



소프트웨어 실습 2

Chapter 1

2022.09

YangMin Lee

S03(공대 1호관), 301-01호실 (yanwenry)

manson23@nate.com

01 안드로이드 개요와 개발 환경 설치



학습목표

- ❖ 안드로이드의 개념과 특징을 이해한다.
- ❖ 안드로이드 개발 환경을 설치한다.

차례

1. 안드로이드의 개요
2. 안드로이드의 개발 환경
3. 안드로이드의 개발 환경 설치

01 스마트폰의 개요

■ 스마트폰

- 통화 기능 + 컴퓨터 + 다양한 기능 내장(MP3, 카메라, DMB, GPS 등).

■ 스마트폰의 역사

- 1992년 IBM사의 사이먼 (최초)
- 1996년 노키아 9000
- 2002년 마이크로소프트 포켓PC
- 2007년 아이폰
- 2008년 안드로이드 폰
- 2010년 윈도우폰7



그림 1-1 스마트폰(픽셀4, 아이폰11, 윈도우폰10)

01 스마트폰의 개요

- 스마트폰 운영체제의 전세계 점유율(2019년 기준)
 - 안드로이드: 85%, 아이폰: 11%, 윈도우폰: 0.01%

표 1-1 스마트폰 개발 환경 비교

구분	안드로이드	아이폰	윈도폰
개발 언어	Java, Kotlin, C++	Objective C	C#, VB,Net
개발 운영체제	Windows, Linux, Mac OS	Mac OS	Windows 8/8.1/10
개발 툴	Eclipse, Android Studio	Xcode	Visual Studio 2013 이상
지원 장치	안드로이드폰, 안드로이드 태블릿, 안드로이드 스마트워치, 안드로이드 TV	아이폰, 아이팟(iPod), 아이패드(iPad), 애플워치	윈도폰
대표 제품	삼성 갤럭시 S/노트 시리즈, LG G/G Pro/V 시리즈, 구글 넥서스/픽셀 시리즈	아이폰 시리즈, 아이패드 시리즈	노키아 루미아(Lumia) 시리즈
최신 개발 버전	Android 10.0(Q)	iOS 13	윈도폰10
앱스토어	구글 플레이, 삼성 Apps, T스토어, 네이버 스토어 등	애플 앱스토어	Windows 스토어

02 안드로이드

■ 구글(Google)이 2007년 안드로이드사를 인수하면서 시작됨

표 1-2 안드로이드 버전의 변천사

출처: android.com

이미지	코드명	버전	API 레벨	발표 일자	비고
	알파(Alpha)	1.0	1	2008년 9월	2008년 9월 최초 발표
	베타(Beta)	1.1	2	2009년 2월	기존 문제 수정, API 변경, 통화 기능 수정
	컵케이크(Cupcake)	1.5	3	2009년 4월	동영상 녹화, 소프트 키보드 지원, 블루투스 지원, 애니메이션 효과
	도넛(Donut)	1.6	4	2009년 9월	안드로이드 마켓 개선, WVGA 해상도 지원, 갤러리 인터페이스, 다중 선택/삭제 지원
	이클레어(Eclair)	2.1	7	2010년 1월	하드웨어 최적화, 많은 해상도 지원, 구글 맵 향상, 가상 키보드 개선, 블루투스 2.1 지원
	프로요(Froyo)	2.2	8	2010년 5월	전반적인 성능 개선, USB 테더링 지원, 업데이트 기능 지원, 플래시 10.1 지원
	진저브레드(Gingerbread)	2.3	10	2010년 12월	UI 성능 개선, 향상된 응용 프로그램 관리, 인터넷 전화, 다양한 센서 지원
	허니콤(Honeycomb)	3.0	11	2011년 2월	태블릿 PC에 최적화
		3.1	12	2011년 5월	
		3.2	13	2011년 7월	
	아이스크림 샌드위치 (Ice Cream Sandwich)	4.0	14	2011년 10월	진저브레드와 허니콤을 통합, 스마트폰과 태블릿을 함께 지원
		4.0.3	15	2011년 11월	

02 안드로이드

- 구글(Google)이 2007년 안드로이드사를 인수하면서 시작됨

이미지	코드명	버전	API 레벨	발표 일자	비고
	젤리빈(Jelly Bean)	4.1 4.2 4.3	16 17 18	2012년 7월 2012년 12월 2013년 6월	아이스크림 샌드위치 기반 위에서 더 빠르고 부드러운 화면, 다양한 액세서리 지원
	킷캣(KitKat)	4.4 4.4w	19 20	2013년 10월 2014년 6월	메모리 관리 강화, GPU 가속화, UI 변경, 4.4w는 웨어러블 확장 지원
	롤리팝(Lollipop)	5.0 5.1	21 22	2014년 11월 2015년 3월	64bit 지원, 매터리얼(material) 디자인, 잠금 중에 알림 영역 표시 등
	마시멜로(Marshmallow)	6.0	23	2015년 10월	앱 권한 설정, 지문 인식 등
	누가(Nougat)	7.0 7.1	24 25	2016년 8월 2016년 10월	가상현실 지원, 3D 게임 최적화, 멀티태스킹 강화, 화면 해상도 조절 등
	오레오(Oreo)	8.0 8.1	26 27	2017년 8월 2017년 10월	PIP, 알림 기능, Java 8 지원, 자동 완성, 어댑티브 아이콘, 배터리 등
	파이(Pie)	9.0	28	2018년 8월	실내 위치 추적, 향상된 알림, 채널 설정, 멀티카메라, 고정 모드, 인공지능 확장 등
	Android Q	10.0	29	2019년 9월	라이브 캡션, 스마트 재생, 청각 보조, 동작 내비게이션, 어두운 테마, 개인 정보 제어 등

02 안드로이드

■ 안드로이드의 주요 기능_(1)

- 애플리케이션 프레임워크를 통해서 제공되는 API를 사용함으로써 코드를 재사용하여 효율적이고 빠른 애플리케이션 개발 가능
- 모바일 기기에 최적화된 달빅 또는 아트런타임 제공
- 2D 그래픽 및 삼차원 그래픽을 최적화하여 표현
- 모바일용 데이터베이스인 SQLite를 제공
- 각종 오디오, 비디오 및 이미지 형식을 지원
- 모바일 기기에 내장된 각종 하드웨어(블루투스, 카메라, 나침반, WiFi 등) 지원
- 이클립스 IDE 또는 Android Studio를 통해서 강력하고 빠른 개발 환경 제공

02 안드로이드

■ 안드로이드의 주요 기능_(2)

- 롤리팝(5.0) : 다양한 안드로이드 기기를 통합 지원
- 마시멜로(6.0) : 앱 권한 설정, 지문 인식 지원
- 누가(7.0) : 가상현실 지원 및 3D 게임, 알림 기 향상, 다중 창 열기 지원
- 오레오(8.0) : PIP, 알림, 자동 채우기, 배터리 강화 등을 지원
- 파이(9.0) : 실내 위치 추적, 향상된 알림, 멀티카메라, 인공지능 확장 등을 지원
- Android 10.0(Q) : 라이브 캡션, 스마트 재생, 청각 보조, 동작 내비게이션, 어두운 테마, 개인 정보 제어 등을 지원

02 안드로이드

■ 안드로이드의 특징

- ① 안드로이드의 핵심 커널(Kernel) : 리눅스(Linux)로 구성
- ② 안드로이드 애플리케이션 개발 언어 : Java 또는 Kotlin
- ③ 안드로이드 SDK에서 많은 라이브러리를 포함하고 있어 개발이 용이
- ④ 오픈 소스를 지향
→ 운영체제부터 관련 문서,
개발 도구 등 거의 모든 것을
무료로 사용 가능
- ⑤ 지속적인 업그레이드 제공

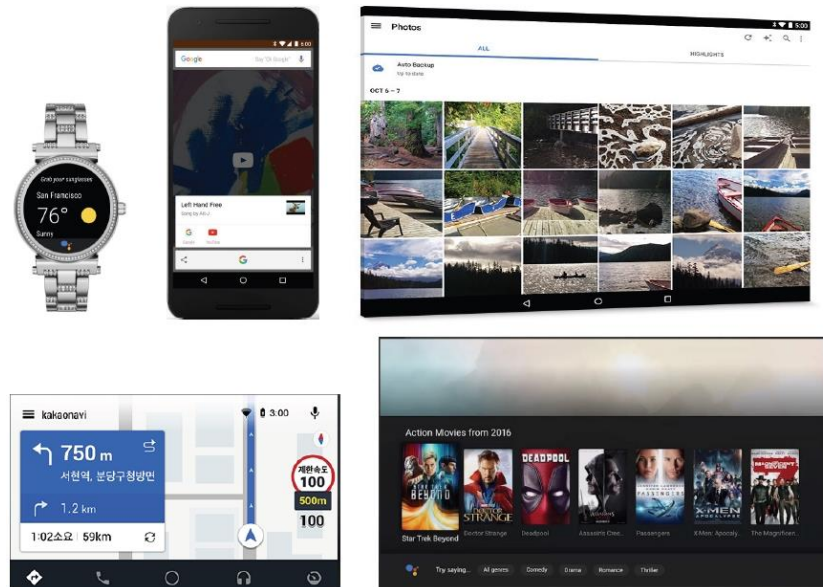


그림 1-2 다양한 안드로이드 기기(스마트워치, 스마트폰, 태블릿, 안드로이드 AUTO, 안드로이드 TV)

02 안드로이드

■ 안드로이드의 구조

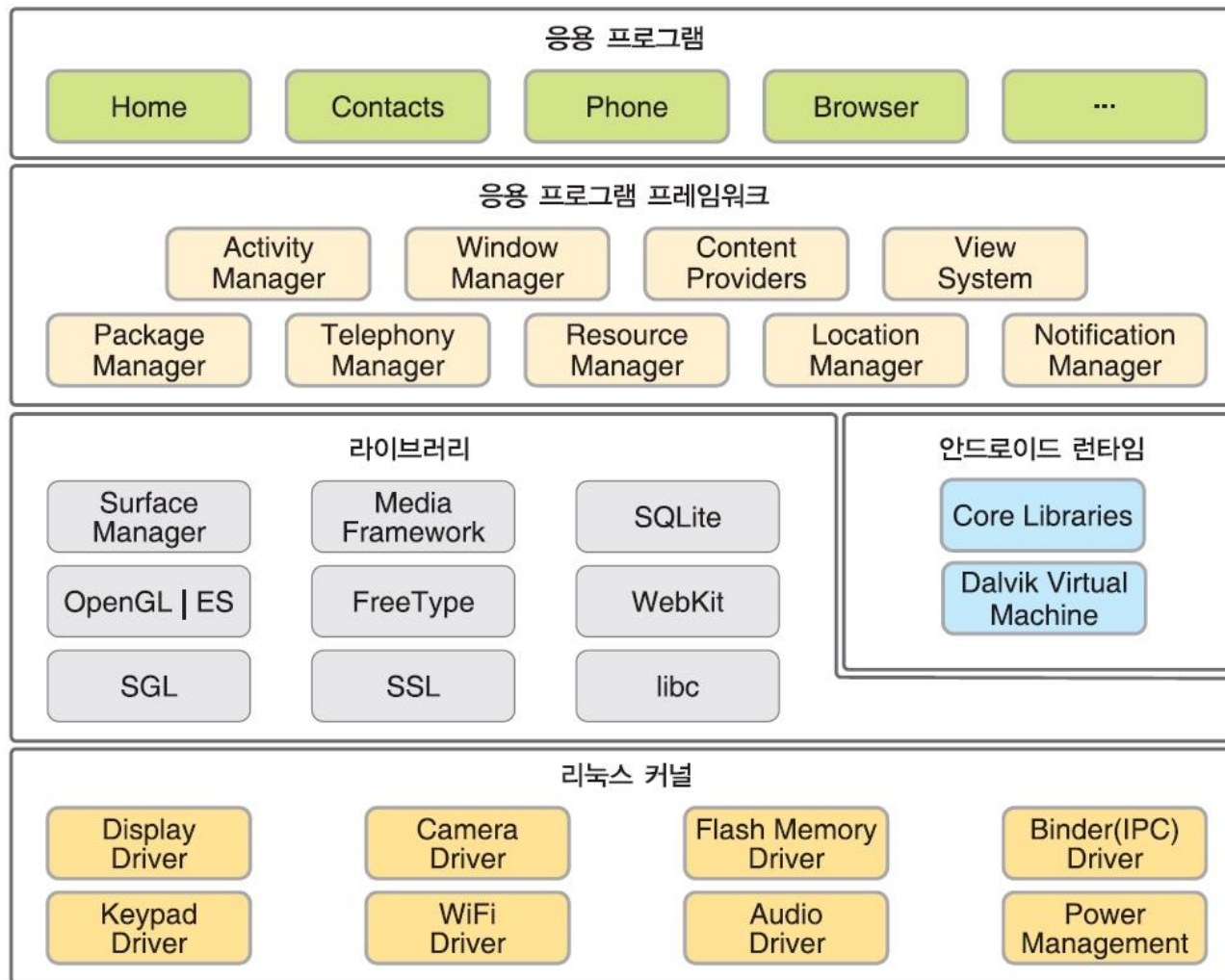


그림 1-3 안드로이드의 구조

02 안드로이드

① 응용 프로그램(Applications)

