

Ch01-안드로이드의 개요와 개발 환경 설치



2024-01-01(분반)

AI 오픈소스 전문프로젝트

Fri 09:00~ (S4-1-204)

학습목표

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

목차

01 안드로이드의 개요

02 안드로이드의 개발 환경

03 안드로이드의 개발 환경 설치

01

안드로이드의 개요

1. 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(픽셀8, 아이폰15, 윈도우폰10)

1. 스마트폰의 개요

■ 스마트폰의 운영체제

- 2021년 2분기 기준 세계시장 점유율
 - 안드로이드: 72%, 아이폰: 26%, 윈도우폰: 0.01%

표 1-1 스마트폰 개발 환경 비교

구분	안드로이드	아이폰	윈도폰(단종)
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/폴드/노트 시리즈, 구글 픽셀/넥서스 시리즈	아이폰 시리즈, 아이패드 시리즈	노키아 루미아 시리즈
최신 개발 버전	Android 14.0(U)	iOS 17	윈도폰10
앱스토어	구글 플레이, 삼성 Apps, T스토어, 네이버 스토어 등	애플 앱스토어	Windows 스토어

2. 안드로이드

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 앱 향상, 가상 키보드 개선, 블루투스 2.1 지원
	프로요(Froyo)	2.2	8	2010년 6월	전반적인 성능 개선, USB 테더링 지원, 업데이트 가능 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	
	젤리빈(Jelly Bean)	4.1	16	2012년 7월	아이스크림 샌드위치 기반 위에서 더 빠르고 부드러워진 화면, 다양한 액세서리 지원
		4.2	17	2012년 12월	
		4.3	18	2013년 6월	
	킷캣(KitKat)	4.4	19	2013년 10월	메모리 관리 강화, GPU 가속화, UI 변경, 4.4w는 웨어러블 확장 지원
		4.4w	20	2014년 6월	

2. 안드로이드

	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
	Android R	11.0	30	2020년 9월	알림과 대화, 미디어 제어, 일화성 권한, 음성 액세스, 전원 메뉴, 스크롤 스트립 등
	Android S	12.0 12L	31 32	2021년 10월 2022년 3월	새로운 UI, 빠른 설정 개선, 성능 및 전력 효율 향상, 개인 정보 강화, 구글 어시스턴트 기능 개선 등(12L은 태블릿 및 폴더블폰 전용)
	Android T	13.0	33	2022년 8월	앱별 언어 환경 설정, 테마 앱 아이콘, 빠른 설정, 텍스트 변환 API 등
	Android U	14.0	34	2023년 10월	사진 선택기, 알림 플래시 카메라 및 배터리 수명 개선, 잠금 화면 사용자 지정, 매직 컴포즈, 앱 패더 등

2. 안드로이드

■ 안드로이드의 주요 기능_(1)

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

2. 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
- Android 11.0(R) : 알림과 대화, 미디어 제어, 일회성 권한, 음성 액세스, 전원 메뉴, 스크롤 스트립 등을 지원
- Android 12.0(S) : 사용자의 UI 커스터마이징 기능 강화, 머신러닝을 카메라 앱에 적용, 제한없는 4K 영상 촬영, Privacy Dashboard 추가, 상단 바에서 센서 해제, 위치 권한을 대략적 위치만 허용 기능, 와이파이 보안 강화 등 지원
- Android 13.0(S) : 앱별 언어 환경 설정, 테마 앱 아이콘, 빠른 설정, 텍스트 변환 API, 새 복사하여 붙여넣기 UI, 예측 뒤로 동작, 더 빠른 하이픈 연결, 블루투스 LE 오디오, MIDI 2.0 등 지원
- **Android 14.0(U)** : 사진 선택기, 알림 플래시, 카메라 및 배터리 수명 등이 추가/개선

2. 안드로이드

■ 안드로이드의 특징

- ① 안드로이드의 핵심 커널(Kernel) : 리눅스(Linux)로 구성되어 있음
- ② 안드로이드 애플리케이션 개발 언어 : Java
- ③ 안드로이드 SDK에서 많은 라이브러리를 포함하고 있어 개발이 용이함
- ④ 오픈 소스를 지향함
- ⑤ 지속적이고 빠른 업그레이드를 제공함

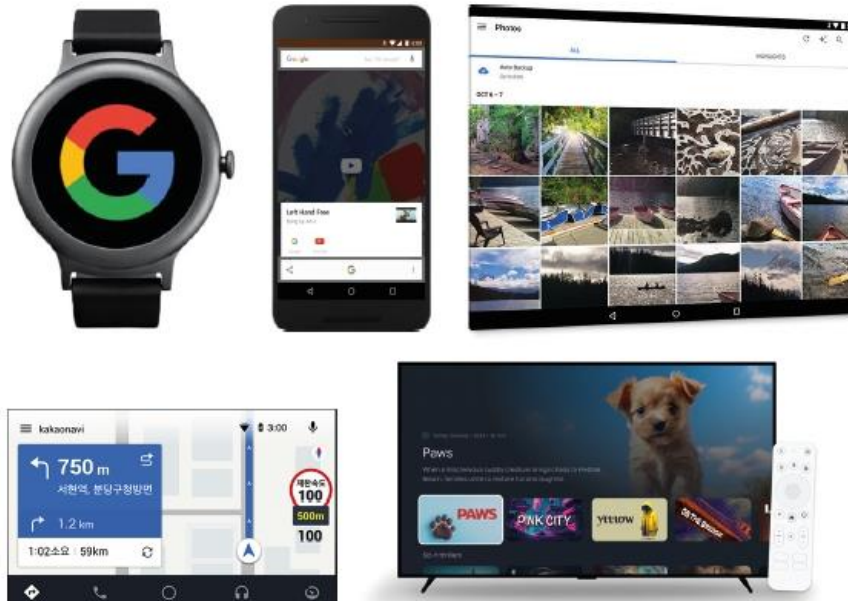


그림 1-2 다양한 안드로이드 기기(스마트워치, 스마트폰, 태블릿, 안드로이드 AUTO, 안드로이드 TV)

출처: google.com

2. 안드로이드

■ 안드로이드의 구조

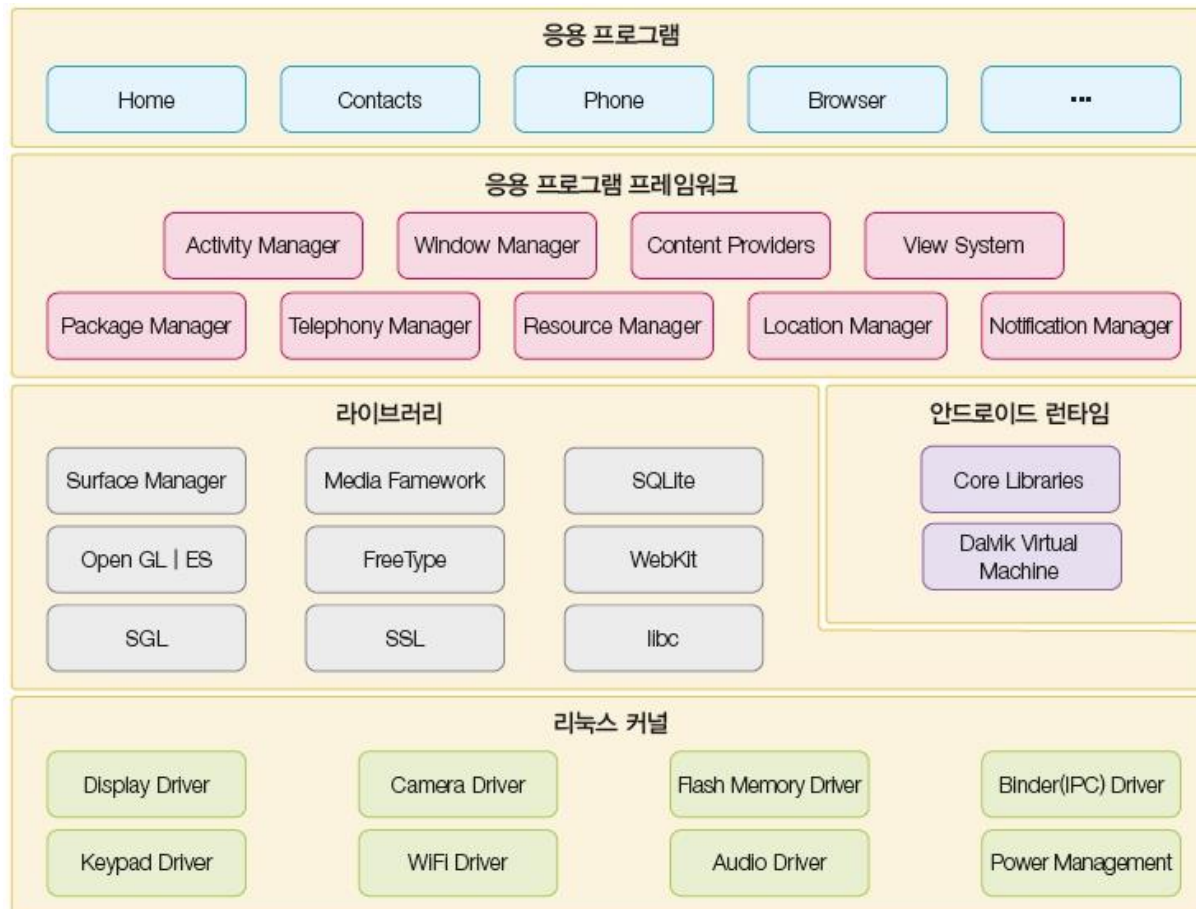


그림 1-3 안드로이드의 구조

출처: developer.android.com

2. 안드로이드

■ 안드로이드의 구조

① 응용 프로그램(Applications)

- 안드로이드 스마트폰에서 사용할 수 있는 일반적인 응용 프로그램
- 웹 브라우저, 달력, 구글맵, 연락처, 게임 등 사용자 입장에서 가장 많이 사용함
- Java로 작성됨

② 응용 프로그램 프레임워크(Application Framework)

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

2. 안드로이드

■ 안드로이드의 구조

③ 안드로이드 런타임(Android Runtime)

- Java 코어 라이브러리와 달빅 가상 머신(Dalvik Virtual Machine) 또는 아트 런타임(ART Runtime)으로 구성됨
- 안드로이드는 Java 문법으로 프로그래밍하지만 Java 가상 머신을 사용하지 않고 이곳의 달빅 가상 머신이나 아트리런타임을 사용함

④ 라이브러리(Libraries)

- 안드로이드에서 사용되는 여러 시스템 라이브러리는 시스템 접근 때문에 Java가 아닌 C로 작성 성능이 뛰어나며 세밀한 조작 가능함

⑤ 리눅스 커널(Linux Kernel)

- 하드웨어의 운영과 관련된 저수준의 관리 기능이 들어있음
 - 메모리 관리, 디바이스 드라이버, 보안 등
- 많은 스마트폰 장치를 지원함
 - 카메라, 터치스크린, GPS, 자이로스코프 등

02

안드로이드 개발 환경

1. 안드로이드 개발을 위한 운영체제

■ 안드로이드 개발을 위한 운영체제

- Windows, Linux, Mac
- Windows 개발 환경이 가장 보편적
- 안드로이드 개발 환경을 구축하면 가상의 안드로이드 장치인 AVD(Android Virtual Device) 제공
- 실제 안드로이드 장비 없이도 안드로이드 장비가 있는 것처럼 작업 가능
- Android Studio Dolphin (2021.3.1) 버전은 Windows 운영체제의 64bit만 설치할 수 있으며 32bit Windows는 지원하지 않음

2. 안드로이드 개발 환경의 구성

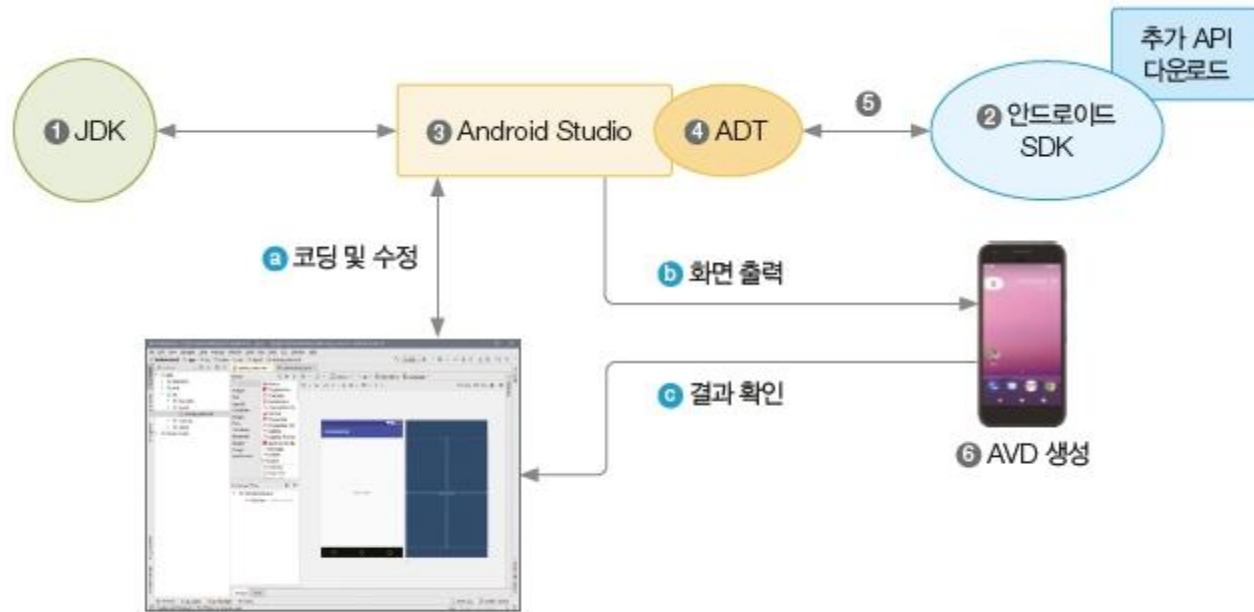


그림 1-4 안드로이드 개발 환경의 구성

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

2. 안드로이드 개발 환경의 구성

여기서 잠깐



안드로이드 개발 환경의 종류

안드로이드 개발 환경은 크게 세 가지로 나뉜다. 이 책은 입문자용이므로 응용 프로그램 개발 환경인 Java와 SDK만을 다룬다. 이후 고급 안드로이드 개발자로 도약하려면 NDK와 PDK도 공부해야 한다.

표 1-3 안드로이드 개발 환경

개발 환경	사용 언어	수준	개발 툴킷	비고
응용 프로그램 개발	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

2. 안드로이드 개발 환경의 구성

여기서 잠깐



Android Studio의 역사

Android Studio는 안드로이드 프로그래밍을 위한 통합개발환경(IDE)이며 구글에서 무료로 배포한다. Android Studio 이전에는 주로 Eclipse가 사용되었으나 2014년 12월에 Android Studio 1.0이 출시되면서 점점 많은 개발자들이 Android Studio를 사용하기 시작하여 현재는 안드로이드 개발환경의 대부분을 차지하고 있다. Android Studio는 버전별로 Canary → Beta → RC(Release Candidate, 출시 후보) → Stable(정식)의 차례로 출시되고 있다. 정식 버전 이후에 업데이트되면 뒤에 Patch라고 명명해서 배포된다. 2024년 1월 초 기준으로 Android Studio Hedgehog(2023.1.1) Stable, Android Studio Iguana(2023.2.1) Beta가, Android Studio Jellyfish(2023.3.1) Canary가 출시되었다.

Android Studio 4.2 이후부터는 표기 방식을 'Android Studio 연도.주버전.부버전'으로 변경했으며, '연도.주버전'은 JetBrains에서 만든 IntelliJ IDEA의 버전을 따르고 있다. 또한 Android Studio 2020.3.1부터는 버전 이름에 알파벳 A부터 동물 이름을 순서대로 붙이고 있다.

