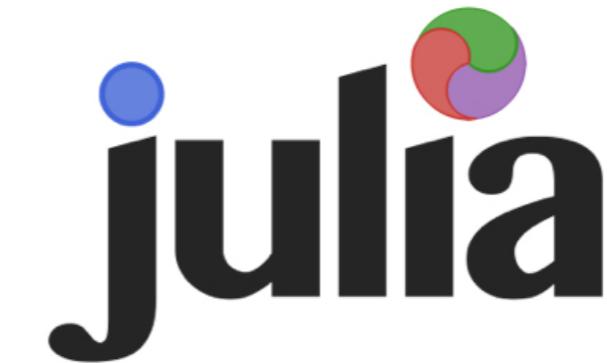


```
# Julia 0.5
* DataFrames.jl
* TensorFlow.jl
```

October 14, 2016

- wookyoung noh
- <https://github.com/wookay>



목차 1

- * 발표자 소개
- * Julia 0.5
- * Julia Packages
 - DataFrames.jl & 경쟁자
 - TensorFlow.jl & 경쟁자

목차 2

- * Julia 커뮤니티
 - 깃헙, 메일링 리스트
 - JuliaCon, Julia in the classroom
- * Julia Korea 커뮤니티
 - 깃헙, 페이스북, 슬랙
 - 줄리아 문서 번역 프로젝트
- * 개발 툴
 - REPL, Jupyter notebook, Juno IDE
- * Julia 0.5
 - 코딩 시간
- * Julia 0.6, 1.0, 2.0

발표자 소개

- <https://github.com/wookay/hackdiary/blob/master/merged/Julia.md>
- <https://github.com/wookay/journal/wiki/언어-덕후의-일기>

Julia 0.5

- * 2016년 9월 19일 릴리즈
 - Version 0.5.0 (2016-09-19 18:14 UTC)
 - <https://github.com/JuliaLang/julia/milestone/14>

Julia 0.5 Highlights

- <http://julialang.org/blog/2016/10/julia-0.5-highlights>

Julia Packages

```
julia> Pkg.add("DataFrames")
```

```
julia> using DataFrames
```

```
julia> Pkg.add("TensorFlow")
```

```
julia> using TensorFlow
```

짤



source: <http://imgur.com/gallery/jzc9UkS>

Julia Statistics

Statistics and Machine Learning made easy in Julia

- <https://github.com/JuliaStats>

DataFrames.jl

library for working with tabular data in Julia

- <https://github.com/JuliaStats/DataFrames.jl>

경쟁자

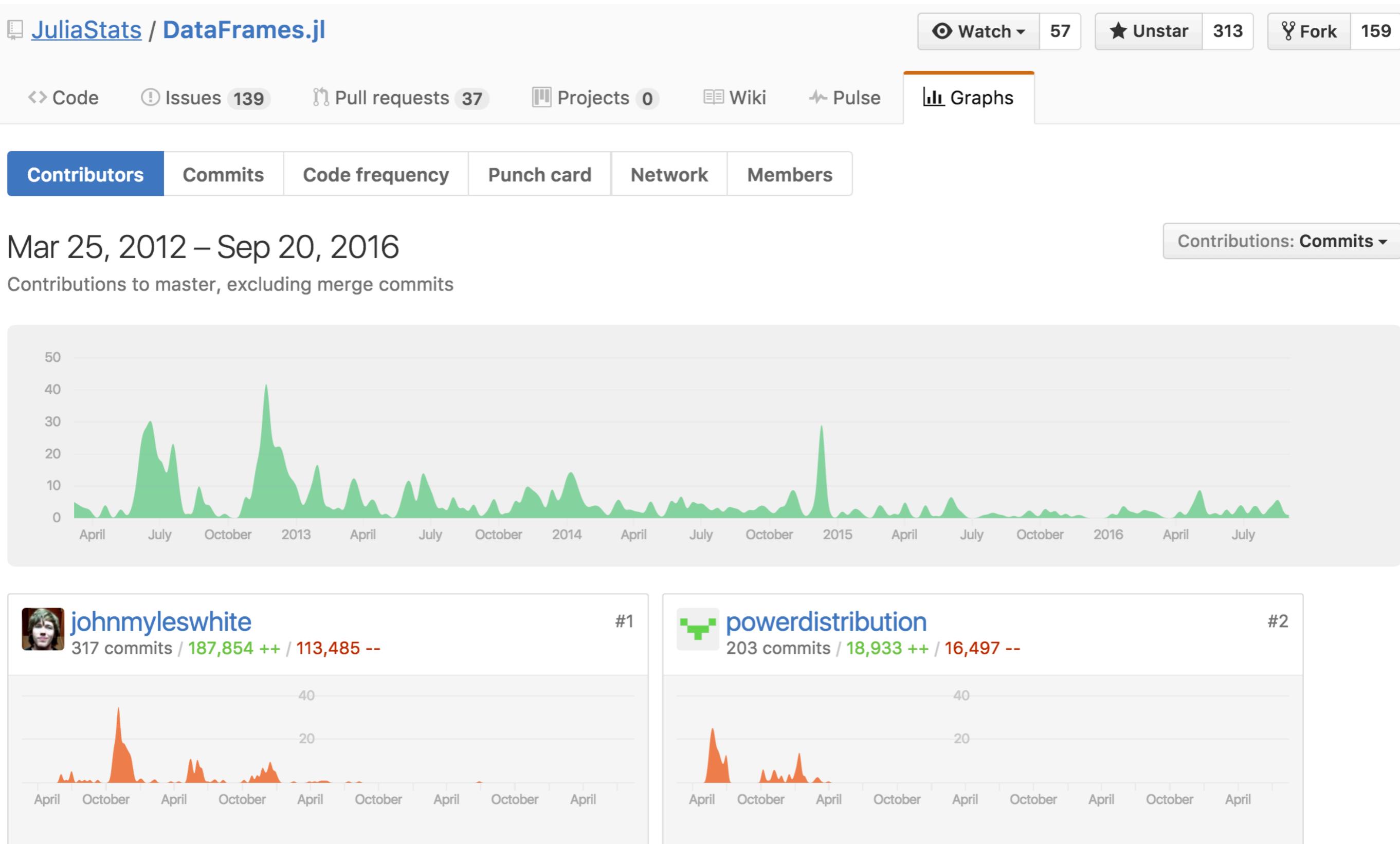
- * R
- * Python pandas

```
# pandas, DataFrames.jl | wc
```

```
~/work/python/pandas $ git shortlog --numbered --summary | wc  
757      2098     14956
```

```
~/.julia/v0.5/DataFrames $ git shortlog --numbered --summary | wc  
105      301     2118
```