- 안드로이드 스마트폰에서 사용할 수 있는 일반적인 응용 프로그램
- 웹 브라우저, 달력, 구글맵, 연락처, 게임 등 사용자 입장에서 가장 많이 사용
- Java 또는 Kotlin으로 제작됨

② 응용 프로그램 프레임워크(Application Framework)

- 안드로이드 API가 존재하는 곳
- 안드로이드폰 하드웨어에 접근할 때 API를 통해서만 가능

02 안드로이드

③ 안드로이드 런타임(Android Runtime)

- Java 코어 라이브러리와 달빅 가상 머신 또는 아트 런타임으로 구성됨
- 안드로이드는 Java 또는 Kotlin 문법으로 프로그래밍하지만 Java 가상 머신을 사용하지 않고 이곳의 달빅 가상 머신이나 아트 런타임을 사용함

④ 라이브러리(Libraries)

- 안드로이드에서 사용되는 여러 시스템 라이브러리는 시스템 접근 때문에 Java 또는 Kotlin이 아닌 C로 작성
 - ➔ 성능이 뛰어나며 세밀한 조작 가능

⑤ 리눅스 커널(Linux Kernel)

- 하드웨어의 운영과 관련된 저수준의 관리 기능
- 메모리 관리, 디바이스 드라이버, 보안 등

01 안드로이드 개발을 위한 운영체제

- **Windows, Linux, Mac**
- Windows 개발 환경이 가장 보편적임
- 안드로이드 개발 환경을 구축하면 가상의 안드로이드 장치인 **AVD(Android Virtual Device)** 제공
- 실제 안드로이드 장비 없이도 안드로이드 장비가 있는 것처럼 작업이 가능함

02 안드로이드 개발 환경의 구성

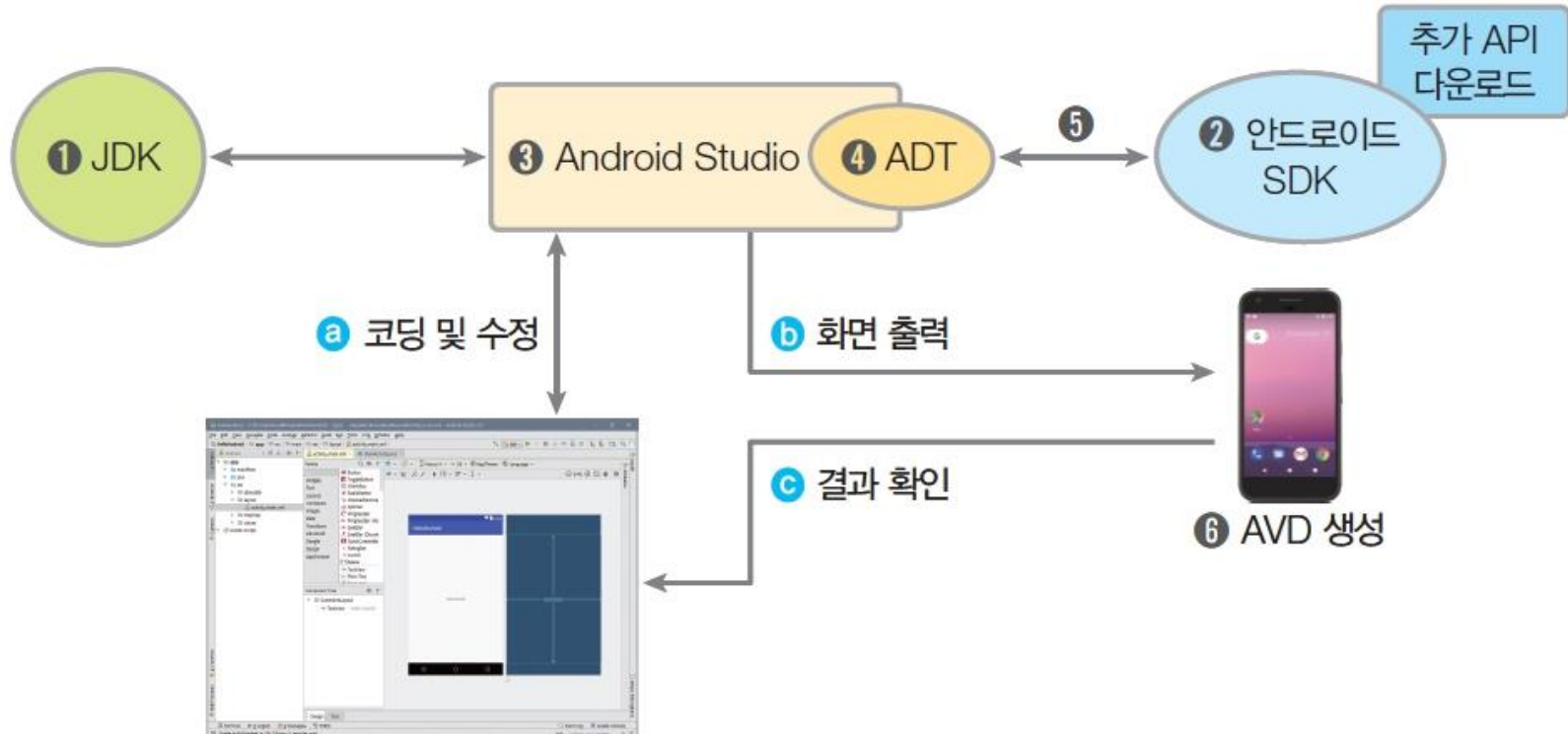


그림 1-4 안드로이드 개발 환경의 구성

- 개발 환경 구성이 완료되면 Android Studio를 실행해서 코딩과 수정(a)을 통해 앱을 개발할 수 있음
- 코딩된 결과는 ⑥에서 생성한 AVD에 출력(b)하고, 개발자는 그 결과를 확인(c)함

02 안드로이드 개발 환경의 구성

저자
한마디

안드로이드 개발 환경의 종류

안드로이드 개발 환경은 크게 세 가지로 나뉜다. 이 책은 입문자용이므로 응용 프로그램 개발 환경인 Kotlin과 Java SDK만을 다룬다. 이후 고급 안드로이드 개발자로 도약하려면 NDK와 PDK도 공부할 필요가 있다.

표 1-3 안드로이드 개발 환경

개발 환경	사용 언어	수준	개발 툴킷	비고
응용 프로그램 개발	Kotlin, Java	초·중급	SDK (Software Development Kit)	일반 응용 프로그램 개발
시스템 응용 프로그램 개발	C, C++	중·고급	NDK (Native Development Kit)	고수준의 응용 프로그램 개발
하드웨어 제어 및 커널 관련	C, C++	고급	PDK (Platform Development Kit)	하드웨어 수준의 강력한 프로그램 개발

01 안드로이드 개발 환경 설치 순서

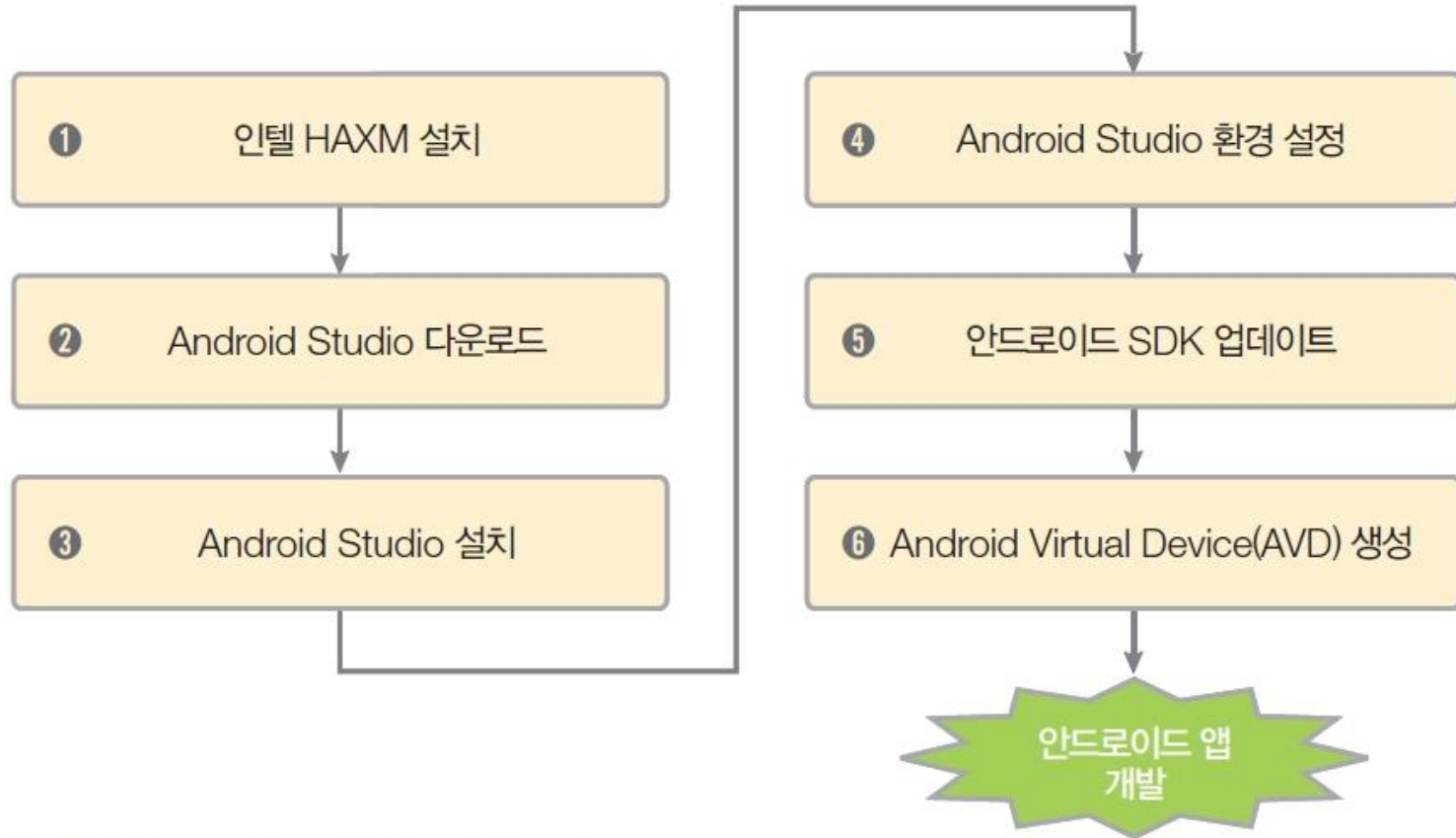


그림 1-5 안드로이드 개발 환경 설치 순서

02 인텔 HAXM 설치

- (1) [제어판]의 [시스템 및 보안]-[시스템] 실행
 - [시스템] 부분에서 CPU, RAM, 시스템 종류 확인



그림 1-6 컴퓨터 정보 확인

02 인텔 HAXM 설치

- (2) <https://github.com/intel/haxm/releases/>에 접속하여 최신 버전의 Windows용 Intel HAXM 압축 파일 다운로드

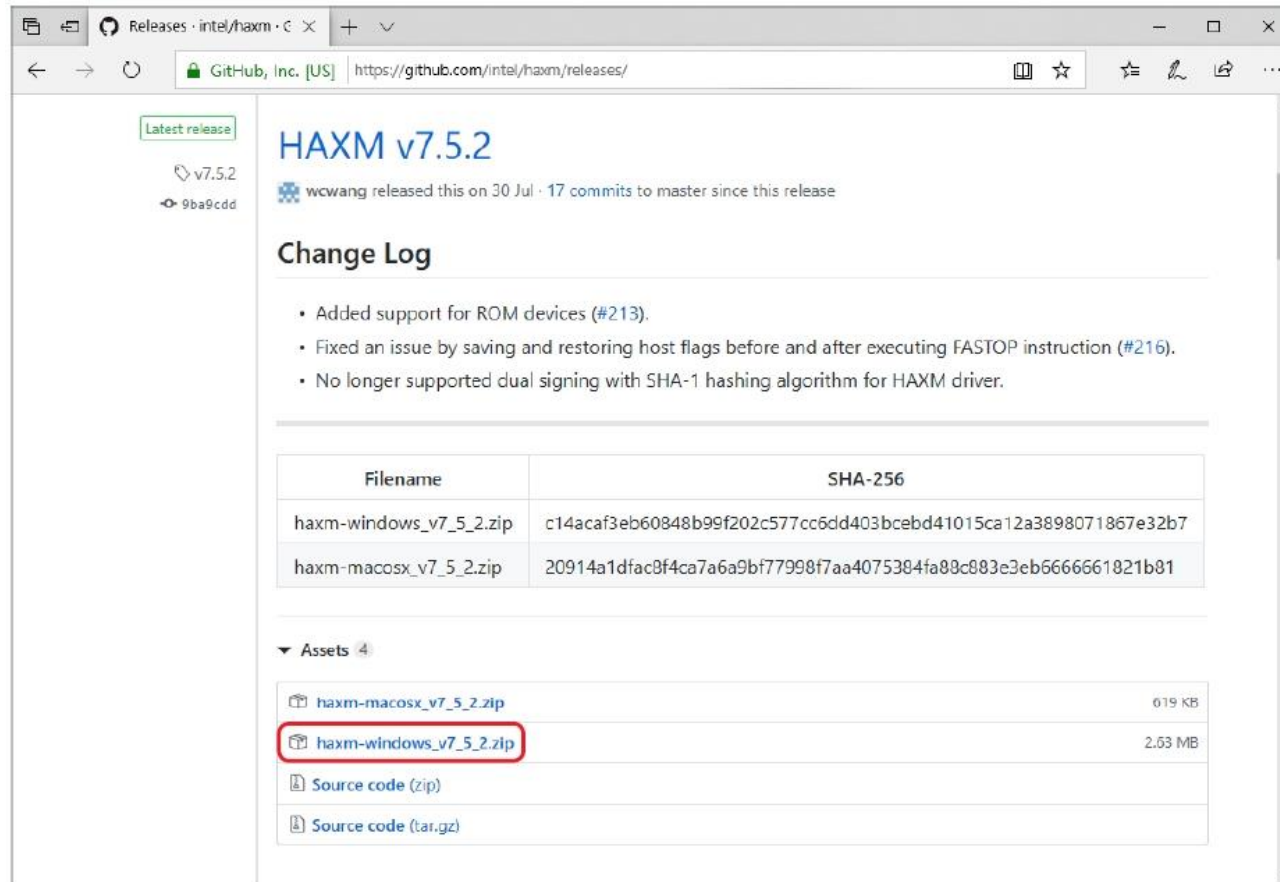


그림 1-7 인텔 HAXM 다운로드

02 인텔 HAXM 설치

- (3) 다운로드한 haxm-windows_버전.zip 파일의 압축 풀기
 - Intelhaxmandroid.exe 파일 실행 후 설치
 - 초기 화면에서 <Install> 클릭



