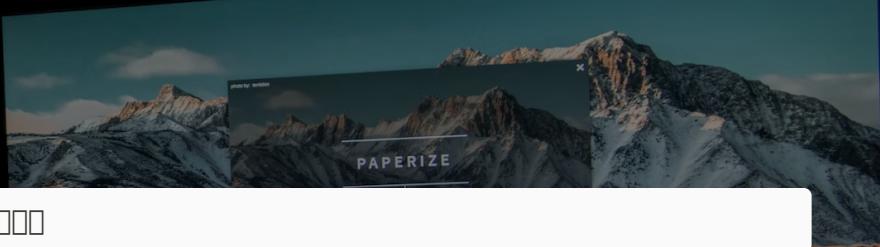


Tools For Developer - □□□□□□□



□□□□□□□□□□

As hackers, we spend a lot of time on our computers. So it makes sense to make that experience as fluid and frictionless as possible.

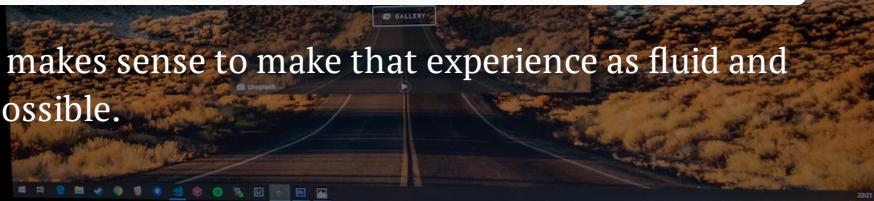


Table of contents

- 1. Editor
 - 1. Actions
 - 2. Moving & Selecting
 - 3. Font
- 2. Shell
- 3. Keys
- 4. Applications & WebSites
 - 1. Extensions
 - 2. Learning
 - 1. Nodejs & JavaScript
 - 2. CSS
 - 3. Tools
 - 4. Others
- 5. More



Editor

1. Actions
2. Moving & Selecting
3. Font

The image shows a MacBook Pro laptop screen with a Python application running in a code editor and a browser. The code editor window displays a file named `routes.py` containing the following code:

```
routes.py
routes.py
from flask import Flask, request
app = Flask('python-api-with-flask')
users = []

@app.route('/olamundo', methods=['GET'])
def olamundo():
    return "OK" + 200

@app.route('/cadastro_usuario', methods=['POST'])
def cadastrar_usuario():
    nome = request.json['name']
    usuarios.append(usuario)
    return {"usuario": usuario}

@app.route('/listar_usuarios', methods=['GET'])
def listar_usuarios():
    return {"usuarios": usuarios}

if __name__ == '__main__':
    app.run()
```

The browser window shows a REST API endpoint at `http://127.0.0.1:5000/listar_usuarios`. The response is a JSON object containing two user entries:

```
[{"usuario": {"id": 1, "name": "Logan", "email": "Logan"}, {"usuario": {"id": 2, "name": "Dennis", "email": "Dennis"}]}
```

The terminal window at the bottom shows the application is running on port 5000, and it lists several incoming requests from the local host (127.0.0.1) on port 49152.

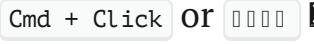
Actions

- Emmet (Zen Coding)

```
(div#emmet>p.bar*4>ul.nameblock>li*5.list)+footer>a[href=meituan.com]{Click me}
```

