
 ATOM

당신도 해킹 가능한 에디터

오픈 프론티어 2기 이종은

<http://yomybaby.github.io/presentation-atom>에서 HTML 버전을 볼 수 있습니다.



이종은

[Twitter](#) [GitHub](#) yomybaby



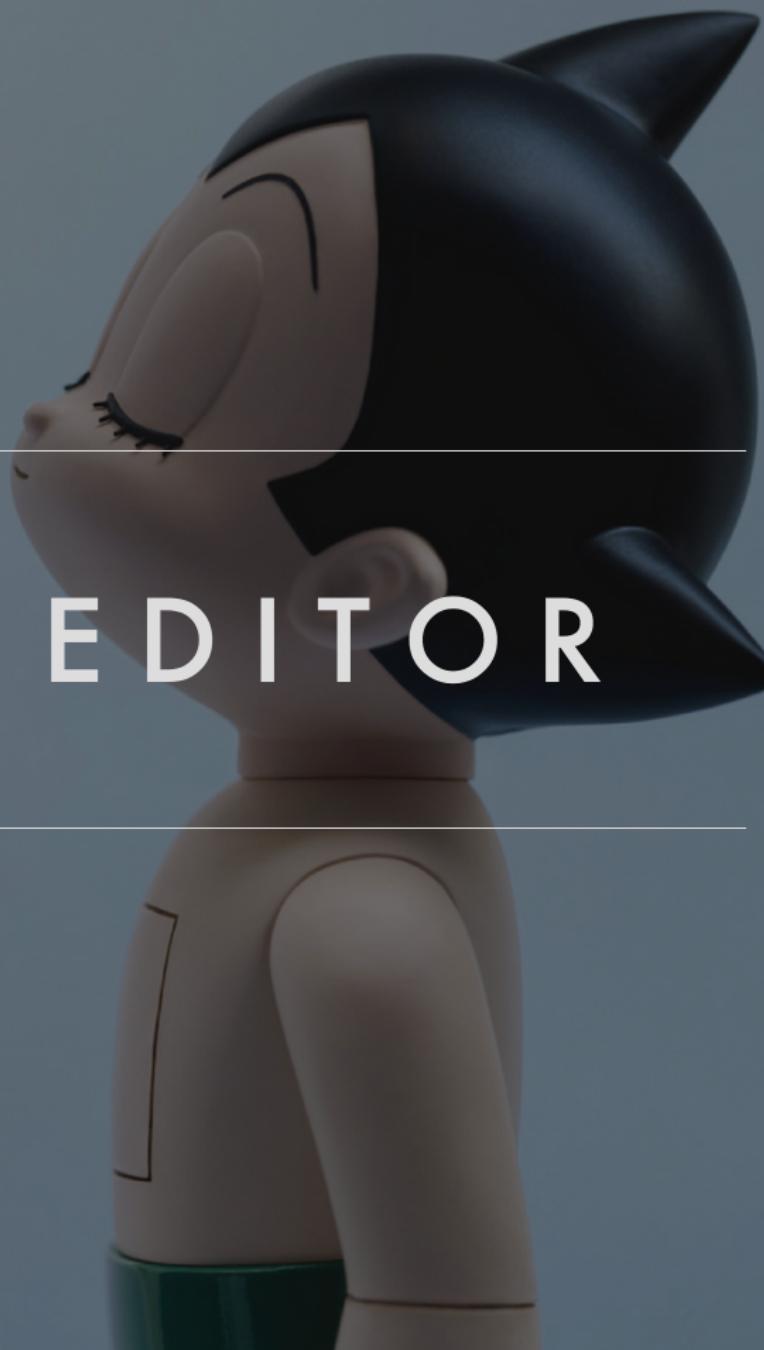
 WEB
 MOBILE



JAVASCRIPT
OPEN SOURCE
CREATIVE COMMONS

목차

- Atom Editor
- Titanium
- Atom Package for Titanium



ATOM EDITOR

PROJECT
ATOM

고백 #1



고백 #2

ATOM으로 갈아탄 지
6개월



WHY ATOM?



Becuase it's

2015

FOR 21ST CENTRY



A HACKABLE TEXT EDITOR

HACKABLE
for 21st Century
by 

HACKABLE

Hackable #1

100% OPEN SOURCE

누구나 자유롭게 개선 할 수 있다.

