

Ionic Study

Day 1

오늘 할 것들

- 스터디 오리엔테이션
- Who am I? (Ice Breaking)
- Ionic 개요
- 개발환경 구성하기

스터디 오리엔테이션

- 준비물 : 노트북, 스마트폰
 - 노트북은 필수! (실습을 위해)
 - 스마트폰은 있어도 좋고 없어도 좋고...
- 스터디 일정 : 총 10주
 - 주 2회 (회당 2시간) 기준
 - 생각보다 길다...
 - 학교 수업처럼 생각하면 힘들. 같이 으쌰으쌰해서 뭔가 결과물을 만들어내는 과정이라고 생각하면 마음이 편할듯
- 중간중간 파티타임도 가져봅시다.
 - 공부만 하면 재미없으니깐...

Who am I?

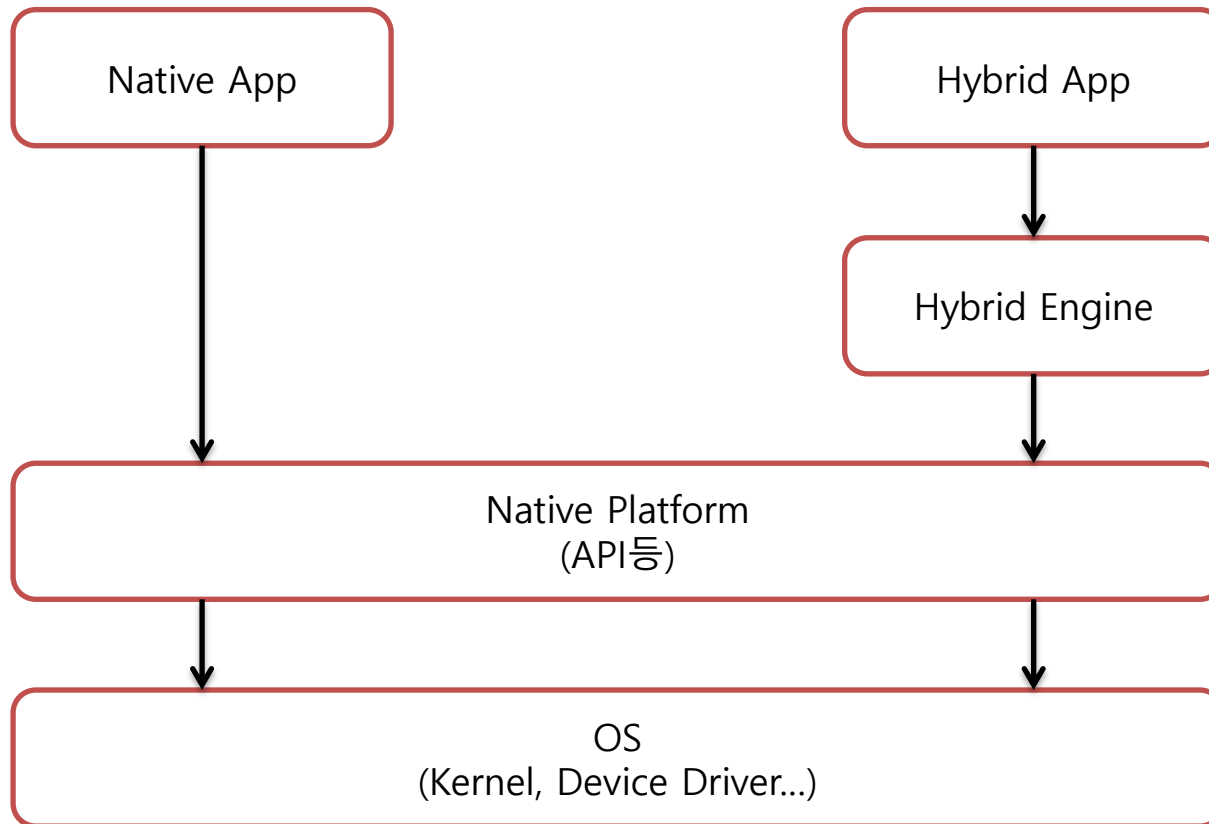
- 10주간 같이 고생할 멤버들에게 나를 소개해봅시다.