표 1-4 Android Studio 주요 버전별 출시일

Android Studio 정식 버전	출시일	기타
1.0~4.0	2014~2020년	버전으로 표기
Arctic Fox(2020.3.1)	2021년 7월	의미: 북극 여우
Bumblebee(2021.1.1)	2022년 1월	의미: 호박벌
Chipmunk(2021.2.1)	2022년 5월	의미: 다람쥐
Dolphin(2021.3.1)	2022년 9월	의미: 돌고래
Electric Eel(2022.1.1)	2023년 1월	의미: 전기 뱀장어
Flamingo(2022.2.1)	2023년 4월	의미: 플라밍고
Giraffe(2022.3.1)	2023년 7월	의미: 기린
Hedgehog(2023.1.1)	2023년 11월	의미: 고슴도치
Iguana(2023.2.1)	2024년 상반기 예상	의미: 이구아나
Jellyfish(2023.1.1)	2024년 중반기 예상	의미: 해파리

03

안드로이드 개발 환경 설치

1. 안드로이드 개발 환경 설치 순서

■ 안드로이드 개발 환경 구축

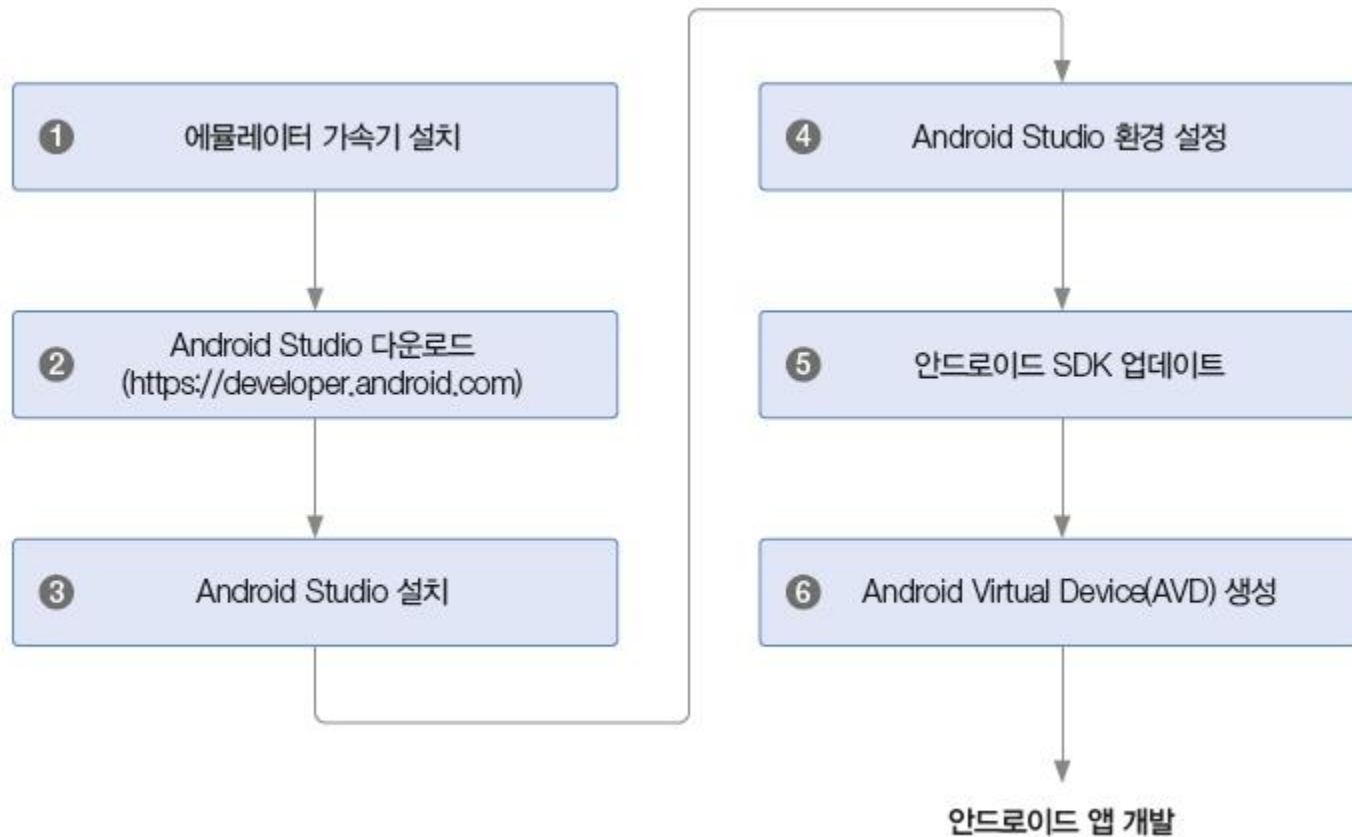


그림 1-5 안드로이드 개발 환경 설치 순서

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-1 에뮬레이터(AVD) 가속기 설치

- (1-1) [제어판]의 [시스템 및 보안]-[시스템] 실행
 - [장치 사양] 부분에서 프로세서, 설치된 RAM, 시스템 종류 확인

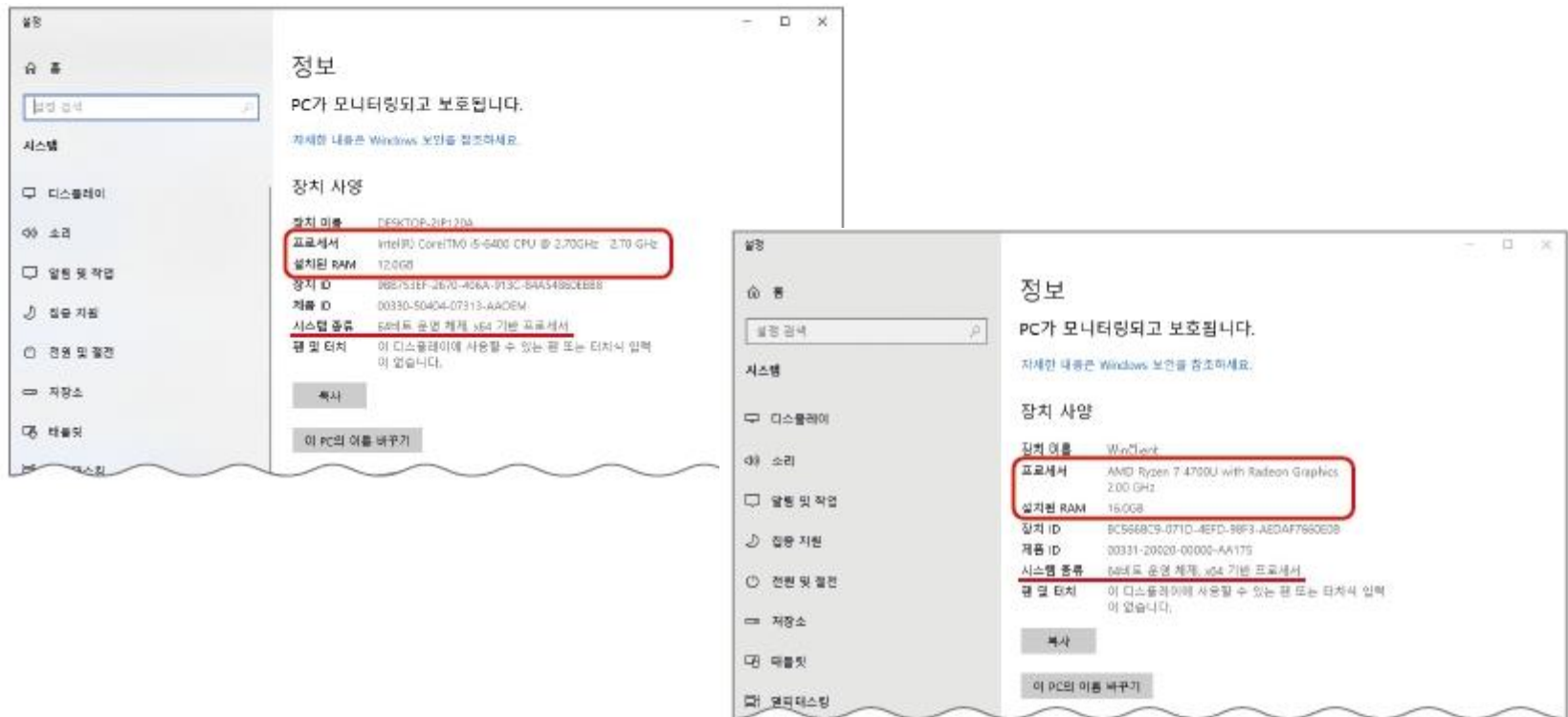


그림 1-6 컴퓨터 정보 확인(인텔 CPU와 AMD CPU)

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-1 에뮬레이터(AVD) 가속기 설치(2 인텔 CPU인 경우)

- (2-1) <https://cafe.naver.com/itCookBook/96>에 접속
 - 인텔 CPU용 가속기인 haxm-7.7.1-setup.exe 파일 다운로드
- (2-2) 다운로드한 haxm-7.7.1-setup.exe 파일의 압축 풀기
 - haxm-7.7.1-setup.exe 파일 실행 후 설치 진행
 - 초기 화면에서 <Next>와 <Install> 클릭
- (2-3) 설치가 완료되면 <Finish>를 클릭하여 설치 종료

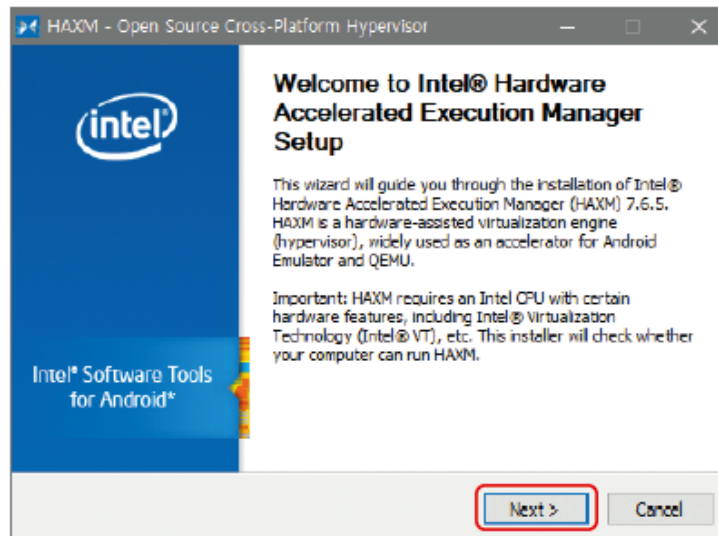


그림 1-7 인텔 HAXM 설치

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-1 에뮬레이터(AVD) 가속기 설치(3. AMD CPU인 경우)

- (3-1) <윈도우 키>+<S> 를 누른 후 “Windows 기능”을 입력해서 [Windows 기능 켜기/끄기]를 실행
- (3-2) [Windows 기능 켜기/끄기] 창에서 [Hyper-V], [Windows 샌드박스], [Windows 하이퍼바이저 플랫폼]이 모두 꺼져 있어야 하는데, 만약 켜져 있다면 끄고 <확인> 클릭

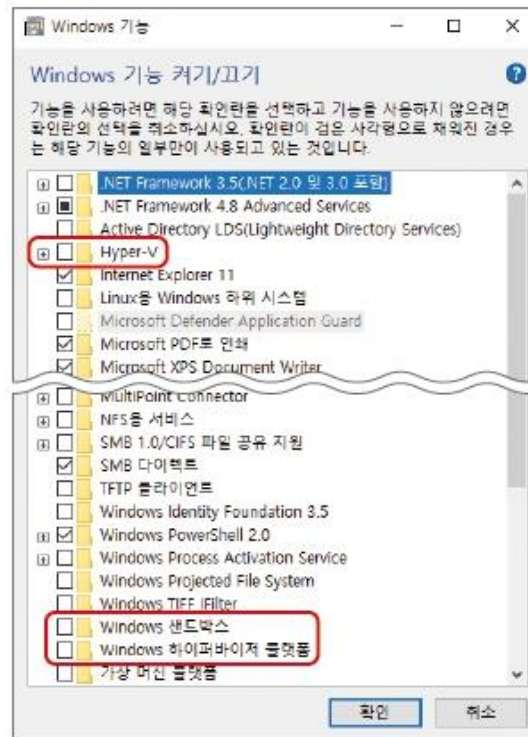


그림 1-8 Windows 기능 켜기/끄기

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-1 에뮬레이터(AVD) 가속기 설치(3_AMD CPU인 경우)

- (3-3) <https://cafe.naver.com/itCookBook/96>에 접속 → AMD CPU용 가속기인 gvm-windows_v1_8_0.zip 파일 다운로드.
- (3-4) 다운로드한 aehd-windows_버전.zip 압축 풀기
→ 압축을 푼 폴더의 silent_install.bat 파일을 실행하여 설치 진행

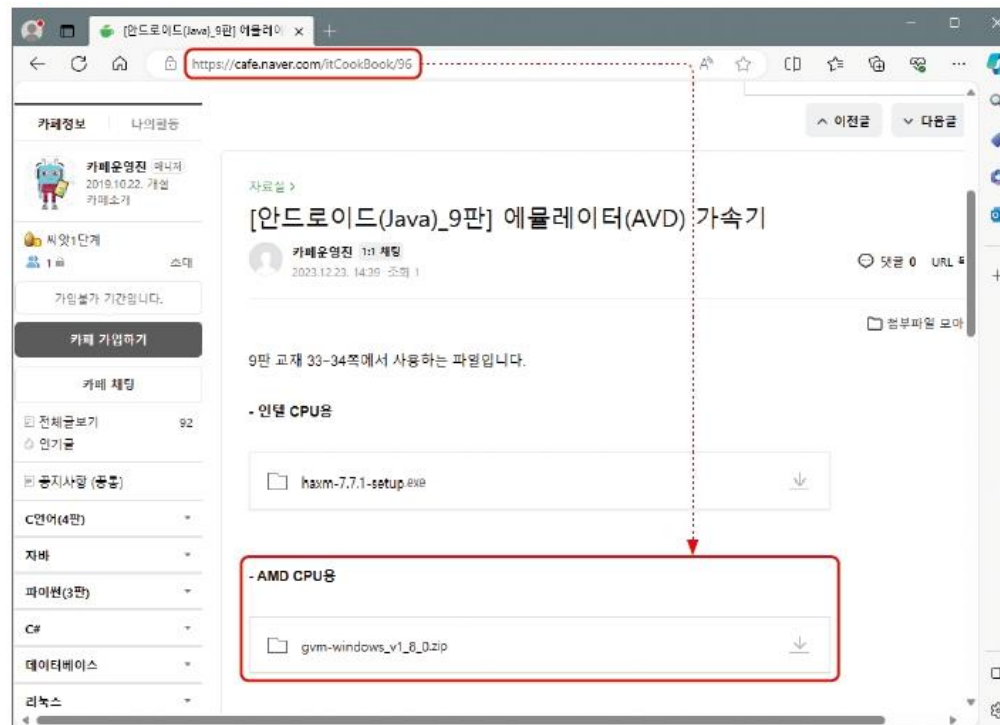


그림 1-9 AMD CPU용 가속기 다운로드

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-2 Android Studio 다운로드하기

- Android Studio Hedgehog (2023.1.1)을 다운로드하고 적당한 폴더에 압축을 풀
- (1) <https://developer.android.com/studio/archive.html> 접속
 - 상단 오른쪽 'English' 선택
 - <I AGREE TO THE TERMS> 클릭
 - Android Studio Hedgehog (2023.1.1) 버전의 파일 다운로드

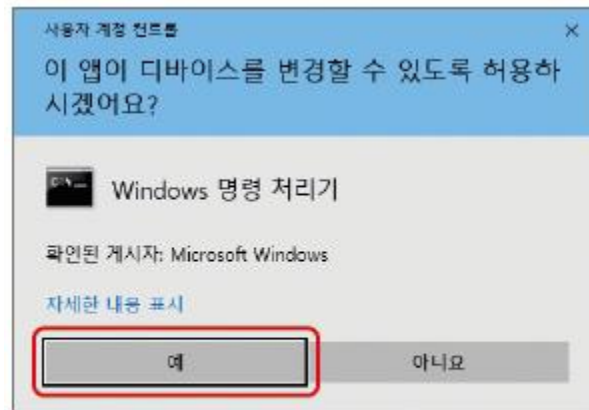


그림 1-10 Android Emulator hypervisor driver 설치

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-2 Android Studio 다운로드하기

- (2) 다운로드한 Android Studio Hedgehog (2023.1.1) 확인



그림 1-11 Android Studio Hedgehog(2023.1.1) 다운로드

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-3 Android Studio 설치하기

- (1) 다운로드 한 Android Studio Hedgehog (2023.1.1) 실행
(android-studio-2023.1.1.00-windows.exe)
 - 초기화면에서 <Next> 클릭