MIT License

Hackable #2

JAVASCRIPT

당신도 읽고 쓸 수 있는 언어

Hackable #3

웹 기반 에디터

HTML, CSS, and JavaScript

The screenshot shows an Atom editor window with the following details:

- Atom Editor:** The main pane displays the `package.json` file for the `titanium-alloy` package. The JSON content includes details like the package name, main file, version, description, license, keywords, repository URL, and provided services.
- Developer Tools:** The bottom half of the window features the Atom DevTools. It includes:
 - Elements Panel:** Shows the DOM structure of the current page, including elements like `<atom-pane>`, `<atom-pane-container>`, and `<atom-panel>`.
 - Console Panel:** Displays log messages from the browser's JavaScript console, including regex patterns and error messages related to the `flavour source.js` file.
 - Styles Panel:** Allows for inspecting and modifying CSS styles applied to the DOM elements.

Hackable #3-1



ELECTRON

- Web Tech
- Open Source
- Cross Platform

Hackable #3-2

그럼 왜 ELECTRON

// 네이티브는 나타나고 사라진다.
하지만 웹은 꾸준히 그 능력이 증가해왔다.

Q1. ATOM은 SUBLIME보다 느리다?

😅 네! 하지만 달라요!





Hello! Thanks for trying out Sublime Text.

This is an unregistered evaluation version, and although the trial is untimed, a license must be purchased for continued use.

Would you like to purchase a license now?

Cancel

Purchase

SUBLIME이 그리울 땐

ATOM 1.0 Packages Themes Documentation Blog Discuss Sign in

unregistered

Are you missing the annoying popup from sublime text?

thiagopnts 0.1.0 113 19

Repo Bugs Versions License

Flag as spam or malicious

unregistered

Are you missing the annoying popup from sublime text?

Hello! Thanks for trying out Atom.

This is an unregistered evaluation version, and although the trial is untimed, a license must be purchased for continued use.

Would you like to purchase a license now?

Cancel OK

NOT ANYMORE

Q2. ATOM은 큰 프로젝트에는 부적합하다?



