



iOS & Android

목차

1. **iOS & Android** 역사
2. **iOS & Android** 특징
3. 모바일 **OS** 점유율 현황
4. **iOS & Android** 하드웨어: 왜 애플이 성능이 더 좋은가?
5. **iOS & Android** 개발 환경
6. 모바일 어플리케이션 마켓 점유율 현황
7. **iOS & Android** 전망

iOS & Android 역사

Android 역사

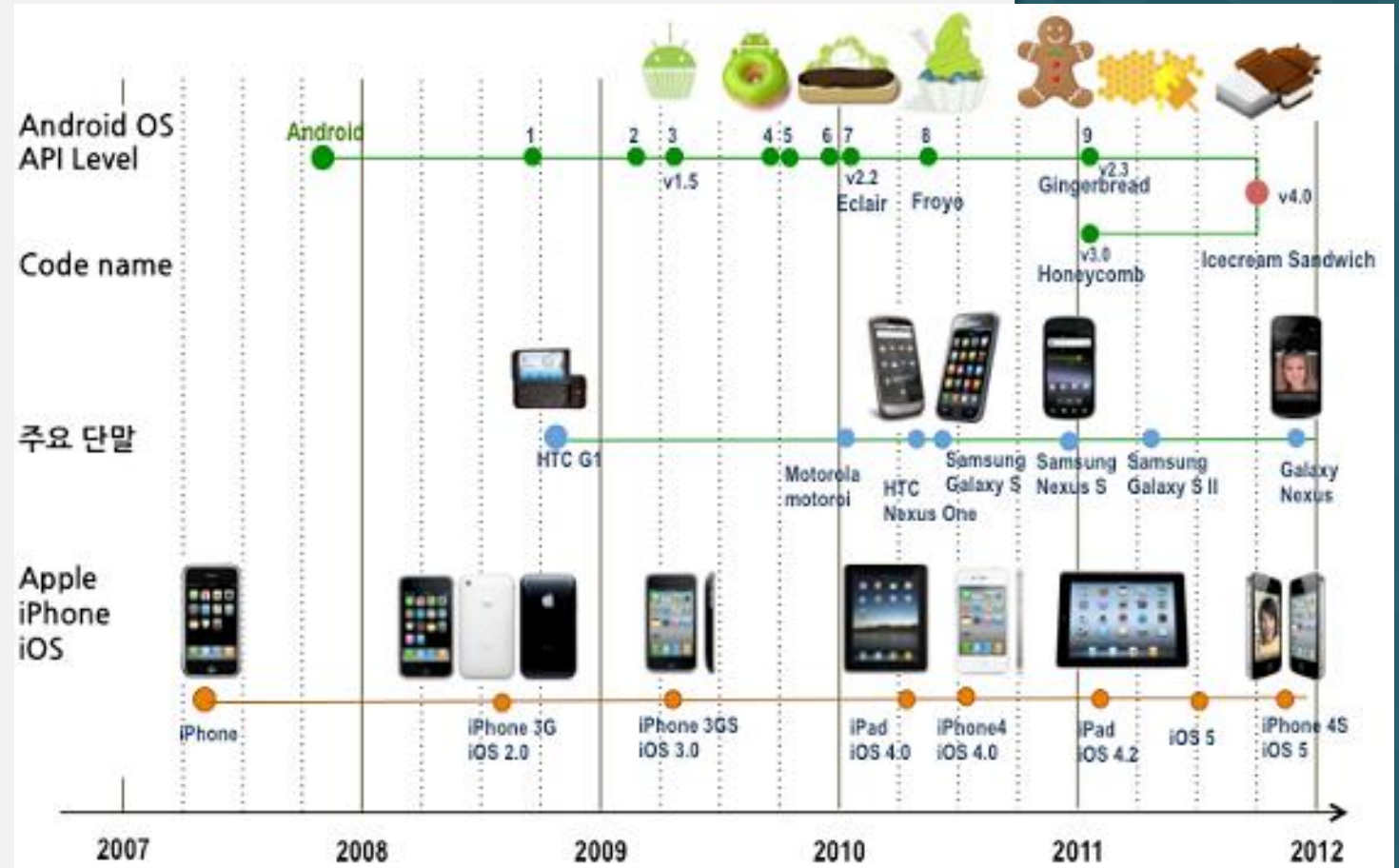
- 2007. 11
- **Linux** 커널 기반
- 안디 루빈의
Android. Inc.
- 이후 구글이
인수