Ionic 개요

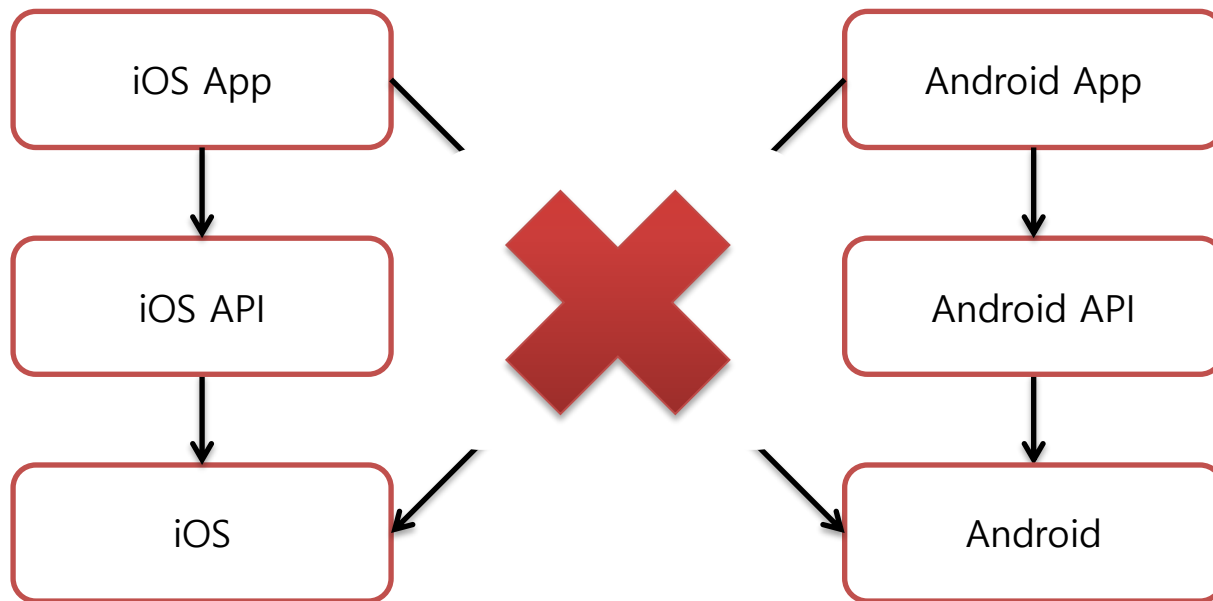
- 어떤 개발 플랫폼을 선정해야 할까?
 - 모든 개발자의 고민 사항.
 - 결과물의 특성에 따라 선택 하는 것이 가장 합리적.
- 네이티브? 하이브리드?
 - 네이티브 앱 : 각 스마트폰의 OS(iOS, Android등)가 지원하는 고유의 언어로 제작하며, 제한된 OS 환경(단일 OS)에서 구동되는 앱.
 - iOS : Objective-C, Swift / Android : Java, Kotlin
 - OS 종속적. (하나의 플랫폼을 타겟으로 만든 앱은 일반적인 방법으론 다른 플랫폼으로 포팅 불가.)
 - 가장 '완벽' 하게 앱을 만들어낼 수 있다. 하지만 그만큼 비용이 많이 들어감...
 - 하이브리드 앱 : 앱과 네이티브 플랫폼 사이에 하이브리드 엔진 존재.
 - OS 독립적. (네이티브 플랫폼이 하이브리드 엔진을 지원하면 OK)
 - 제한된 성능... (네이티브 플랫폼 ↔ 하이브리드 엔진간 상호작용이 속도저하의 주 원인)