아니요! *Facebook : Xcode to Atom(Nuclide)*



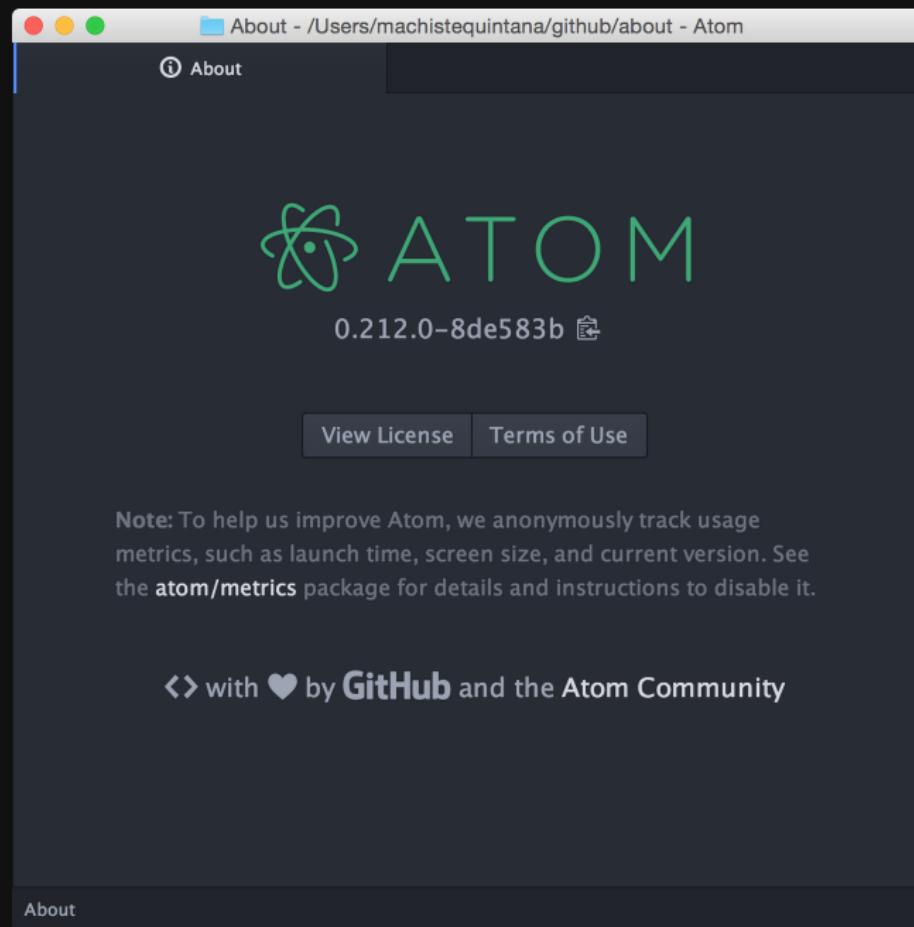
BUILDING STUFF
FOR
BUILDING STUFF

에디터를 만드는 에디터

철학

**LIGHT CORE
MORE EXTENSIBILITY**

About까지도 패키지



WELCOME TO THE PKG WORLD

apm install <whatever-you-want>

유용한 패키지 1

MERGE CONFLICTS

```
_diffs.html.haml - /Users/smash/code/gitlabhq
```

```
commits.scss x notes.scss x _diffs.html.haml o
```

```
We're done here
```

```
You've dealt with all of the conflicts in this file.
```

```
Save and stage this file for commit?
```

```
Mark Resolved
```

```
74 .diff-content
    -# Skipp all non non-supported blobs
    - next unless file.respond_to?('text?')
    - if file.text?
    - if params[:view] == 'parallel'
        = render "projects/commits/parallel_view", diff: diff, project: project
    - else
        = render "projects/commits/text_file", diff: diff, index: index
    - elsif file.image?
        - old_file = project.repository.blob_at(@commit.parent_id, diff.old_path)
        = render "projects/commits/image". diff: diff, old_file: old_file, file: file
```

```
app/views/projects/commits/_diffs.html.haml* 74,1 Main Text ↵ c8e1c53 +17 Send Feedback
```

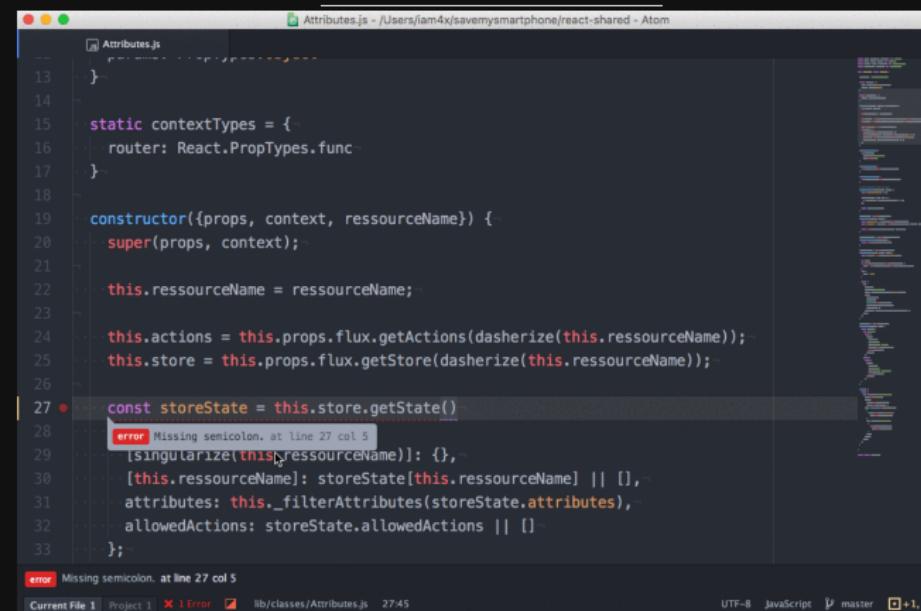
File	Status
app/assets/stylesheets/sections/commits.scss	modified by both
app/assets/stylesheets/sections/notes.scss	modified by both
app/views/projects/commits/_diffs.html.haml	modified by both
features/steps/project/merge_requests.rb	modified by both

```
Conflicts Hide
```

```
Quit
```

유용한 패키지 2

LINTER



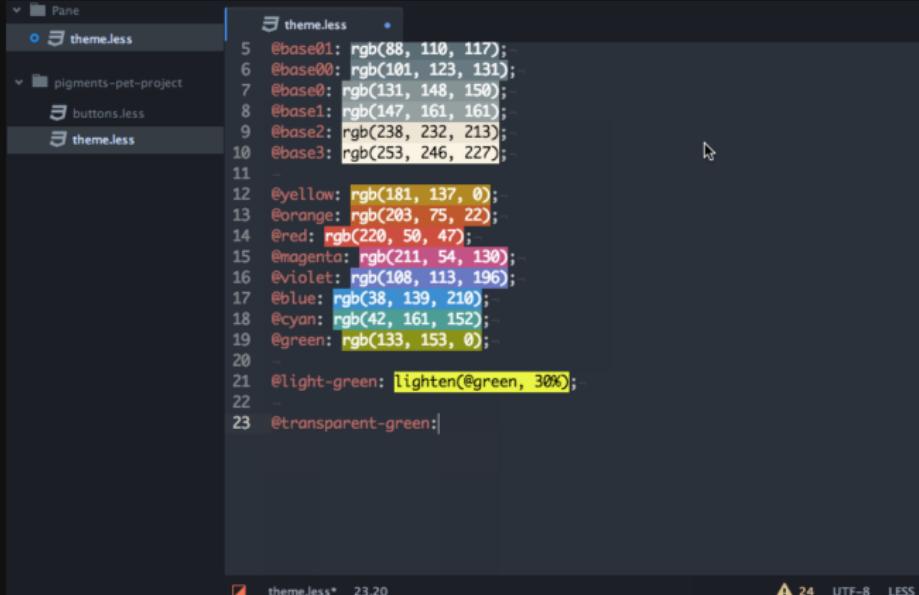
The screenshot shows a dark-themed Atom code editor window. The title bar reads "Attributes.js - /Users/jam4x/savemysmartphone/react-shared - Atom". The main editor area displays a block of JavaScript code. On line 27, there is a red error marker with the message "error Missing semicolon. at line 27 col 5". The code is as follows:

```
13
14
15 static contextTypes = {
16   router: React.PropTypes.func
17 }
18
19 constructor({props, context, ressourceName}) {
20   super(props, context);
21
22   this.ressourceName = ressourceName;
23
24   this.actions = this.props.flux.getActions(dasherize(this.ressourceName));
25   this.store = this.props.flux.getStore(dasherize(this.ressourceName));
26
27 •   const storeState = this.store.getState()
28   error Missing semicolon. at line 27 col 5
29   [singularize(this.ressourceName)]: {},
30   [this.ressourceName]: storeState[this.ressourceName] || [],
31   attributes: this._filterAttributes(storeState.attributes),
32   allowedActions: storeState.allowedActions || []
33 };
```