<https://github.com/JuliaStats/DataFrames.jl/graphs/contributors>



* John Myles White

<http://www.oreilly.com/pub/au/4730>



John Myles White

Statistics and machine learning educator

jmw@johnmyleswhite.com | @johnmyleswhite

Princeton, New Jersey

Areas of Expertise:

- machine learning
- statistics
- data science
- R

- consulting
- speaking
- training
- writing

Biography

Books

Multimedia

Praise



Bandit Algorithms for Website Optimization

by John Myles White

December 2012

Print: \$19.99

Ebook: \$16.99

 4.0



Machine Learning for Hackers

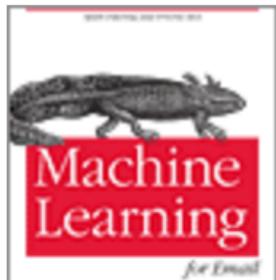
by Drew Conway, John Myles White

February 2012

Print: \$49.99

Ebook: \$42.99

 3.6



Machine Learning for Email

by Drew Conway, John Myles White

October 2011

Print: \$24.99

Ebook: \$20.99

Julia for Data Science 책 두권

- <http://julialang.org/learning/#books>



짤



source: <https://giphy.com/gifs/southparkgifs-10H1TAPBSZPd7438k>

TensorFlow.jl

Julia wrapper for TensorFlow

- <https://github.com/malmaud/TensorFlow.jl>

경쟁자

- * Python
- * C++

TensorFlow

TensorFlow is an Open Source Software Library for Machine Intelligence

- <https://www.tensorflow.org/>

“ TensorFlow was originally developed by researchers and engineers working on the Google Brain Team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research, but the system is general enough to be applicable in a wide variety of other domains as well. ”

```
# tensorflow, tensorflow/python, TensorFlow.jl | wc
```

```
~/work/tensorFlow $ git shortlog --numbered --summary | wc  
457    1299    9199
```

```
~/work/tensorFlow/tensorflow/python $ git shortlog --numbered --  
summary . | wc  
173    506    3529
```

```
~/.julia/v0.5/TensorFlow $ git shortlog --numbered --summary | wc  
5      14      96
```

```
~/work/tensorFlow $ git shortlog --numbered --summary .
```

2702 A. Unique TensorFlower
638 Vijay Vasudevan
363 Illia Polosukhin
215 Martin Wicke
211 Benoit Steiner
183 terrytangyuan
179 Derek Murray

...

```
~/work/tensorFlow/tensorflow/python $ git shortlog --numbered --summary .
```

638 A. Unique TensorFlower
263 Vijay Vasudevan
114 Eugene Brevdo
100 Derek Murray
83 Geoffrey Irving
67 Benoit Steiner
63 Martin Wicke

TensorFlow KR

- <https://www.facebook.com/groups/TensorFlowKR/>

짤 - 러닝머신



Julia 커뮤니티

<http://julialang.org/community/>

- Github, Mailing lists, Gitter, IRC
- JuliaCon
- Julia in the classroom

Julia 커뮤니티 - Github

<https://github.com/JuliaLang/julia>

- Code
- Issues
- Pull requests

Julia 커뮤니티

<http://julialang.org/community/>

- Mailing lists

<https://groups.google.com/forum/#!forum/julia-users>

- Gitter

<https://gitter.im/JuliaLang/julia>

- IRC

<http://webchat.freenode.net/?channels=julia>

JuliaCon

- * 줄리아 컨퍼런스

- JuliaCon 2014
- JuliaCon 2015
- JuliaCon 2016

JuliaCon 2014 - 시카고 Gleacher Center

- <http://juliacon.org/2014/>
- 유튜브 [https://www.youtube.com/playlist?
list=PLP8iPy9hna6TSRouJfvobfxkZFYiPSvPd](https://www.youtube.com/playlist?list=PLP8iPy9hna6TSRouJfvobfxkZFYiPSvPd)