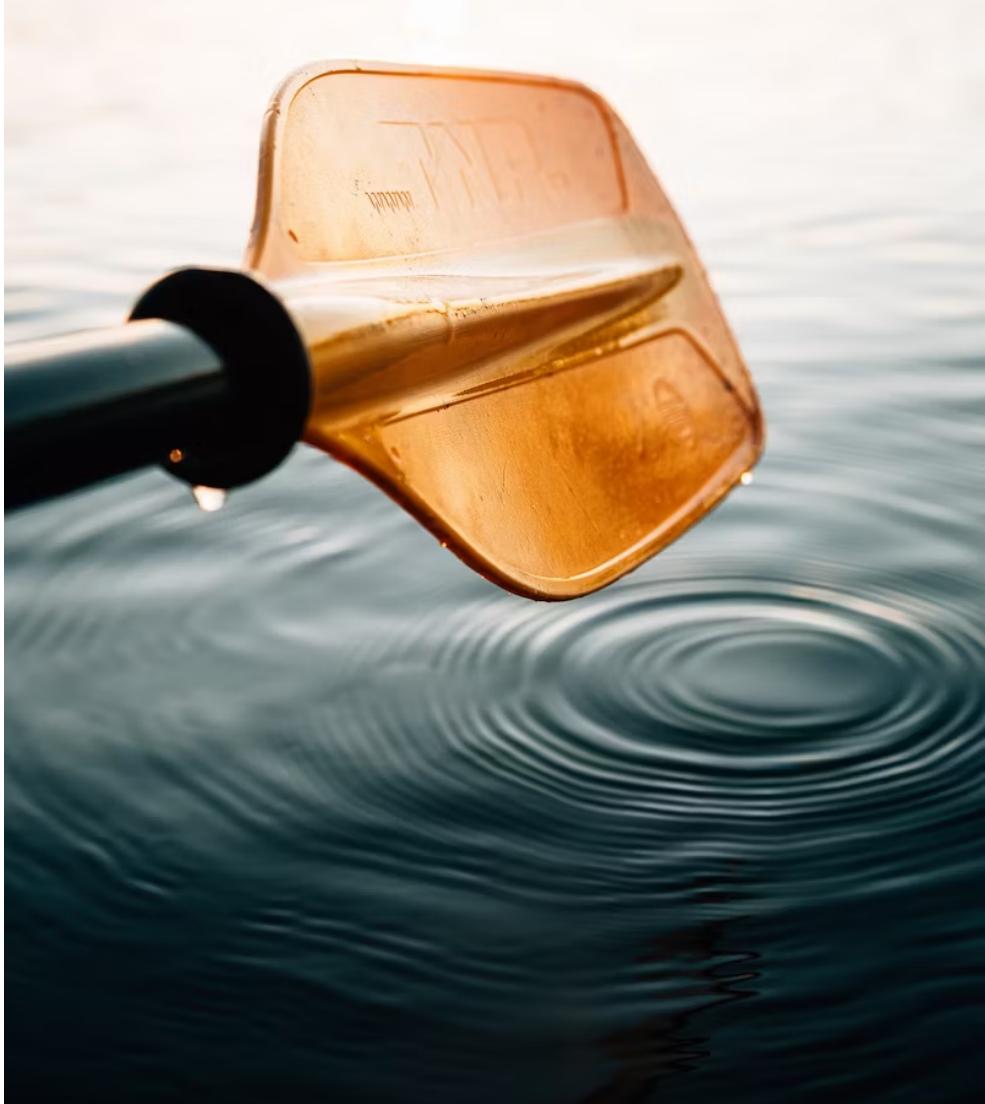
```
<style>
#emmet{
  m4-6
  w100p
  h100r
}
</style>
```

- Multi Cursor

- Cmd + d 
- Cmd + Click or 



```
export const urls = [
  { key: 'drivers', path: '/drivers' },
  { key: 'riders', path: '/riders' },
  {
    key: 'news',
    path: '/news'
  }
]
```

- Folding/Unfolding Block
 - `⌘ + ⌘ + [/ ⌘ + ⌘ +]`
- Parenthesis Matches / Matching Pair / Closiong Tag
 - `Ctrl + c`
- Highlight cursor column
- Go To Definition/References
- `⌘⌘:`
 - `Ctrl-d` `Meta-d`
 - Tips 

Moving & Selecting

- **XXXX:**
 - **XX:** Ctrl-f/b Meta-f/b
 - **XX:** Ctrl-a Ctrl-e
 - **XX shift** 𩫔𩫔𩫔
 - **XX:** Ctrl-n Ctrl-p
 - **XX:** Ctrl-v Meta-v
 - **XXXXXX:** Meta-g g
 - **XXXX:** Command-↑ Command-↓
 - Expand Region: Ctrl + w
 - 𩫔𩫔𩫔𩫔𩫔𩫔𩫔𩫔𩫔𩫔
- { Vim\Emacs Mode



Font

- monospace

ChatGPT - AI bot BETA
An AI bot writing helpful answers

Why are monospaced typefaces used for programming?

