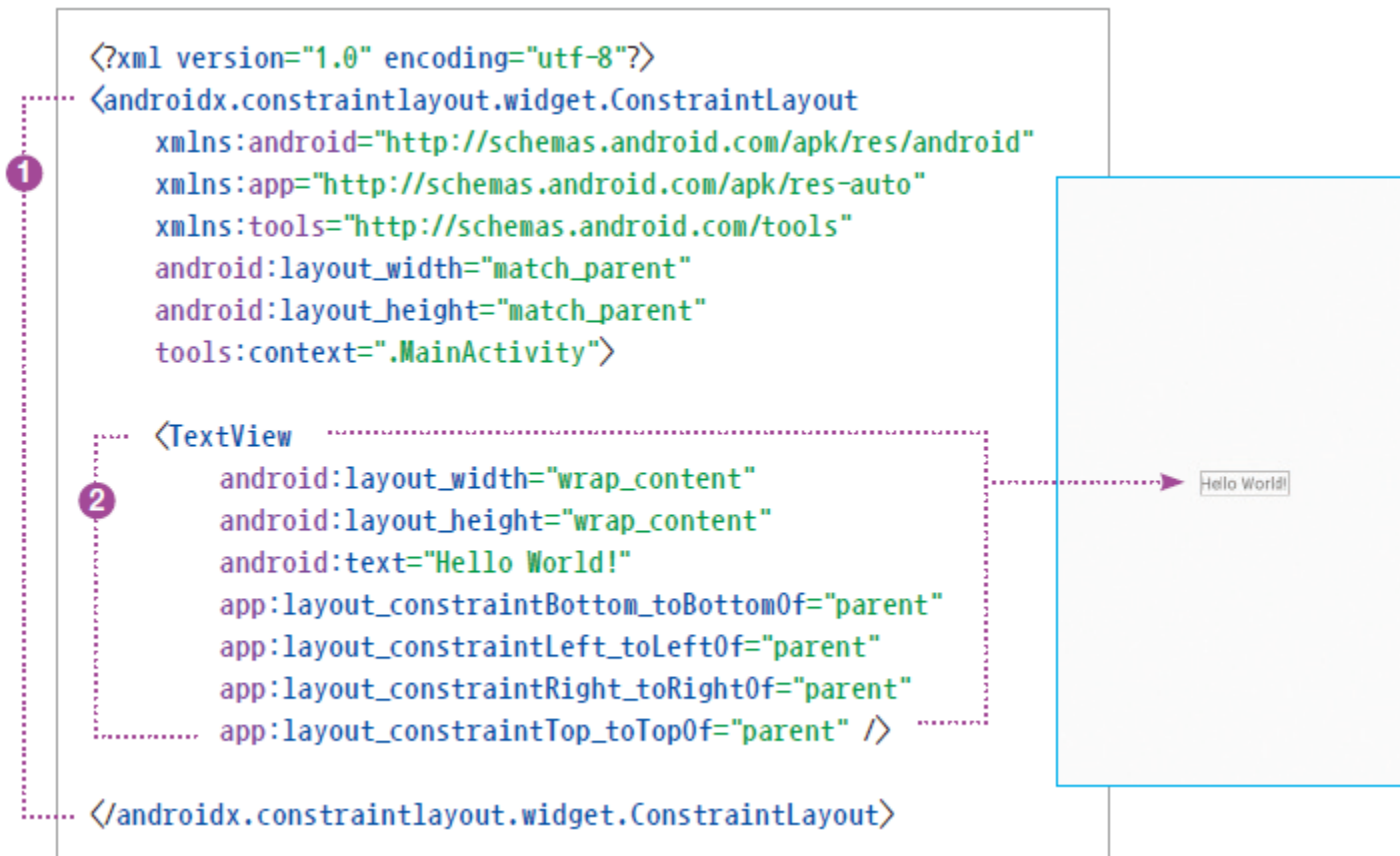
1 public class MainActivity extends AppCompatActivity {

    @Override
2     protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
3         setContentView(R.layout.activity_main);
    }
}
```


4. 안드로이드 프로젝트 구조 살펴보기

■ 리소스

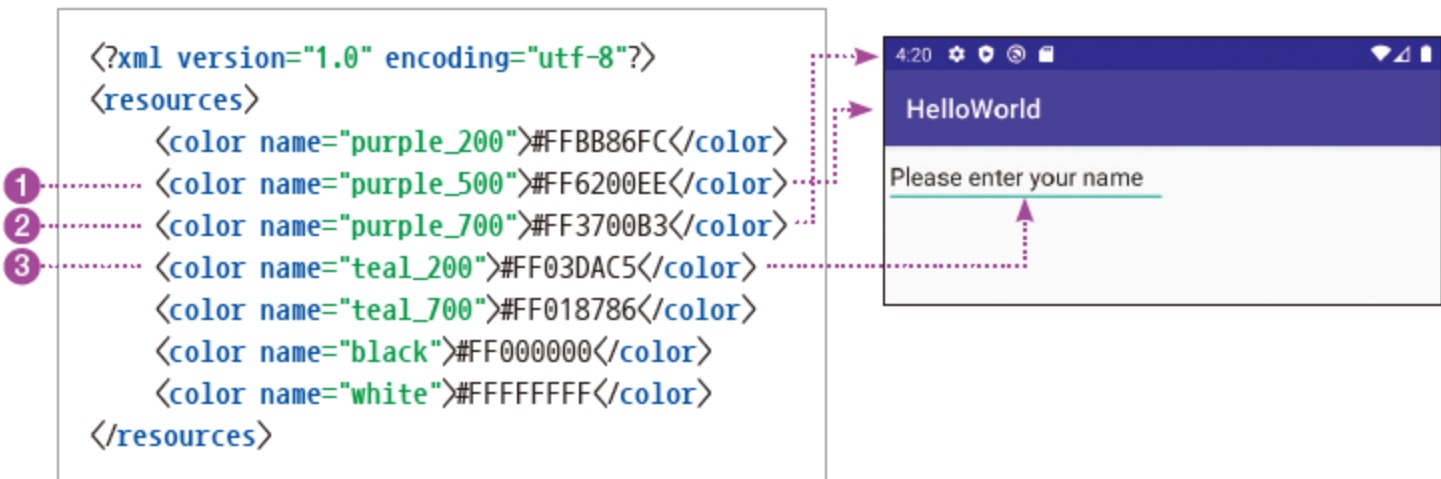
- 레이아웃 파일 - layout\activity_main.xml



4. 안드로이드 프로젝트 구조 살펴보기

■ 리소스

- 색상 리소스 파일 - values\colors.xml



4. 안드로이드 프로젝트 구조 살펴보기

■ 리소스

- 문자열 리소스 파일 - values\strings.xml