그림 1-12 Android Studio 다운로드 완료



그림 1-13 Android Studio 설치 초기화면

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-3 Android Studio 설치하기

- (2) 'Android Virtual Devices'의 체크 끄기 - <Next> 클릭

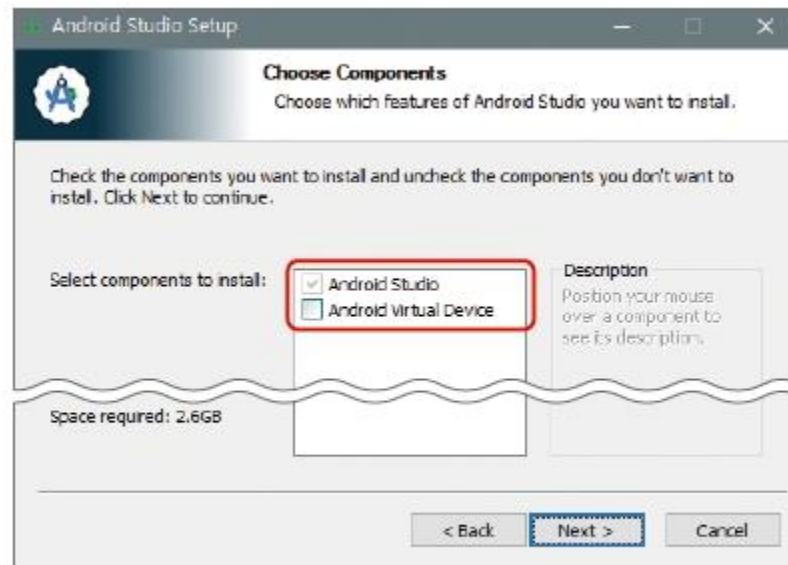


그림 1-14 Android Studio 설치 항목 선택

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-3 Android Studio 설치하기

- (3) Android Studio가 설치될 폴더 선택
 - 기본값으로 두고 <Next> 클릭

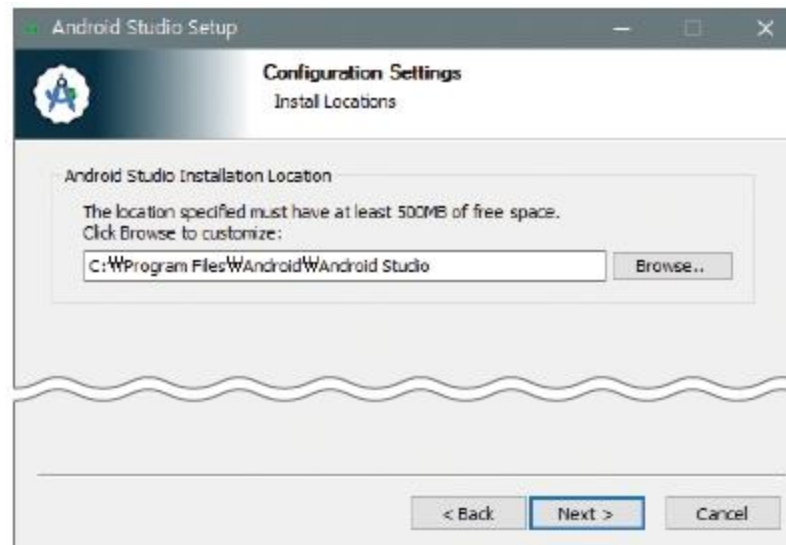


그림 1-15 설치 경로 설정

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-3 Android Studio 설치하기

- (4) 시작 메뉴 폴더 생성 화면에서도 기본값으로 두고 <Install> 클릭



그림 1-16 시작 메뉴 폴더 생성

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-3 Android Studio 설치하기

- (5) Android Studio 설치 진행 → (6) 설치 완료되면 <Next> 클릭

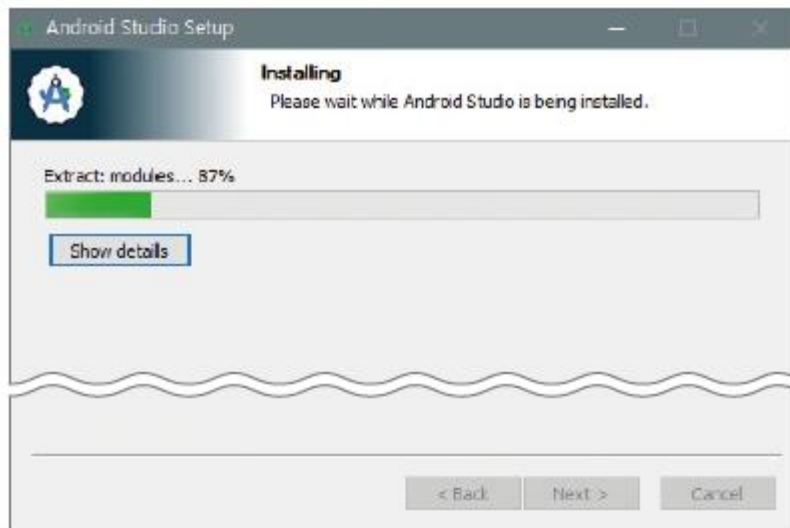


그림 1-17 설치 진행

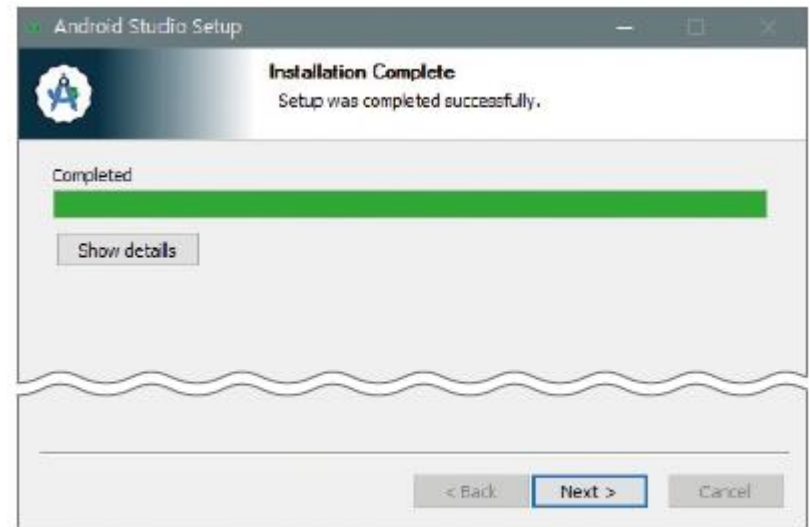


그림 1-18 설치 완료 1

2. 에뮬레이터(AVD) 가속기 설치

■ 실습 1-3 Android Studio 설치하기

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



그림 1-19 설치 완료 2

3. Android Studio 설정

■ 실습 1-4 Android Studio 몇 가지 설정하기

- (1) [Windows 시작]-[Android Studio]의 Android Studio 실행
 - 'Do not import settings' 선택된 상태에서 <OK> 클릭

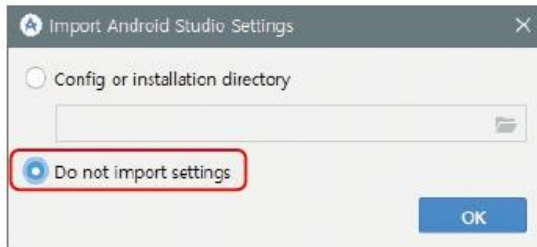


그림 1-20 Android Studio 이전 환경 가져오기 설정

- (2) 로고 화면이 나옴



그림 1-21 Android Studio 로고

3. Android Studio 설정

- 실습 1-4 Android Studio 몇 가지 설정하기
 - (3) [Android Studio Setup Wizard] 실행 - <Next> 클릭

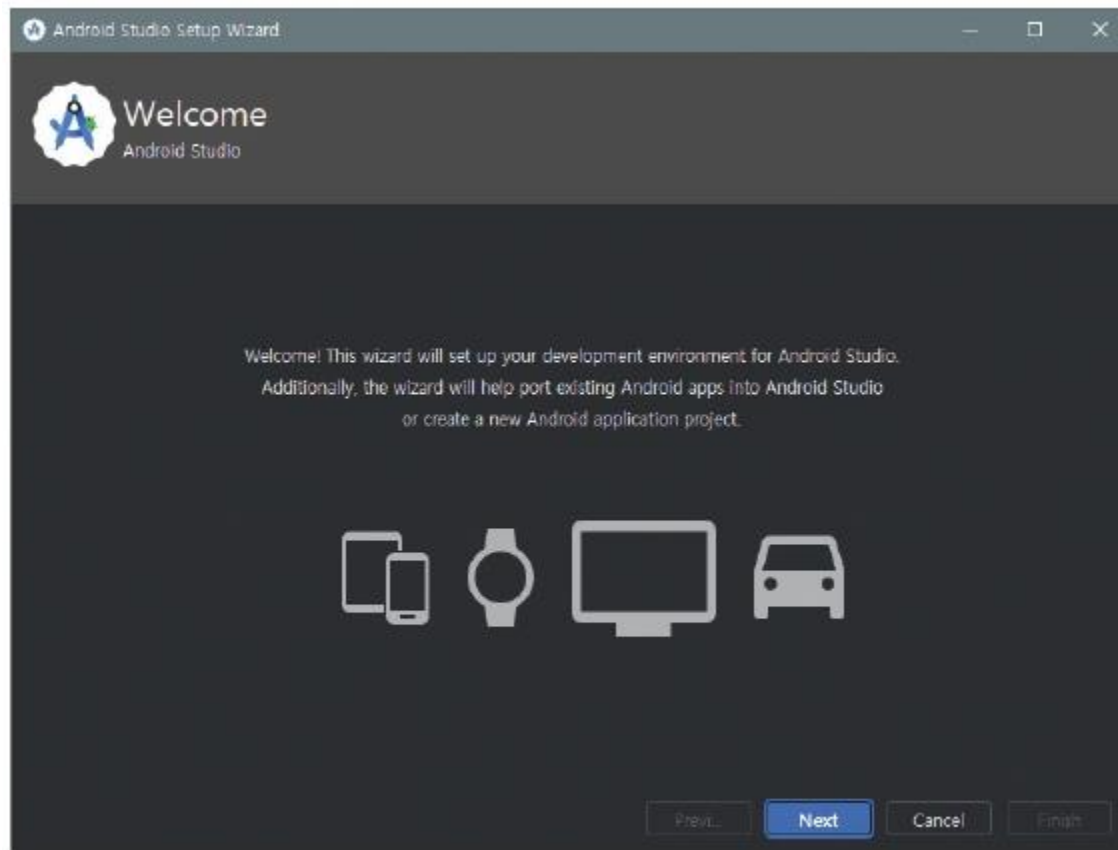


그림 1-22 Android Studio Setup Wizard 실행 1

3. Android Studio 설정

- 실습 1-4 Android Studio 몇 가지 설정하기
 - (4) [Install Type] : 'Standard' 선택 - <Next> 클릭