JuliaCon 2015 - MIT

- <http://juliacon.org/2015/>
- 유튜브 [https://www.youtube.com/playlist?
list=PLP8iPy9hna6Sdx4soiGrSefrm0PdUWixM](https://www.youtube.com/playlist?list=PLP8iPy9hna6Sdx4soiGrSefrm0PdUWixM)

JuliaCon 2015 - MIT



source: <https://twitter.com/acidflask/status/633349038226690048>

JuliaCon 2016 - MIT

- <http://juliacon.org/2016/>
- 유튜브 https://www.youtube.com/playlist?list=PLP8iPy9hna6SQPwZUDtAM59-wPzCPyD_S

JuliaCon 2016 - MIT

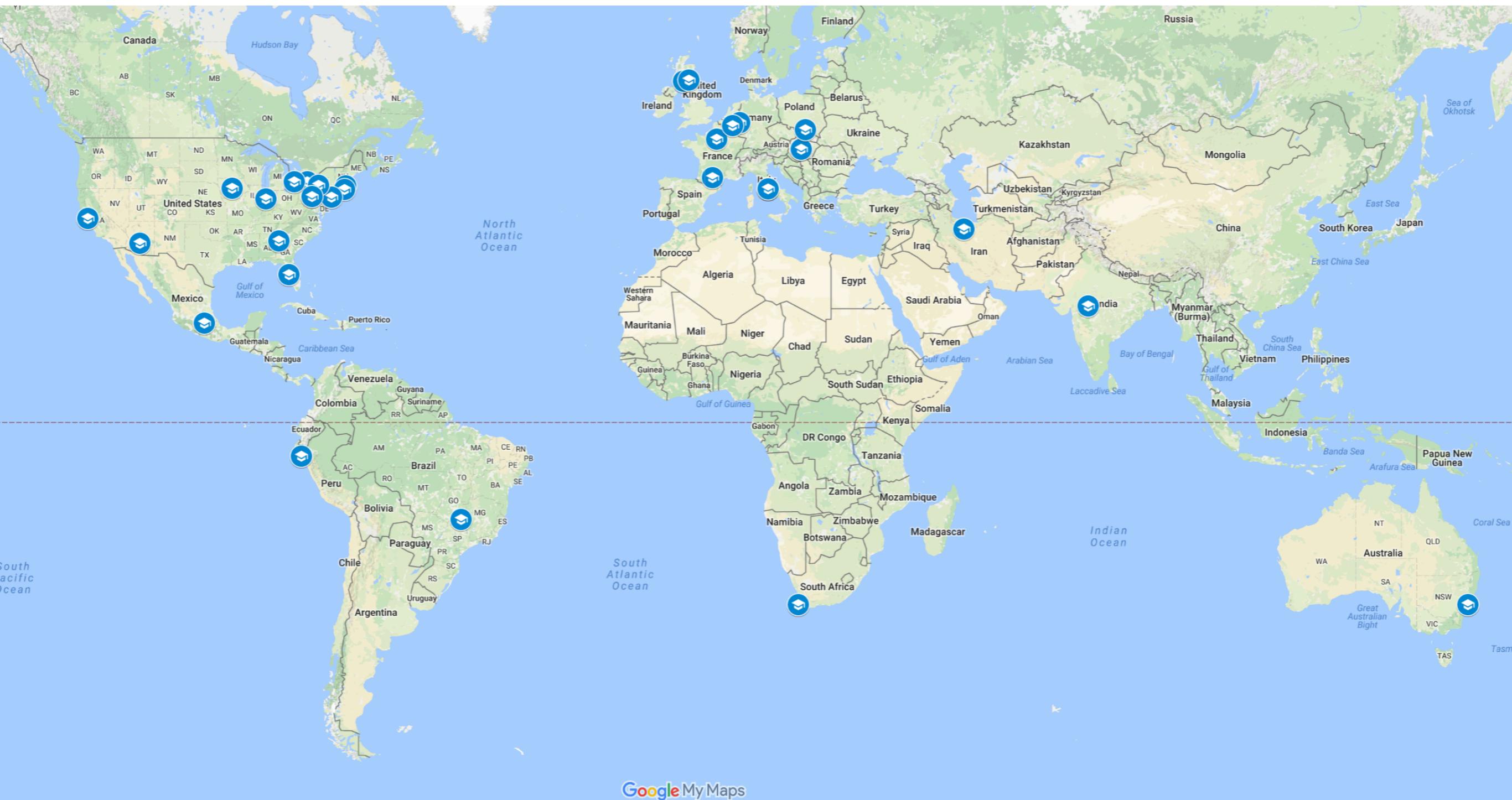


source: <https://twitter.com/ScienceMIT/status/75588576871583344>

Julia in the classroom

- * 줄리아를 강의에 사용하는 대학
 - <https://drive.google.com/open?id=1w6BwQzf87CxKyEGjRD321W93174&usp=sharing>