```
<resources>  
    <string name="app_name">HelloWorld</string>  
</resources>
```

4. 안드로이드 프로젝트 구조 살펴보기

■ 리소스

- 테마 리소스 파일 - themes\themes.xml

```
<resources xmlns:tools="http://schemas.android.com/tools">
  <!-- Base application theme. -->
  1 <style name="Theme.HelloWorld" parent="Theme.MaterialComponents.DayNight.DarkActionBar">
    2 <!-- Primary brand color. -->
    <item name="colorPrimary">@color/purple_500</item>
    <item name="colorPrimaryVariant">@color/purple_700</item>
    <item name="colorOnPrimary">@color/white</item>
    <!-- Secondary brand color. -->
    <item name="colorSecondary">@color/teal_200</item>
    <item name="colorSecondaryVariant">@color/teal_700</item>
    <item name="colorOnSecondary">@color/black</item>
    <!-- Status bar color. -->
    <item name="android:statusBarColor" tools:targetApi="1">?attr/colorPrimaryVariant</item>
    <!-- Customize your theme here. -->
  </style>
</resources>
```



4. 안드로이드 프로젝트 구조 살펴보기

■ 리소스

- 안드로이드 정의 파일 - AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.androidbook.helloworld" >

    1 <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        2 android:label="HelloWorld"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Helloworld" >

        <activity
            android:name=".MainActivity"
            android:exported="true" >
            3 <intent-filter>
                5 <action android:name="android.intent.action.MAIN" />
                4 6 <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

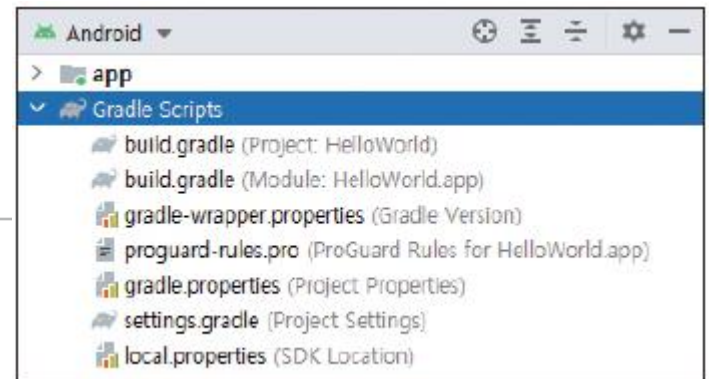
</manifest>
```

4. 안드로이드 프로젝트 구조 살펴보기

■ 리소스

- 모듈 수준의 그레이들
 - build.gradle(Module HelloWorld.app)

```
plugins {  
    id 'com.android.application'  
}  
  
android {  
    ① ..... compileSdk 32  
  
    defaultConfig {  
        ② ..... applicationId "com.androidbook.helloworld"  
        ③ ..... minSdk 21  
        ④ ..... targetSdk 32  
        ⑤ ..... versionCode 1  
        ..... versionName "1.0"  
  
        testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"  
    }  
}
```



4. 안드로이드 프로젝트 구조 살펴보기

■ 리소스

- 모듈 수준의 그레이들 - build.gradle(Module HelloWorld.app)

```
}

buildTypes {
    release {
        minifyEnabled false
        proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'),
            'proguard-rules.pro'
    }
}

compileOptions {
    sourceCompatibility JavaVersion.VERSION_1_8
    targetCompatibility JavaVersion.VERSION_1_8
}

6 dependencies {

    implementation 'androidx.appcompat: appcompat:1.4.1'
    implementation 'com.google.android.material: material:1.5.0'
    implementation 'androidx.constraint layout: constraint layout: 2.1.3'
    testImplementation 'junit:junit:4.13.2'
    androidTestImplementation 'androidx.test.ext:junit:1.1.3'
    androidTestImplementation 'androidx.test.espresso: espresso-core: 3.4.0'
}
```

PROJECT. [도서 쇼핑몰] 프로젝트 시작하기