The status bar at the bottom shows "Current File 1 Project 1 1 Error lib/classes/Attributes.js 27:45" and "UTF-8 JavaScript master □ +1, -1".

유용한 패키지 3

PIGMENTS



```
theme.less
5 @base01: #88, 110, 117;
6 @base00: #101, 123, 131;
7 @base0: #131, 148, 150;
8 @base1: #147, 161, 161;
9 @base2: #238, 232, 213;
10 @base3: #253, 246, 227;
11 -
12 @yellow: #181, 137, 0;
13 @orange: #203, 75, 22;
14 @red: #220, 50, 47;
15 @magenta: #211, 54, 130;
16 @violet: #108, 113, 196;
17 @blue: #38, 139, 210;
18 @cyan: #42, 161, 152;
19 @green: #133, 153, 0;
20 -
21 @light-green: lighten(@green, 30%);
22 -
23 @transparent-green:
```

유용한 패키지 4

BEAUTIFY

The screenshot shows a dark-themed code editor with a context menu open over a block of JavaScript code. The code is as follows:

```
1  $('a[href*="#"]:not([href="#">')).click(function() {if (location.pathname.replace(/^\//, '') == this.pathname.replace(/^\//, '') && location.hostname == this.hostname) {var target = $(this.hash);target = target.length ? target : $('[name=' + this.hash.slice(1) + ']');$(target).offset().top - 0}if (!target) return false;var scroll = target.offset().top - 0;$(window).scrollTop(scroll);return false}});
```

The context menu is divided into several sections:

- Top-level items: Undo, Redo, Cut, Copy, Paste, Delete, Select All.
- Section separator.
- Split Up, Split Down, Split Left, Split Right, Close Pane.
- Section separator.
- Bottom-level items: Beautify editor contents, Debug Atom Beautify, Color Picker, Git add file.

유용한 패키지 5

expand-selection-to-quotes

```
section
h3 유용한 패키지 1
h2
  a(href="https://atom.io/packages/merge-conflicts") Merge Conflicts
  img(src="./images/pkg-merge.gif",width="400px")

section
h3 유용한 패키지 2
h2
  a(href="https://atom.io/packages/linter") Linter
  img(src="./images/pkg-linter.gif",width="400px")

section
```

유용한 패키지 6

regex-railroad-diagram

```
31  ··· serialize: ->-
32  -
33  ··· # Tear down any state and detach
34  ··· destroy: ->-
35  ··· ···@detach()-
36  -
37  ··· getRegexScope: (scope) ->-
38  ··· ···scopeName = []
39  ··· ···for name in scope
40  ··· ···· scopeName.push name
41  -
42  ··· ···if /^string\.regexp/.test name
43  ··· ···· scopeName =
44  -
45  ··· x = /hello(or|foo|bar|(:fo|bar){4})x{4}/-
46  -
47  ··· false-
48  -
49  -
50  ··· showRailRoadDiagram: (regex) ->-
51  ··· rr = atom.workspaceView.find '.regex-railroad-diagram'-
52  ··· if not rr.length-
53  ··· ···# create current diff-
54  ··· ···@hide()-
55  -
56  -
57  -
58  -
59  -
60  -
61  -
62  -
63  -
64  -
65  -
66  -
67  -
68  -
69  -
70  -
71  -
72  -
73  -
74  -
75  -
76
```

The railroad diagram illustrates the regex pattern:

```
graph LR
    hello[hello] --> or1((or))
    hello --> foo1((foo))
    hello --> bar1((bar))
    or1 --> fo1((fo))
    or1 --> bar2((bar))
    foo1 --> fo2((fo))
    foo1 --> bar3((bar))
    fo1 --- fo2
    fo1 --- bar3
    bar2 --- fo2
    bar2 --- bar3
    fo2 --- capture1[capture 1]
    bar3 --- capture1
    capture1 --- atLeast4Times[x at least 4 times]
```

A callout labeled "capture 1" points to the "fo" and "bar" nodes under the "foo" node. A note "at least 4 times" is placed next to the "x" node.