네이티브 vs 하이브리드



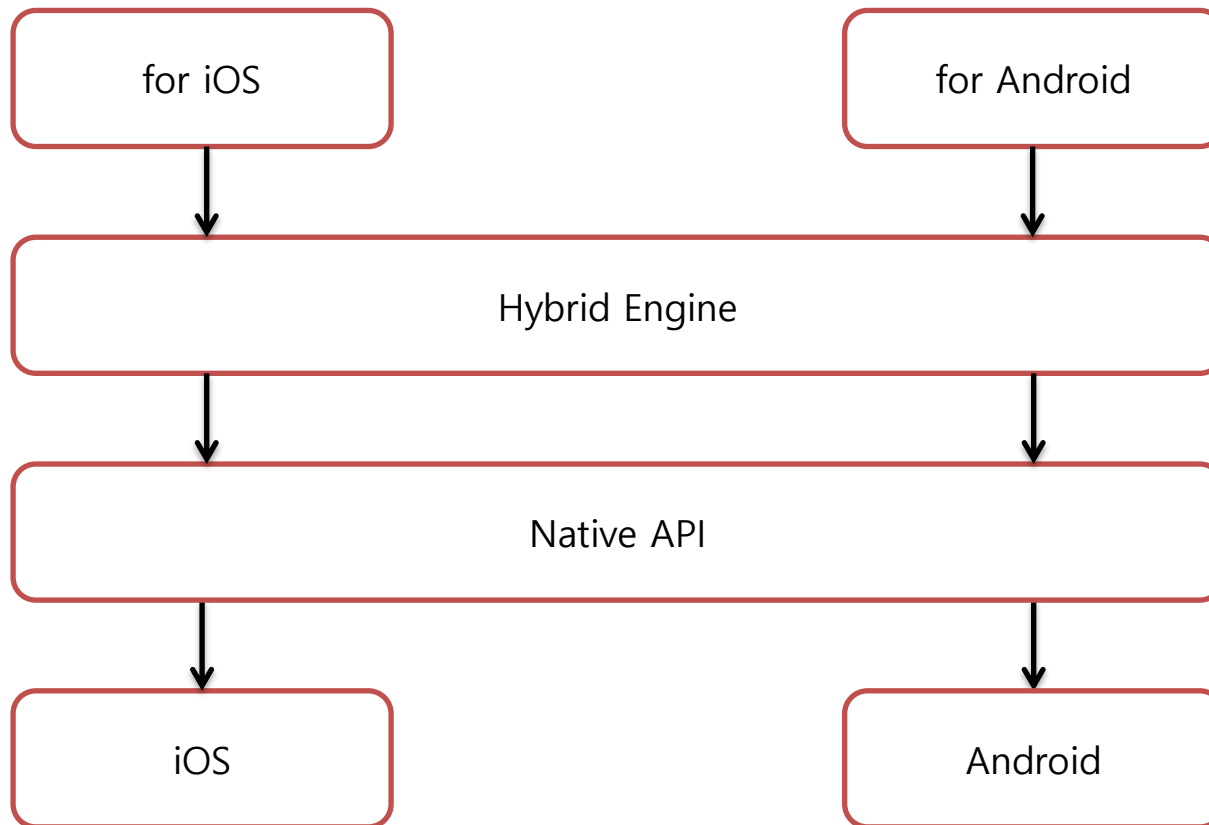
네이티브 vs 하이브리드

- 이기종간 상호 포팅 불가.
 - 해야한다면 크로스 컴파일러를 지원하는 개발환경이 필요. (ex: 자마린)



네이티브 vs 하이브리드

- OS 독립적.
 - OS가 하이브리드 엔진을 지원하게 만들면 됨 (ex: cordova)



하이브리드 앱

- 장점
 - 하이브리드 엔진이 거의 다 해먹음
 - 입력을 받아서 앱으로 전달
 - UI를 네이티브 플랫폼에서 렌더링
- 단점
 - 네이티브 앱보다는 느려...
 - 반응속도가 생명인 앱(게임?)에서는 아마 힘들수도 있음.
 - UI도 제한적임... (여러 플랫폼을 지원해야 하므로 특정 네이티브 플랫폼의 UI 특성만을 반영하기는 매우 어려움)

Ionic 소개

- 하이브리드 앱 개발 플랫폼
 - 라이브러리만을 지칭하기보단, 전체 개발 프로세스를 포함하는 플랫폼의 성격이 강함.
- Angular (AngularJS가 아님) 기반
 - Typescript-based
 - Not same Ionic 1. but, it similar to Ionic 2
 - Ionic 2 != Ionic 3, Ionic 2 == Ionic 3 (not ===)

Ionic 장, 단점

- 장점

- 하이브리드 엔진이 거의 다 해먹음
 - 입력을 받아서 앱으로 전달
 - UI를 네이티브 플랫폼에서 렌더링
- 높은 코딩 생산성
- 깔끔한 빌드 시스템
- Ionic 1에 비해 더 향상된 성능 (자체 최적화 Great!)

- 단점

- 네이티브 앱보다는 느려...
 - 반응속도가 생명인 앱(게임?)에서는 아마 힘들수도 있음.
- UI도 제한적임... (여러 플랫폼을 지원해야 하므로 특정 네이티브 플랫폼의 UI 특성만을 반영하기는 매우 어려움)
- 적은 레퍼런스 (Critical! => 구글링으로 어렵게 레퍼런스를 찾아도 이게 100% 돌아갈거라는 보장을 해선 안됨)
- 한국어 자료 전무 (일하다가 트러블 터지면 어따 하소연해도 받아줄 곳이 음...)
- 러닝 커브에 대한 두려움.
 - Angular 고유 특성. 초반에는 무난하게 배우다가 갑자기 어느 시점부터 탁 막혀옴... -_-;;

개발환경 구성하기

- Ionic를 사용하기 위해 필요한 것들:
 - Node, node, node.js, Node.js!
 - Ionic, cordova 라이브러리
 - 에디터
- 그 외 있어도 되고 없어도 되고.
 - iOS, Android Native Hardware
 - 에뮬레이터로 대체할 수 있음 (단, 환경이 제한적임: 하드웨어 의존적인 기능은 에뮬레이터로 사용하기 불가능하거나 매우 곤란할 수 있음)
 - 블루스택, 지니모션
 - iOS Simulator, Android Virtual Device (AVD)

개발환경 구성하기

- Node.js 설치
 - <https://nodejs.org>
 - 입맛에 맞는 버전 선택.
 - LTS (Longterm Supported) : 가장 안정적인 버전. 안정성과 보안성에 초점을 둠.
 - Current (Stable) : 추가기능 개발 및 기존 API의 기능 개선에 우선하는 버전.
 - LTS 버전 기준으로 진행.

개발환경 구성하기

- Node.js 설치



The screenshot shows the Node.js website's download page. At the top, the Node.js logo is centered, with a navigation bar containing links: HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, NEWS, and FOUNDATION. Below the navigation bar, a paragraph describes Node.js as a JavaScript runtime built on Chrome's V8 JavaScript engine, highlighting its event-driven, non-blocking I/O model and the npm package ecosystem. A green banner announces 'Node 4.x is End Of Life – April Release Updates'. The main section is titled 'Download for macOS (x64)' and features two green buttons: '8.11.1 LTS Recommended For Most Users' (highlighted with a red box) and '10.0.0 Current Latest Features'. Below these buttons are links for 'Other Downloads | Changelog | API Docs'. A link to 'Or have a look at the LTS schedule.' is provided, followed by a sign-up prompt for 'Node.js Everywhere, the official Node.js Weekly Newsletter.'

node

HOME | ABOUT | DOWNLOADS | DOCS | GET INVOLVED | SECURITY | NEWS | FOUNDATION

Node.js® is a JavaScript runtime built on [Chrome's V8 JavaScript engine](#). Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, [npm](#), is the largest ecosystem of open source libraries in the world.