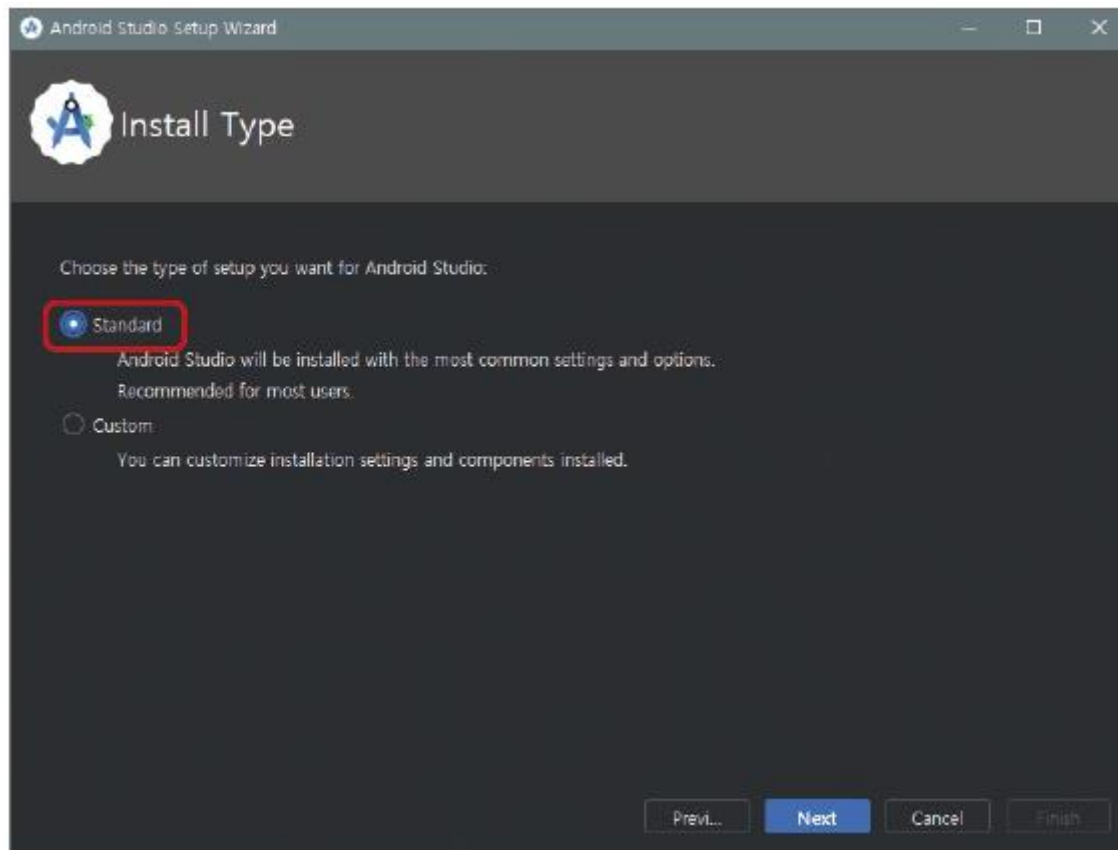


그림 1-23 Android Studio Setup Wizard 실행 2

3. Android Studio 설정

■ 실습 1-4 Android Studio 몇 가지 설정하기

- (5) [Verify Settings]에서 다운로드 크기 확인 → <Finish> 클릭

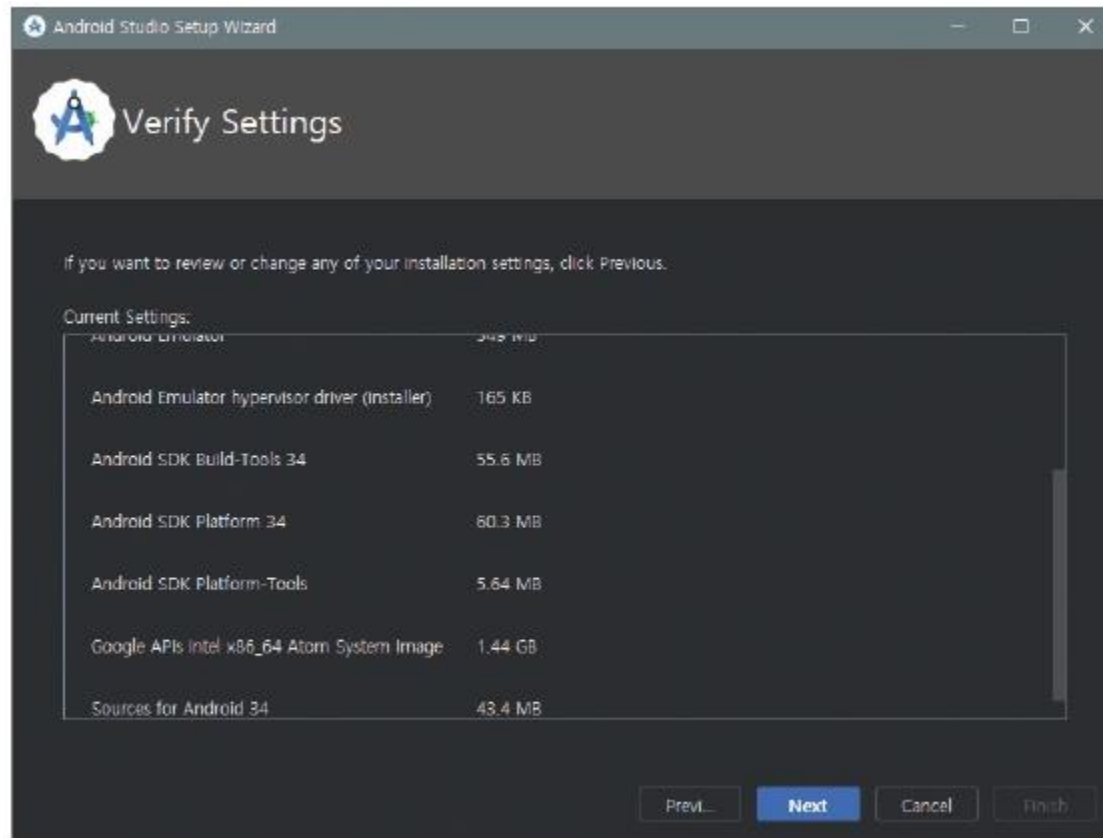


그림 1-24 Android Studio Setup Wizard 실행 3

3. Android Studio 설정

■ 실습 1-4 Android Studio 몇 가지 설정하기

- (6) [License Agreement] 창에서 각 항목 선택 → <Agree>를 클릭 → <Finish> 클릭

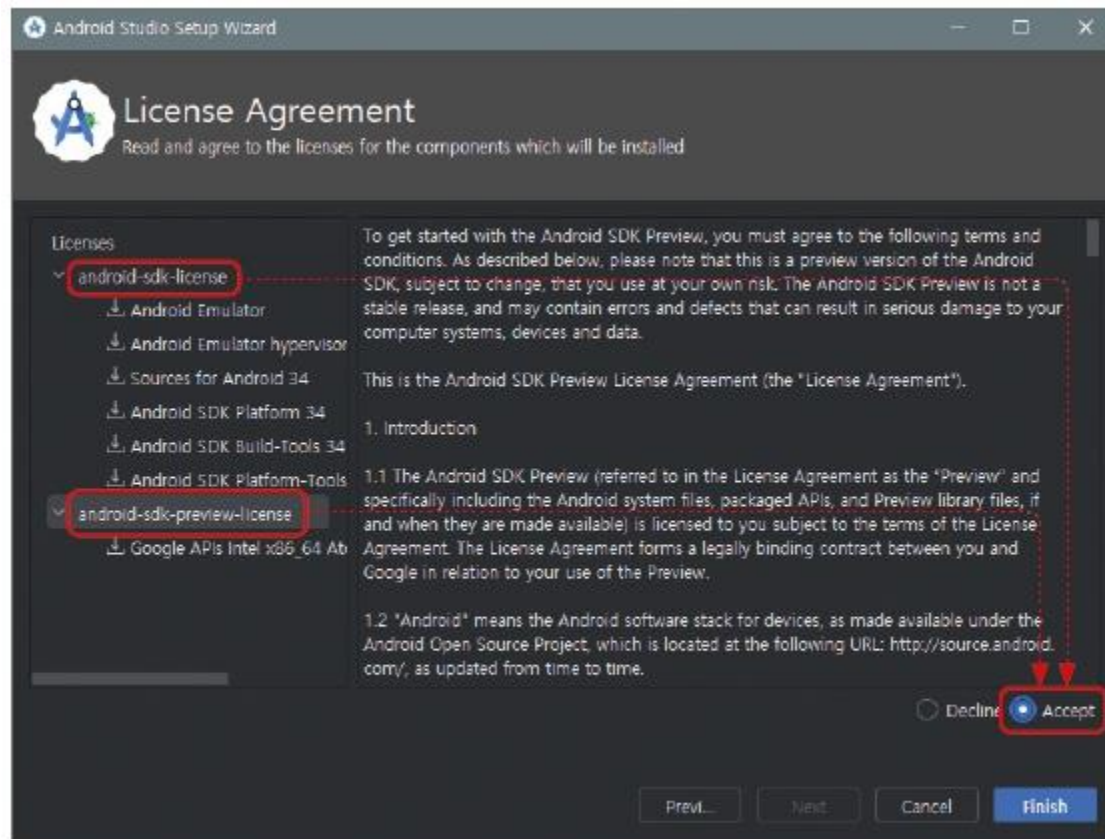


그림 1-25 Android Studio Setup Wizard 실행 4

3. Android Studio 설정

- 실습 1-4 Android Studio 몇 가지 설정하기
 - (7) 안드로이드 SDK의 다운로드 및 설치 진행

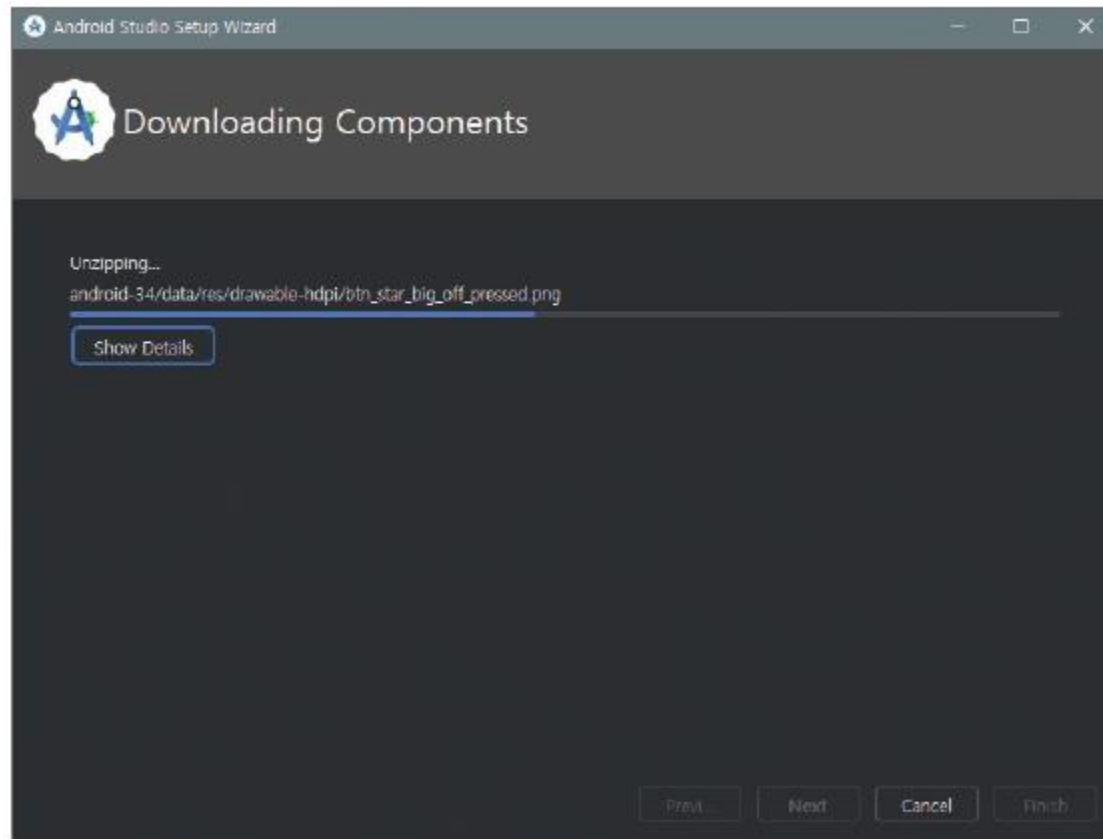


그림 1-26 Android Studio Setup Wizard 실행 5

3. Android Studio 설정

- 실습 1-4 Android Studio 몇 가지 설정하기
 - (8) <Finish> 클릭해서 종료

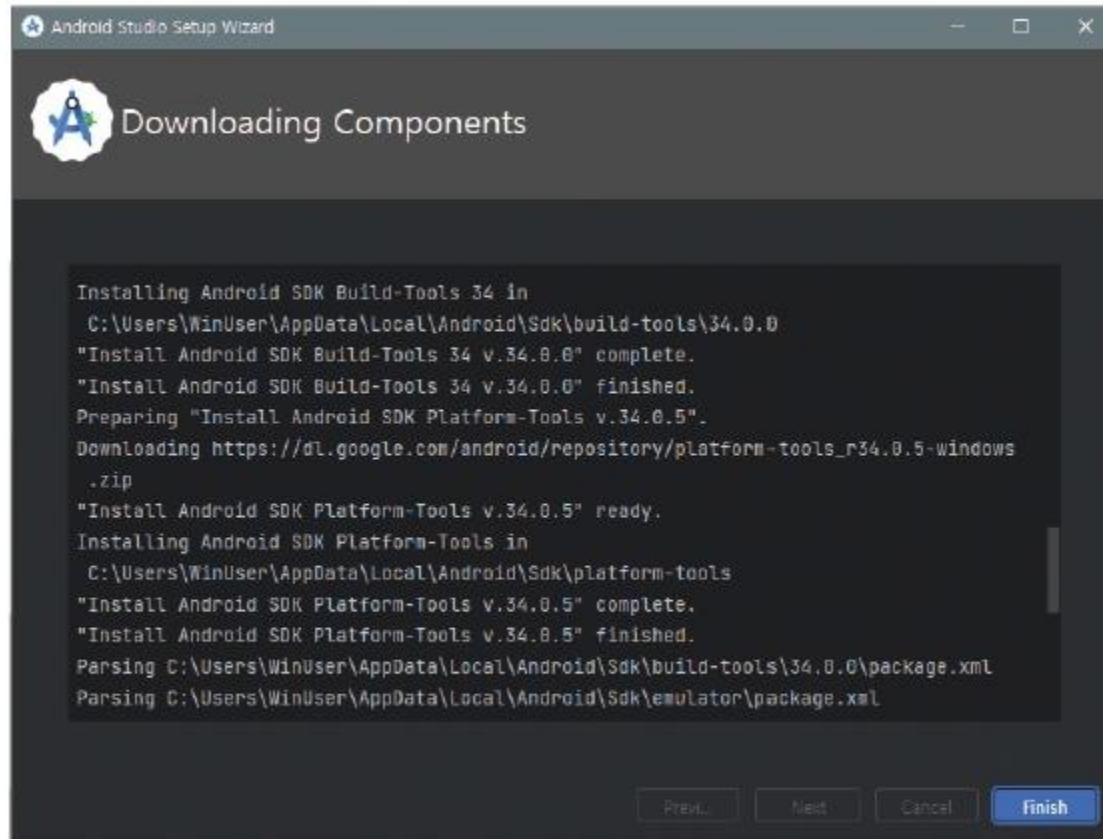


그림 1-27 Android Studio Setup Wizard 실행 6

3. Android Studio 설정

■ 실습 1-4 Android Studio 몇 가지 설정하기

- (9) Android Studio 초기화면에서 [Customize] 선택 → 'All Settings' 클릭

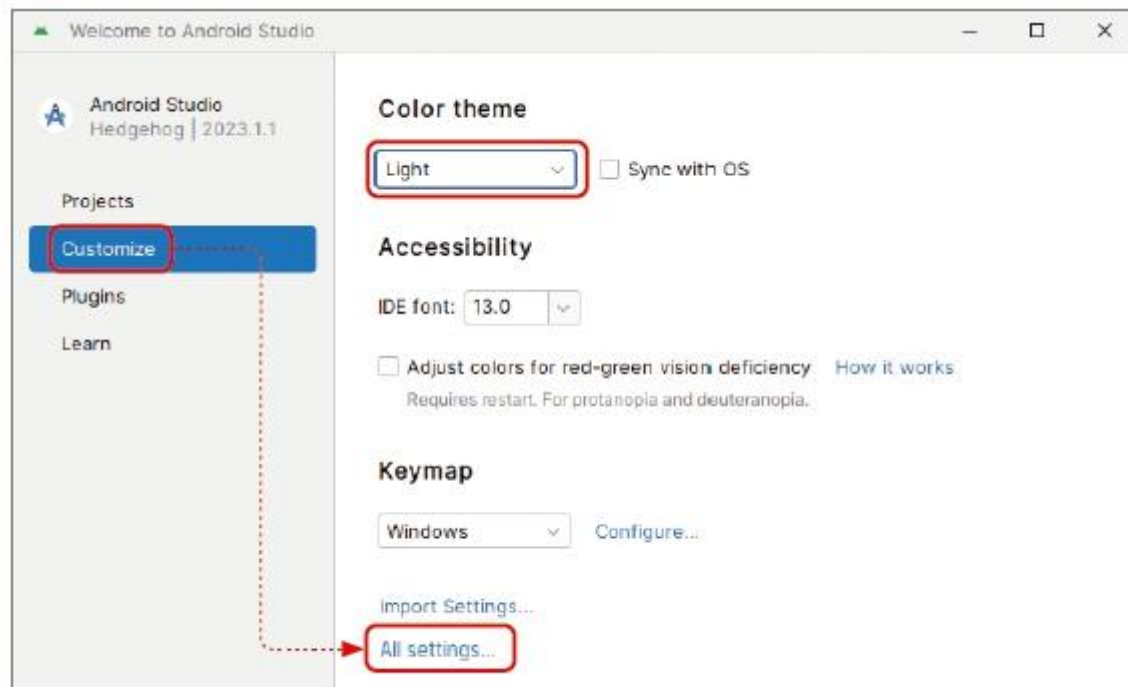


그림 1-28 Android Studio 초기화면

3. Android Studio 설정

■ 실습 1-4 Android Studio 몇 가지 설정하기

- (10) 자동 업데이트 기능 끄기
 - 1. 왼쪽의 [Appearance & Behavior]-[System Settings]-[Updates] 선택
 - 2. 오른쪽의 업데이트와 관련된 체크박스 끄기

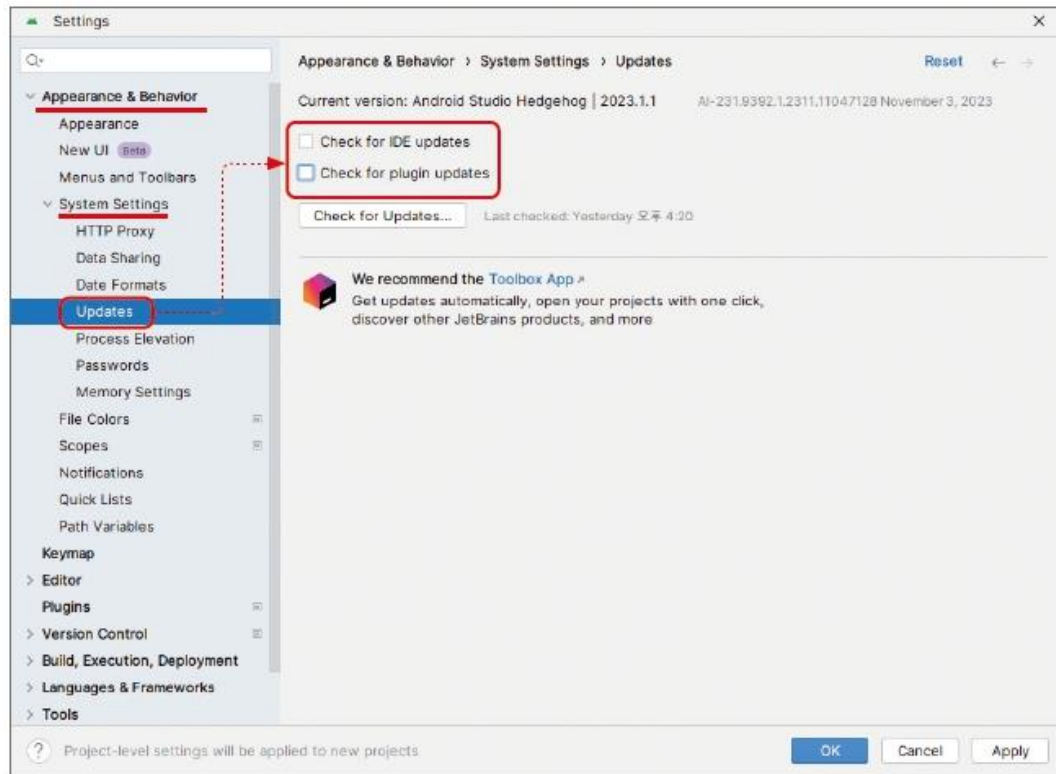


그림 1-29 업데이트 기능 끄기

3. Android Studio 설정

■ 실습 1-4 Android Studio 몇 가지 설정하기

- (11) [Appearance & Behavior]-[Appearance] 선택
→ 'Show main menu in a separate toolbar' 체크

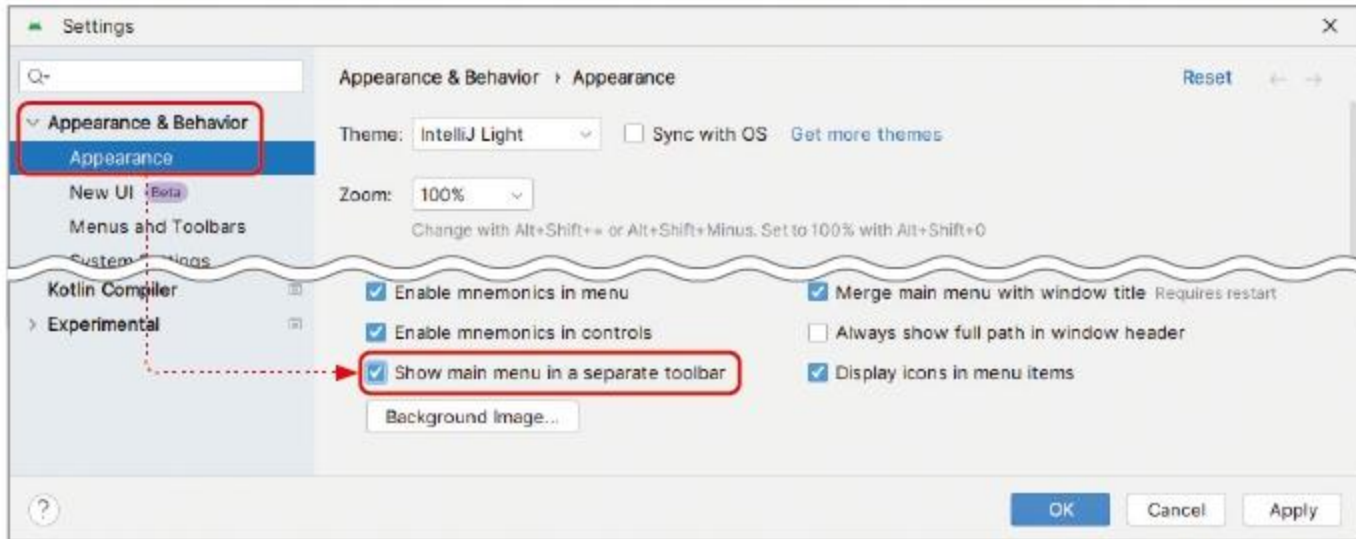


그림 1-30 메뉴 보이기 설정

3. Android Studio 설정

■ 실습 1-5 안드로이드 SDK 업데이트하기

- (1) Android Studio 초기화면에서 왼쪽 [Project]를 선택 →
오른쪽 아래 'More Actions'를 클릭 → [SDK Manager]를 선택