유용한 패키지 7

imdone-atom

The screenshot shows the Atom code editor interface. On the left is a file tree with the following directory structure:

- imdone-core
- imdone-atom
 - imdone
 - keymaps
 - lib
 - config-view.coffee
 - imdone-atom-view.coffee
 - imdone-atom.coffee
 - imdone-helper.coffee
 - js-utils.js
 - menu-view.coffee
 - plugin-manager.coffee
 - menus
 - spec
 - styles
 - .gitignore
 - CHANGELOG.md
 - LICENSE.md
 - package.json
 - README.md- imdone-atom-github
- innobuilt.github.io
- imdone-web

The main editor area displays the contents of the `lib/plugin-manager.coffee` file:

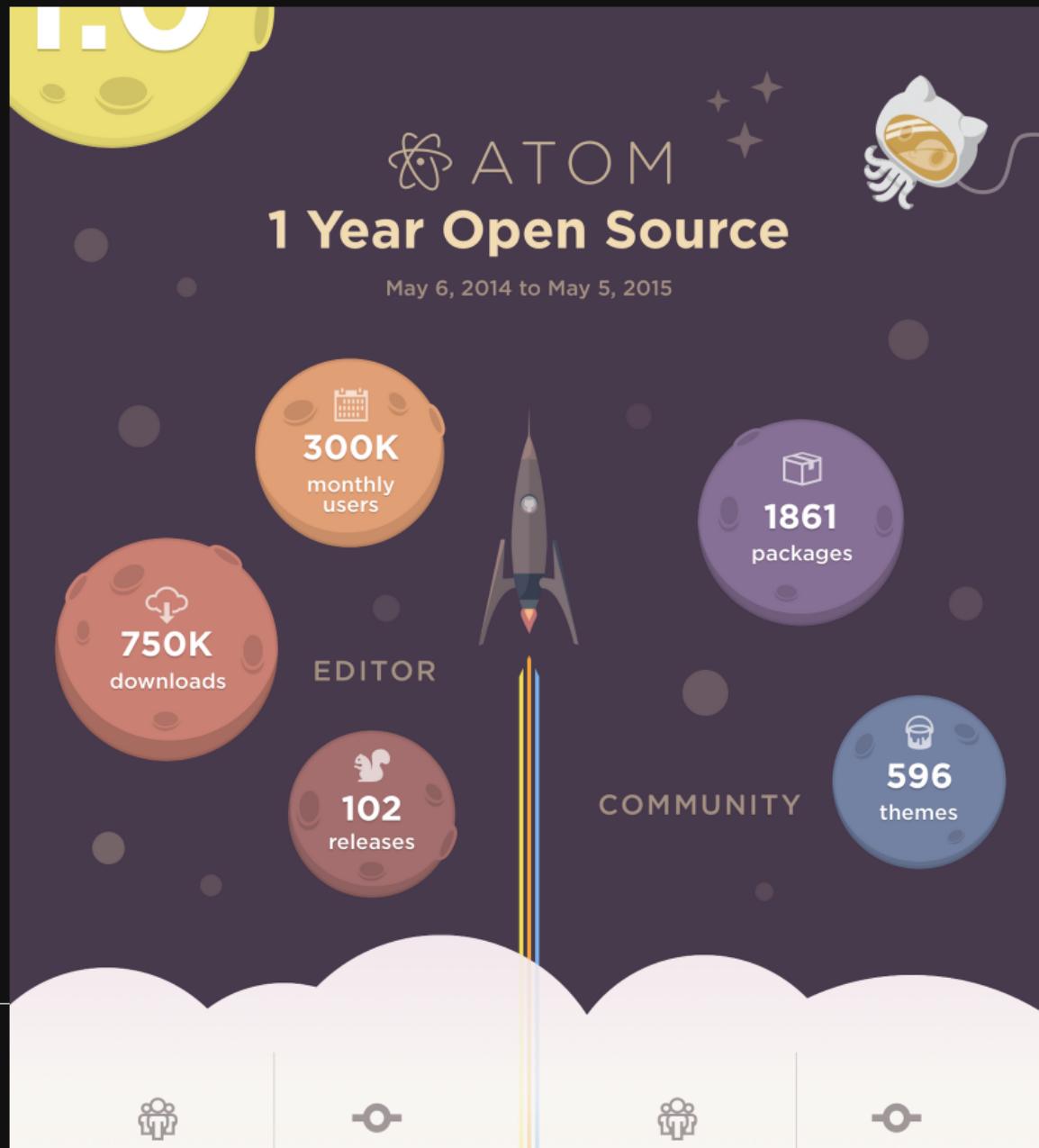
```
1  {Emitter} = require 'atom'
2
3  # TODO
4  module.exports =
5    emitter: new Emitter
6    plugins: {}
7
8    addPlugin: (Plugin) ->
9      return unless Plugin && Plugin.pluginName
10     @plugins[Plugin.pluginName] = Plugin
11     @emitter.emit 'plugin.added', Plugin
12
13    removePlugin: (Plugin) ->
14      return unless Plugin && Plugin.pluginName
15      delete @plugins[Plugin.pluginName]
16      @emitter.emit 'plugin.removed', Plugin
17
18    getAll: ->
19      (val for key, val of @plugins)
```