그림 1-8 인텔 HAXM 설치 1

- (4) 'Launch Intel HAXM Documentation'의 체크를 해제하고 <Finish> 클릭



그림 1-9 인텔 HAXM 설치 2

03 Android Studio 설치

- (1) <https://developer.android.com/studio/archive.html> 접속
 - <I AGREE TO THE TERMS> 클릭
 - Android Studio 3.5.1 버전의 파일 다운로드

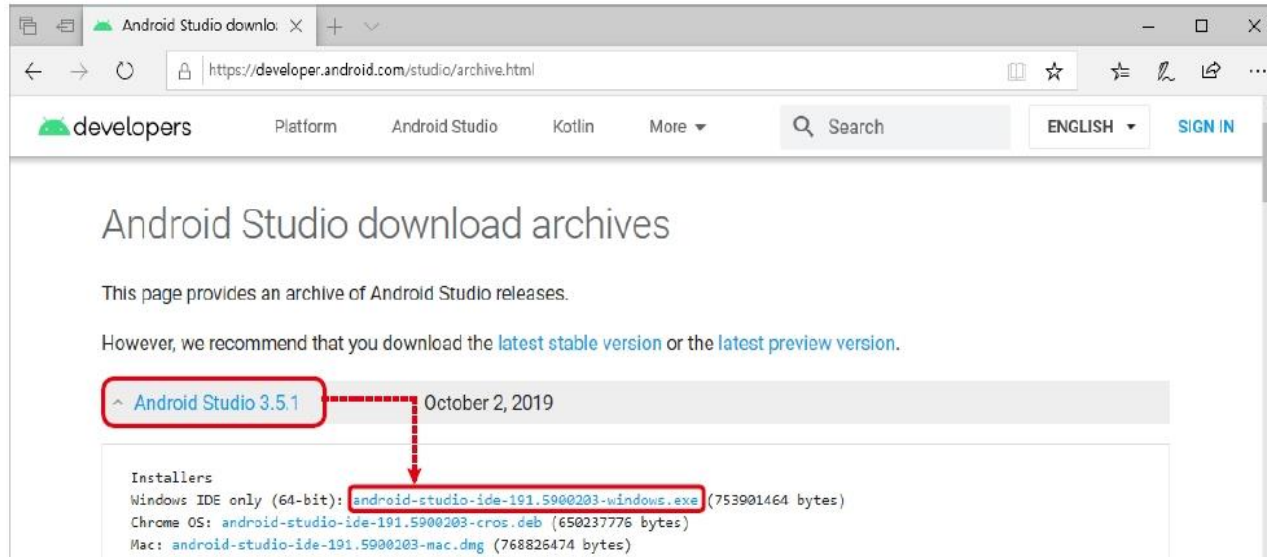


그림 1-10 Android Studio 3.5.1 다운로드

03 Android Studio 설치

- (2) 다운로드 받은 Android Studio 3.5.1 버전 확인
(android-studio-ide-191.5900203-windows.exe, 약 736MB)



그림 1-11 Android Studio 다운로드 완료

- (3) Android Studio 3.5.1 실행하고
초기화면에서 <Next> 클릭



그림 1-12 Android Studio 설치 초기화면

03 Android Studio 설치

- (4) 'Android Virtual Devices'의 체크 끄고 <Next> 클릭



그림 1-13 Android Studio 설치 항목 선택

- (5) Android Studio가 설치될 폴더는 기본값으로 두고 <Next> 클릭

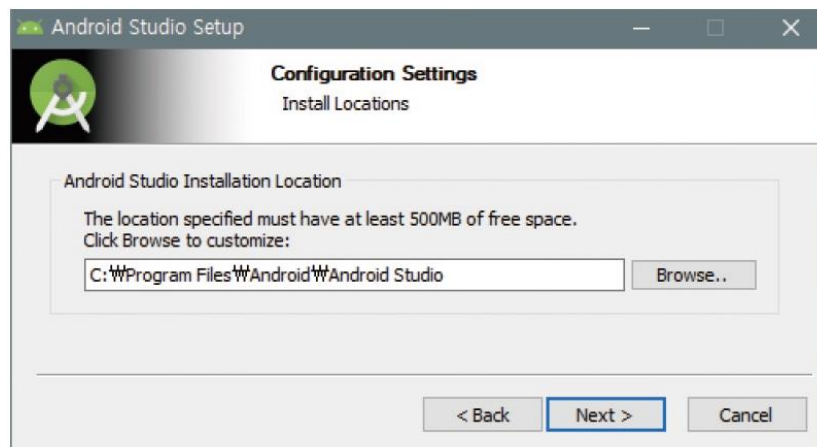


그림 1-14 설치 경로 설정

03 Android Studio 설치

- (6) 시작 메뉴 폴더 생성 화면 : 기본값으로 두고 <Install> 클릭

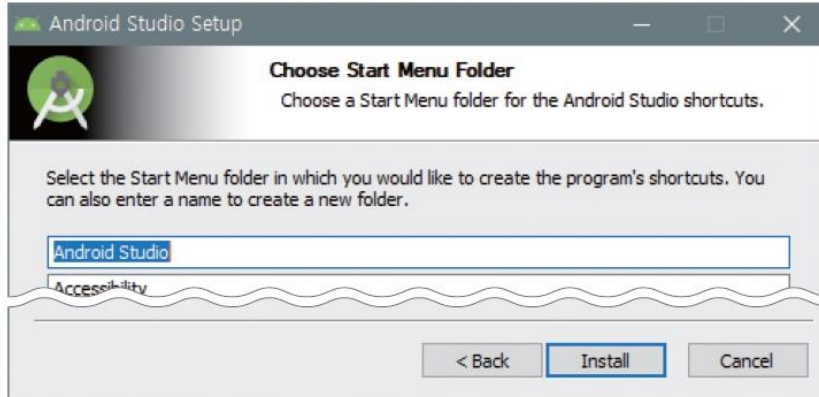


그림 1-15 시작 메뉴 폴더 생성

- (7) Android Studio 설치 진행 완료

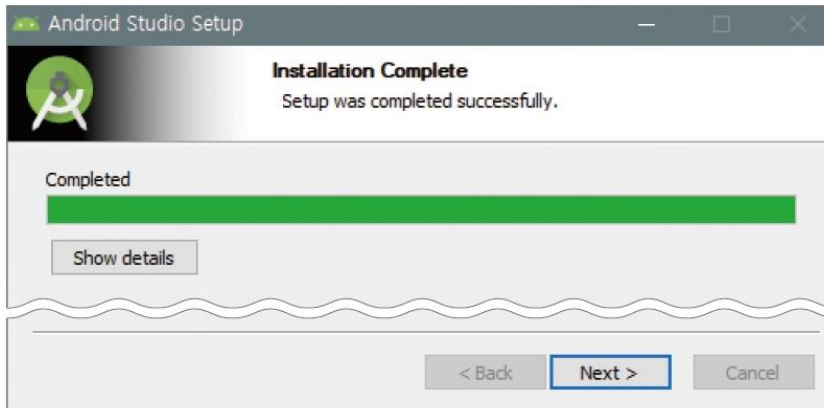


그림 1-17 설치 완료 1

03 Android Studio 설치

- (8) 설치 완료 화면 :
'Start Android Studio'의 체크 해제하고 <Finish> 클릭

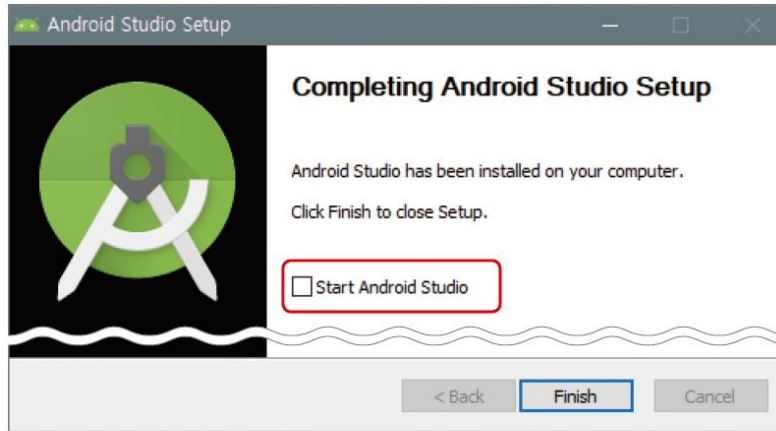


그림 1-18 설치 완료 2

04 Android Studio 설정

- (1) [Windows 시작]-[Android Studio]를 선택하여 Android Studio를 실행하여 'Do not import settings'가 선택된 상태에서 <OK> 클릭