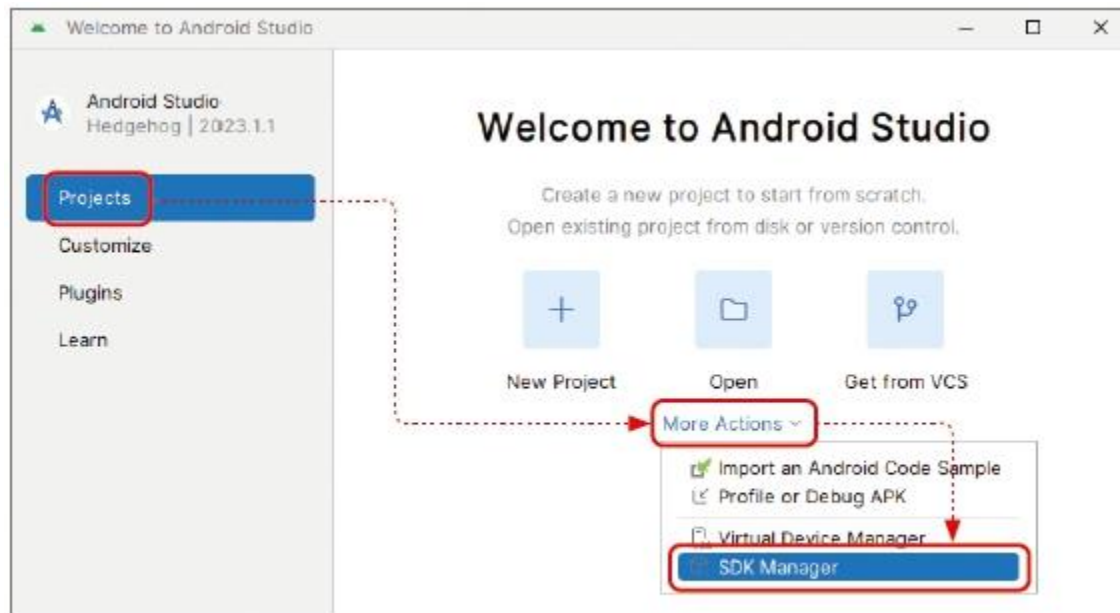


그림 1-31 안드로이드 SDK 추가 설치 1

3. Android Studio 설정

■ 실습 1-5 안드로이드 SDK 업데이트하기

- (2) 오른쪽 아래 'Show Package Details'에 체크 후, 기존의 체크 항목 모두 끄기
 - Android 14.0(U) 아래의 'Android S아 Platform 34'와 'Google APIs Intel x86 Atom_64 System Image'만 체크
 - 아직 <OK> 클릭은 하지 않음