At the bottom of the editor, the status bar shows: lib/plugin-manager.coffee* 3:7 + LF UTF-8 CoffeeScript master

모르고 지나갈 만한

기본 패키지들

- [bracket-matcher](#)
- [markdown-preview](#)
- [Open on Github](#)
- ...





3339 PACKAGES



968 THEMES



259 RELEASE

TITANIUM



a

JAVASCRIPT로
NATIVE 앱을 만드는
플랫폼



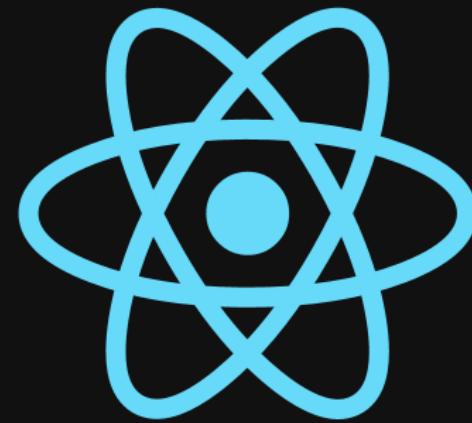
CROSS PLATFORM



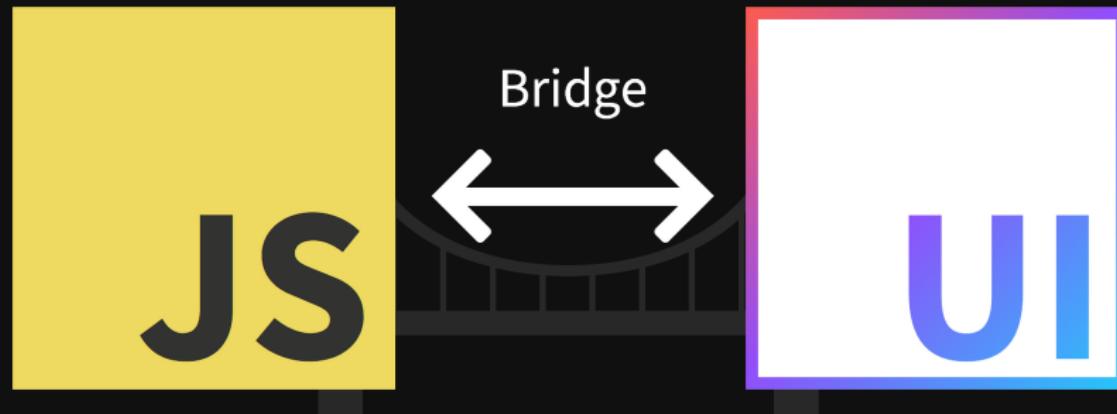
OPEN SOURCE

Apache License 2

JS TO NATIVE



FACBOOK - REACT NATIVE



Q.

왜 *Titanium*을 택했나?

A. 생산성

- 수많은 API, 다양한 관련 오픈소스
- JAVASCRIPT
- 개인이나 소규모팀에서도 빠르게 서비스 개발!

LESS CODING
MORE THINKING

개발보다는 제품자체에 집중

TITANIUM 둘러보기

titanium-alloy

All-in-One package for Titanium Alloy

#titanium #alloy #jump-to-definition #autocomplete

 yomybaby

 1,103



16

ATOM PACKAGE

 Repo

 Bugs

 Versions

 License

Flag as spam or malicious

FOR TITANIUM ALLOY

All-in-One package for Titanium Alloy

This is a Atom package for [Titanium Alloy](#)

```
$ apm install titanium-alloy
```

결심
STUDIO에서 탈출하자!