ChatGPT

Monospaced typefaces, also known as fixed-width or non-proportional typefaces, are used in programming because they ensure that each character takes up the same amount of horizontal space. This makes it easier for programmers to read and write code, as the indentation and alignment of characters is consistent, regardless of the letter or symbol. Additionally, monospaced typefaces make it easier to spot errors, such as missing or extra characters, in code.

-

- ProgrammingFonts
 - best-programming-fonts

- - ■ ■ ■ ■

FiraCode

Fira Code v6

ligatures: ON

Fira Mono

ligatures: NO

Common

Arithmetics	<code>++ -- += & && =</code>	<code>++ -- /= && =</code>
Scope	<code>→ ⇒ :: _</code>	<code>-> => :: _</code>
Equality	<code>= == ≠ ≠ == == ≠ ≠</code>	<code>= == != /= == == != !=</code>
Comparisons	<code>≤ ≥ ≤ ≥ ⇌</code>	<code><= >= <= >= <=</code>
Comments	<code>/* */ // ///</code>	<code>/* */ // ///</code>
Escaped chars	<code>\n \\</code>	<code>\n \\</code>
Bit operations	<code><< <<< <= >> >> = ^=</code>	<code><< <<< <= >> >> = ^=</code>
Hexadecimal Ex	<code>0xFF 1920x1080</code>	<code>0xFF 1920x1080</code>

Language-specific

JavaScript	<code>**</code> <code>==</code> <code>!=</code> <code>?.</code>	<code>**</code> <code>==</code> <code>!=</code> <code>?.</code>
HTML	<code></</code> <code><!--</code> <code></></code> <code>→</code> <code>/></code> <code>www</code>	<code></</code> <code><!--</code> <code></></code> <code>--></code> <code>/></code> <code>www</code>
Markdown	<code>#</code> <code>##</code> <code>###</code> <code>####</code> <code>#####</code> <code>#####</code>	<code>#</code> <code>##</code> <code>###</code> <code>####</code> <code>#####</code> <code>#####</code>
	<code>-</code> <code>--</code> <code>---</code> <code>—</code> <code>—</code> <code>—</code>	<code>-</code> <code>--</code> <code>---</code> <code>—</code> <code>—</code> <code>—</code>
	<code>=</code> <code>=</code> <code>==</code> <code>==</code> <code>==</code> <code>==</code>	<code>=</code> <code>=</code> <code>==</code> <code>==</code> <code>==</code> <code>==</code>
	<code> </code> <code> </code> <code> </code> <code> </code> <code> </code> <code> </code>	<code> </code> <code> </code> <code> </code> <code> </code> <code> </code> <code> </code>
Java	<code>◊</code> <code>↔</code>	<code><></code> <code>~></code>
C#	<code>?.</code> <code>??</code>	<code>?.</code> <code>??</code>
Ruby	<code>..</code> <code>...</code> <code>≈</code> <code>≠</code> <code>↔</code>	<code>..</code> <code>...</code> <code>=~</code> <code>!~</code> <code><=</code>
Go	<code>:=</code>	<code>:=</code>
Swift	<code>..<code></code></code>	<code>..<code></code></code>
Kotlin	<code>!!</code> <code>?.</code> <code>..</code>	<code>!!</code> <code>?.</code> <code>..</code>
R	<code>:::</code>	<code>:::</code>
Scala	<code>!=</code> <code>==</code> <code>:::</code> <code><:<</code>	<code>!=</code> <code>==</code> <code>:::</code> <code><:<</code>
Rust	<code>..=</code> <code>::<</code>	<code>..=</code> <code>::<</code>
Clojure	<code>#{} #_ #_(#? #: .- ;</code> <code>~@</code>	<code>#{} #_ #_(#? #: .- ;</code> <code>~@</code>
Elixir	<code>← → #{} ▷ ◊</code>	<code><- → #{} > <></code>
Haskell	<code>⇒⇒ ⇐⇒ ⇑⇒ ⇒⇒ ⇓⇒</code>	<code>=>> =<< >=> <=< =>= =<<</code>

Shell

- `|` operator lets you “chain” programs such that the output of one is the input of another:

```
cat package.json | grep lint  
ls | tail -n 3  
ls | head -n 3  
ls -rSl | tail -n 3 | pbcopy  
pbpaste | head -n1  
ll | grep nest
```

- `xargs` xargs translates arguments to commands. This effectively means you can run one command per argument, or forward all arguments to one command.

```
cd ~/work/base  
ls | code  
ls | xargs code -  
ls | xargs code  
ls | xargs -L1 code
```

- `>` overwirte output to a file:

```
curl wttr.in > test.txt  
curl cip.cc > test.txt  
cat test.txt
```

- `>>` append output to a file:

```
curl wttr.in >> test.txt  
curl cip.cc >> test.txt  
cat test.txt
```

- Always try to use RegExp.