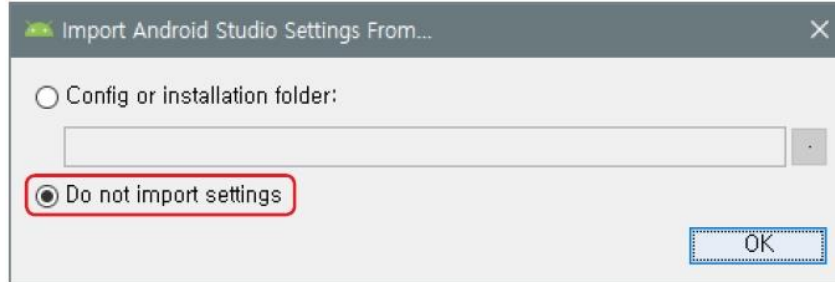


그림 1-19 Android Studio 이전 환경 가져오기 설정

- (2) 로고 화면이 나옴



그림 1-20 Android Studio 로고

04 Android Studio 설정

- (3) [Android Studio Setup Wizard]가 실행됨



그림 1-21 Android Studio Setup Wizard 실행 1

- (4) [Install Type]은 기본값인 'Standard'를 선택

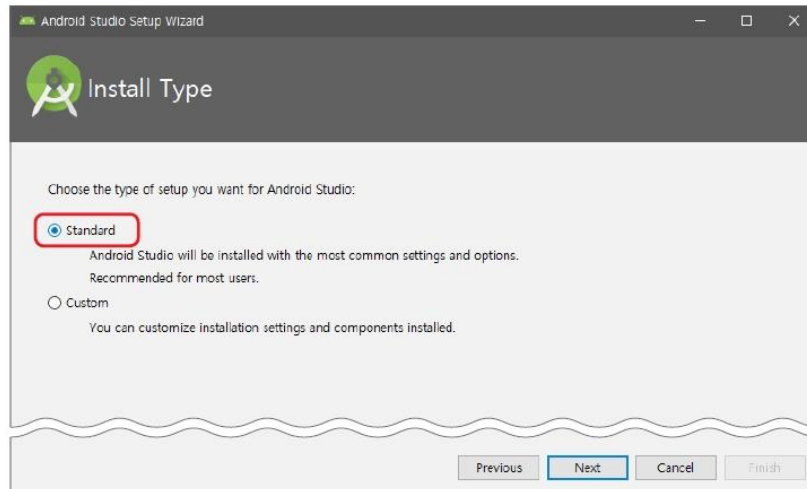


그림 1-22 Android Studio Setup Wizard 실행 2

04 Android Studio 설정

- (5) [Select UI Theme]는 자유롭게 선택
 - 책에서는 'Light'로 실습 진행

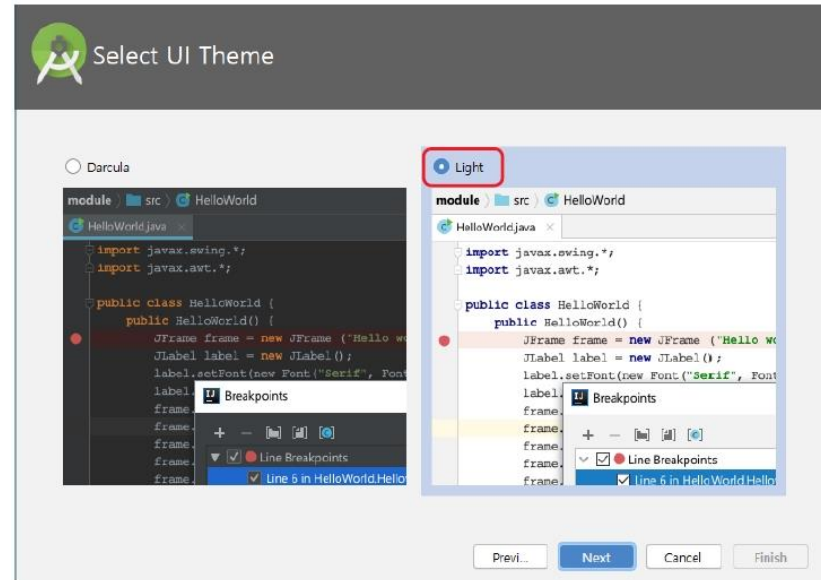


그림 1-23 Android Studio Setup Wizard 실행 3

- (6) [Verify Settings]에서 다운로드 크기를 확인하고 <Finish> 클릭

04 Android Studio 설정

- (7) 안드로이드 SDK의 다운로드 및 설치 진행

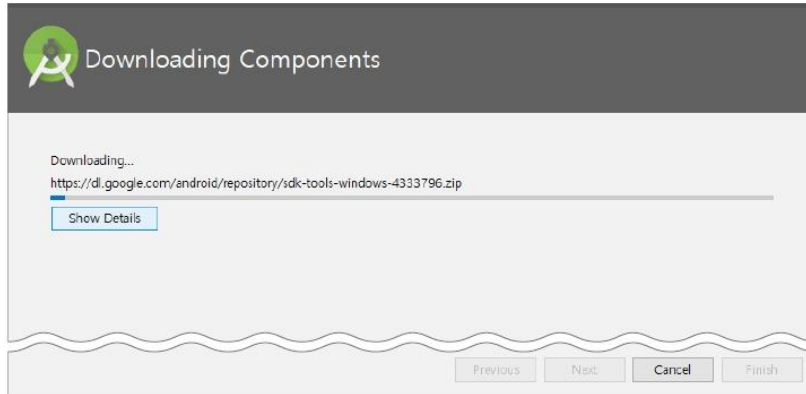


그림 1-25 Android Studio Setup Wizard 실행 5

- (8) 설치가 완료되면 <Finish>를 클릭하여 종료

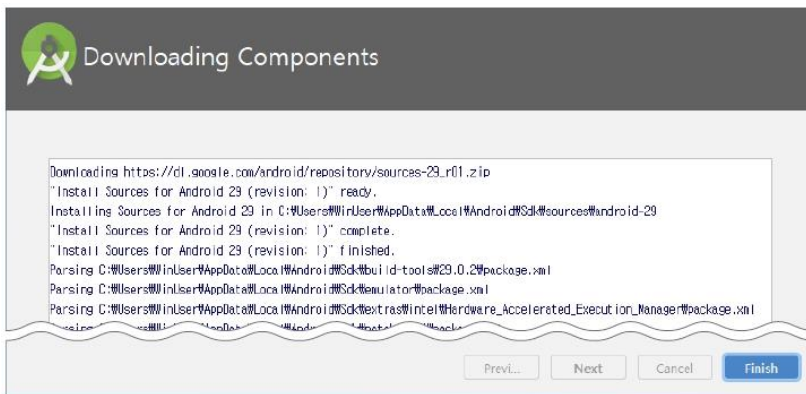


그림 1-26 Android Studio Setup Wizard 실행 6

04 Android Studio 설정

- (9) Android Studio 초기화면에서 'Configure' - 'Settings' 선택

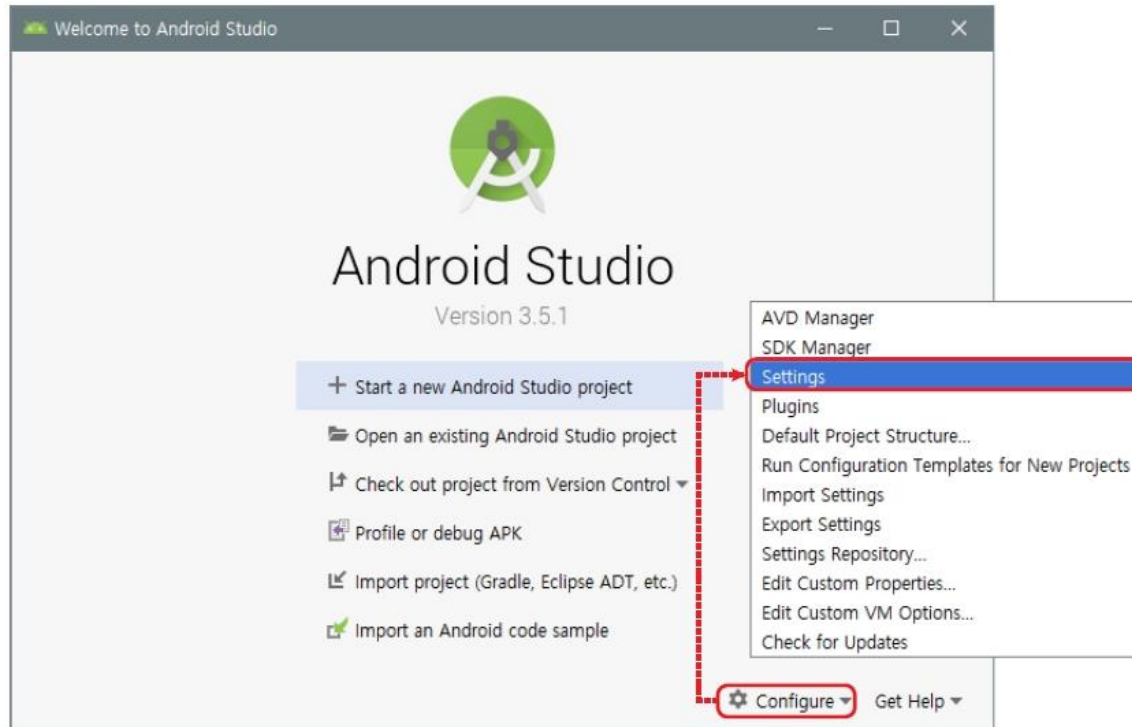


그림 1-27 Android Studio 초기화면

04 Android Studio 설정

- (10) 자동 업데이트 기능 끄기
 - [Appearance & Behavior]-[System Settings][Updates]를 선택하고 오른쪽의 'Automatically check updates for'의 체크 해제

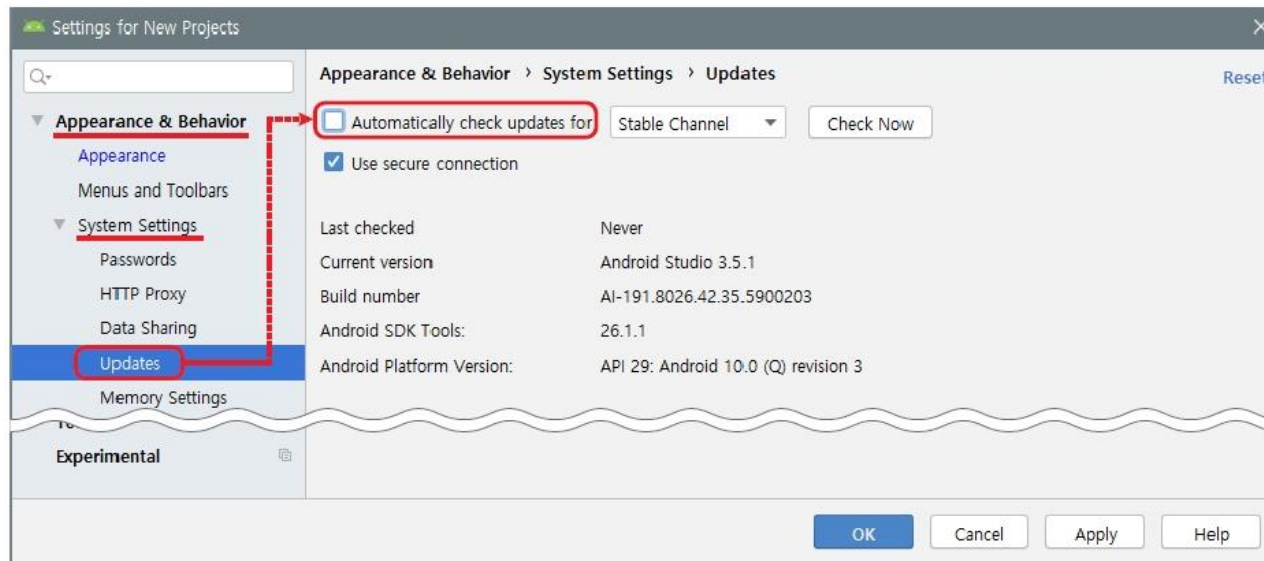


그림 1-28 업데이트 기능 끄기

04 Android Studio 설정

- (11) 행 번호 설정
 - [Editor]-[General][Appearance]를 선택하고 오른쪽의 'Show line numbers' 체크

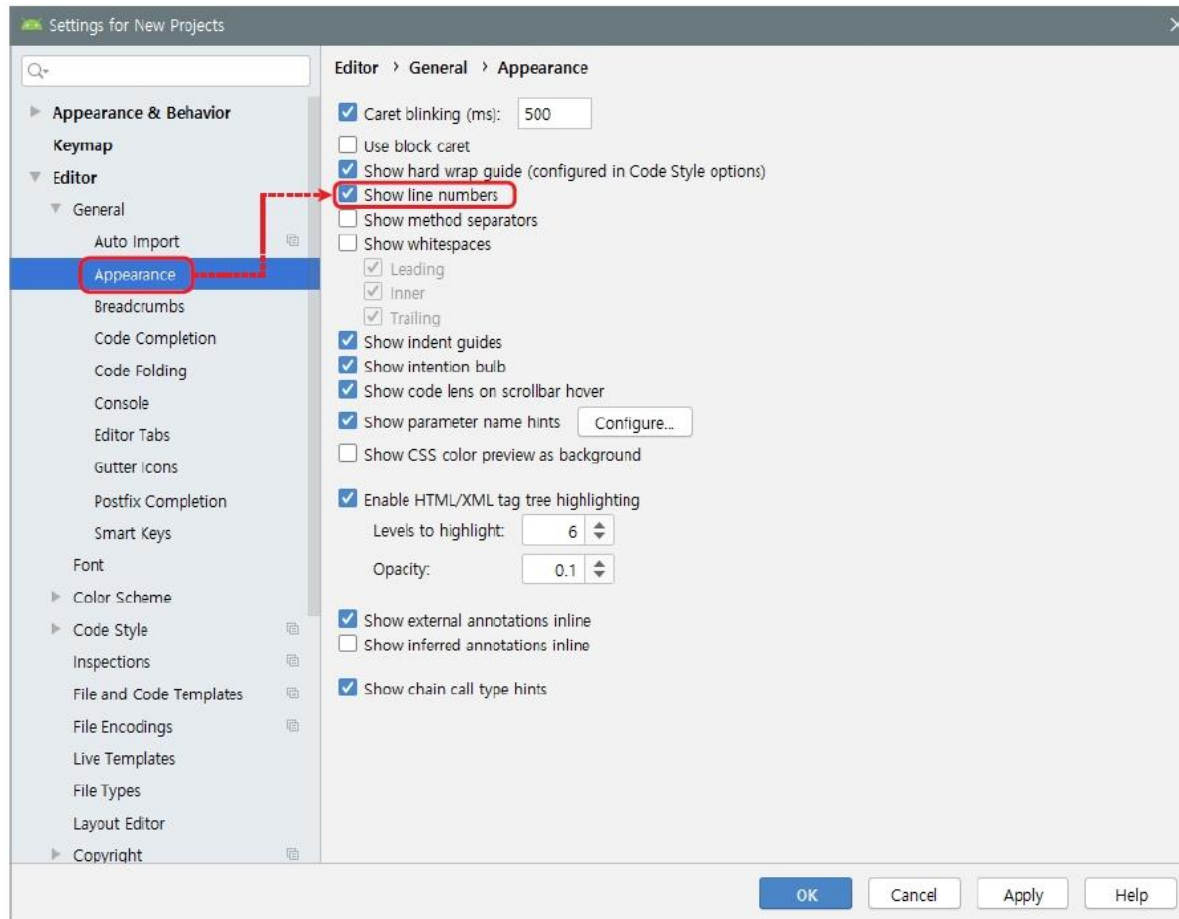


그림 1-29 행 번호 설정

04 Android Studio 설정-SDK 업데이트하기

- (1) 안드로이드 버전별 SDK 추가/삭제 창 확인
 - Android Studio 초기화면에서 'Configure' - 'SDK Manager' 선택