TI 개발에서 느끼는 어려움 #1

STUDIO가 무겁다

Eclipse 기반이기에

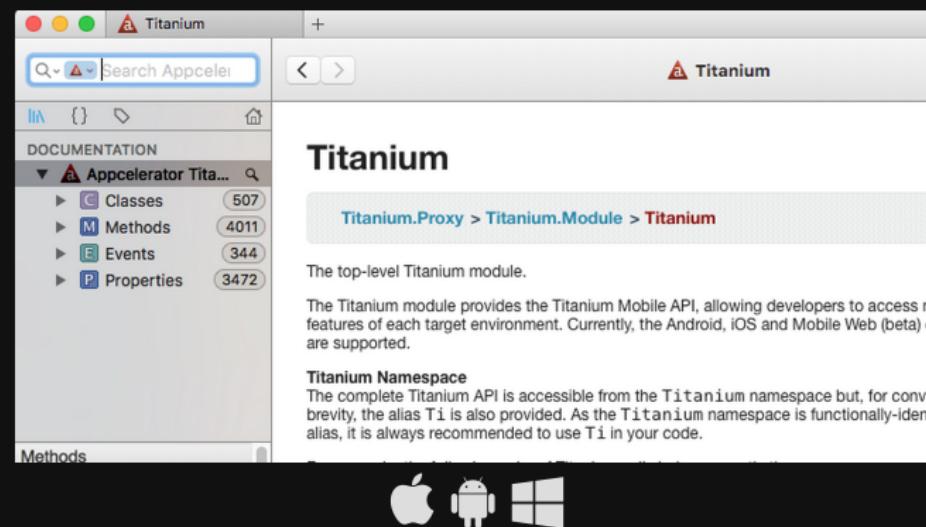
TI 개발에서 느끼는 어려움 #2

반복적 타이핑

```
1 <Alloy>
2   <ItemTemplate class="itemTemplate" name="postItemTemplate">
3     <View>
4       <ImageView class="photo" bindId="photo"></ImageView>
5     </View>
6     <View class="opacityScreen"></View>
7
8     <Label class="photoTitle" bindId="title"></Label>
9     <Label class="time" bindId="time"></Label>
10    <Label class="userName" bindId="userName"></Label>
11    <View class="verticalMetaWrap">
12      <Label class="commentCount" bindId="commentCount"></Label>
13      <Label class="likeCount" bindId="likeCount"></Label>
14    </View>
```

TI 개발에서 느끼는 어려움 #3

수많은 API



TI 개발에서 느끼는 어려움 #4

오타는 좀 치명적

Because JS.



TI 개발에서 느끼는 어려움 #5

작은 파일 이동

View, Style, Controller..

패키지 개발 절차 #1

비슷한 놈을 찾아라

패키지 개발 절차 #2

비슷한 패키지를 DEV로 설치한다.

```
apm dev <package_name> [<directory>]
```

패키지 개발 절차 #3

분석하고 수정한다

Pull Request 혹은 새로운 패키지로

시연

转折



Christian Brousseau
@TheBrousse

This guy is making something better than Titanium Studio ever was. Kudos man!



Jong Eun Lee @yomybaby
#titanium View Autocomplete for tag,event,property,value! It's v0.6.1 of #ti @atom package. git.io/vIEUd

 1,144in 7
people 4

처음 시작할 때

- Coffee script
- Babel
- RegExp

■ TIP 1

수정 후 RELOAD EDITOR!