<그림 1: Android 타임라인 (<https://m.blog.naver.com/alsemfsla88/221800319158>)>

iOS 역사

- 2007. 01.
- macOS 기반
- 스티븐 잡스의
Apple. Inc.
- 초기 이름은 OS X



〈그림 2: iOS 타임라인 (<https://ko.wikipedia.org/wiki/iOS>)〉

iOS & Android 특징

특징

Android

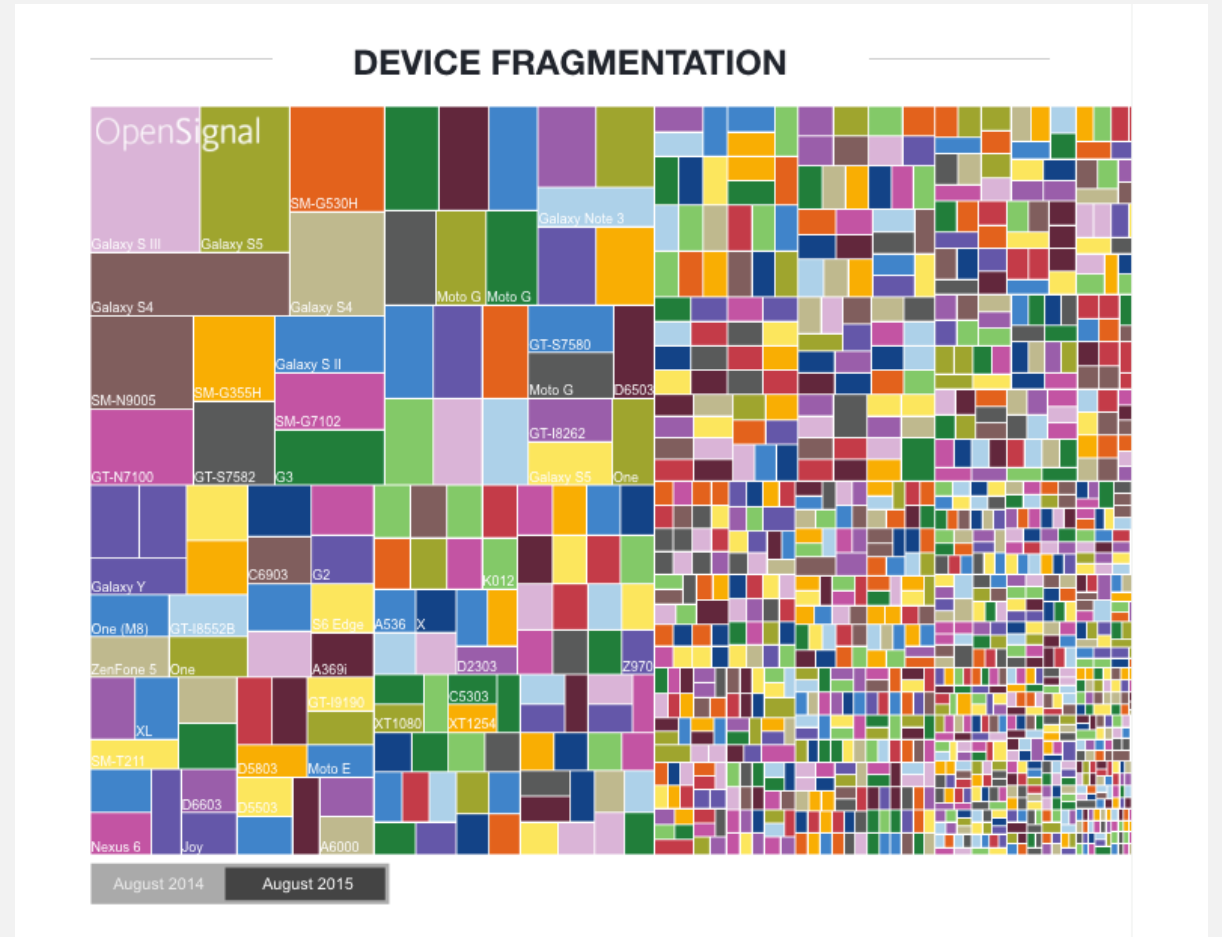
- 개방형 플랫폼
- **Google** 서비스와의 통합
- 고도의 이식성

iOS

- 안정적이고 빠른 성능
- 높은 보안 수준
- 통합된 인터페이스
- 생태계 일관성

OS 파편화란?