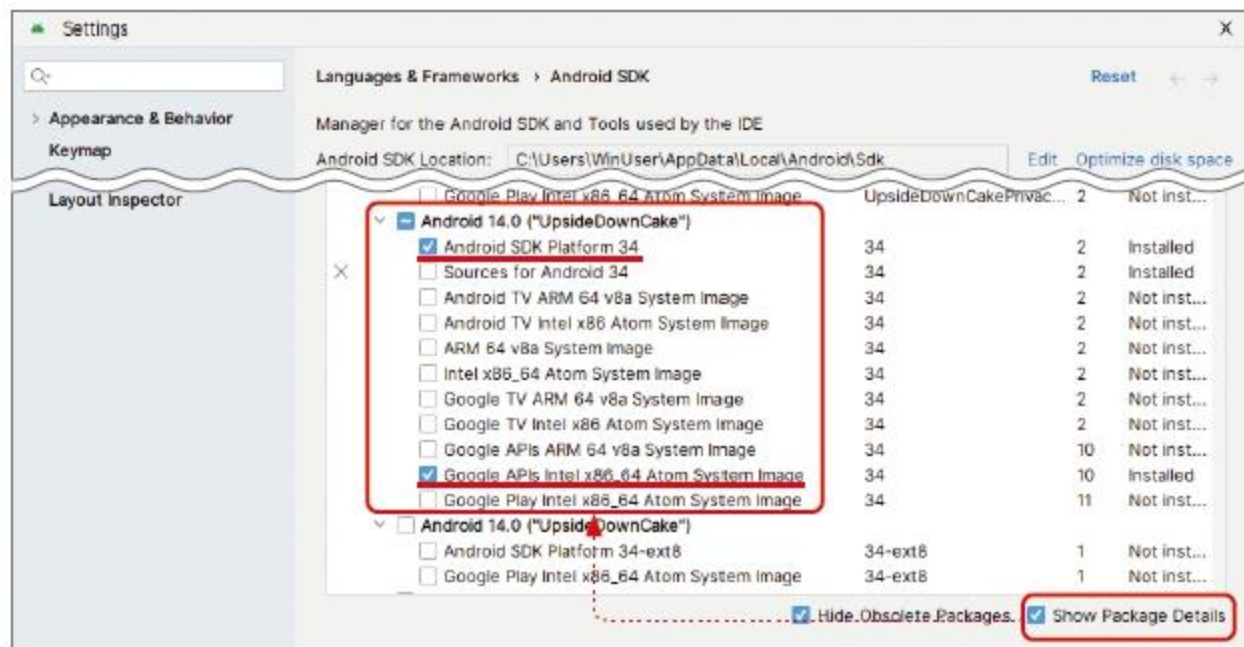


그림 1-32 안드로이드 SDK 추가 설치 2

3. Android Studio 설정

■ 실습 1-5 안드로이드 SDK 업데이트하기

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

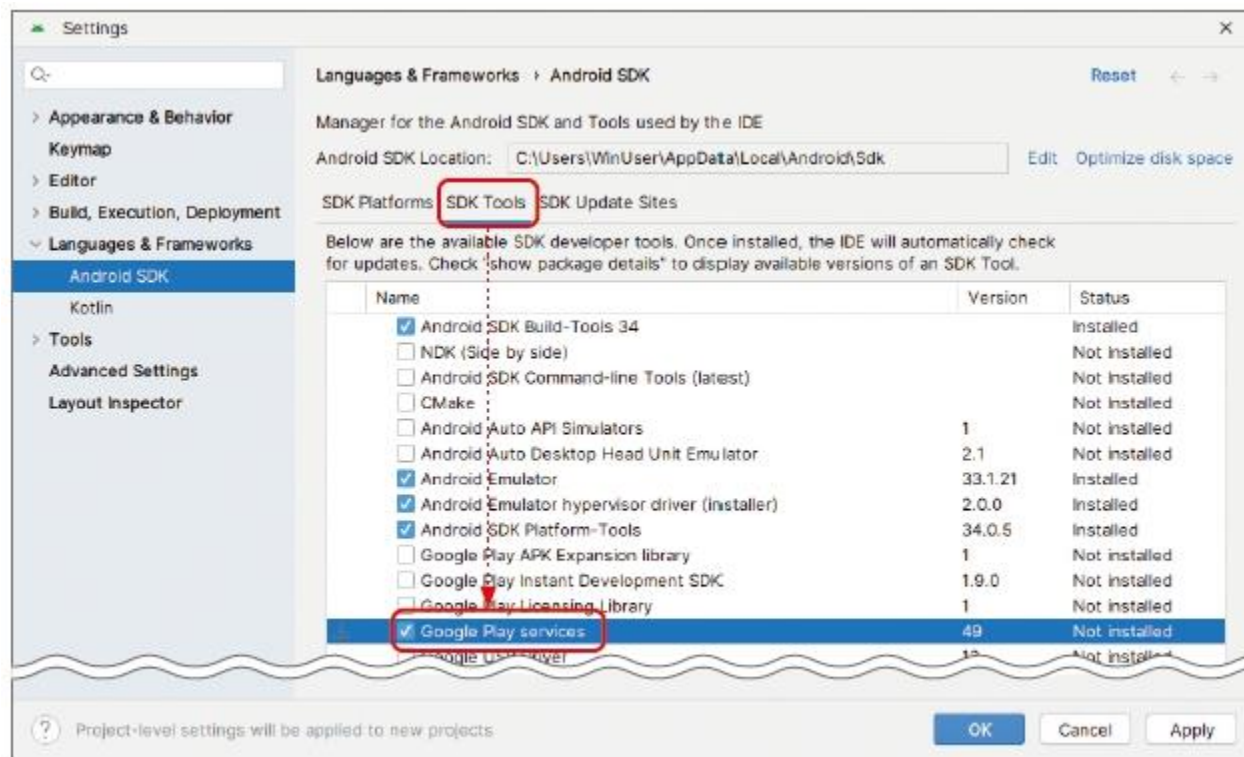


그림 1-33 안드로이드 SDK 추가 설치 3

3. Android Studio 설정

■ 실습 1-5 안드로이드 SDK 업데이트하기

- (4) [SDK Component Installer] 창이 나오고 잠시 설치 진행

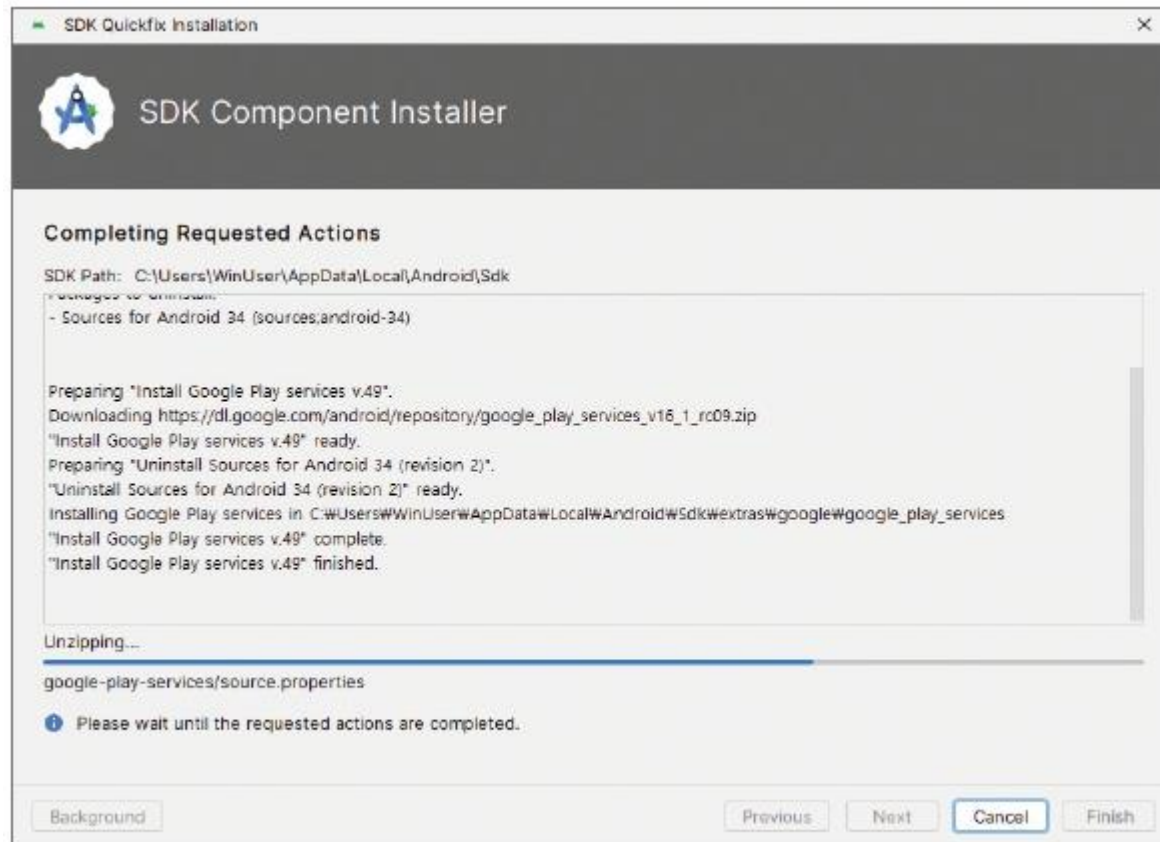


그림 1-34 다운로드 및 설치 진행

3. Android Studio 설정

- 실습 1-5 안드로이드 SDK 업데이트하기
 - (5) <Finish>를 클릭하여 설치를 종료

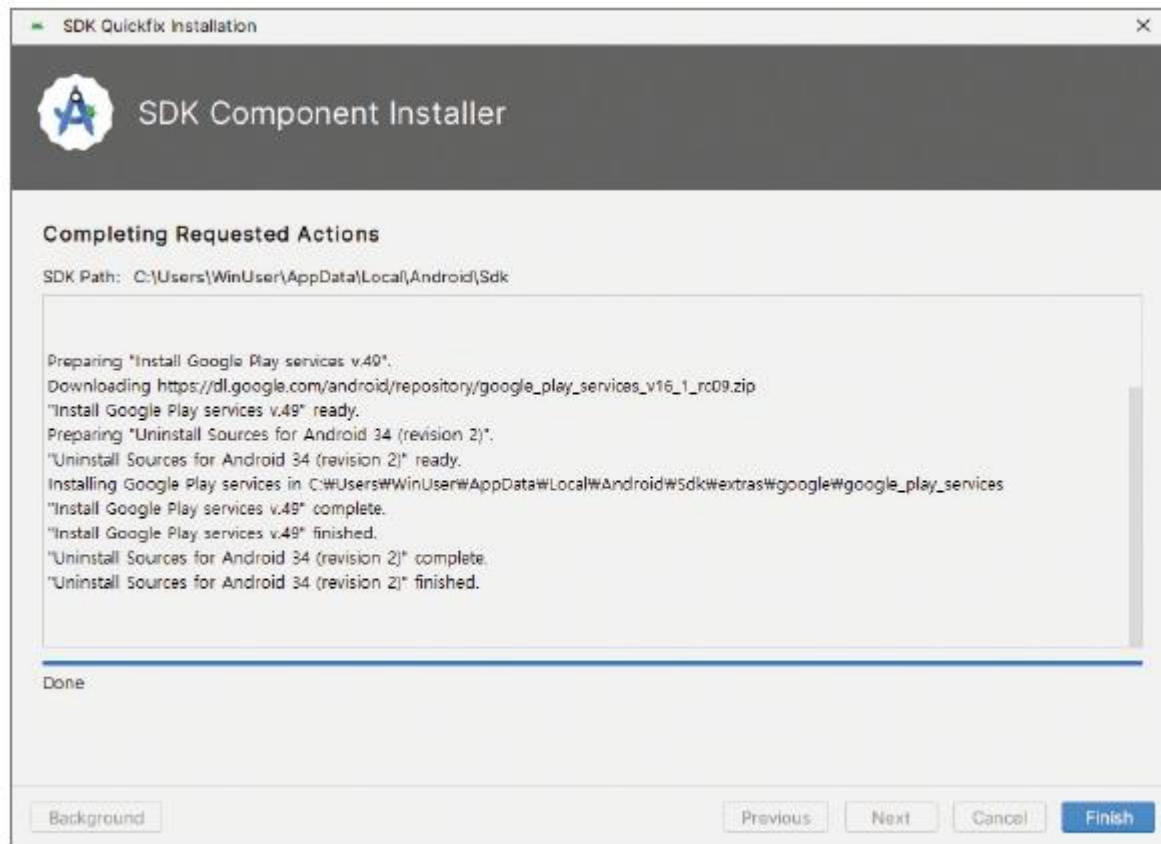


그림 1-35 SDK 설치 완료

3. Android Studio 설정

■ 실습 1-5 안드로이드 SDK 업데이트하기

- (6) Android Studio 및 안드로이드 SDK의 설치가 완료됨 → Android Studio 종료

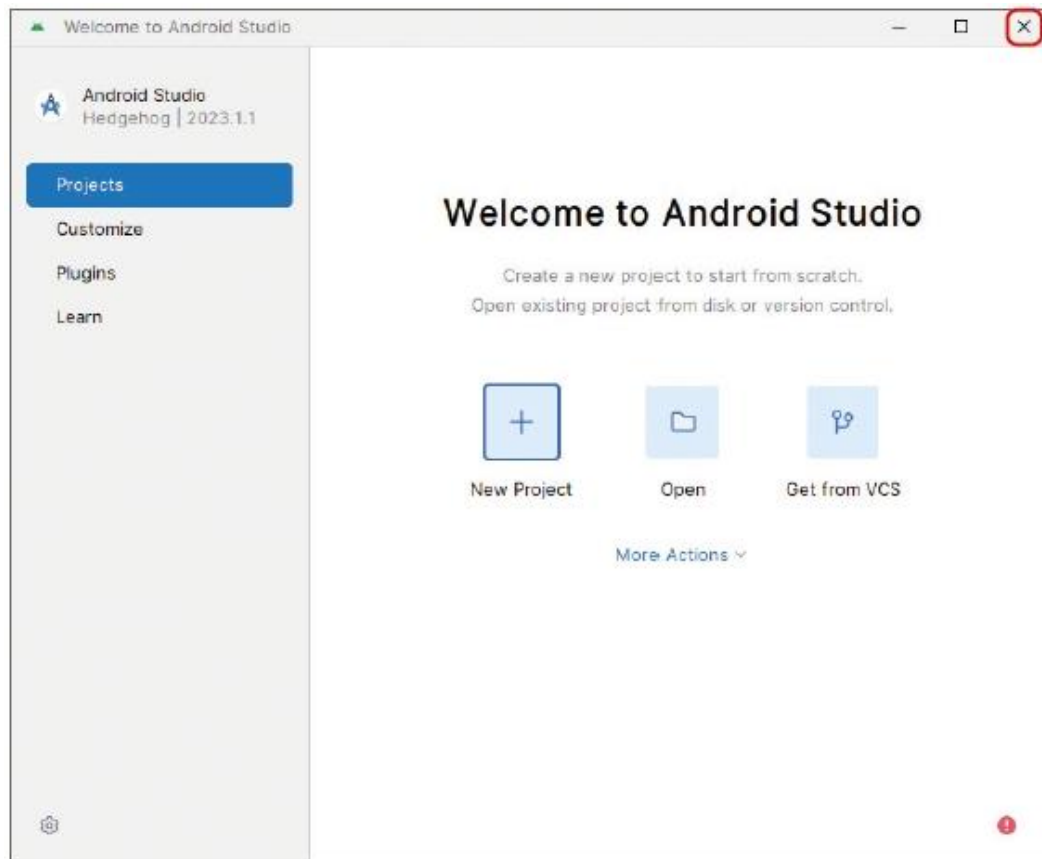


그림 1-36 Android Studio 종료

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- AVD는 안드로이드폰의 기능을 90% 정도 사용할 수 있지만 실제 안드로이드폰보다 느리다는 것을 감수해야 함
- (1) Windows [시작]을 클릭해서 Android Studio를 실행

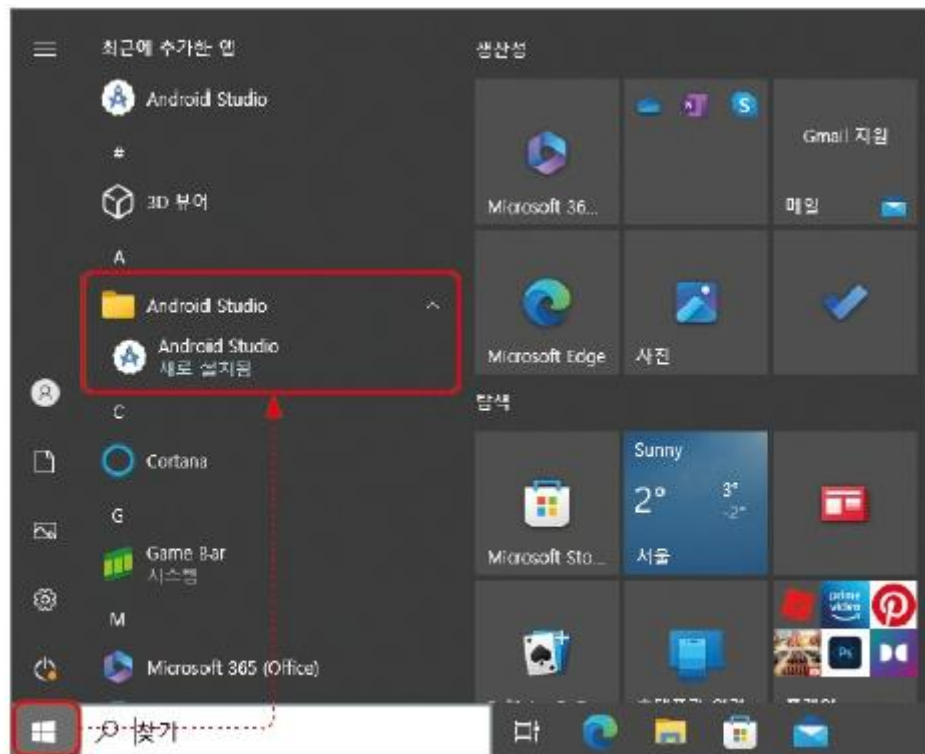


그림 1-37 Android Studio 실행

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (2) 잠시 Android Studio의 로고가 등장
- (3) 예전 버전의 Android Studio는 AVD를 생성하기 위해서 프로젝트를 생성해야 했지만, 현재의 Android Studio는 초기화면에서 AVD의 생성/삭제/수정 등의 관리가 가능
- (4) Android Studio 초기화면의 왼쪽 [Project]를 클릭하고, 오른쪽 아래의 'More Actions'를 클릭하여 [Virtual Device Manager]를 선택

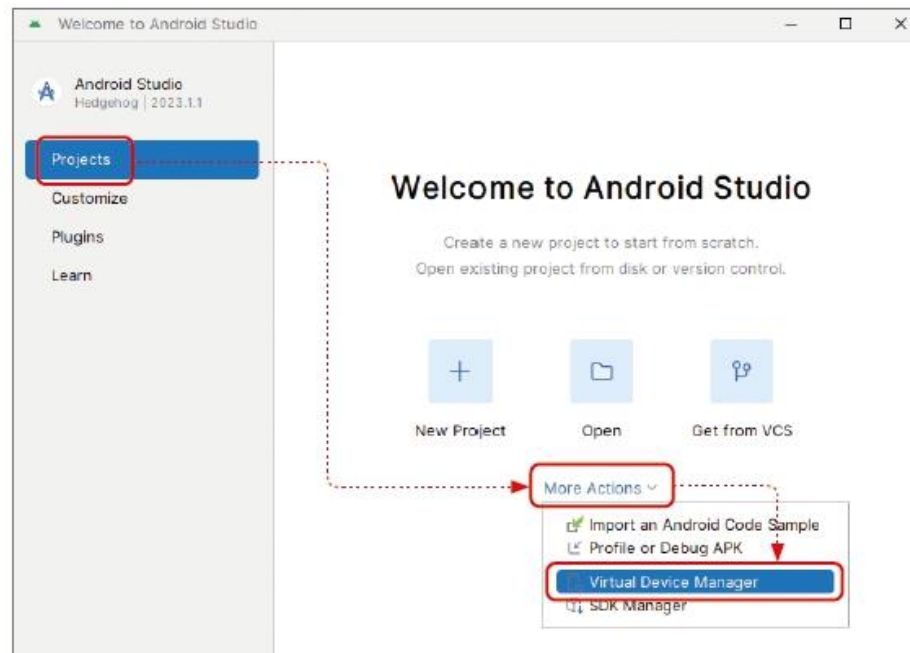


그림 1-38 Virtual Device Manager(AVD Manager) 실행

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (5) [Device Manager]가 실행 → 만약 자동으로 생성된 AVD가 있다면 선택하고, 오른쪽 [:]를 클릭한 후 [Delete]를 선택해서 삭제
- (6) 경고 메시지가 나오면 <Yes>를 선택

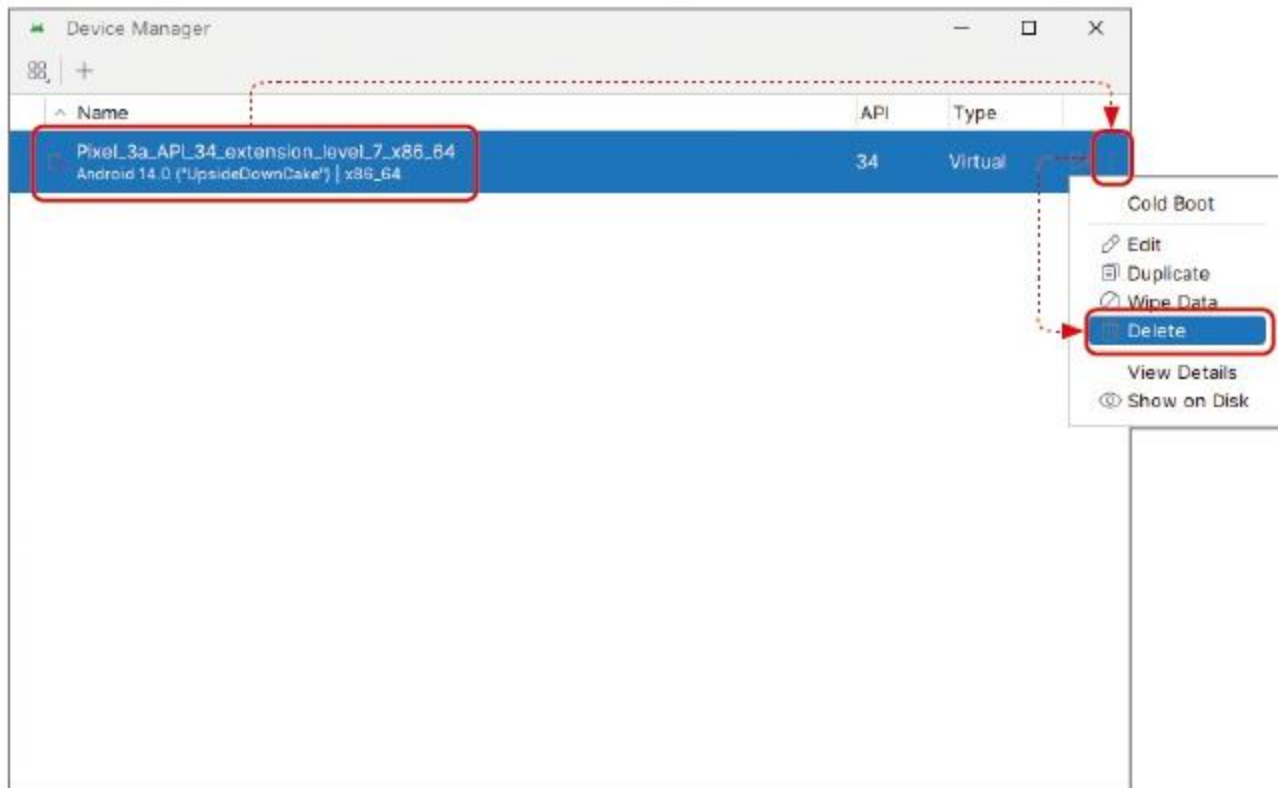


그림 1-39 자동 생성된 AVD 삭제

4. AVD 생성

- 실습 1-6 개발 테스트를 위한 AVD 생성하기
 - (7) <Create virtual device>를 클릭한다.

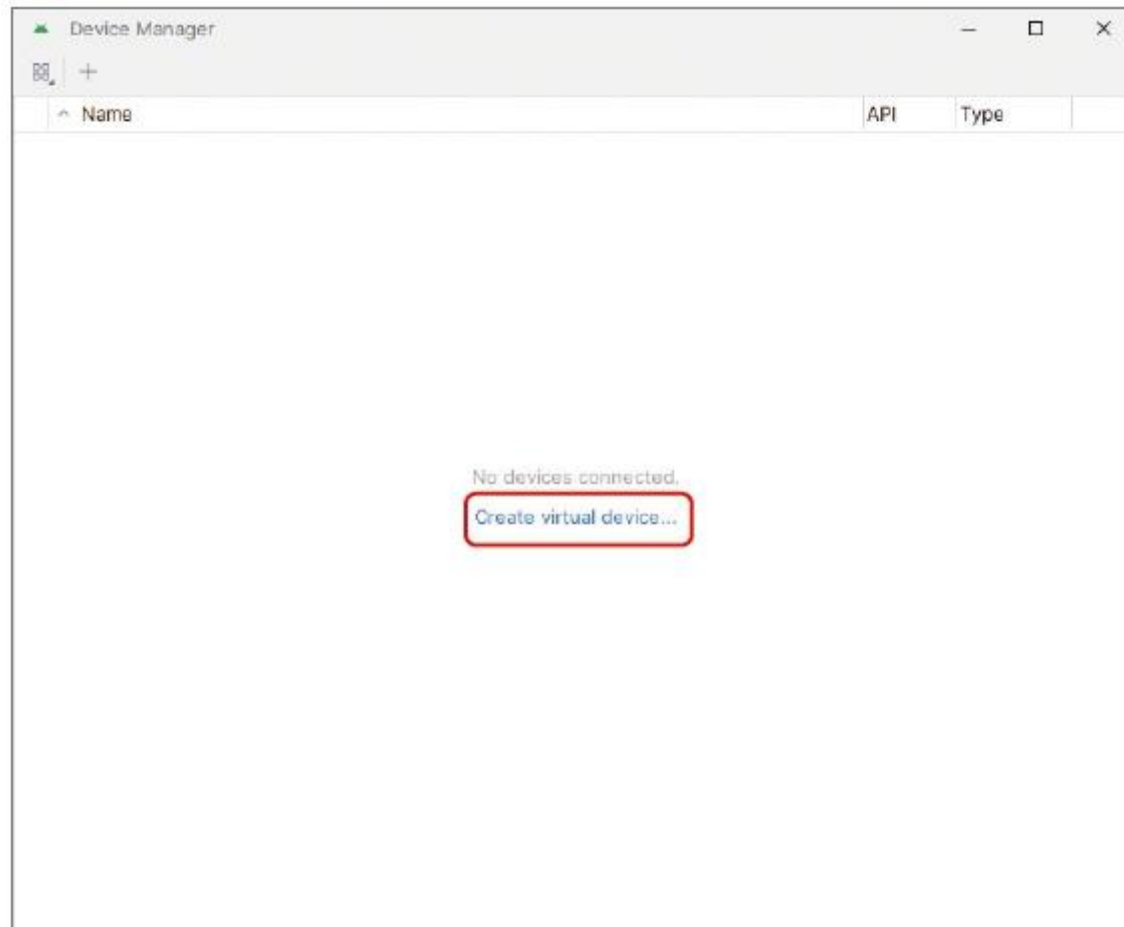


그림 1-40 AVD 생성 1

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (8) [Select Hardware] 창에서 왼쪽의 [Phone]이 선택된 상태
 - 구글의 레퍼런스폰인 'Pixel2' 선택하고 <Next> 클릭