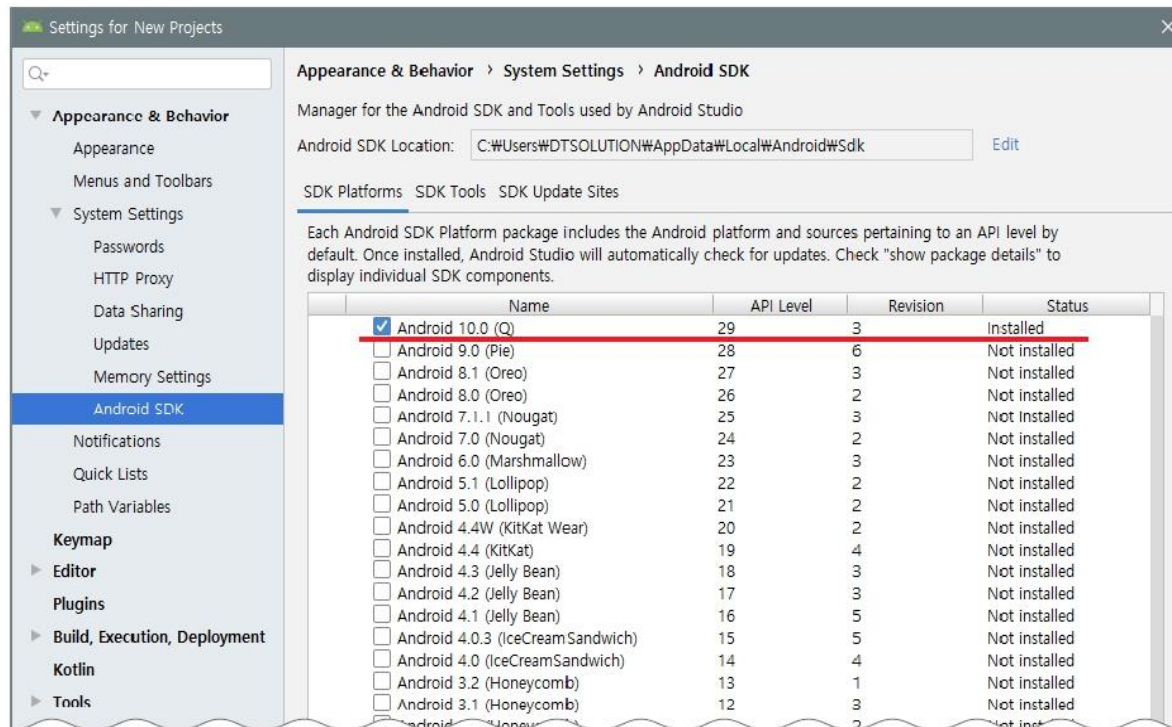


그림 1-30 안드로이드 SDK 추가 설치 1

04 Android Studio 설정-SDK 업데이트하기

- (2) 'Show Package Details'에 체크 후 기존에 체크되어 있는 항목을 모두 해제
 - Android 10.0(Q) 아래의 'Android SDK Platform 29'와 'Google APIs Intel x86 Atom System Image'에만 체크.

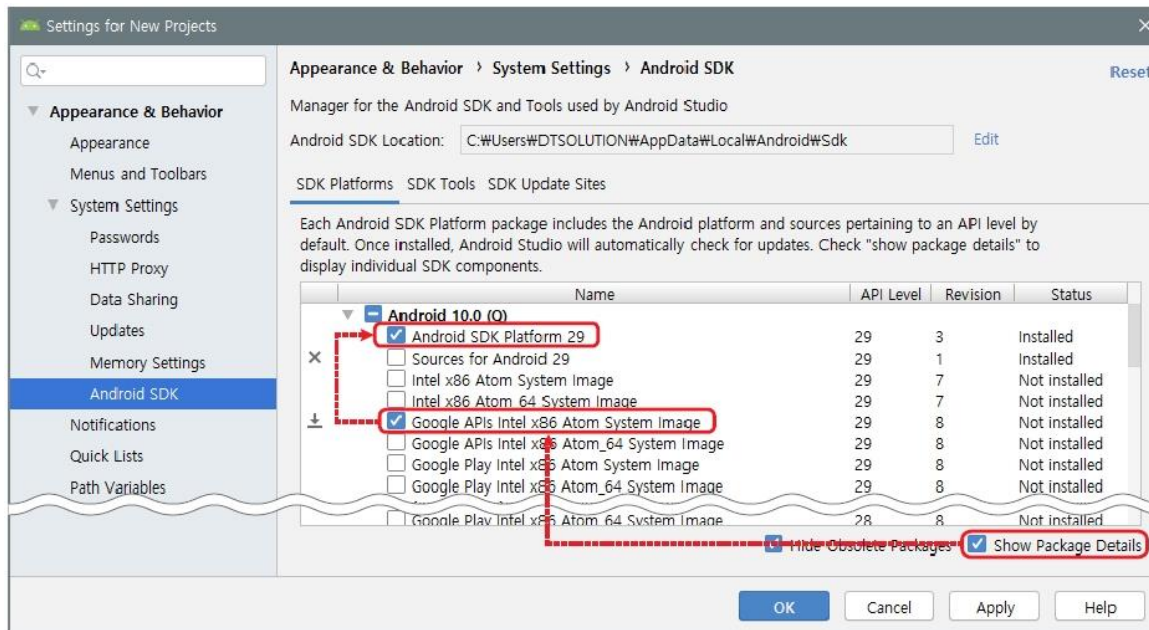


그림 1-31 안드로이드 SDK 추가 설치 2

04 Android Studio 설정-SDK 업데이트하기

- (3) [SDK Tools] 탭 클릭하여 추가로 'Google Play services'에 체크
 - [Confirm Change] 창이 나오면 <OK> 클릭