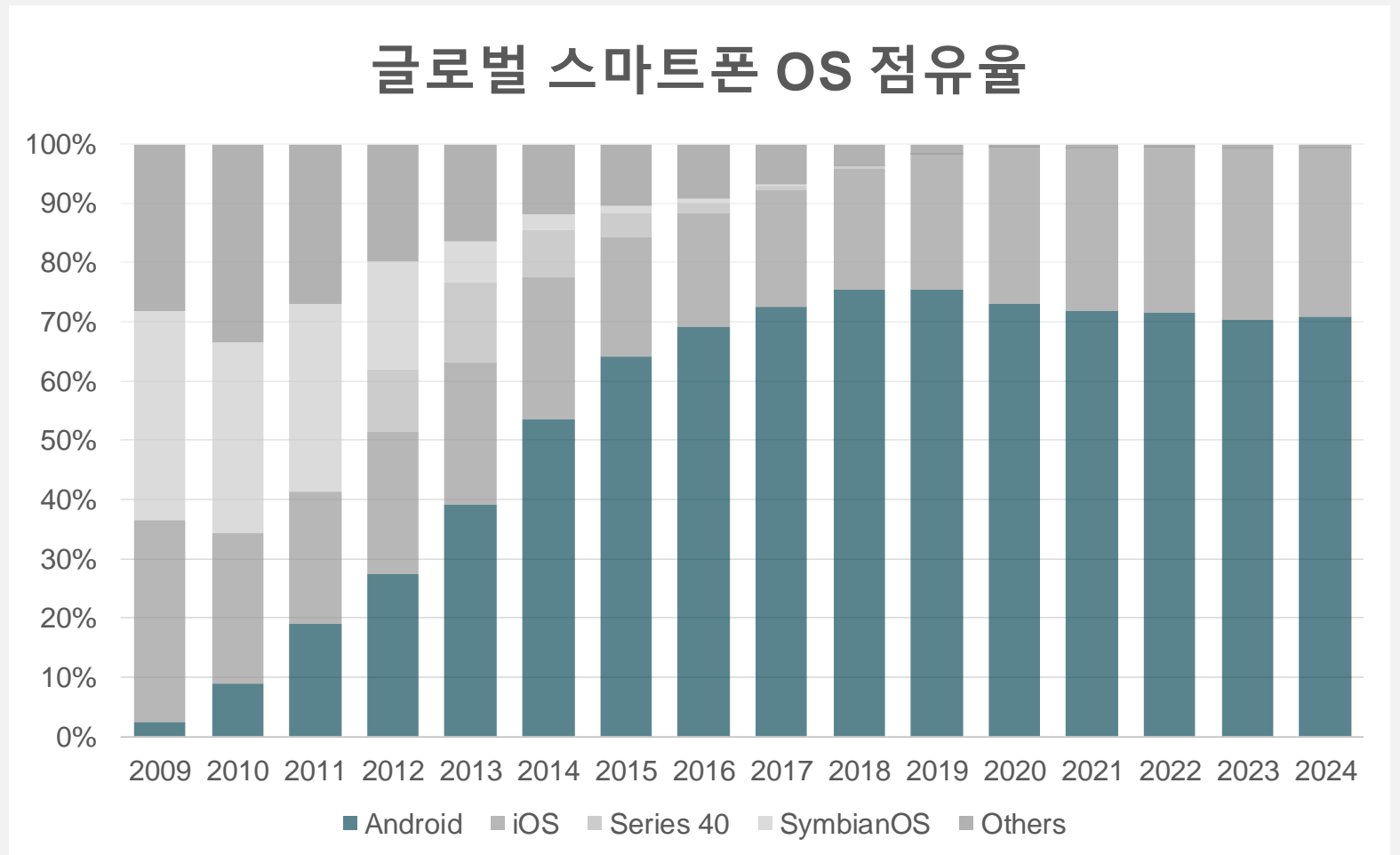
- 다양한
디바이스의 OS
호환성 문제
- 모바일 기기, 특히
안드로이드에서
발생



<그림 3: 디바이스 파편화 (<https://tech.buzzvil.com/blog/tech-blog-%EC%95%88%EB%93%9C%EB%A1%9C%EC%9D%B4%EB%93%9C-android-%ED%8C%8C%ED%8E%B8%ED%99%94-fragmentation/>)>

모바일 **os** 점유율 현황

점유율













〈그림 4: 글로벌 스마트폰 OS 점유율 (<https://gs.statcounter.com>)〉

iOS & Android 하드웨어: 왜 애플이 성능이 더 좋은가?






벤치마크 결과






• 싱글 코어 점수

Device	Score
 Samsung Galaxy S24 Ultra Qualcomm Snapdragon 8 Gen 3 @ 2.3 GHz	2169
 Samsung Galaxy S24+ Qualcomm Snapdragon 8 Gen 3 @ 2.3 GHz	2157
 Samsung Galaxy S24 Qualcomm Snapdragon 8 Gen 3 @ 2.3 GHz	2125
 Samsung Galaxy S24+ Samsung Exynos 2400 @ 2.0 GHz	2087
 Samsung Galaxy S24 Samsung Exynos 2400 @ 2.0 GHz	2004

Processor	Score
 iPhone 15 Pro Apple A17 Pro @ 3.8 GHz	2908
 iPhone 15 Pro Max Apple A17 Pro @ 3.8 GHz	2899
 iPhone 14 Pro Apple A16 Bionic @ 3.5 GHz	2568
 iPhone 15 Plus Apple A16 Bionic @ 3.5 GHz	2565
 iPhone 14 Pro Max Apple A16 Bionic @ 3.5 GHz	2563