Julia in the classroom



머시기 Korea 페이스북 커뮤니티

* Python Korea

* TensorFlow KR

* R Korea

머시기 Korea 페이스북 커뮤니티

Python Korea 멤버 13,033명

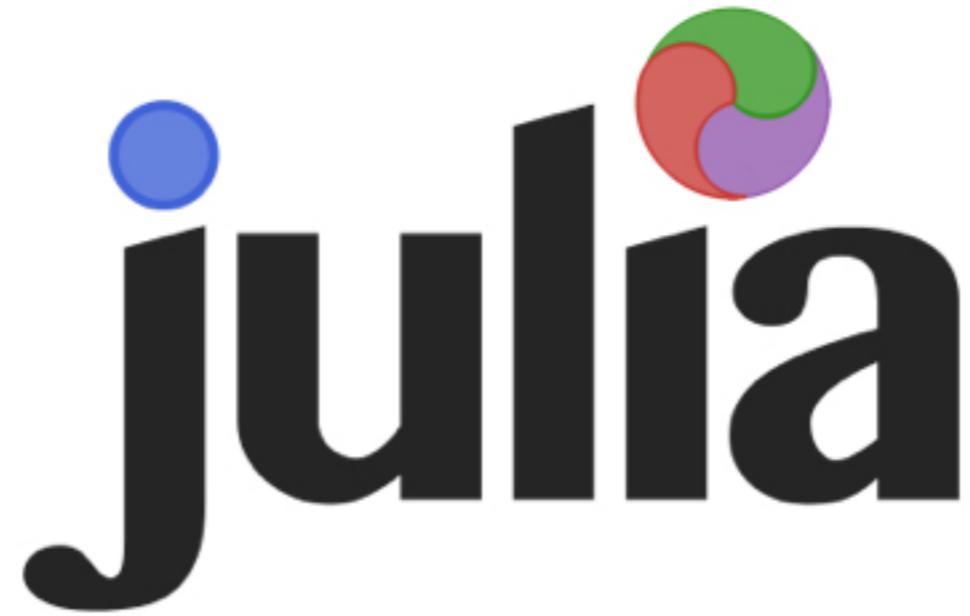
- <https://www.facebook.com/groups/pythonkorea/>

TensorFlow KR 멤버 5,204명

- <https://www.facebook.com/groups/TensorFlowKR/>

R Korea 멤버 2,972명

- <https://www.facebook.com/groups/KoreaRUsers/>



- * Julia Korea 멤버 237명
 - <https://www.facebook.com/groups/juliakorea/>

Julia Korea 커뮤니티

- * Github
 - <https://github.com/juliakorea>
- * Facebook Group
 - <https://www.facebook.com/groups/juliakorea/>
- * Slack
 - <http://juliakorea.slack.com/>

Julia Korea 커뮤니티 - Github

- * 줄리아 문서 번역 프로젝트
 - <https://github.com/juliakorea/doc>
- * 줄리아 매뉴얼 (일부 번역)
 - <http://juliakorea.github.io/latest/>