cmd + alt + ctrl + l

개선 PR : reload package without reloading editor

■ TIP 2

커서위치의 SCOPE 확인

cmd + alt + p

■ TIP 3

특정폴더에서 먹통일 때

`rm ~/.atom/storage/*`



ATOM의 미래

From

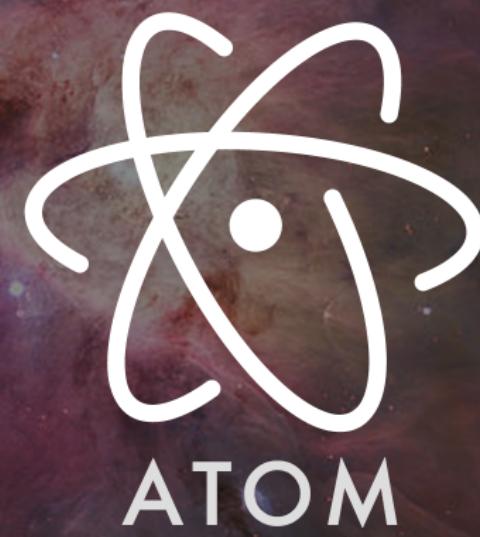
초등학교에서 **처음 코딩을 배우는 날** 사용한 에디터

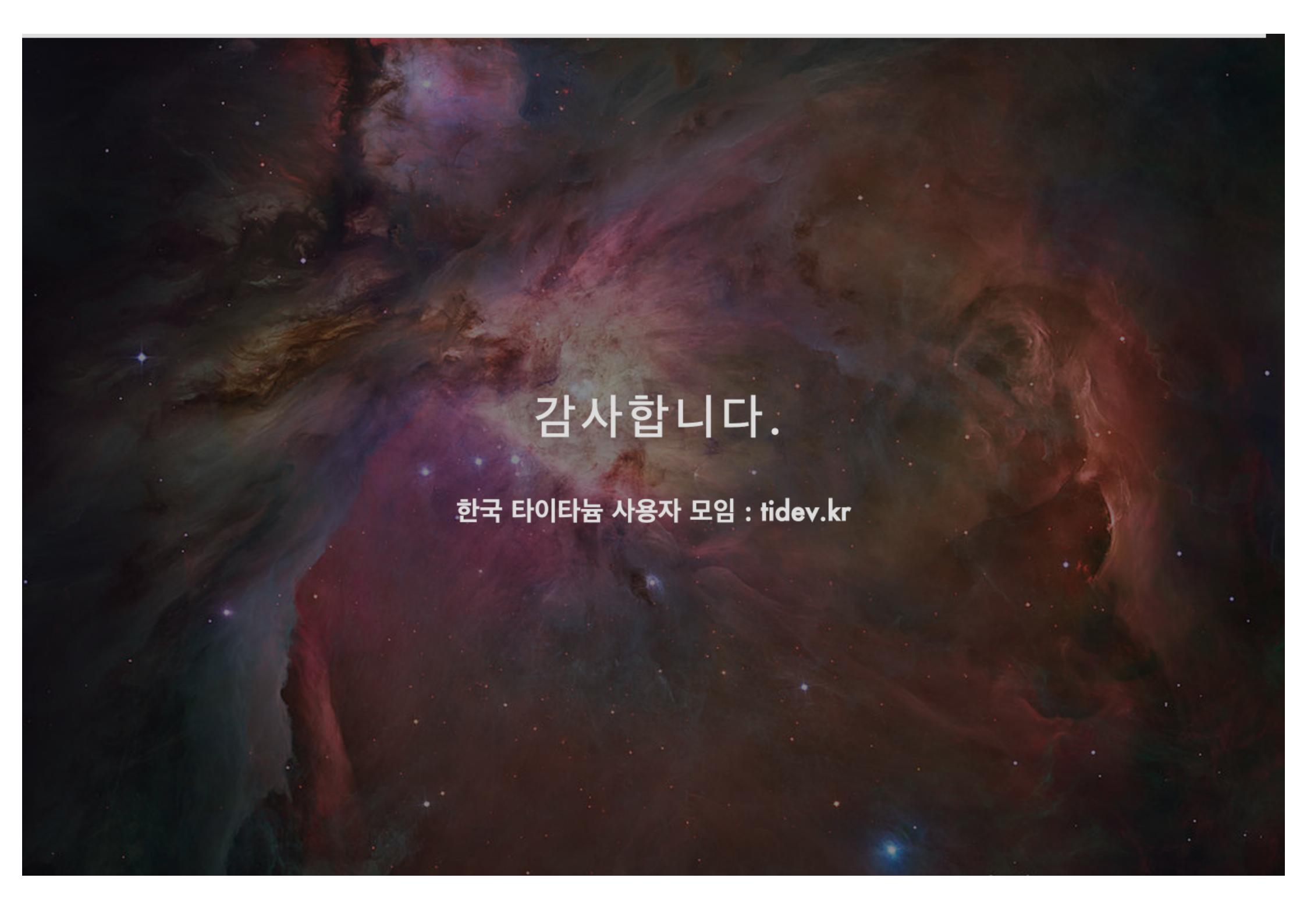
To

노련한 개발자가 되어도 계속 쓸 수 있는 에디터

기억할 3가지

- Open Source
- JavaScript
- Light core + Pakcakges





감사합니다.

한국 타이타늄 사용자 모임 : tidev.kr

사용 이미지

- bridge icon by freepik (CC BY)
- Atom 캐릭터 이미지 by Tokyo Denki University

참조

- <http://www.slideshare.net/ateevexes/an-intro-to-atom-editor>
- <http://www.wired.com/2015/06/github-atoms-code-editor-nerds-take-universe/>
- <http://code.tutsplus.com/tutorials/check-out-atom-githubs-new-development-editor--net-37030>
- <http://johnnunemaker.com/atom-a-hackable-text-editor-for-the-21st-century/>