Node 4.x is End Of Life – April Release Updates

Download for macOS (x64)

8.11.1 LTS
Recommended For Most Users

10.0.0 Current
Latest Features

[Other Downloads](#) | [Changelog](#) | [API Docs](#) [Other Downloads](#) | [Changelog](#) | [API Docs](#)

Or have a look at the [LTS schedule](#).

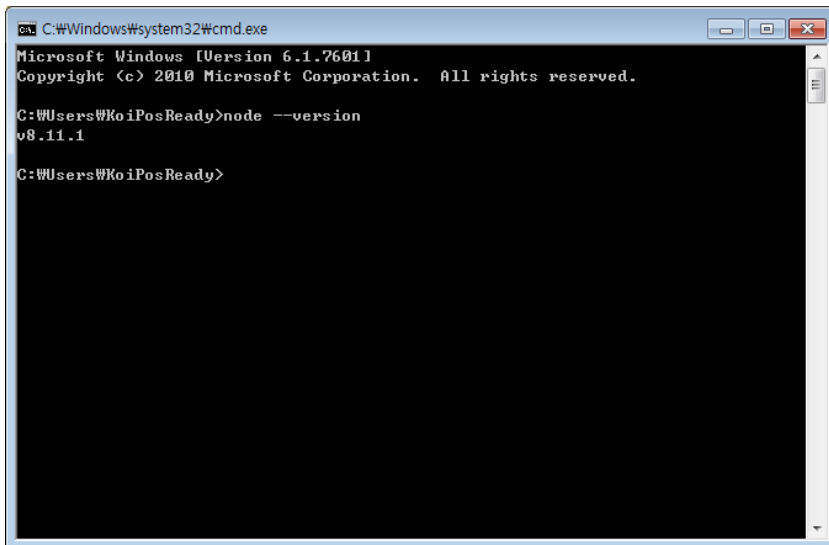
Sign up for [Node.js Everywhere](#), the official Node.js Weekly Newsletter.

개발환경 구성하기

- Node.js 설치
 - 설치가 잘 되었는지 확인해보기.
 - 터미널 (명령 프롬프트) 실행
 - Win : 시작 > 실행 > cmd 입력 후 엔터 (관리자 권한으로 실행해야함.)
 - Mac : Finder > 응용 프로그램 > 유틸리티 > 터미널

개발환경 구성하기

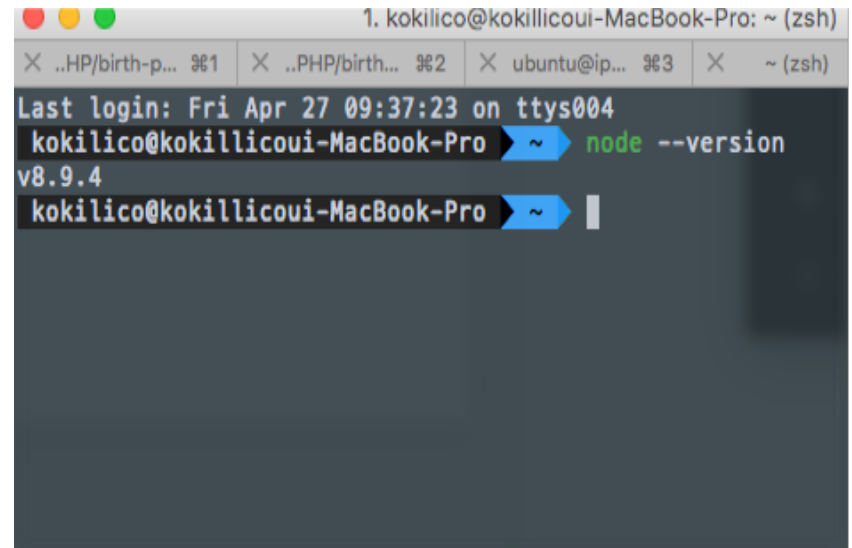
- Node.js 설치
 - node --version
 - 설치된 Node.js의 버전을 확인하는 명령어



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601.1
Copyright (c) 2010 Microsoft Corporation. All rights reserved.

C:\Users\WkoiPosReady>node --version
v8.11.1

C:\Users\WkoiPosReady>
```



```
1. kokilico@kokillicoui-MacBook-Pro: ~ (zsh)
X ..HP/birth-p... %1 X ..PHP/birth... %2 X ubuntu@ip... %3 X ~ (zsh)