• 멀티 코어 점수

Device	Score
 Samsung Galaxy S24 Ultra Qualcomm Snapdragon 8 Gen 3 @ 2.3 GHz	6772
 Samsung Galaxy S24+ Samsung Exynos 2400 @ 2.0 GHz	6651
 Samsung Galaxy S24+ Qualcomm Snapdragon 8 Gen 3 @ 2.3 GHz	6638
 Samsung Galaxy S24 Qualcomm Snapdragon 8 Gen 3 @ 2.3 GHz	6511
 Samsung Galaxy S24 Samsung Exynos 2400 @ 2.0 GHz	6358

Processor	Score
 iPhone 15 Pro Apple A17 Pro @ 3.8 GHz	7234
 iPhone 15 Pro Max Apple A17 Pro @ 3.8 GHz	7203
 iPhone 14 Pro Apple A16 Bionic @ 3.5 GHz	6567
 iPhone 14 Pro Max Apple A16 Bionic @ 3.5 GHz	6538
 iPhone 15 Plus Apple A16 Bionic @ 3.5 GHz	6413

<그림 5: iOS & Android 디바이스 벤치마크 결과 (<https://browser.geekbench.com>)>

성능 차이 요인

iOS

- **Objective-C, Swift** 사용
- **OS**와 하드웨어 양쪽을 직접 설계

Android

- **Java, Kotlin** 사용
- 가상머신 오버헤드
- 가비지 컬렉션
- **ARM**의 아키텍처 성능에 의존

프로그래밍 시 주의점: Out of Order Execution

Memory Ordering reordering

Reordering 종류

Load Load

```
if (x == 10) {}
if (y == 20) {}
```

Load Store

```
if (x == 10) {}
y = 10;
```

Store Load

```
y = 10;
if (x == 10) {}
```

Store Store

```
x = 10;
y = 20;
```

152

- Apple⁰ | Out of Order Execution 을

Snapdragon / Exynos 보다 더

효율적으로 수행한다.

Memory Ordering reordering

아키텍처 별 Reordering

Memory ordering in some architectures^{[7][8]}

Type	Alpha	ARMv7	PA-RISC	POWER	SPARC RMO	SPARC PSO	SPARC TSO	x86	x86 oostore	AMD64	IA-64	zSeries
Loads reordered after loads	Y	Y	Y	Y	Y				Y		Y	
Loads reordered after stores	Y	Y	Y	Y	Y				Y		Y	
Stores reordered after stores	Y	Y	Y	Y	Y	Y			Y		Y	
Stores reordered after loads	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Atomic reordered with loads	Y	Y		Y	Y						Y	
Atomic reordered with stores	Y	Y		Y	Y	Y					Y	
Dependent loads reordered	Y											
Incoherent instruction cache pipeline	Y	Y		Y	Y	Y	Y	Y	Y		Y	Y