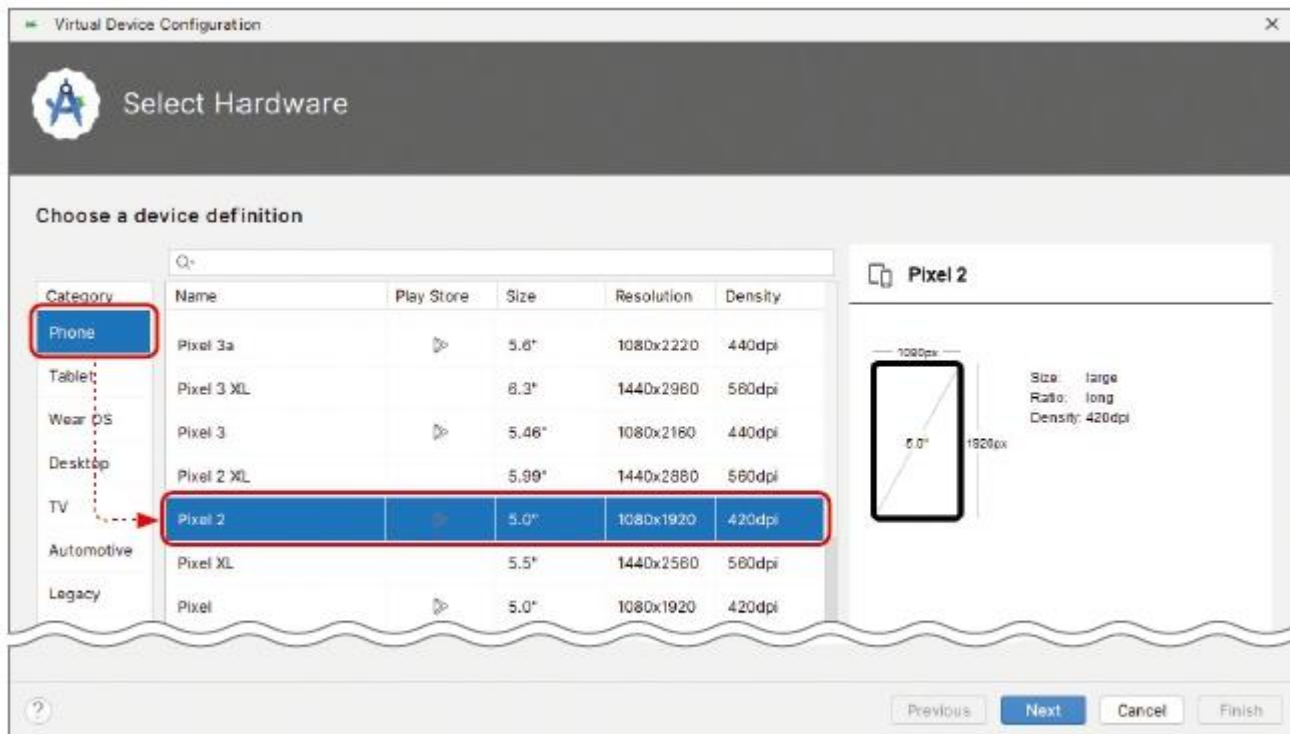


그림 1-41 AVD 생성 2

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (9) [x86 Images] 탭 :
'UpsideDownCake(34, x86_64, Android 14.0 (Google APIs))' 선택 → <Next> 클릭

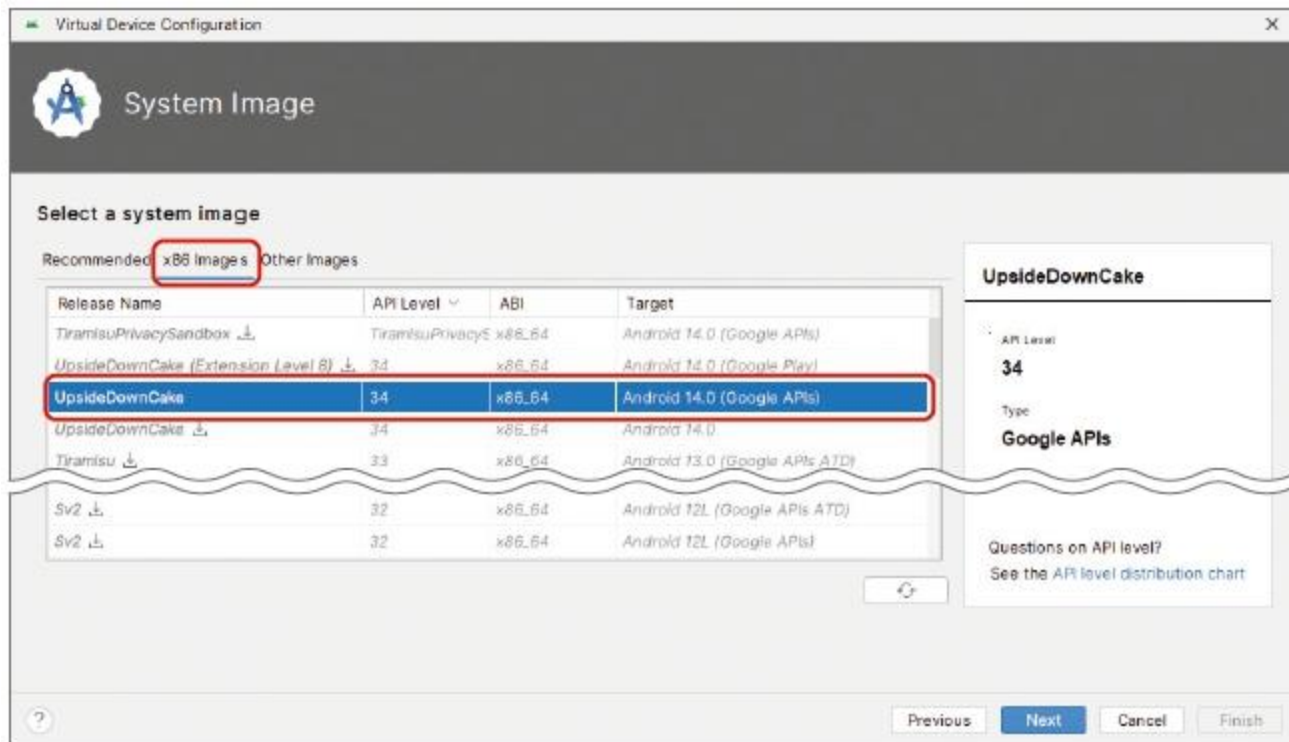


그림 1-42 AVD 생성 3

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (10) AVD Name은 'U(Pixel 2)'로 임의 설정하고 <Finish> 클릭

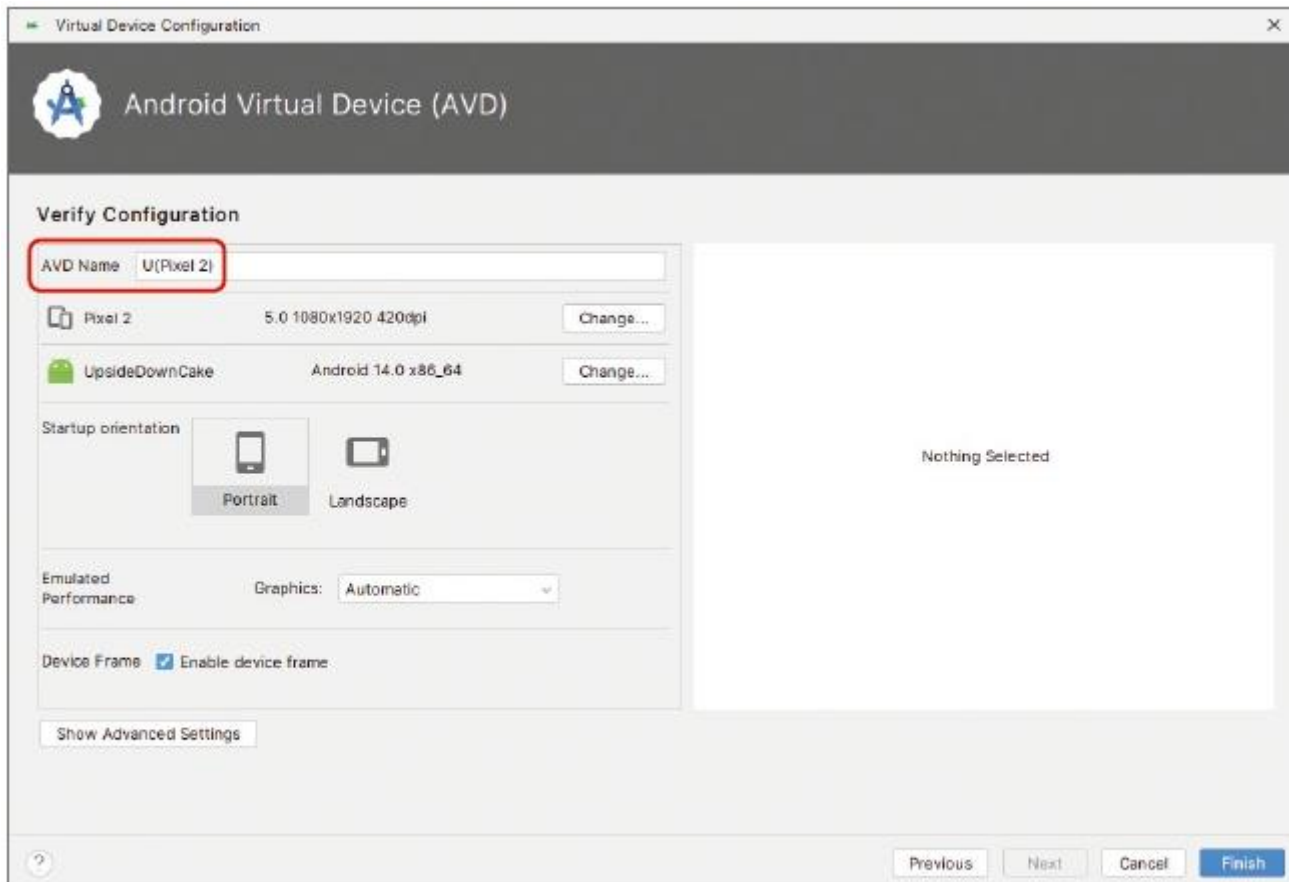


그림 1-43 AVD 생성 4

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (11) 최종적으로 완성된 AVD의 모습
 - <Launch> 버튼을 클릭하여 AVD 부팅

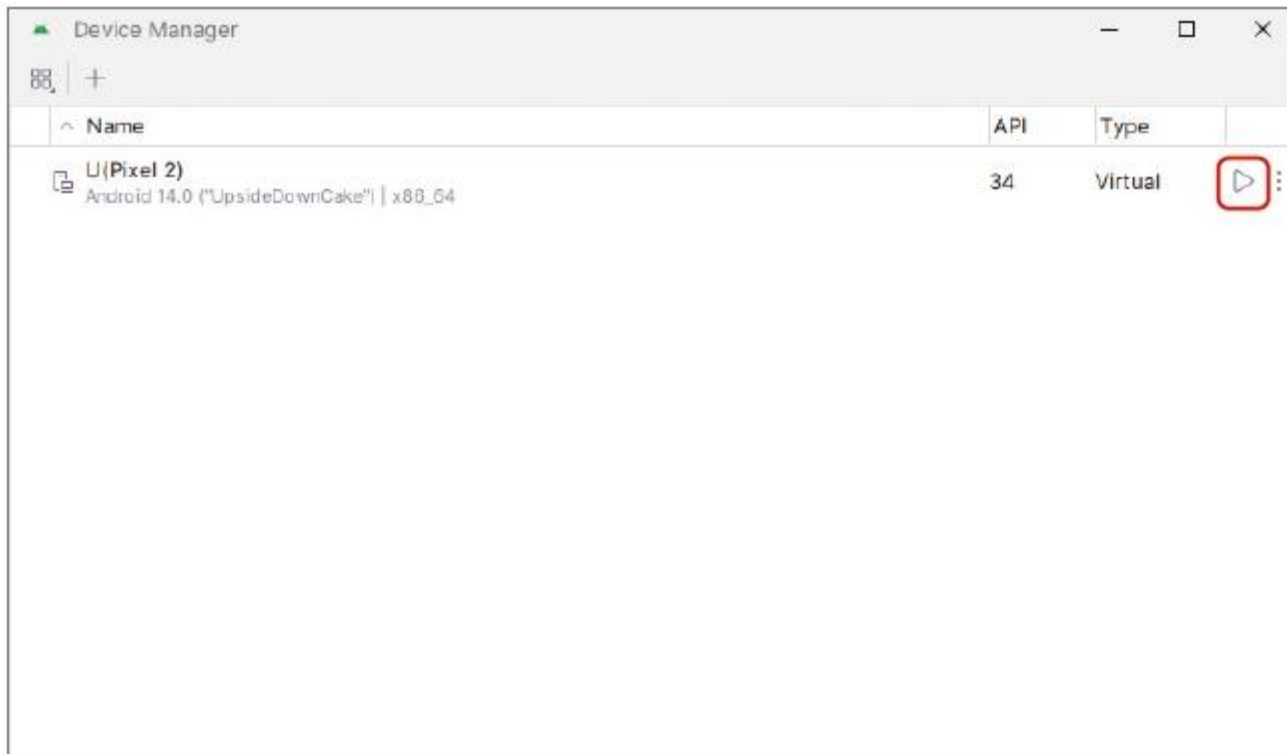


그림 1-44 생성된 AVD

4. AVD 생성

- 실습 1-6 개발 테스트를 위한 AVD 생성하기
 - (12) 잠시 후에 부팅된 AVD 확인할 수 있음



그림 1-45 AVD 가동 화면

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (13) 화면 한글 설정하기-1
 - 화면을 마우스로 클릭 - 위쪽으로 스와이프
 - 'Settings' 아이콘 클릭



그림 1-46 화면 한글 설정 1

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (14) 화면 한글 설정하기-2
 - 화면을 위로 스와이프하여 [System] 클릭 → [Languages] 클릭

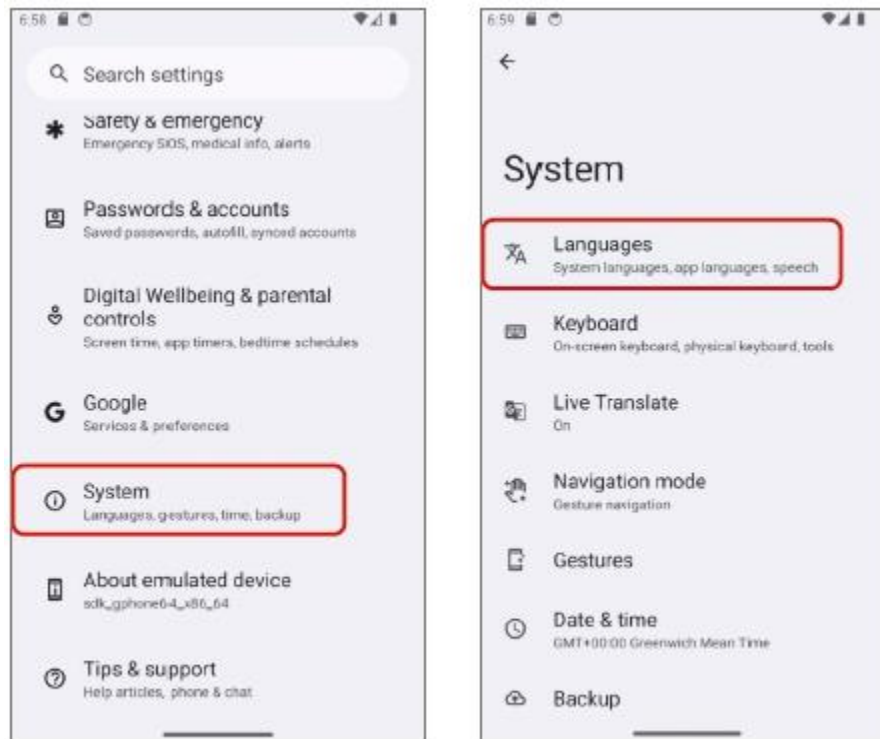


그림 1-47 화면 한글 설정 2

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (15) 화면 한글 설정-3
 - [System Languages]-[Add a language] 클릭 → 제일 아래쪽으로 스와이프 → '한국어' 클릭