Last login: Fri Apr 27 09:37:23 on ttys004
kokilico@kokillicoui-MacBook-Pro ~ % node --version
v8.9.4
kokilico@kokillicoui-MacBook-Pro ~ %
```


개발환경 구성하기

- NPM – Node.js의 기본 패키지 관리자
 - Node.js에서 사용 가능한 모듈들을 패키지화시켜 모아놓은 것.
 - 일일이 개발하지 않아도 이미 누군가가 만들어둔 모듈을 다운로드 받는것으로 대신할 수 있음.
 - 직접 빌드, 의존성 및 버전관리로부터 해방



개발환경 구성하기

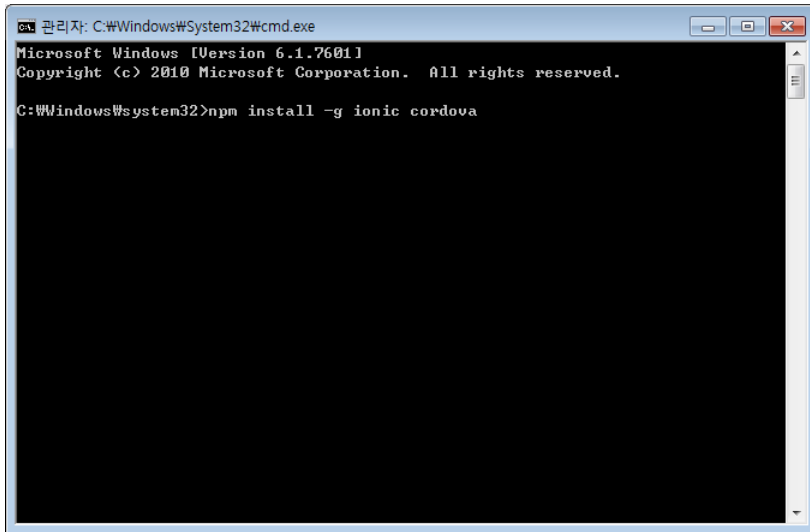
- Ionic 3 CLI 설치
 - Command Line Interface : 실행, 테스트, 배포등을 위한 명령어 인터페이스
 - 명령 프롬프트 상에서 다음과 같은 명령어 입력 후 실행

sudo npm install -g ionic cordova
(mac)

* Win에서는 sudo를 빼고 입력

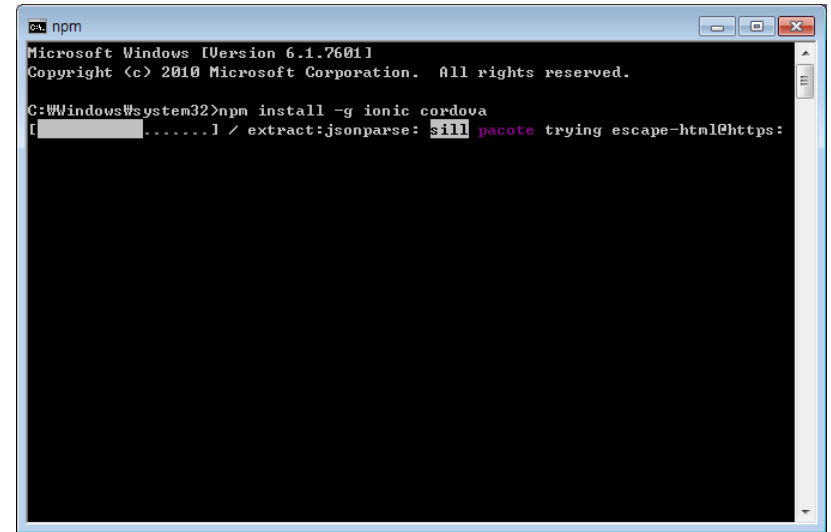
개발환경 구성하기

- Ionic 3 CLI 설치
 - 시간이 쪼끔 걸릴 수 있으니 인내심을 가지고 기다려봅시다.



```
관리자: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2010 Microsoft Corporation. All rights reserved.

C:\Windows\system32>npm install -g ionic cordova
```

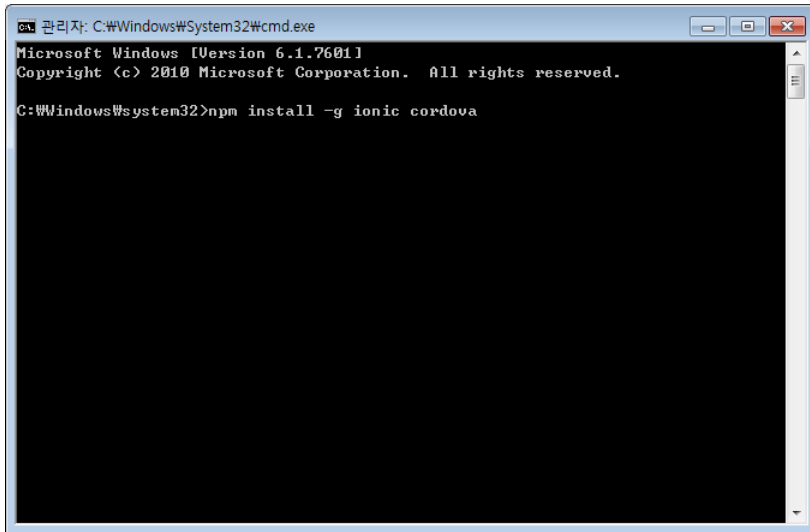


```
npm
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2010 Microsoft Corporation. All rights reserved.