https://en.wikipedia.org/wiki/Memory_ordering

x86은 Store Load 일 때에만 reordering
ARM은 네 경우 모두 reordering

153

C++ Concurrency in Action Study
C++ Korea

<그림 6: Reordering 규칙 (<https://www.slideshare.net/seao/c-atomic>)>

iOS & Android 개발 환경

IDE

Android 

iOS

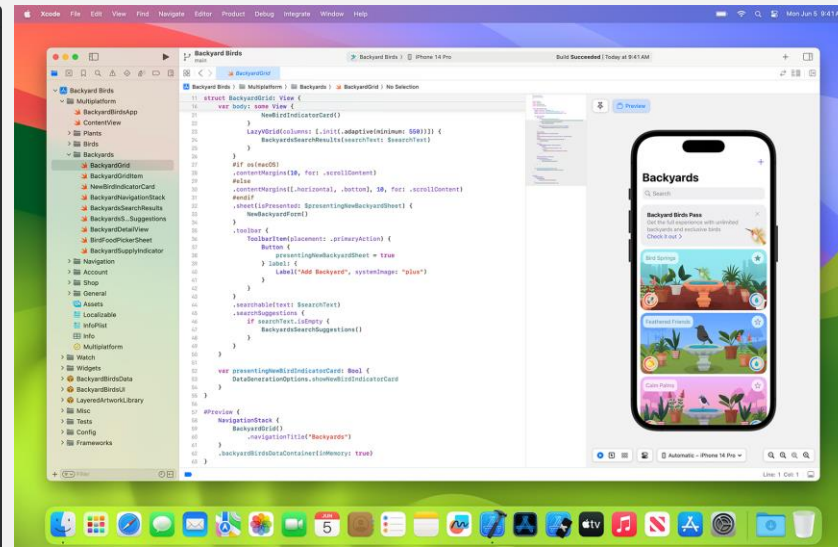
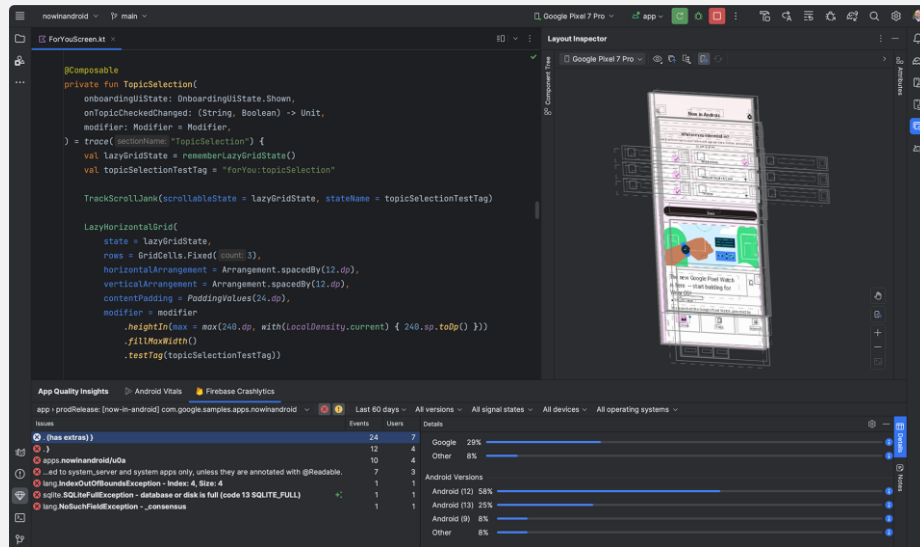


Android



iOS

IDE



Windows



macOS

macOS

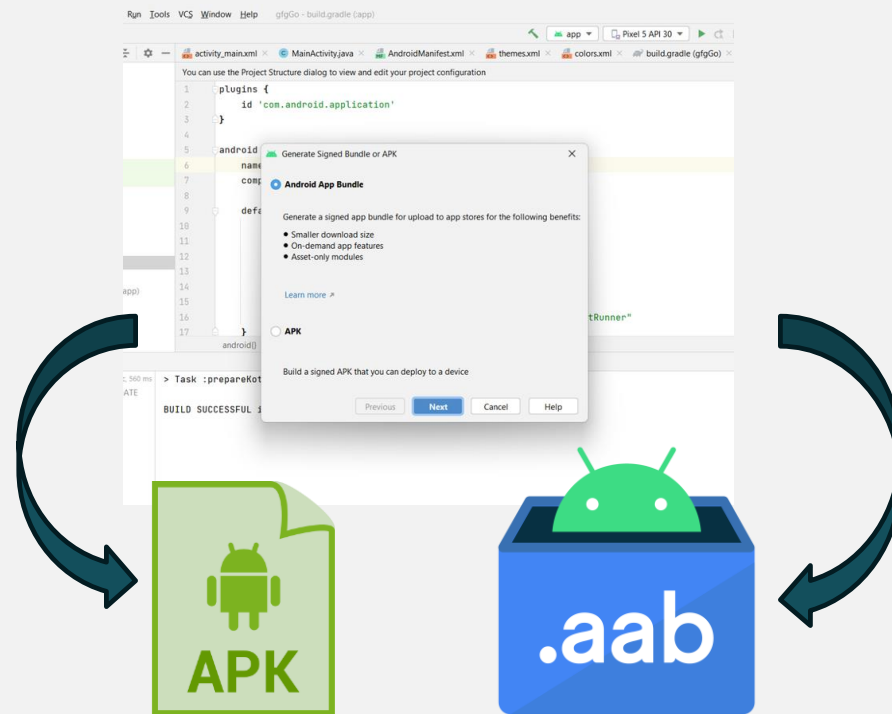
〈그림 7: Android Studio 개발 환경 (<https://developer.android.com/studio>)〉