Julia Korea 커뮤니티 - 슬랙

- <http://juliakorea.slack.com/>

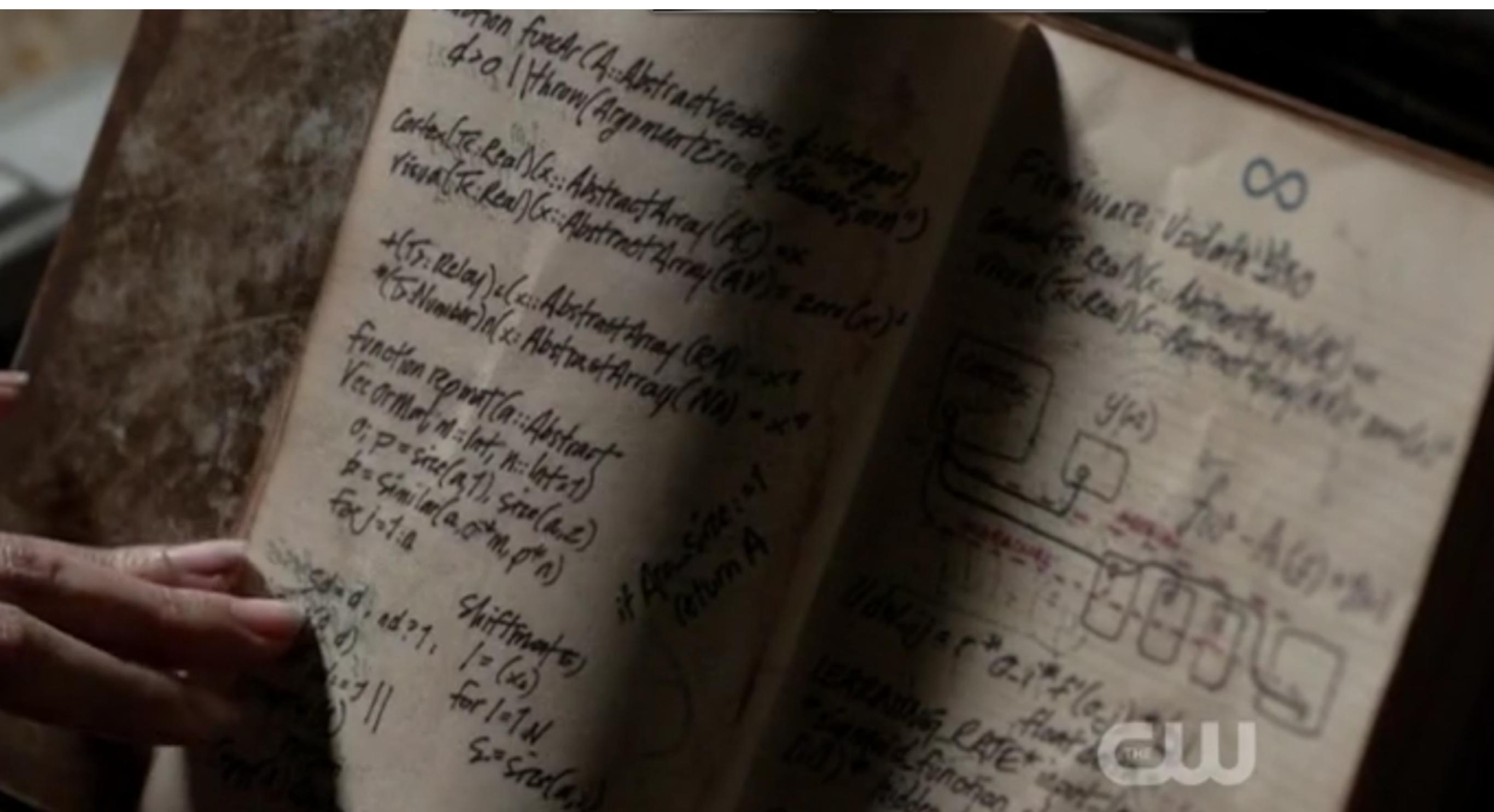
juliakorea 슬랙 가입하기

- <http://juliakorea-slack.herokuapp.com/>

A.L.I.E. has been coded in Julia

https://www.reddit.com/r/The100/comments/4gt7o0/alie_has_been_coded_in_julia/

- [https://en.wikipedia.org/wiki/The_100_\(TV_series\)](https://en.wikipedia.org/wiki/The_100_(TV_series))



개발 툴

- REPL
- Jupyter notebook
- Juno

REPL

- * 건 乾 ☰
- * 리 離 ☲
- * 감 坎 ☵
- * 곤 坤 ☷

```
# REPL - 0.3.12, 0.4.7, 0.5.0, 0.6.0-dev
```

```
λ ~ ≡≡  
_ _ _ _ _ _ _ _ _ _ | A fresh approach to technical computing  
_ _ _ _ _ _ _ _ _ _ | Documentation: http://docs.julialang.org  
_ _ _ _ _ _ _ _ _ _ | Type "?help" for help.  
_ _ _ _ _ _ _ _ _ _ | Version 0.6.0-dev.716 (2016-09-20 08:45 UTC)  
_ _ _ _ _ _ _ _ _ _ | Commit 721ba7a* (0 days old master)  
_ _ / \ _ _ _ _ _ _ | x86_64-apple-darwin16.0.0
```

Jupyter notebook

- Project Jupyter <http://jupyter.org/>
- Julia python r

JuliaBox - Run Julia from the Browser. No setup.