C:\Windows\system32>npm install -g ionic cordova
[.....] / extract:jsonparse: sill pacote trying escape-html@https:
```

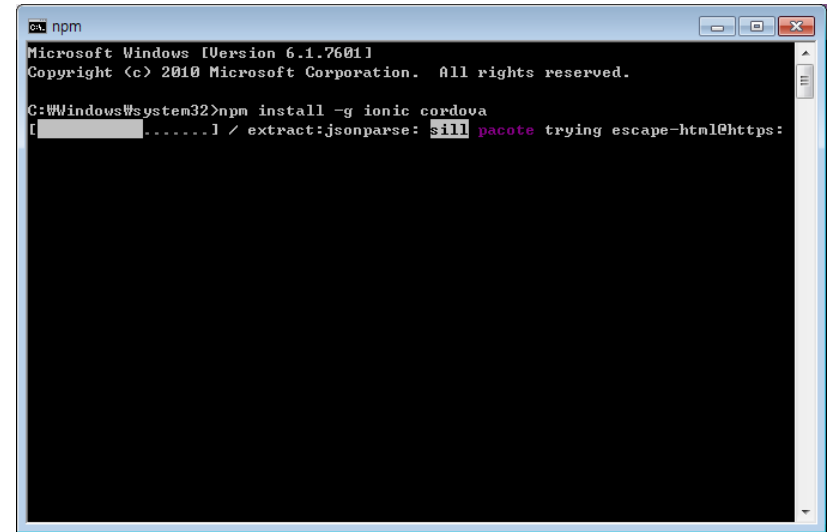
개발환경 구성하기

- Ionic 3 CLI 설치
 - 시간이 쪼끔 걸릴 수 있으니 인내심을 가지고 기다려봅시다.



```
관리자: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2010 Microsoft Corporation. All rights reserved.

C:\Windows\system32>npm install -g ionic cordova
```

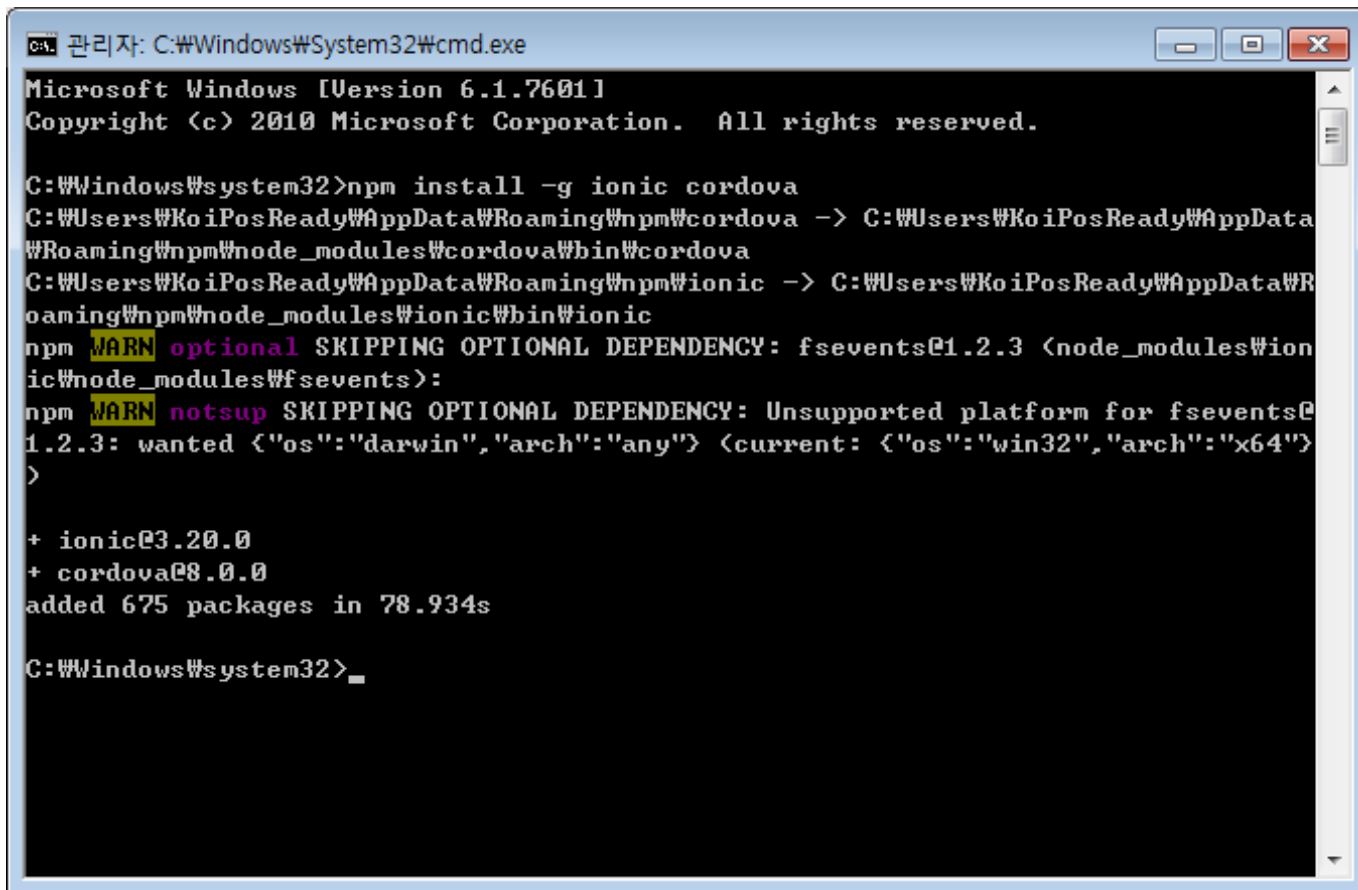


```
npm
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2010 Microsoft Corporation. All rights reserved.