〈그림 8: Xcode 개발 환경 (<https://developer.apple.com/xcode>)〉

Android

iOS

빌드 파일

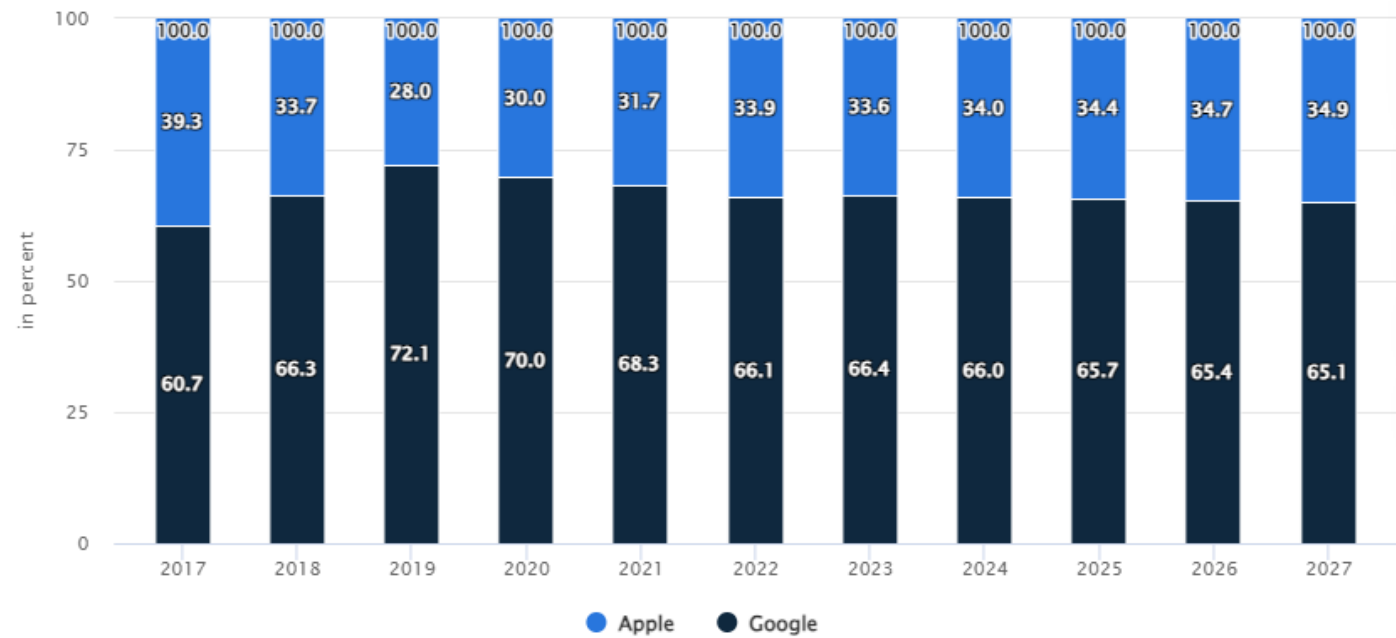


<그림 9: Android App Bundle 빌드 (<https://www.geeksforgeeks.org/how-to-generate-signed-aab-file-in-android-studio/>)>

모바일 어플리케이션 마켓 점유율 현황

스토어 별 점유율 (국내)