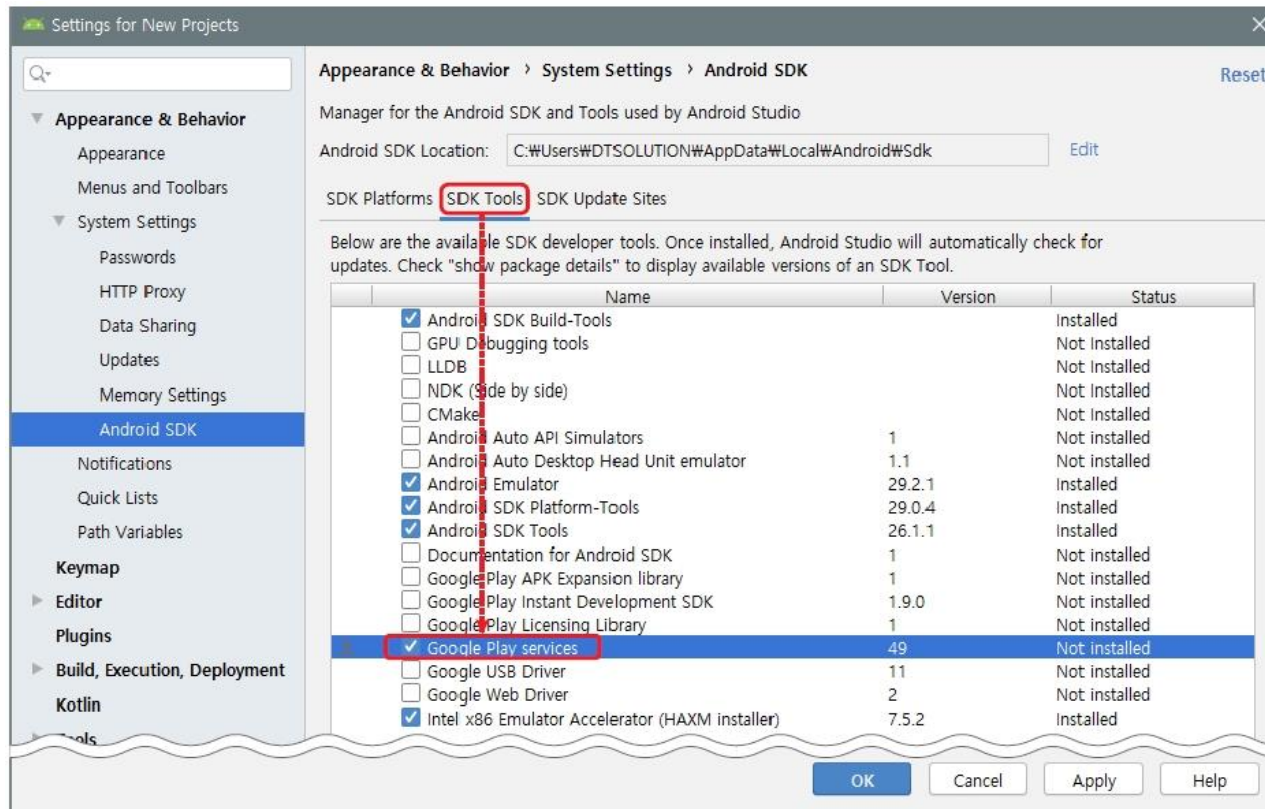


그림 1-32 안드로이드 SDK 추가 설치 3

04 Android Studio 설정-SDK 업데이트하기

- (4) [License Agreement] 창에서 각 상위 항목을 클릭하여 'Accept' 선택

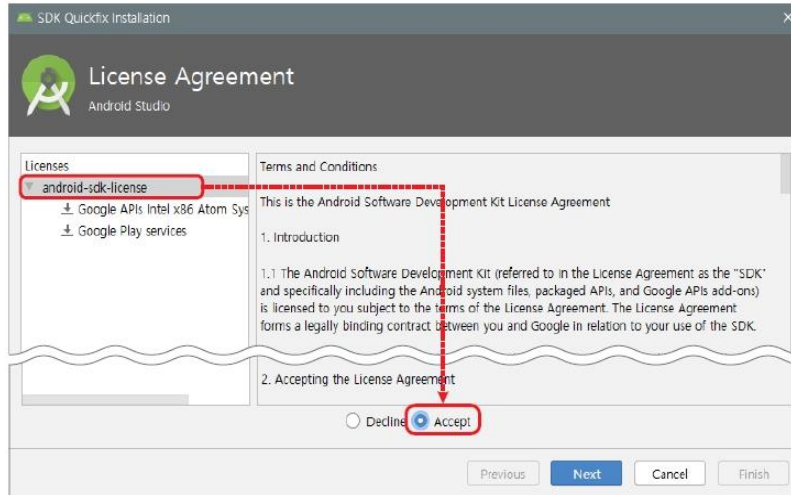


그림 1-33 라이선스 동의

- (5) 설치가 완료되면 왼쪽 아래에 'Done'이 표시됨

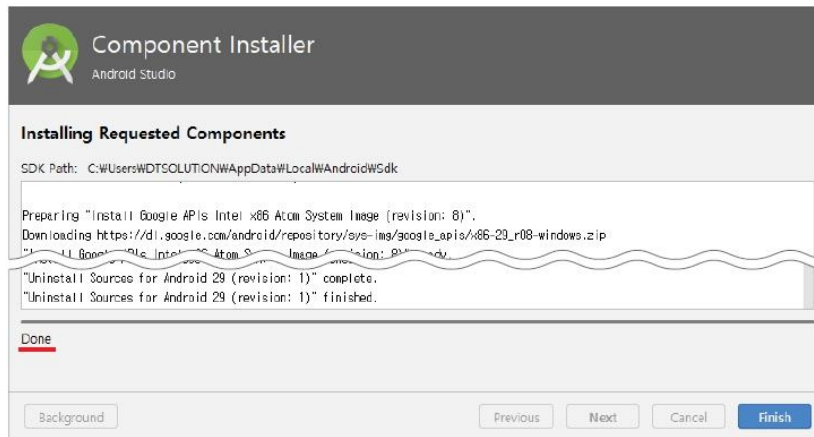


그림 1-35 설치 완료

05 AVD 생성

- (1) Android Studio 초기화면에서 [Start a new Android Studio project] 클릭

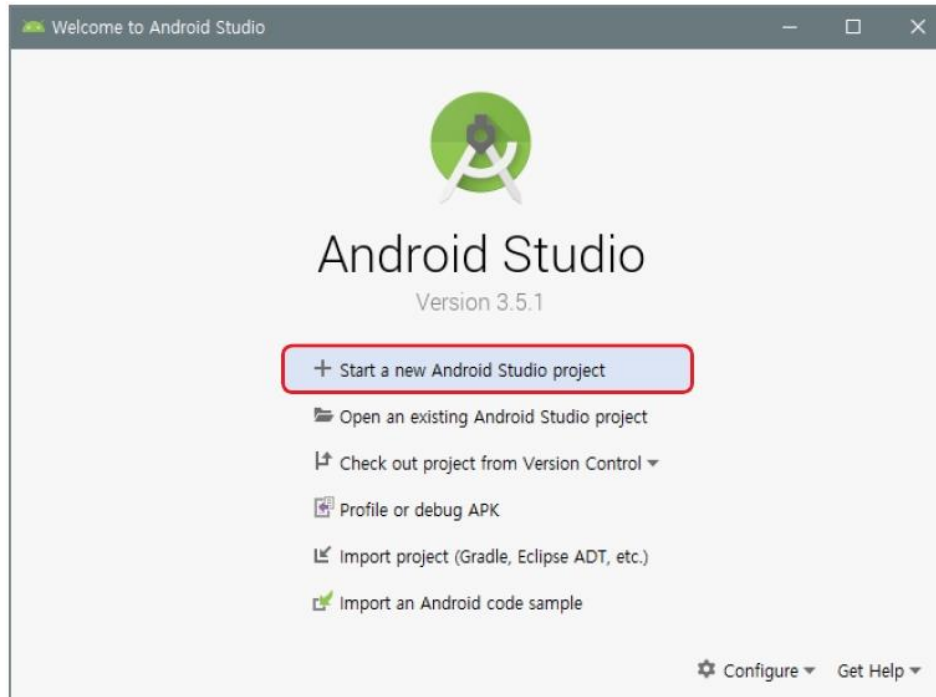


그림 1-36 AVD 생성을 위한 임시 프로젝트 생성

05 AVD 생성

- (2) [Choose a Project]에서 'Empty Activity'를
- (3) 아래 창에서 다운로드와 빌드 등의 작업이 진행

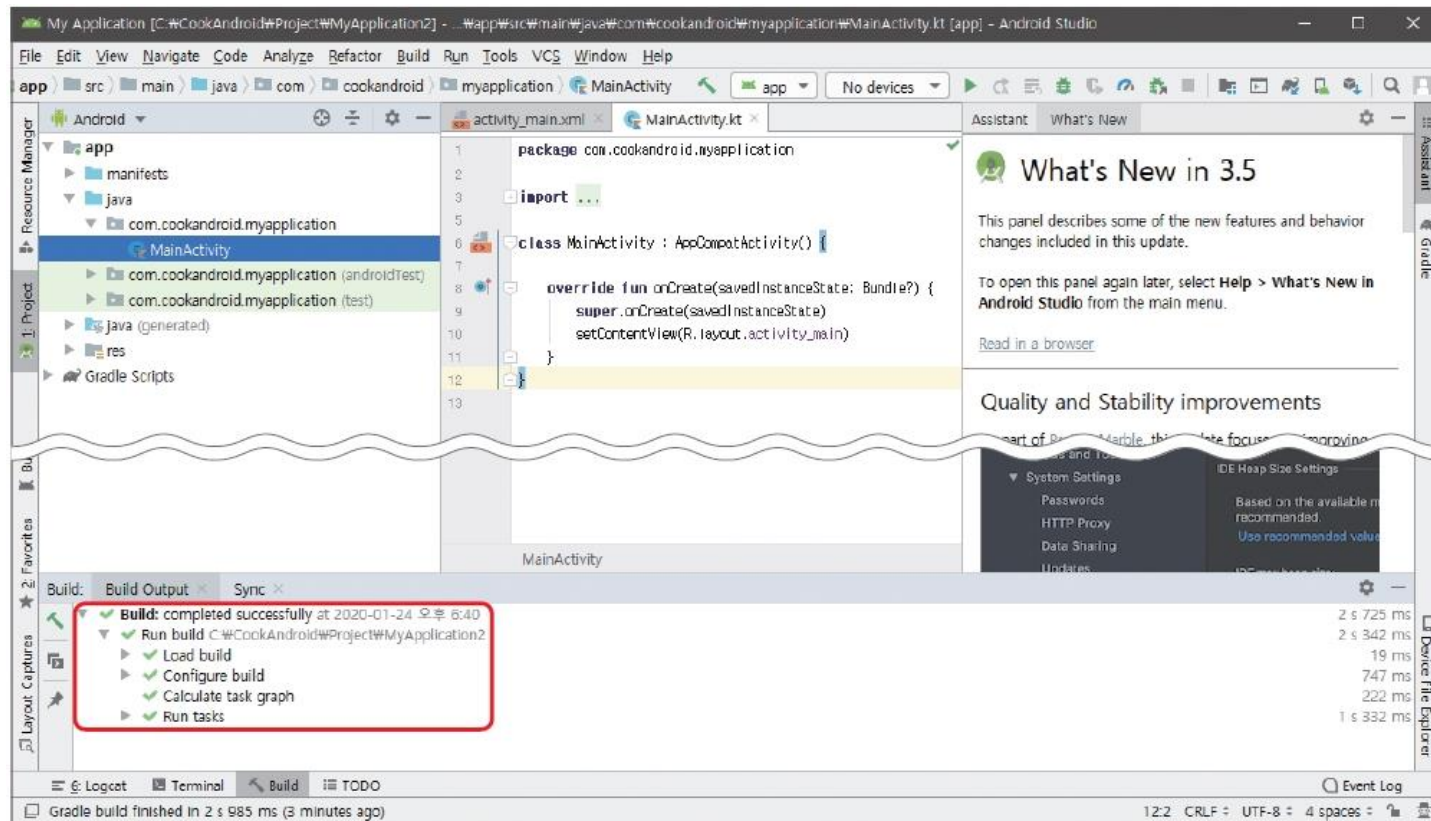
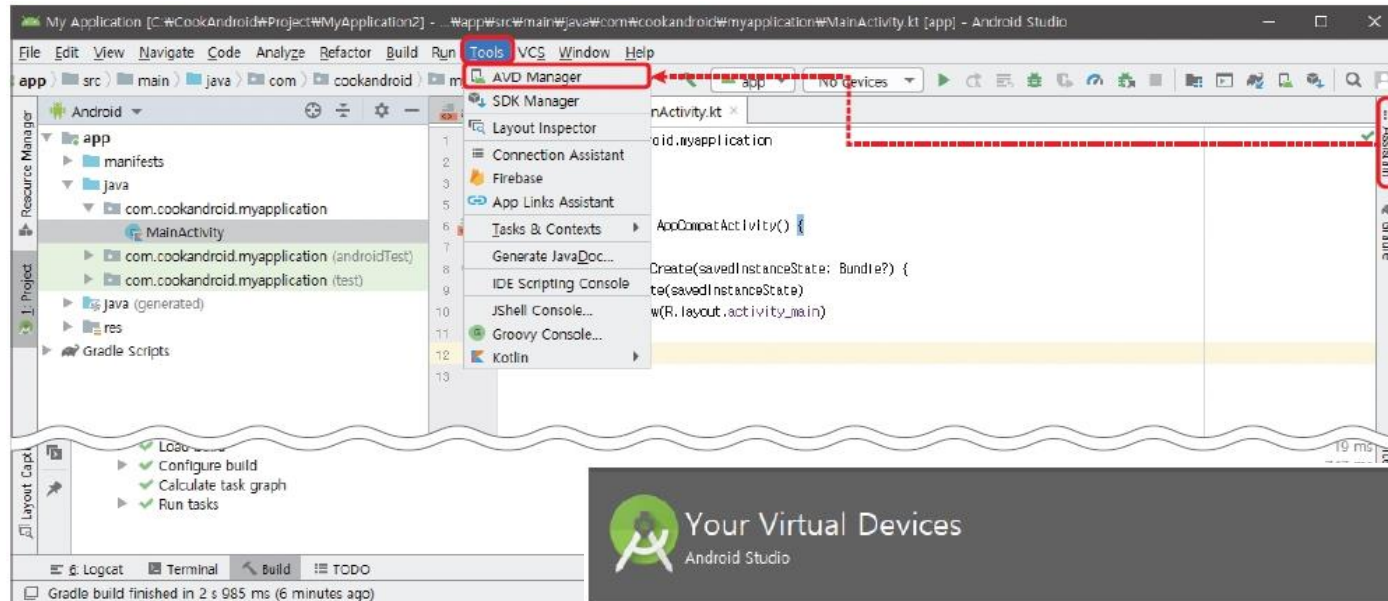


그림 1-37 프로젝트 생성 완료

05 AVD 생성

- (4) [Tools]-[AVD Manager] 선택



- (6) [Android Virtual Device Manager]가 실행됨



그림 1-39 AVD 생성 1

05 AVD 생성

- (7) 'Pixel 2'를 선택하고 <Next> 클릭

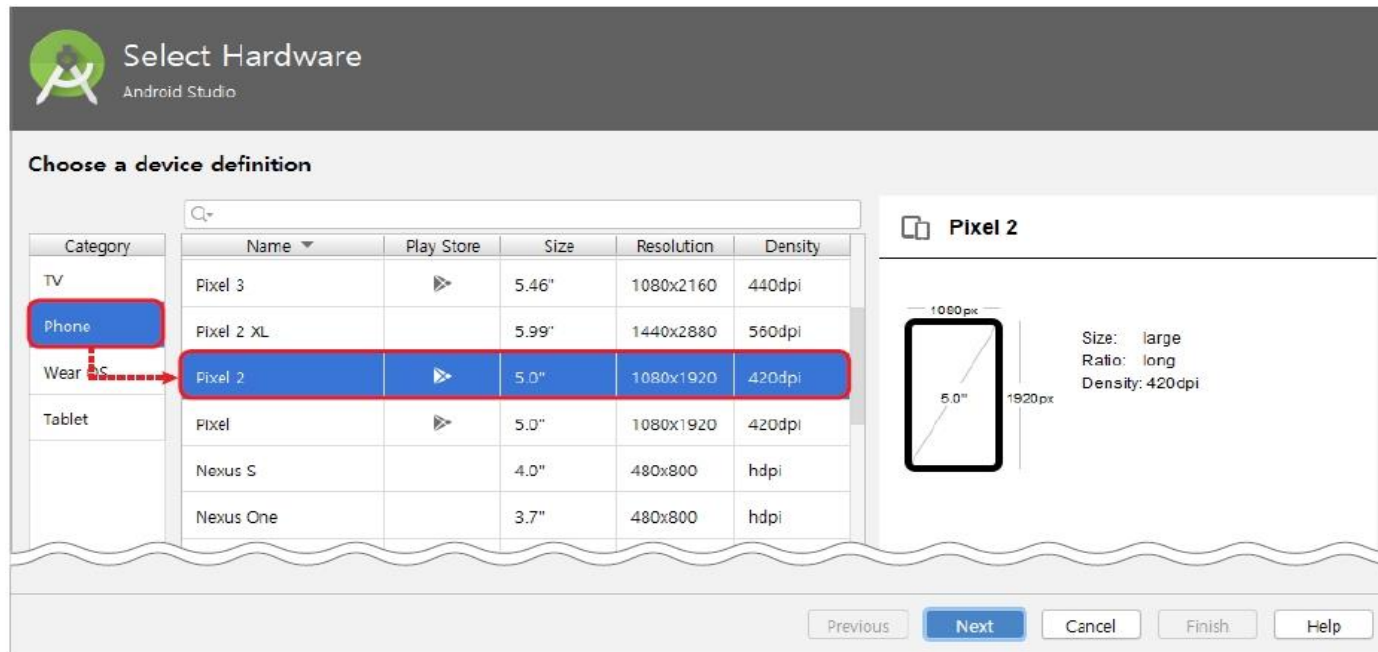


그림 1-40 AVD 생성 2

05 AVD 생성

- (8) [x86 Images] 탭 클릭 - 'Q[API 29, x86, Android 10.0(Google APIs)]' 선택하고 <Next> 클릭