- <https://juliabox.com/>

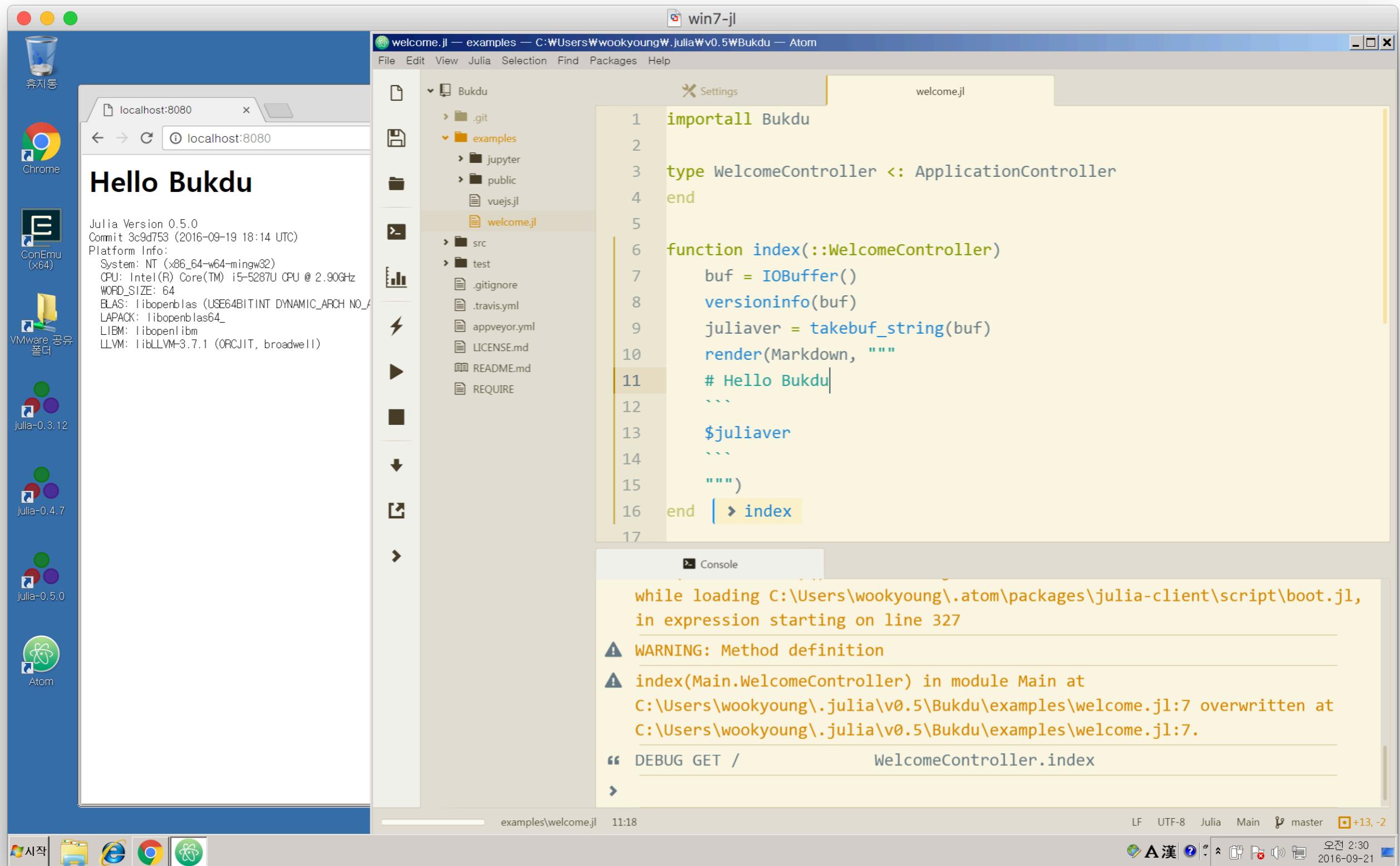
Juno IDE - 맥

The screenshot shows the Juno IDE interface on a Mac OS X system. The title bar reads "welcome.jl — /Users/wookyung/.julia/v0.6/Bukdu". The left sidebar contains a file tree for the "Bukdu" directory, which includes subfolders ".git", "examples" (containing "jupyter" and "public"), and files "vuejs.jl" and "welcome.jl". The main editor area displays the "welcome.jl" file content:

```
1 importall Bukdu
2
3 type WelcomeController <: ApplicationController
4 end
5
6 index(::WelcomeController) = "Hello Julia $VERSION" | > index
7
8 Router() do
9     get("/", WelcomeController, index)
10 end
11
12 Bukdu.start(8080)
13
14 Endpoint() do
15     plug(Plug.Logger)
16     plug(Router)
17 end
18
19 wait()
20
21 # Bukdu.stop()
22
```

The status bar at the bottom shows "examples/welcome.jl*" 6:52 (1, 51) and icons for LF, UTF-8, Julia, Main, and master.

Juno IDE - 원도우



Julia v0.5.0 Release Notes

- * <https://github.com/JuliaLang/julia/blob/v0.5.0/NEWS.md>

Julia 0.5 Highlights

- <http://julialang.org/blog/2016/10/julia-0.5-highlights>

New language features

- * Generator expressions: `f(i) for i in 1:n` ([#4470]). This returns an iterator that computes the specified values on demand. This is useful for computing, e.g. `sum(f(i) for i in 1:n)` without creating an intermediate array of values.
- * Generators and comprehensions support filtering using `if` ([#550]) and nested iteration using multiple `for` keywords ([#4867]).

New language features

- * Fused broadcasting syntax: ```f.(args...)``` is equivalent to ```broadcast(f, args...)``` ([#1508]). and nested ``f.(g.(args...))`` calls are fused into a single `broadcast` loop ([#17300]). Similarly, the syntax ``x .= ...`` is equivalent to a `broadcast!(identity, x, ...)` call and fuses with nested "dot" calls; also, ``x .+= y`` and similar is now equivalent to ``x .= x .+ y``, rather than ``x = x .+ y`` ([#17510]).