REVENUE SHARE BY STORE

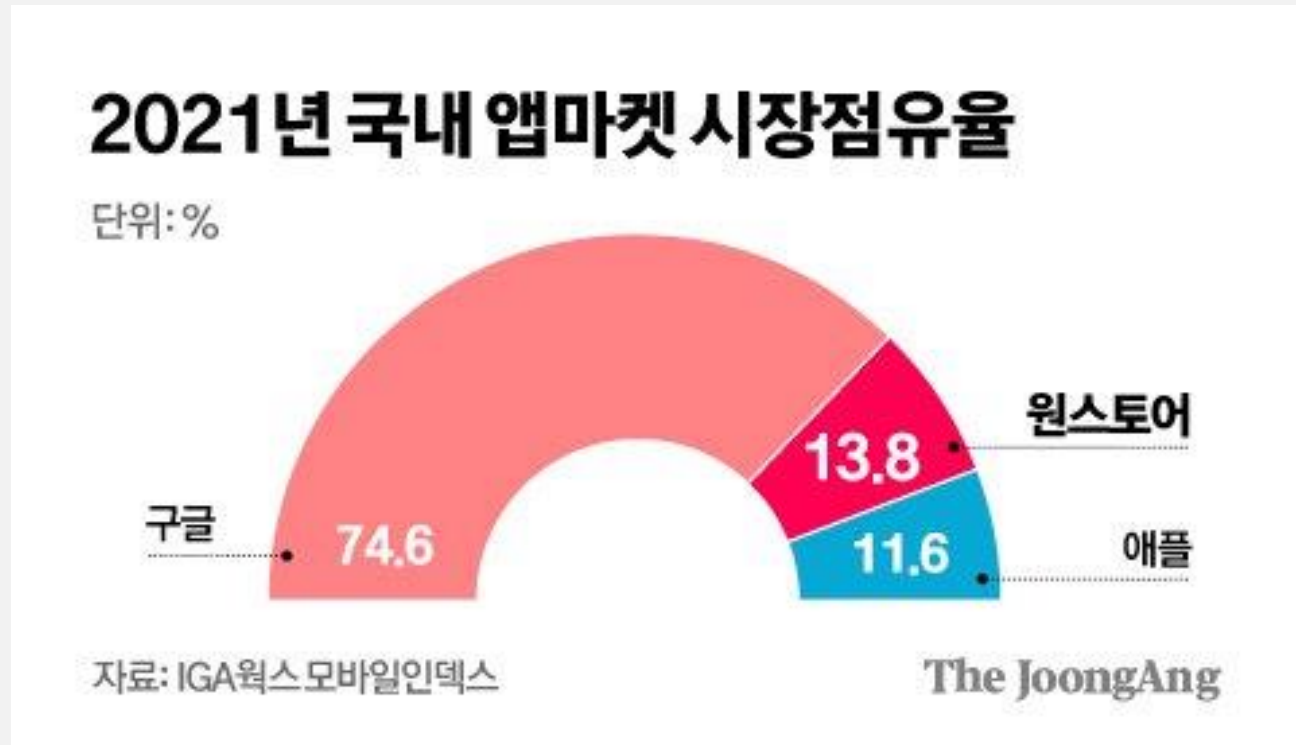


Most recent update: Mar 2024

Source: Statista Market Insights

〈그림 10: 스토어 별 수익 점유율(국내), Statista, Market : App, Region : Korea〉

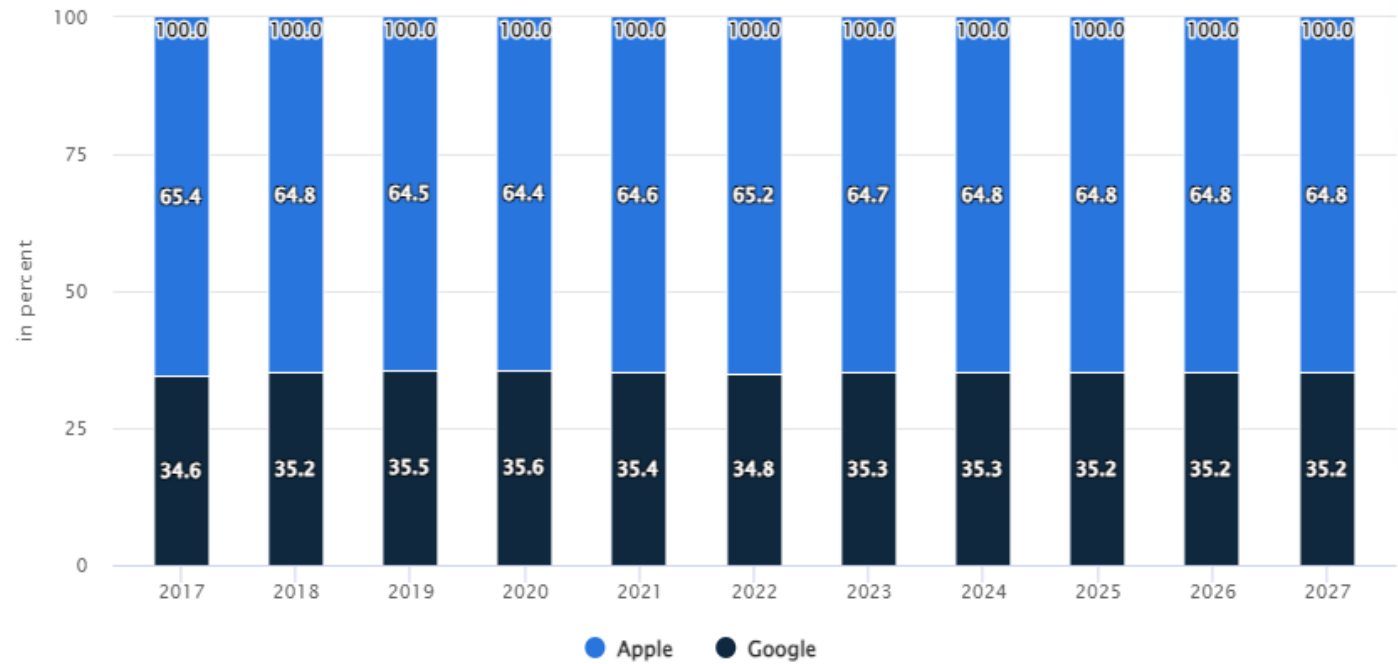
스토어 별 점유율 (국내)



〈그림 11: 2021년 국내 앱마켓 시장 점유율, IGA웍스모바일인덱스, 중앙일보 (<https://www.joongang.co.kr/article/25069830#home>)〉

스토어 별 점유율 (세계)