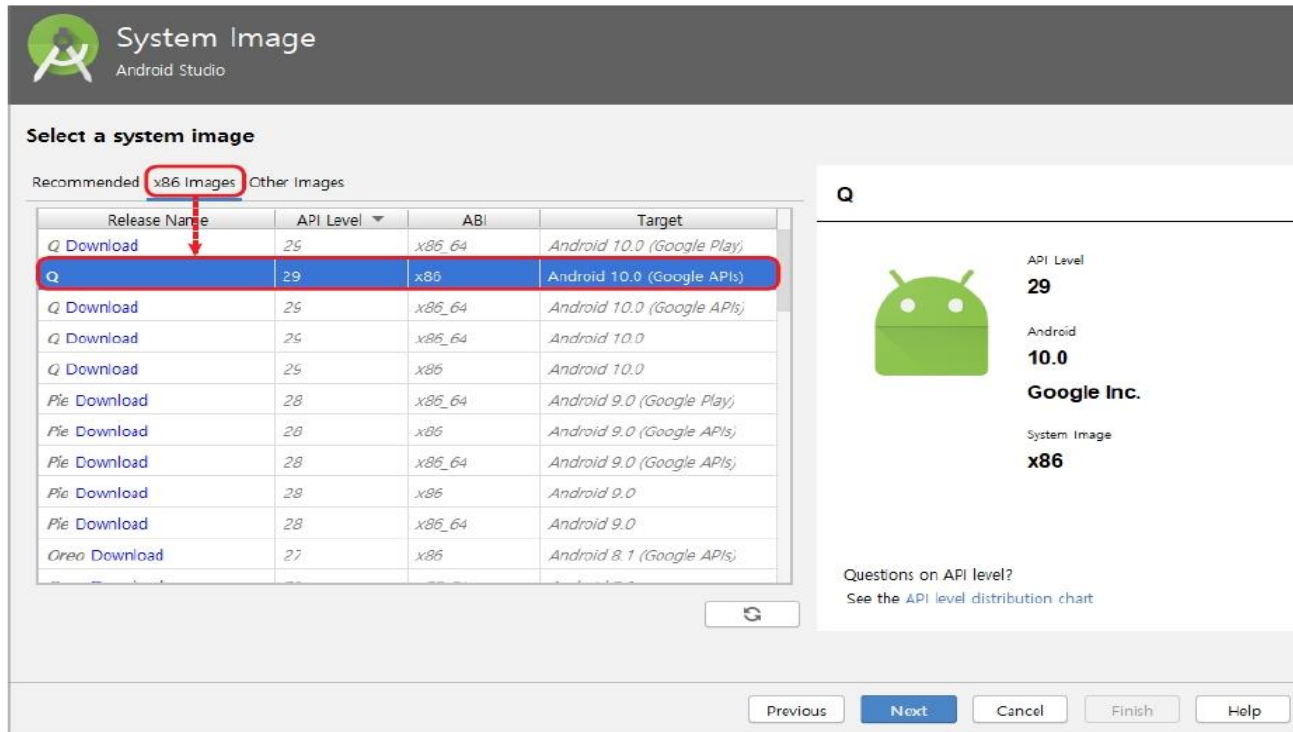


그림 1-41 AVD 생성 3

05 AVD 생성

- (9) AVD Name은 'Q(Pixel 2)'로 하고 <Finish> 클릭

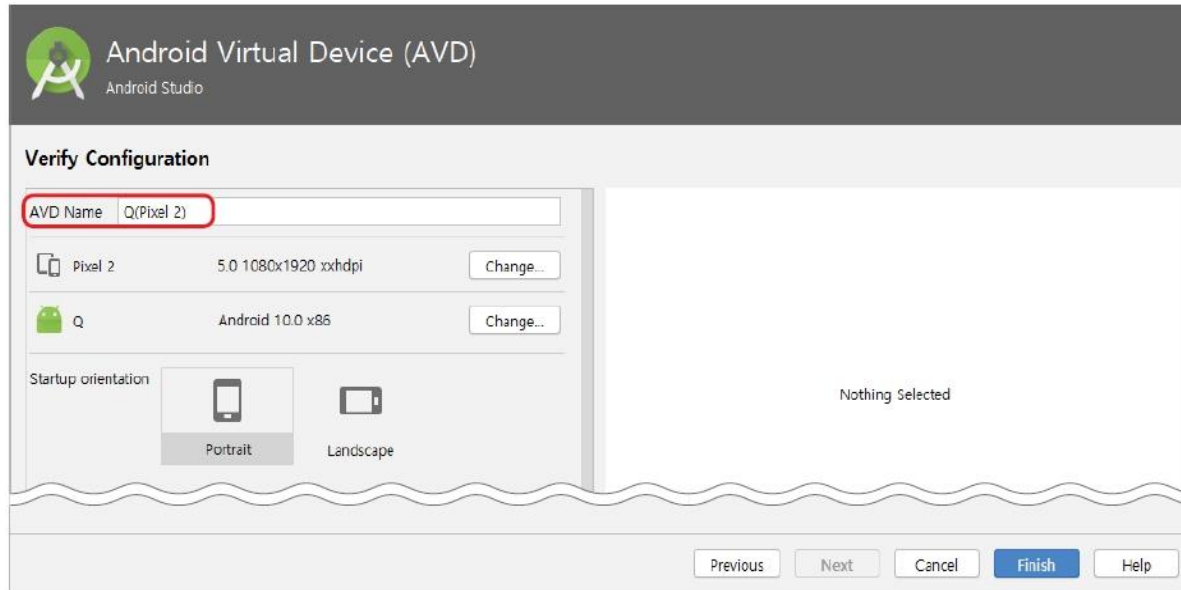


그림 1-42 AVD 생성 4

05 AVD 생성

- (10) <Launch...> 아이콘을 클릭하여 AVD 부팅

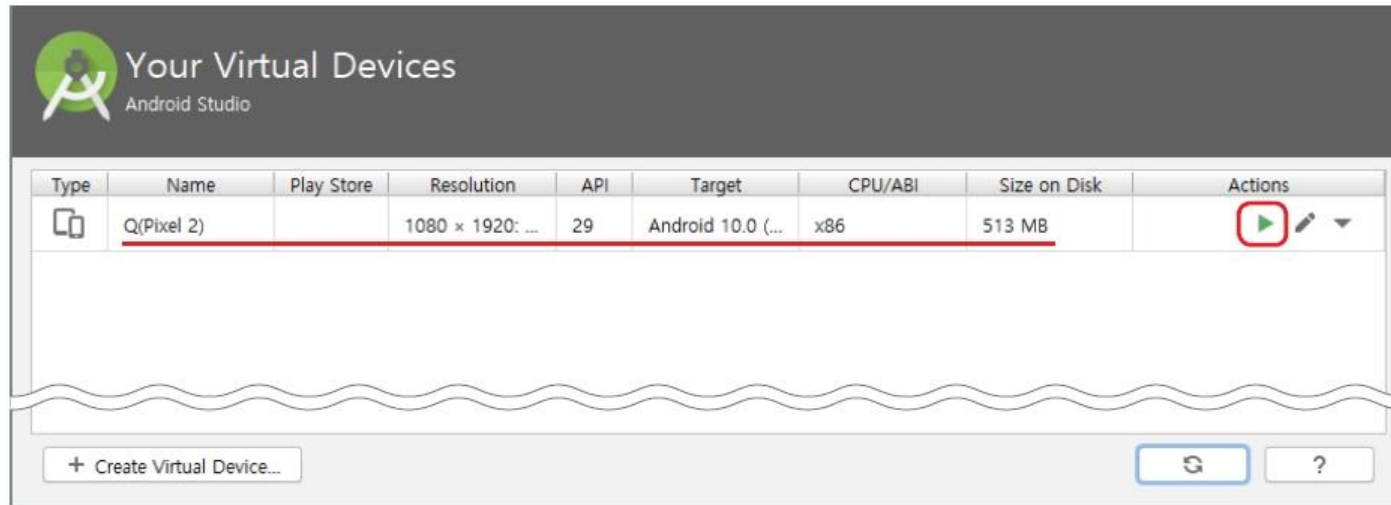


그림 1-43 생성된 AVD

05 AVD 생성

- (11) 부팅된 AVD 확인



그림 1-44 AVD 가동 화면

05 AVD 생성

- (12) 화면이 한글로 표시되도록 설정
 - 화면을 마우스로 클릭한 후 위쪽으로 스와이프하고 Settings 아이콘 클릭

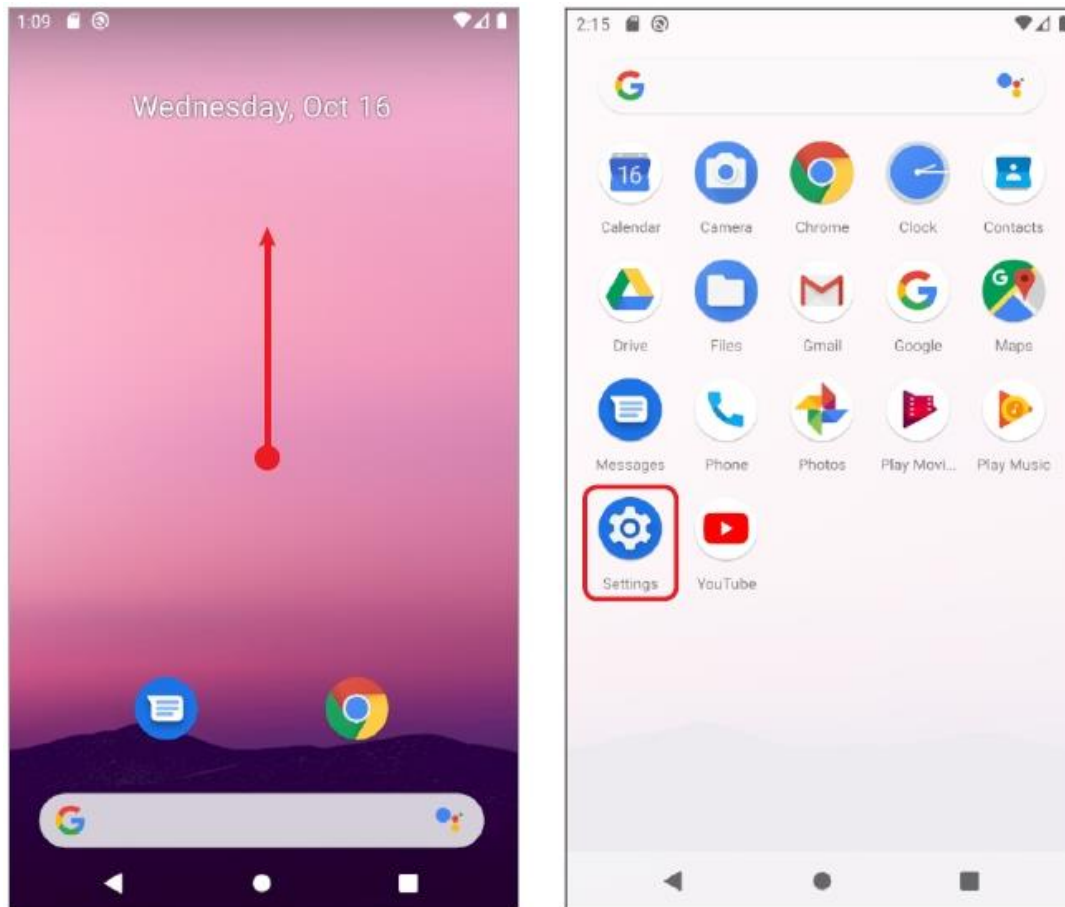


그림 1-45 화면 한글 설정 1

05 AVD 생성

- (13) [System] 클릭 - [Languages & input] 클릭

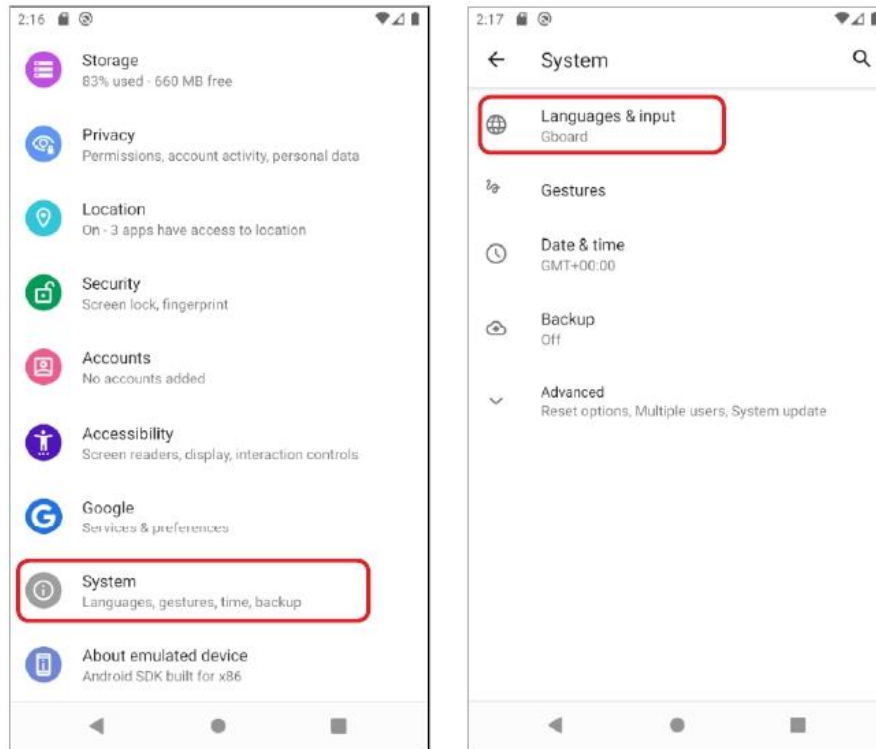


그림 1-46 화면 한글 설정 2

05 AVD 생성

- (14) [Languages]-[Add a language] - '한국어' 클릭

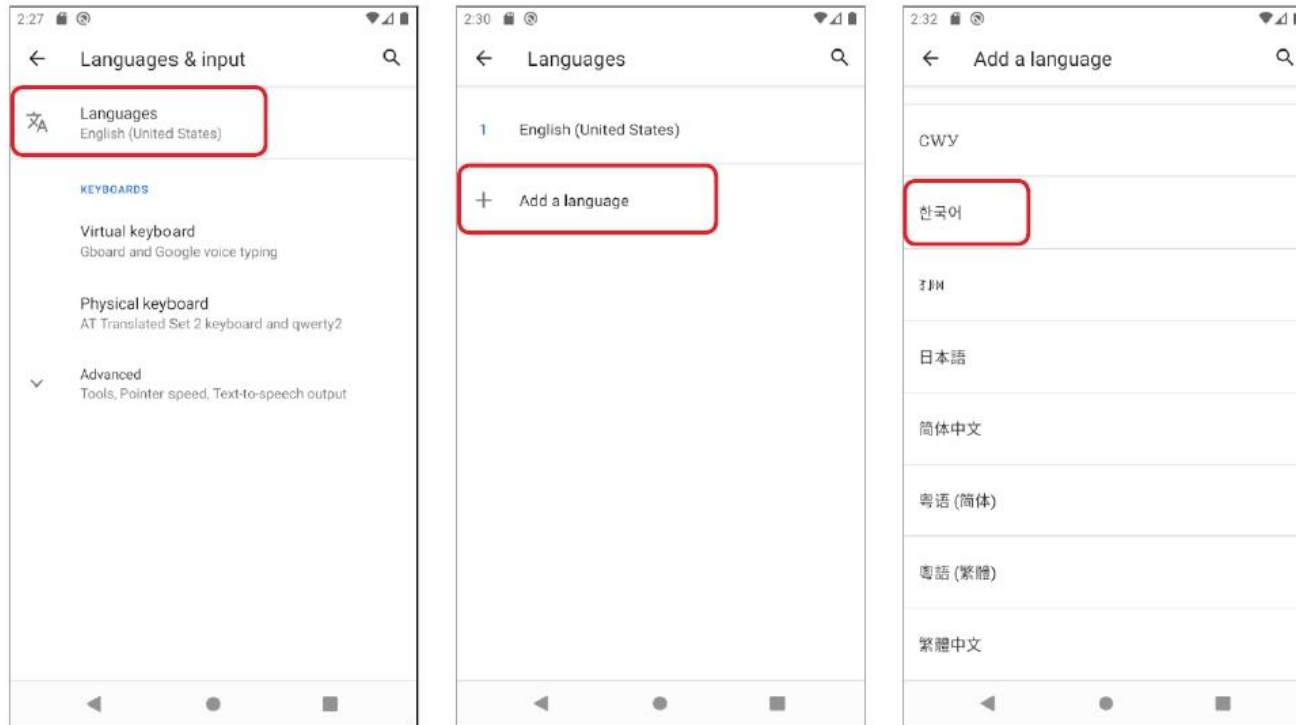


그림 1-47 화면 한글 설정 3

05 AVD 생성

- (15) '대한민국'을 클릭하면 두 번째 위치에 한국어가 추가된 것을 확인
 - '한국어(대한민국)'을 꼭 누른 채 위로 올려서 첫 번째로 만들기