New language features

- * Macro expander functions are now generic, so macros can have multiple definitions (e.g. for different numbers of arguments, or optional arguments) ([#8846], [#9627]). However note that the argument types refer to the syntax tree representation, and not to the types of run time values.
- * Varargs functions like `foo{T}(x::T...)` may now restrict the number of such arguments using `foo{T,N}(x::Vararg{T,N})` ([#11242]).
- * `x ∈ X` is now a synonym for `x in X` in `for` loops and comprehensions, as it already was in comparisons ([#13824]).

New language features

- * The `PROGRAM_FILE` global is now available for determining the name of the running script ([#14114]).
- * The syntax `x.:sym` (e.g. `Base.:+`) is now supported, while using `x.(:sym)` or `x.(i)` for field access are deprecated in favor of `getfield` ([#15032]).
- * Function return type syntax `function f()::T` has been added ([#1090]). Values returned from a function with such a declaration will be converted to the specified type `T`.

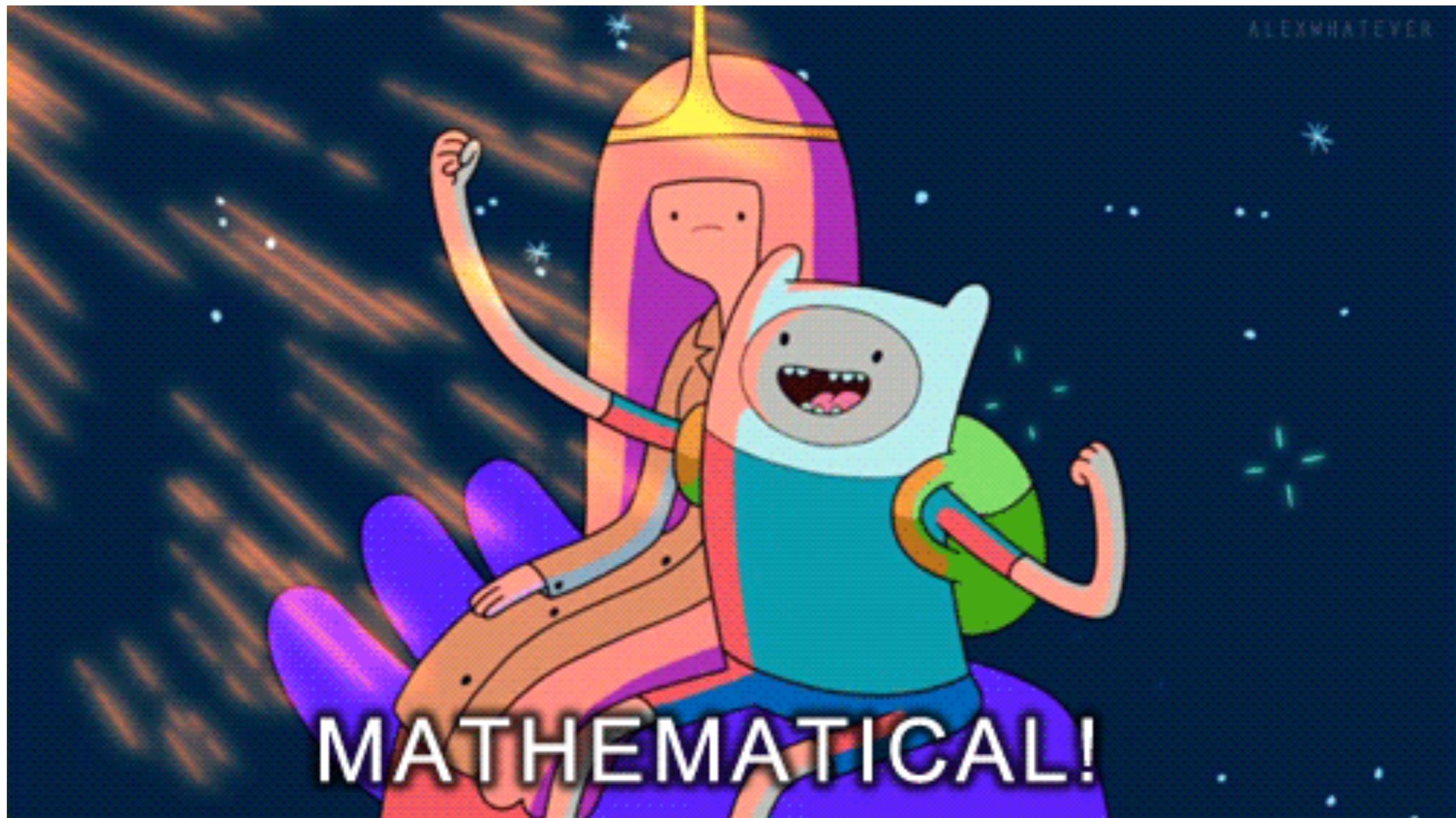
New language features

- * Many more operators now support `.` prefixes (e.g. `. \leq `) ([#17393]). However, users are discouraged from overloading these, since they are mainly parsed in order to implement backwards compatibility with planned automatic broadcasting of dot operators in Julia 0.6 ([#16285]). Explicitly qualified operator names like `Base. \leq ` should now use `Base. $\mathbf{:}\leq$ ` (prefixed by `@compat` if you need 0.4 compatibility via the `Compat` package).