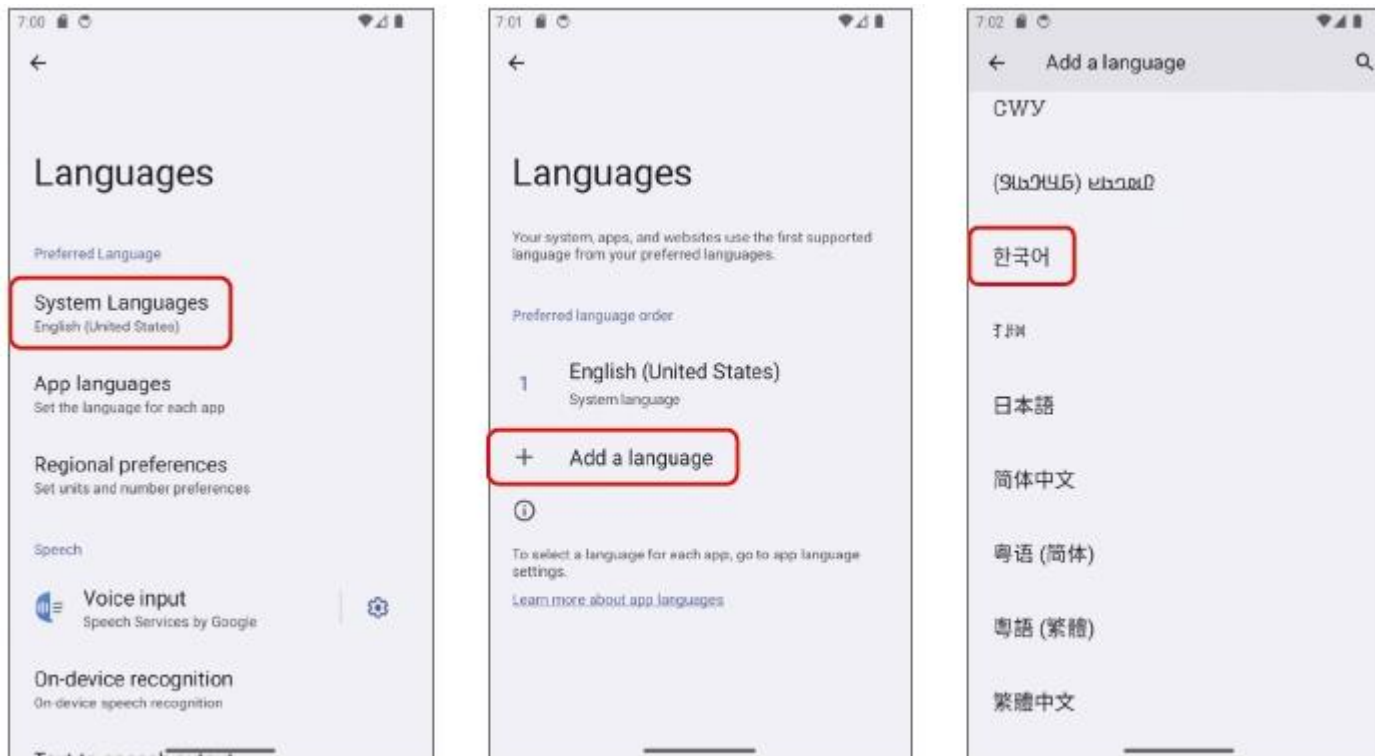


그림 1-48 화면 한글 설정 3

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (16) 화면 한글 설정-4
 - '한국어(대한민국)'을 꼭 누른 채 위로 올려서 첫 번째로 만들기
 - 메시지 창이 나오고 <Change>를 클릭하면 화면이 한글로 변경됨
 - <<를 2회 연속 클릭하여 [시스템] 화면으로 돌아가기

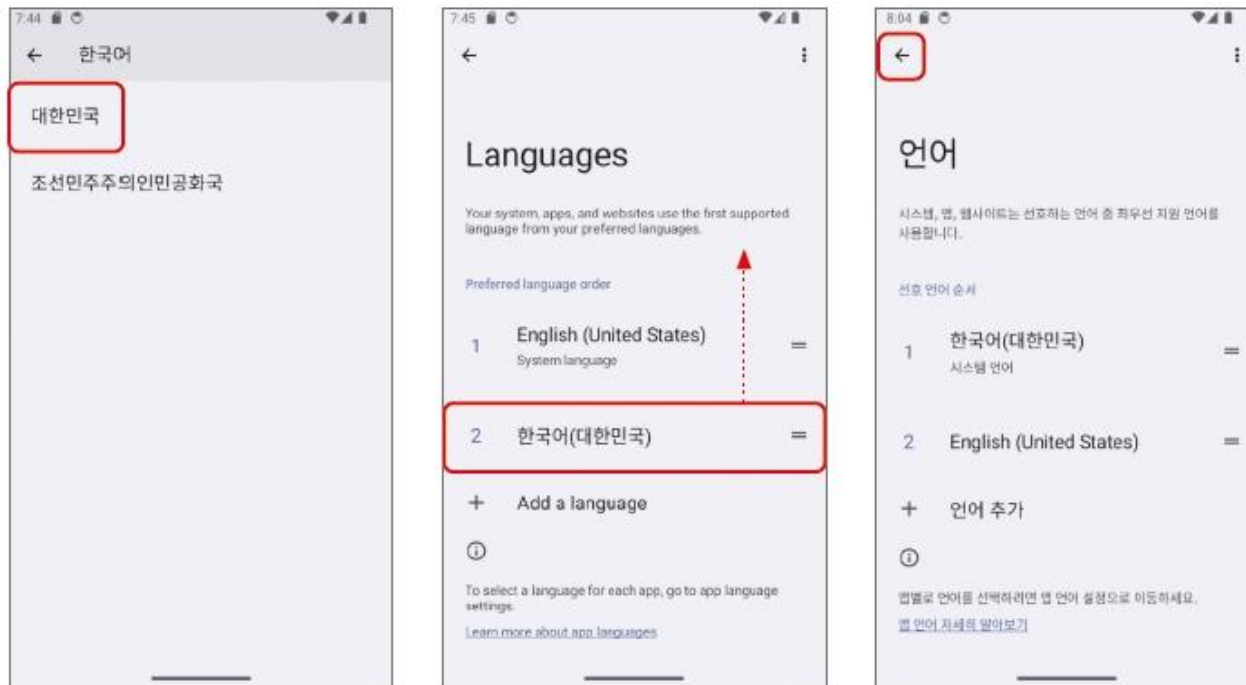


그림 1-49 화면 한글 설정 4

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

■ (17) 맞춤법 기능 끄기

- [키보드]를 클릭 → 화면을 아래로 스크롤 → [맞춤법 검사기] 클릭
→ '맞춤법 검사기 사용' 끄기 → <←> 클릭하여 앞 화면으로 돌아가기

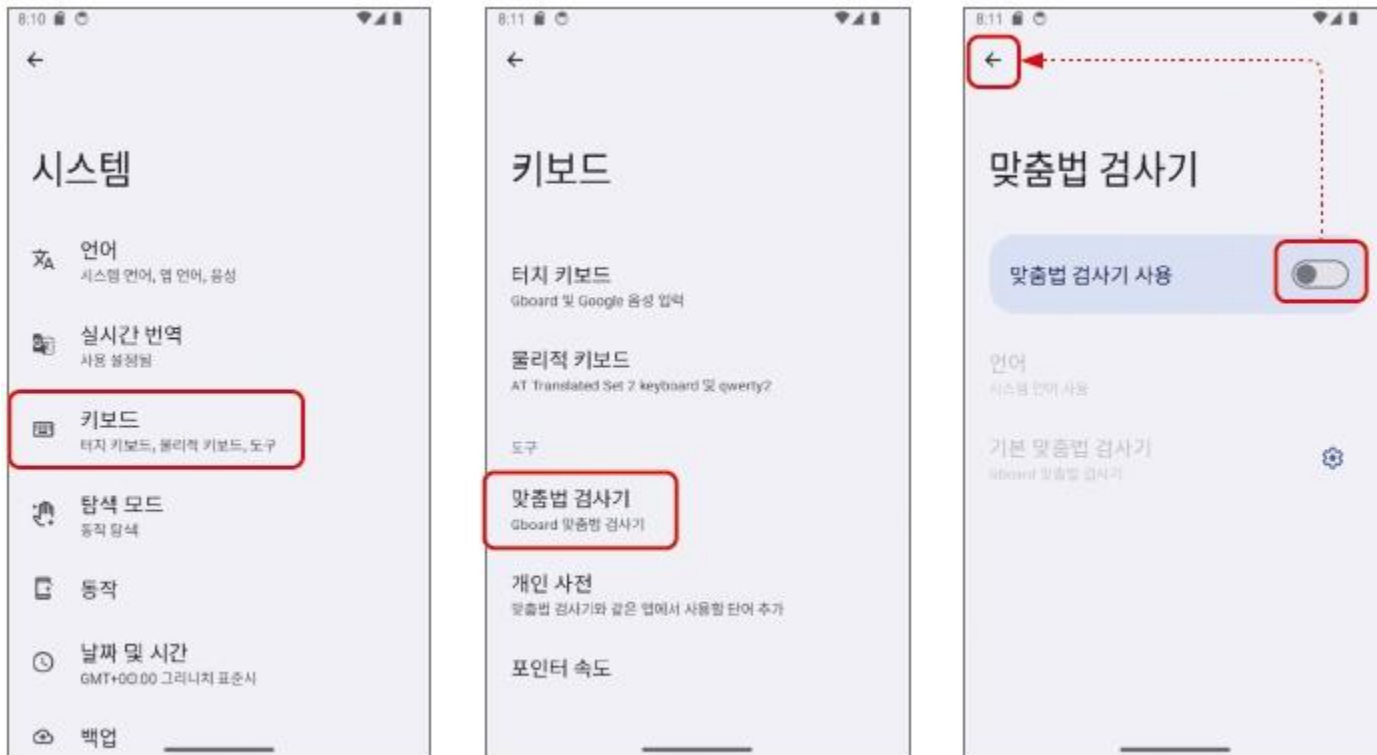


그림 1-50 맞춤법 기능 끄기

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (18) 절전 모드 시간 늘리기
 - <←>를 연속 2회 클릭 - [설정] 화면 - [디스플레이] 선택
 - [절전 모드] 클릭하여 '30분' 선택 : 30분마다 화면이 잠김

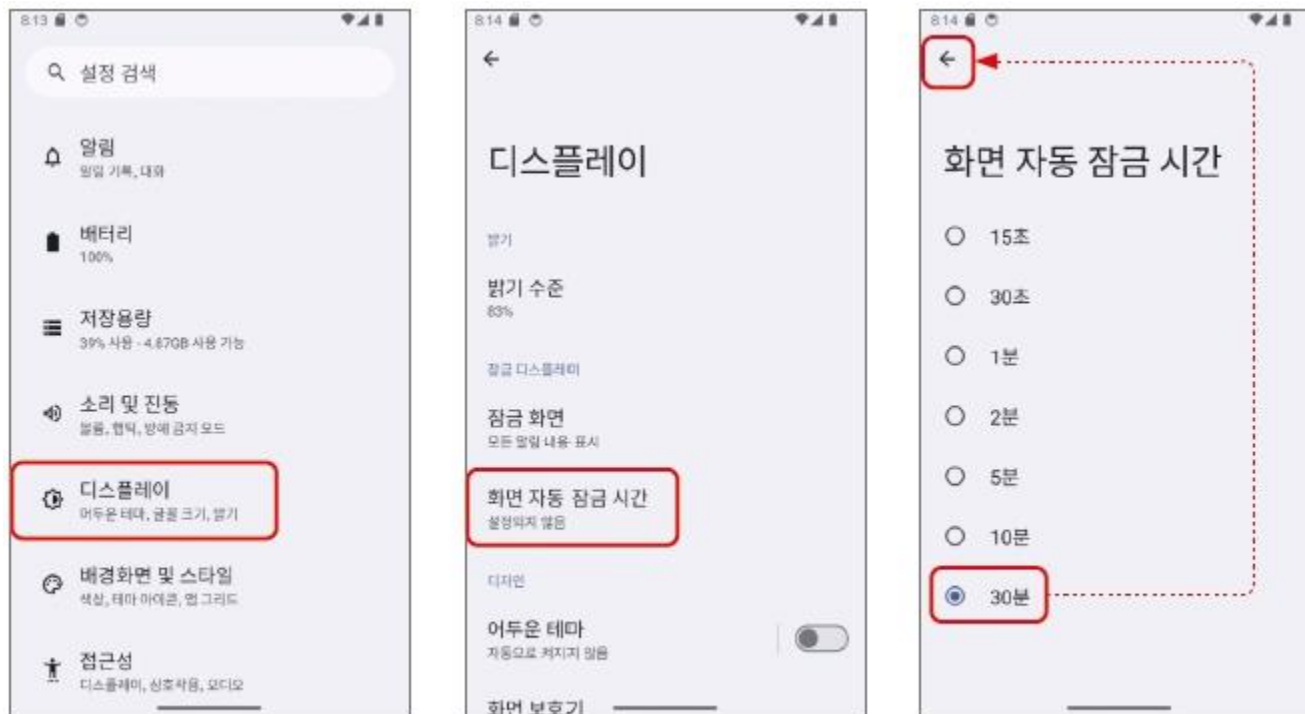


그림 1-51 화면 잠금 시간 늘리기

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (19) 스마트폰과 같은 기능 제공
 - 오른쪽 키패드의 홈 아이콘(○) 클릭 : 초기화면으로 돌아감
 - 위쪽 상태 바를 잡아서 아래로 스와이프 : 현재 알림 확인 가능
 - 오른쪽 키패드의 돌아가기 아이콘(<)을 클릭 : 닫힘

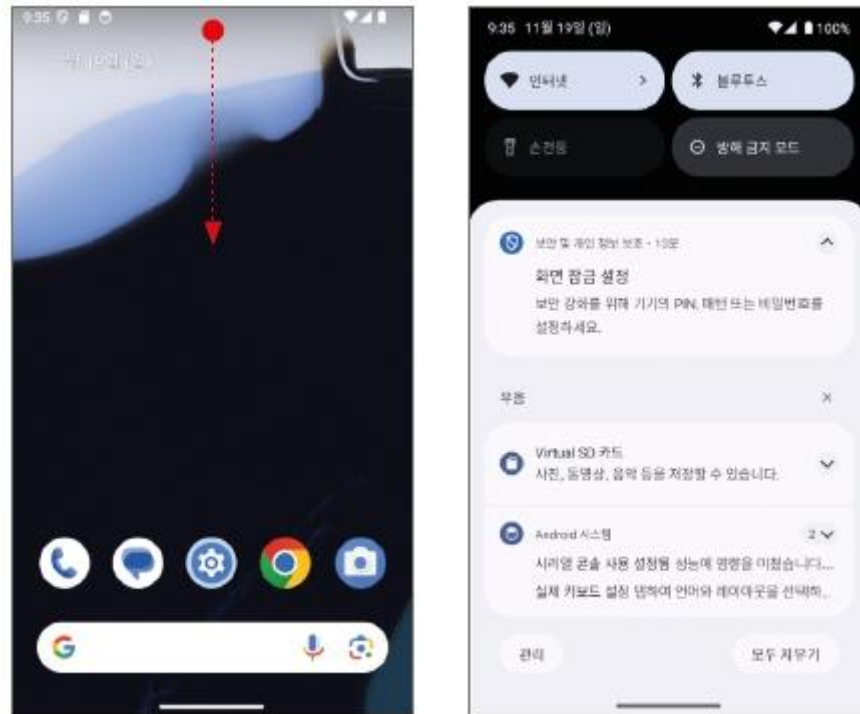


그림 1-52 알림 확인

4. AVD 생성

■ 실습 1-6 개발 테스트를 위한 AVD 생성하기

- (20) AVD 종료하기
 - 상태 바를 다시 잡아서 연속으로 끌어내린 후, 아래쪽 전원 버튼을 클릭하고 [전원 끄기]를 클릭하면 스마트폰이 종료

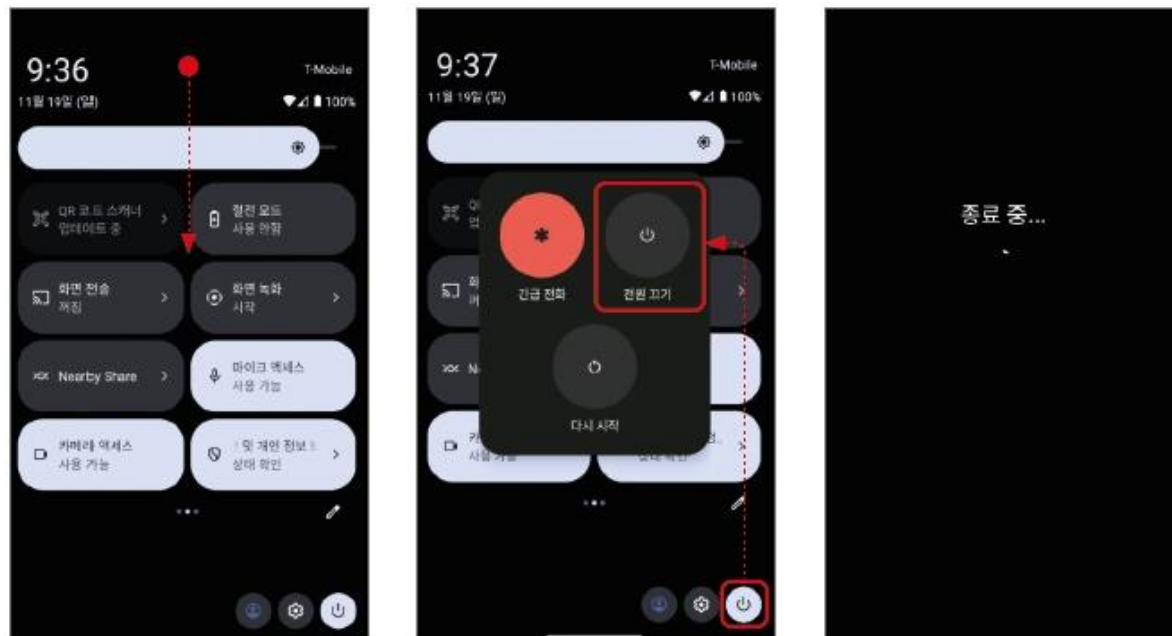


그림 1-53 AVD 종료

- (21) [Device Manager] 창을 닫고, Android Studio 초기화면도 닫기



감사합니다.