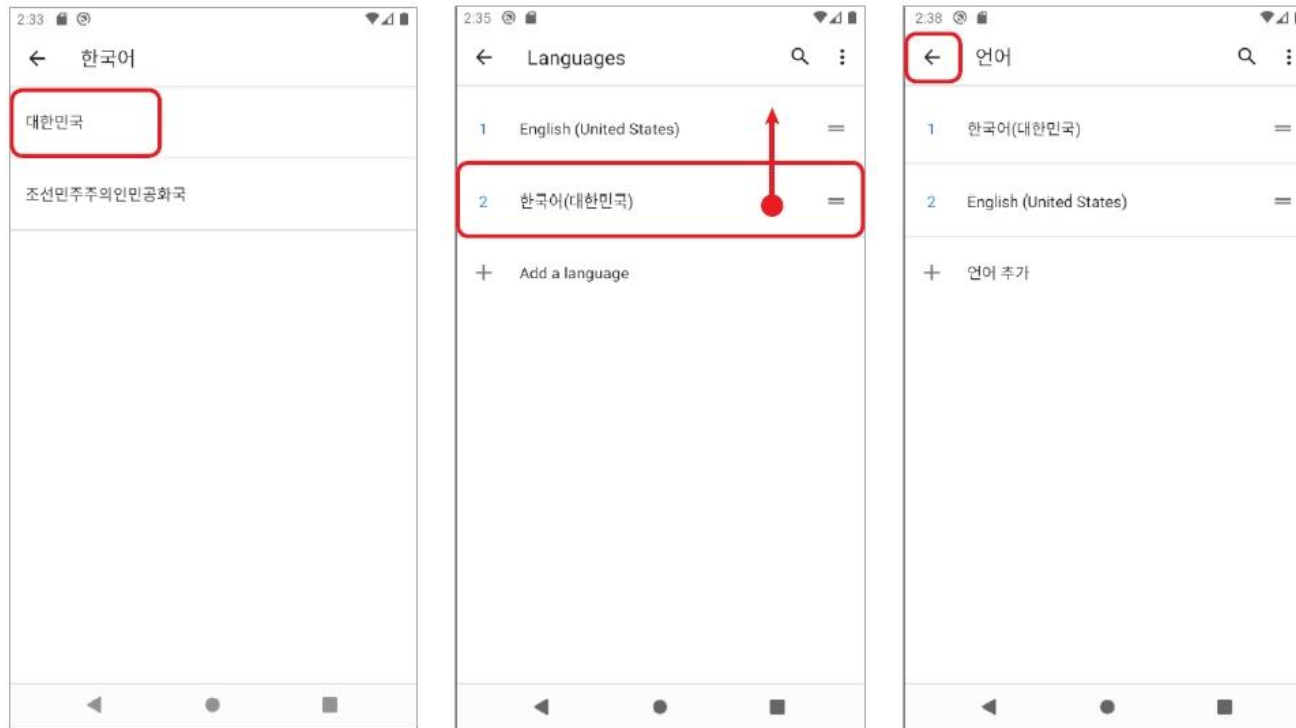


그림 1-48 화면 한글 설정 4

05 AVD 생성

- (16) 맞춤법 검사기 기능 해제
 - [고급]-[맞춤법 검사기]를 클릭 - '맞춤법 검사기 사용' 끄기

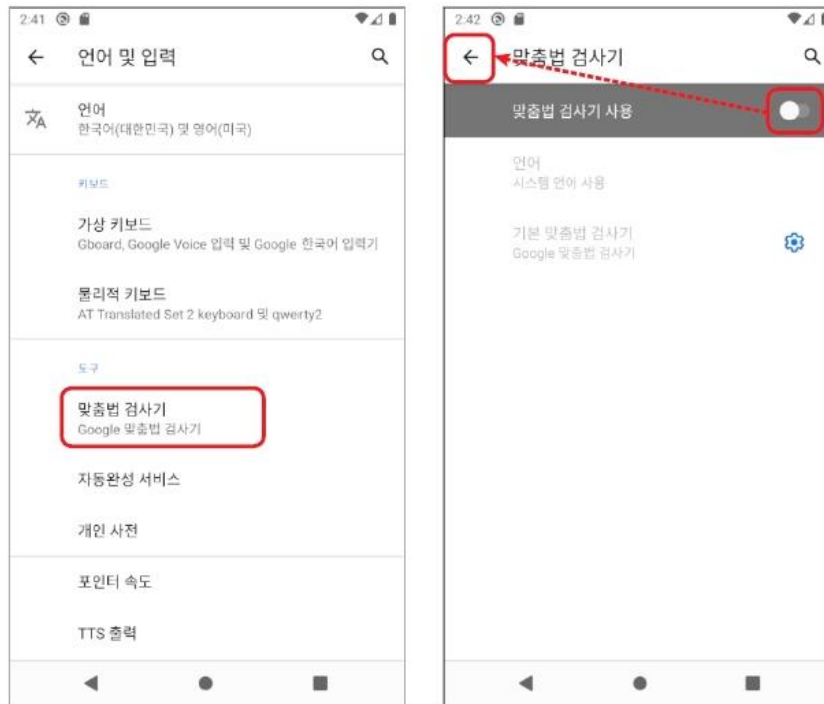


그림 1-49 맞춤법 기능 끄기

05 AVD 생성

- (17) 화면 자동 잠금 시간 늘리기
 - [설정] 화면에서 [디스플레이] 선택
 - [화면 자동 잠금 시간]을 클릭한 후 '30분' 선택 : 30분마다 화면이 잠김

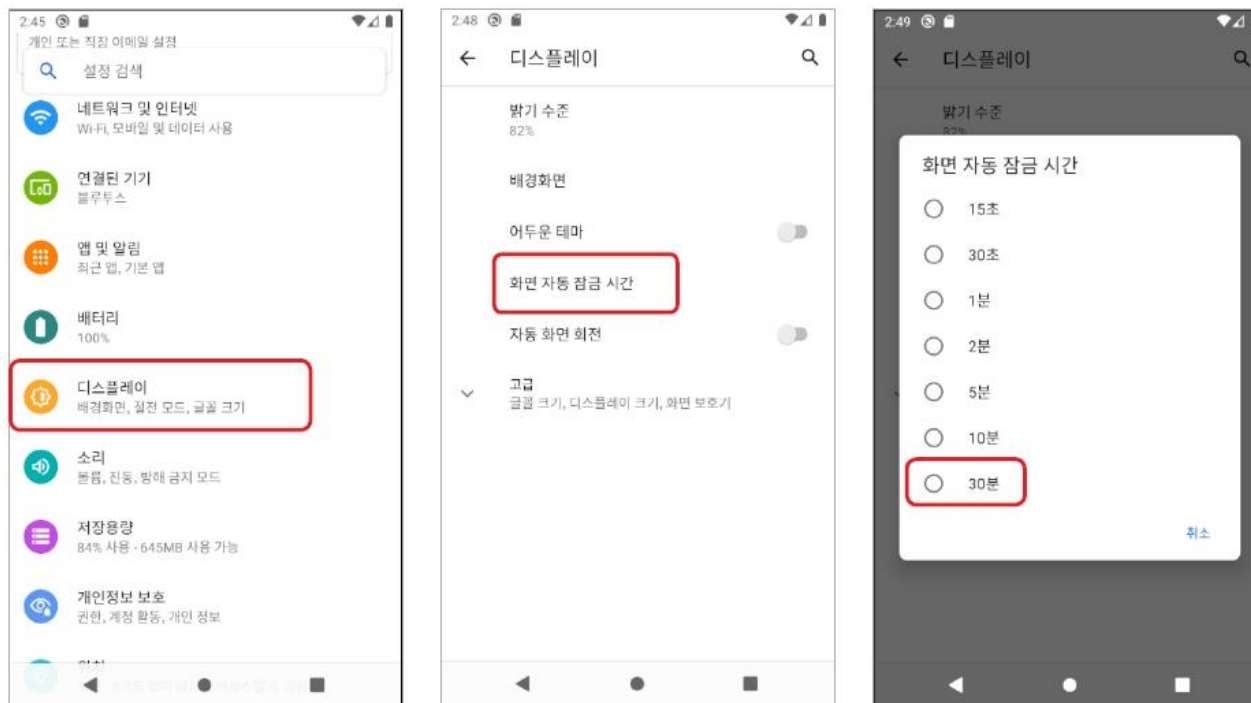


그림 1-50 화면 잠금 시간 늘리기

05 AVD 생성

- (18) 아래쪽 홈 아이콘(●)을 연속 클릭하면 초기화면으로 돌아감
- (19) 위쪽 상태 바를 잡아서 아래로 끌어 내리면 현재 알림 확인 가능
- (20) 아래쪽 돌아가기 아이콘(◀)을 클릭하면 닫힘

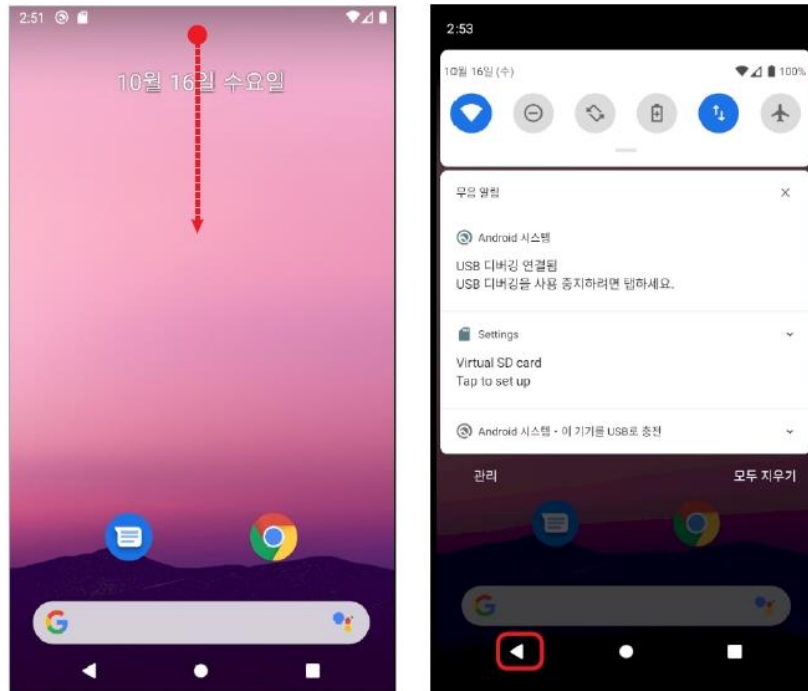
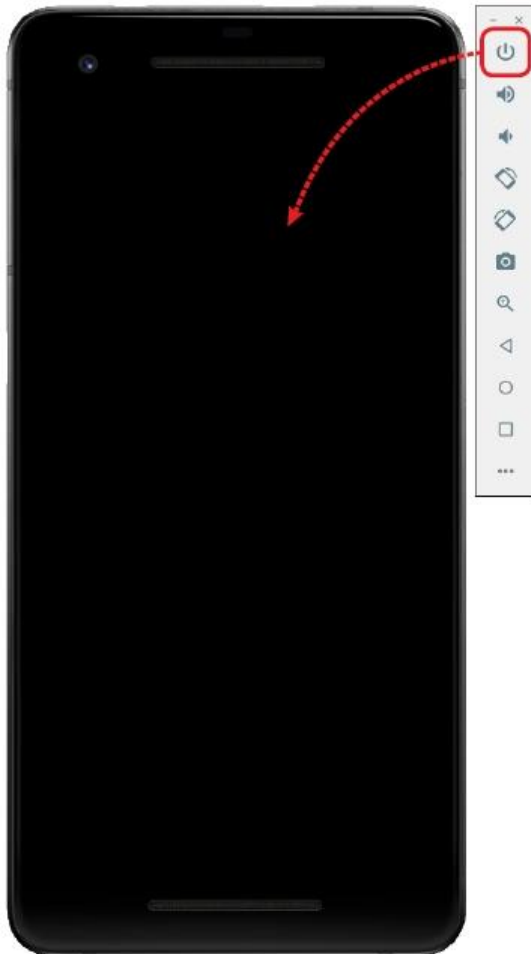


그림 1-51 알림 확인

05 AVD 생성

- (21) 오른쪽 위의 파워 버튼을 누르면 스마트폰이 종료됨
- (22) [File]-[Close Project]를 선택하여 프로젝트 종료



TIP X를 눌러도 바로 AVD가 종료되지 않을 수 있다. 잠시 기다린다.

그림 1-52 AVD 종료

Thank You !