C:\Windows\system32>npm install -g ionic cordova
[.....] / extract:jsonparse: sill pacote trying escape-html@https:
```

개발환경 구성하기

- Ionic 3 CLI 설치



```
C:\> 관리자: C:\Windows\System32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2010 Microsoft Corporation. All rights reserved.

C:\Windows\system32>npm install -g ionic cordova
C:\Users\KoiPosReady\AppData\Roaming\npm\cordova -> C:\Users\KoiPosReady\AppData\Roaming\npm\node_modules\cordova\bin\cordova
C:\Users\KoiPosReady\AppData\Roaming\npm\ionic -> C:\Users\KoiPosReady\AppData\Roaming\npm\node_modules\ionic\bin\ionic
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.3 (node_modules\ionic\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.3: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

+ ionic@3.20.0
+ cordova@8.0.0
added 675 packages in 78.934s

C:\Windows\system32>
```

개발환경 구성하기

- Ionic 3 CLI 설치 확인
 - `ionic --version`

```
C:\Windows\system32>ionic --version
3.20.0

C:\Windows\system32>_
```

```
kokilico@kokillicoui-MacBook-Pro ~ ➤ ionic --version
3.19.0
kokilico@kokillicoui-MacBook-Pro ~ ➤
```

개발환경 구성하기

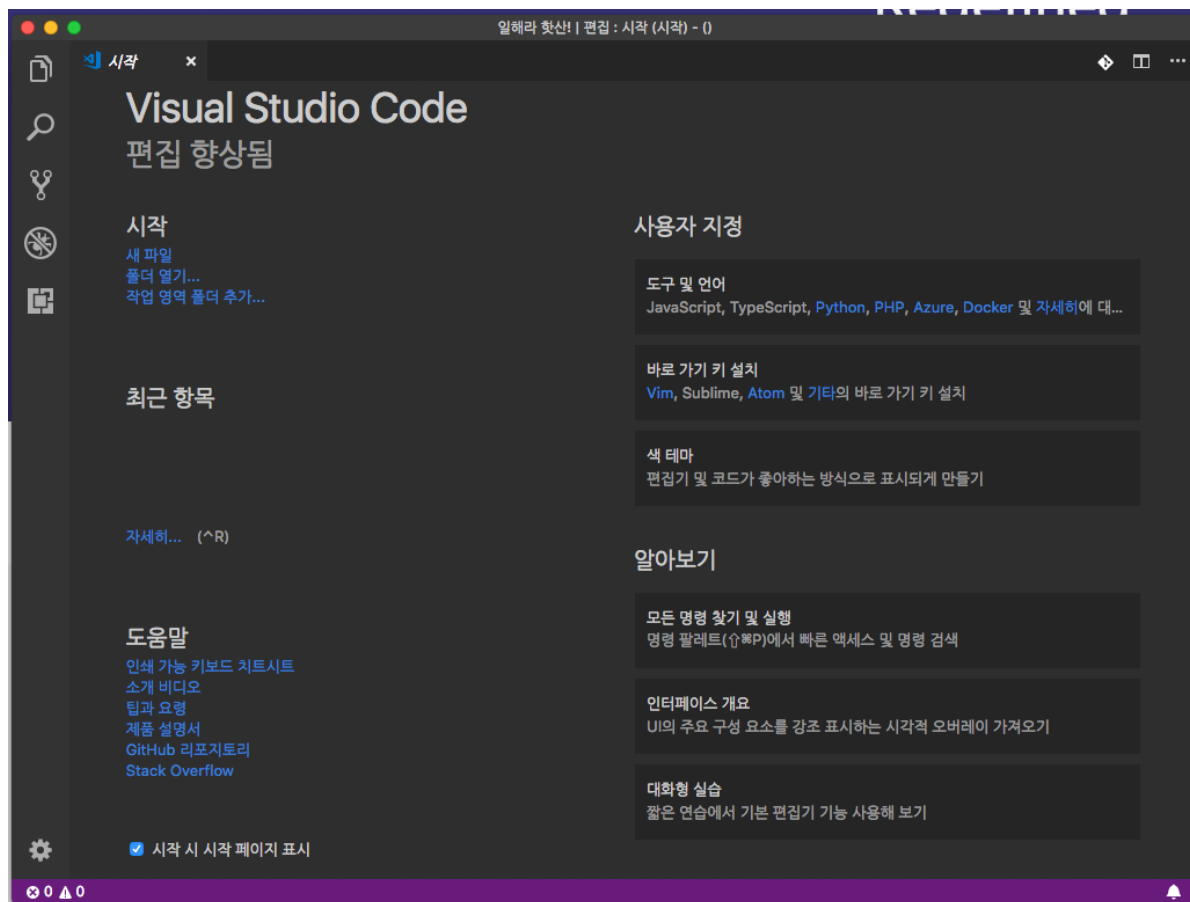
- 에디터 설치
 - 많고 많은 에디터가 있음.
 - Sublime Text, Atom, Visual Studio Code, Bracket, Editplus, Notepad++, Ultraedit, Coda, Acroedit등등등등!
 - 우리는 Visual Studio Code를 사용.
 - MS에서 무료로 제공
 - Ionic에서 사용하는 Typescript를 제대로 지원하는 몇 안되는 에디터 중 하나 (Typescript는 MS에서 만들었으니 당연한걸지도?)

개발환경 구성하기

- Visual Studio Code 설치
 - <https://code.visualstudio.com/> 접속
 - 사용중인 OS에 맞는 버전 설치.
 - 인스톨러가 제공되기 때문에 어려움 없이 설치 가능.

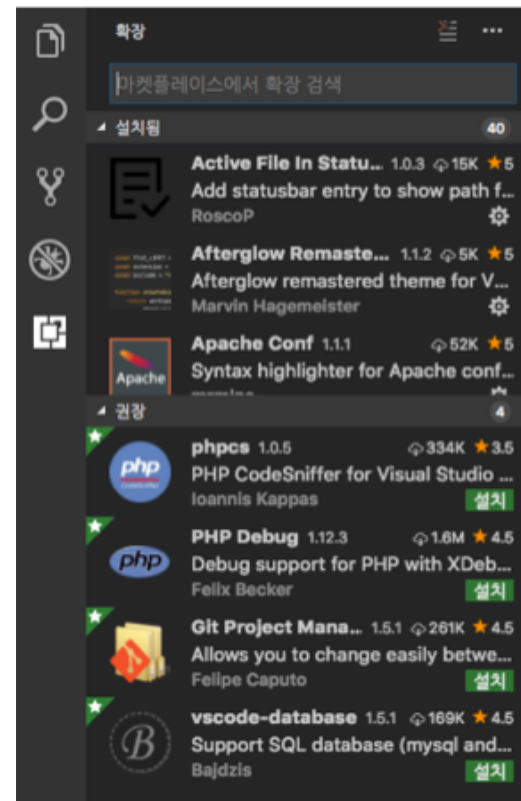
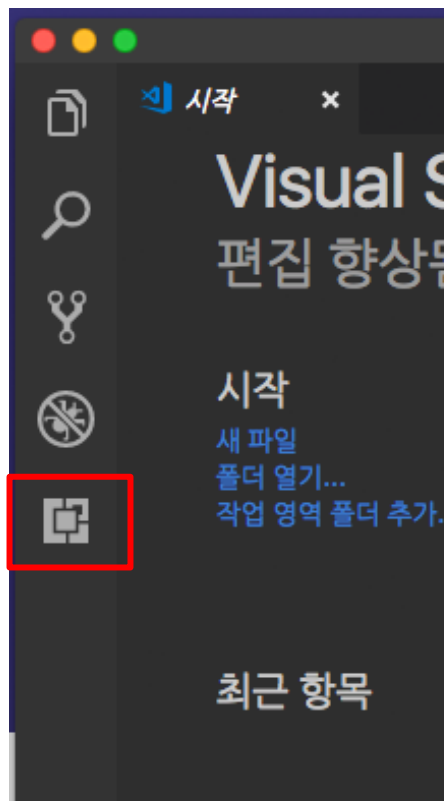
개발환경 구성하기

- Visual Studio Code 셋팅



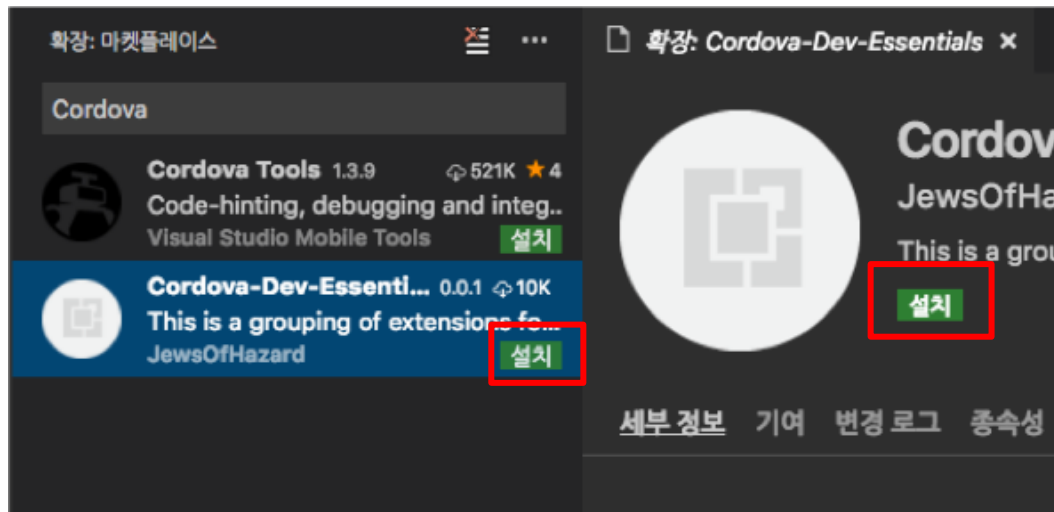
개발환경 구성하기

- 플러그인 설치
 - 메인 화면 왼쪽에서 플러그인 메뉴 클릭



개발환경 구성하기

- 플러그인 설치
 - Cordova-Dev-Essentials 플러그인 설치
 - 해당 검색어의 일부만 입력 후, 엔터키를 누르면 검색.
 - 설치하려는 플러그인 목록 선택 후, [설치] 버튼 클릭.



오늘은 여기까지~

See you next day!