REVENUE SHARE BY STORE



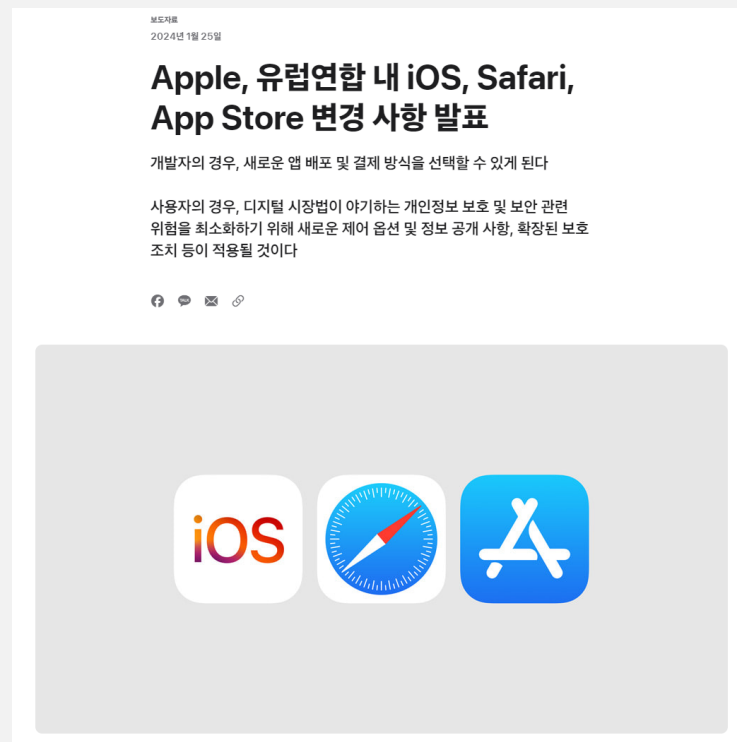
Most recent update: Mar 2024

Source: Statista Market Insights

〈그림 12: 스토어 별 수익 점유율(세계), Statista, Market : App, Region : WorldWide〉

iOS & Android 전망

전망



" 외부 앱 마켓플레이스에 **iOS** 앱을 배포할 수 있는 새로운 선택지 - 개발자가 **iOS** 앱을 외부 앱 마켓플레이스에서 제공할 수 있도록 만들어주는 새로운 **API** 및 도구를 포함한다. "

[Apple, 유럽연합 내 iOS, Safari, App Store 변경 사항 발표, 2024.01.25, Apple Newsroom]

iOS

Android

게임 개발자로 서

- 적은 기기 호환 오류

- 에뮬레이터의 적극적인 활용



Thank you!

참고 문헌

<https://dataonair.or.kr/db-tech-reference/d-lounge/expert-column/?mod=document&uid=52770>

<https://maily.so/grabnews/posts/dc711e>

<https://docs.swift.org/swift-book/documentation/the-swift-programming-language/automaticreferencecounting/>

<https://topdigital.agency/ios-vs-android-a-detailed-comparison-of-mobile-operating-systems/>

https://www.quora.com/How-is-a-bionic-chip-different-from-Snapdragon-chips?force_dialog=1&redirect_to_mweb=1

<https://www.quora.com/How-does-the-Apple-A15-Bionic-compare-to-the-Snapdragon-888>

<https://www.quora.com/Why-did-Apple-never-join-the-developers-of-Vulkan-in-favor-of-developing-their-Metal-graphics-API>

<https://debugger.medium.com/why-is-apples-m1-chip-so-fast-3262b158cba2>

<https://pmtchlessons.com/why-are-m1-macs-so-fast/>

<https://source.android.com/docs/core/runtime?hl=ko>

<https://www.intel.com/content/www/us/en/developer/articles/technical/increase-performance-of-vm-workloads-with-thp.html>

<https://developer.android.com/guide/components/fundamentals>

https://en.wikipedia.org/wiki/Android_software_development#:~:text=Android%20software%20development%20is%20the,other%20languages%20is%20also%20possible

.

[https://en.wikipedia.org/wiki/Kotlin_\(programming_language\)#cite_note-auto-6](https://en.wikipedia.org/wiki/Kotlin_(programming_language)#cite_note-auto-6)

<https://techcrunch.com/2019/05/07/kotlin-is-now-googles-preferred-language-for-android-app-development/>

[https://en.wikipedia.org/wiki/Swift_\(programming_language\)](https://en.wikipedia.org/wiki/Swift_(programming_language))

https://en.wikipedia.org/wiki/Android_SDK

https://en.wikipedia.org/wiki/Android_NDK

<https://en.wikipedia.org/wiki/Gradle>

<https://docs.unity3d.com/Manual/iphone-BuildProcess.html>

<https://developer.android.com/guide/app-bundle>

<https://www.theverge.com/2024/3/5/24091370/microsoft-windows-11-android-apps-end-of-support>

<https://www.apple.com/kr/newsroom/2024/01/apple-announces-changes-to-ios-safari-and-the-app-store-in-the-european-union/>