New language features

- * User-extensible bounds check elimination is now possible with the new `@boundscheck` macro ([#14474]). This macro marks bounds checking code blocks, which the compiler may remove when encountered inside an `@inbounds` call.

코딩 시간



source: <https://media.giphy.com/media/cc08MSKkjHE2c/giphy.gif>

```
# Julia | wc
```



```
~/work/julia master git shortlog --numbered --summary | wc
 621    1799    12624                                # Sep 30, 2016
```



```
~/work/julia tags/v0.5.0 git shortlog --numbered --summary | wc
 614    1778    12478                                # Sep 19, 2016
```



```
~/work/julia tags/v0.4.0 git shortlog --numbered --summary | wc
 465    1358    9485                                # Oct 2015
```



```
~/work/julia tags/v0.3.0 git shortlog --numbered --summary | wc
 310     909    6339                                # Aug 2014
```



```
~/work/julia v0.2.0 git shortlog --numbered --summary | wc
 186     550    3799                                # Nov 2013
```



```
~/work/julia tags/v0.1 git shortlog --numbered --summary | wc
 112     339    2314                                # Feb 2013
```

~/work/julia master git shortlog --numbered -summary

7887 Jeff Bezanson
3605 Stefan Karpinski
2724 Viral B. Shah
2165 Jameson Nash
1420 Tony Kelman
1213 Keno Fischer
1199 Tim Holy
861 Yichao Yu
730 Andreas Noack Jensen
724 Mike Nolta
720 Jiahao Chen
683 Steven G. Johnson
632 Jake Bolewski
571 Carlo Baldassi
560 Elliot Saba
409 Katharine Hyatt
333 Amit Murthy
273 Isaiah Norton
262 Simon Kornblith

~/work/julia master git log -reverse

commit a9cbc036ac62dc5ba5200416ca7b40a2f9aa59ea

Author: Stefan Karpinski <stefan.karpinski@gmail.com>

Date: Sat Aug 22 20:39:06 2009 -0700

Initial empty commit.

commit eb256df11428c8ce63f6cb6ae0bc495645c6eec5

Author: Jeff Bezanson <bezanson@post.harvard.edu>

Date: Sun Aug 23 02:02:49 2009 -0400

beginning work on parser

commit c474155897f42940d76b3ef8087ab7749e7a3a6b

Author: Jeff Bezanson <bezanson@post.harvard.edu>

Date: Sun Aug 23 02:46:56 2009 -0400

adding basic [vector] and {list} syntax

commit 1565577e1041bb2dfe310d31b004ead3a906a1c3

Author: Jeff Bezanson <bezanson@post.harvard.edu>

Date: Sun Aug 23 22:23:02 2009 -0400

Julia 0.6

- * prompt pasting
- * <https://github.com/JuliaLang/julia/milestone/20>

Julia 1.0

- * 2017년
- * <https://github.com/JuliaLang/julia/milestone/4>

Julia 2.0

* <https://github.com/JuliaLang/julia/milestone/23>

참여하기

- Notes for Julia Contributors

<https://github.com/JuliaLang/julia/blob/master/CONTRIBUTING.md>

- julia doc 번역

<https://github.com/juliakorea/doc>

- julialang-web 번역

<https://www.transifex.com/julialang-i18n/julialang-web/>

참여하기

- Notes for Julia Contributors

<https://github.com/JuliaLang/julia/blob/master/CONTRIBUTING.md>

참여하기

- julia doc 번역

<https://github.com/juliakorea/doc>

- GitHub Flow에 대한 설명과 스쿼시, forked repository update하는 방법

https://www.youtube.com/watch?v=x-b_ij22vWg

참여하기

- julialang-web 번역

<https://www.transifex.com/julialang-i18n/julialang-web/>

- julialang-web : Languages - Korean

<https://www.transifex.com/julialang-i18n/julialang-web/language/ko/>

참여하기

- * Facebook Group
 - <https://www.facebook.com/groups/juliakorea/>
- * Slack
 - <http://juliakorea.slack.com/>

References

- * Julia <http://julialang.org/>
- * JuliaCon <http://juliacon.org/>
- * Packages
 - DataFrames.jl <https://github.com/JuliaStats/DataFrames.jl>
 - TensorFlow.jl <https://github.com/malmaud/TensorFlow.jl>
- * Project Jupyter <http://jupyter.org/>
- * Juno <http://junolab.org/>
- * John Myles White <http://www.johnmyleswhite.com/>
- * TensorFlow <https://www.tensorflow.org/>

۱۱

* ㄱㅅㅎㄴㄷ

* ㄱㅁㅅㄴㄷ

* ㄸㅋㅂㄹㅁㅊ