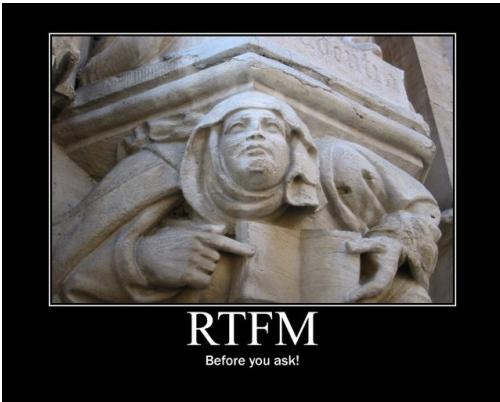
```
ls -l *.json
```

- alias

```
→ alias mnpm  
mnpm='npm --registry=http://r.npm.sankuai.com --cache=/Users/chaofeis/.cache/mnpm --disturl=http://npm.sankuai.com/mirro  
→ alias gp  
gp='git push'  
→ alias gma  
gma='git add . && gm'
```



- man



TL;DR

- tldr

```
+ ~ tldr tar  
  
tar  
  
Archiving utility.  
Often combined with a compression method, such as gzip or bzip2.  
More information: https://www.gnu.org/software/tar/.  
  
- [c]reate an archive from [f]iles:  
  tar cf target.tar file1 file2 file3  
  
- [c]reate a g[z]ipped archive from [f]iles:  
  tar czf target.tar.gz file1 file2 file3  
  
- [c]reate a g[z]ipped archive from a directory using relative paths:  
  tar czf target.tar.gz --directory=path/to/directory .  
  
- E[x]tract a (compressed) archive [f]ile into the current directory:  
  tar xf source.tar[.gz|.bz2|.xz]  
  
- E[x]tract a (compressed) archive [f]ile into the target directory:  
  tar xf source.tar[.gz|.bz2|.xz] --directory=directory  
  
- [c]reate a compressed archive from [f]iles, using [a]rchive suffix to determine the compression program:  
  tar caf target.tar.xz file1 file2 file3  
  
- Lis[t] the contents of a tar [f]ile [v]erbosely:  
  tar tvf source.tar  
  
- E[x]tract [f]iles matching a pattern:  
  tar xf source.tar --wildcards "*.*"
```

- oh-my-zsh

- change directory:

Command	Result
folder/	write the folder name with / at the end
...	go back one folder (parent dir)
...	go back two folders (parent from parent dir)
...	go back five folders
/	go to root
~	go to home
■ `zsh_stats`	Frequency statistic

- `TAB` for completion

```
ls -  
echo $
```

```
agkoz ~[zsh-z] (develop)  
[2] % █
```

- alias:

```
alias | grep git
```

- plugins\themes
- More about changing directory

- j
- z

07:50:54

```
1 [██████] 38.9% Tasks: 227 total, 5 running
2 [███] 13.9% Uptime: 28:38:19
3 [██████] 27.8% Battery: 98.8% (Running on A)
4 [███] 15.7% Mem [██████████] (2791/4896MB)
```

```
3:42 G-Eazy All I Could Do ft. Ski The Endless Summer
3:10 G-Eazy Acting Up ft. Devon Ba The Endless Summer
4:39 G-Eazy Run ft. Team Robot The Endless Summer
3:05 G-Eazy Outta Pocket RMX ft. S The Endless Summer
3:25 G-Eazy The Outsider The Outsider
3:34 G-Eazy Dear Ms. Rose (ft. Phe The Outsider
3:21 G-Eazy You Were Up To Somethi The Outsider
3:33 G-Eazy Outta Pocket The Outsider
3:19 G-Eazy Blazin On A Sunny Afte The Outsider
4:10 G-Eazy Wasp The Outsider
3:20 G-Eazy A Thing For Me The Outsider
3:44 G-Eazy Right Now The Outsider
2:26 G-Eazy Kings The Outsider
4:01 G-Eazy California Culture The Outsider
3:09 G-Eazy Netflix The Outsider
3:50 G-Eazy My Life Is a Party The Outsider
4:01 G-Eazy Nose Goes The Outsider
4:50 G-Eazy Hello Must Be Nice
6:07 G-Eazy Plastic Dreams (feat. Must Be Nice
6:22 G-Eazy Lady Killers (feat. Ho Must Be Nice
5:23 G-Eazy Mad (feat. Devon Baldwin Must Be Nice
2:53 G-Eazy Interlude Must Be Nice
5:31 G-Eazy Marilyn (feat. Dominic Must Be Nice
5:49 G-Eazy Stay High (feat. Mod S Must Be Nice
5:04 G-Eazy Breathe Must Be Nice
5:39 G-Eazy Must Be Nice (feat. Jo Must Be Nice
4:49 G-Eazy Loaded (feat. DJ Carna Must Be Nice
3:39 G-Eazy Endless Summer ft. Eri The Endless Summer
2:47 G-Eazy Hang Ten The Endless Summer
4:04 G-Eazy Runaround Sue ft. Greg The Endless Summer
2:54 G-Eazy Reefer Madness The Endless Summer
3:38 G-Eazy Make-Up Sex The Endless Summer
3:16 G-Eazy Well Known Ft. KAM Roy The Endless Summer
3:42 G-Eazy All I Could Do ft. Ski The Endless Summer
3:56 G-Eazy Wasp The Endless Summer
3:10 G-Eazy Acting Up ft. Devon Ba The Endless Summer
4:39 G-Eazy Run ft. Team Robot The Endless Summer
3:05 G-Eazy Outta Pocket RMX ft. S The Endless Summer
3:25 G-Eazy The Outsider The Outsider
3:34 G-Eazy Dear Ms. Rose (ft. Phe The Outsider
3:21 G-Eazy You Were Up To Somethi The Outsider
3:33 G-Eazy Outta Pocket The Outsider
```

```
3:38 #out civis
3:31 tput clear
3:32 while :
3:33 do
3:34 # New position:
3:35 ((\$1>2)) && ((x+($1==1)?1:-1))
3:36 ((($1>2))) && ((y+($1==2)?1:-1))
3:37
3:38 # Loop on edges (change color on loop):
3:39 ((x>(5+$v)) || $x<0 || $y>$h || $y<0)?($RANDOM%7):()
3:40 ((x>(5+$v)?0:(($x<0)?$v:$x)))
3:41 ((y>($y+$h)?0:(($y<0)?$h:$y)))
3:42
3:43 # Key random direction:
3:44 ((n=$RANDOM%5-1))
3:45 ((n>(5-1)|$n>=0)?$1:$1+$n)
3:46 ((n>(5-8)?3:$n%4))
3:47
3:48 # Print:
3:49 tput cup $y $x
3:50 echo "\033[1;${c}m\033[0m\033[0;${c}m"
3:51 ((t>$r)) && tput clear && t=0 || ((t++))
3:52 t=$n
3:53
3:54 # Wait for next frame:
```

Tmux Tips

38.1 93%

[1/101] >>>

Hostname: vpn3-144170.near.uiuc.edu
 Distro: OS X 10.9.1
 Kernel: Darwin
 Uptime: 28:27
 Shell: /bin/zsh
 Terminal: screen
 Packages:
 CPU:
 Memory: 4 GB
 Disk: 90%

>/tweet

Library	bin	home	private	var
Network	cores	item-colors	sbin	
System	dev	each_kernel	tsp	
Users	efi_changed	net	tweepy	



- `git log` + `git show COMMIT` -> tig
- serve
 - ███python -m SimpleHTTPServer 9000 -> python3 -m http.server 3000 -> serve
 - `npm install -g serve`



#LBNL

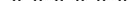
- Ctrl + r

Keys

- ChromeXXXXXX: Command + Shift + T

- □□□

- Cmd + Tab
 - Cmd + `
 - Ctrl + Up Ctrl + Down

-  Command + Shift + v

- **X** Meta/option/Alt □

-

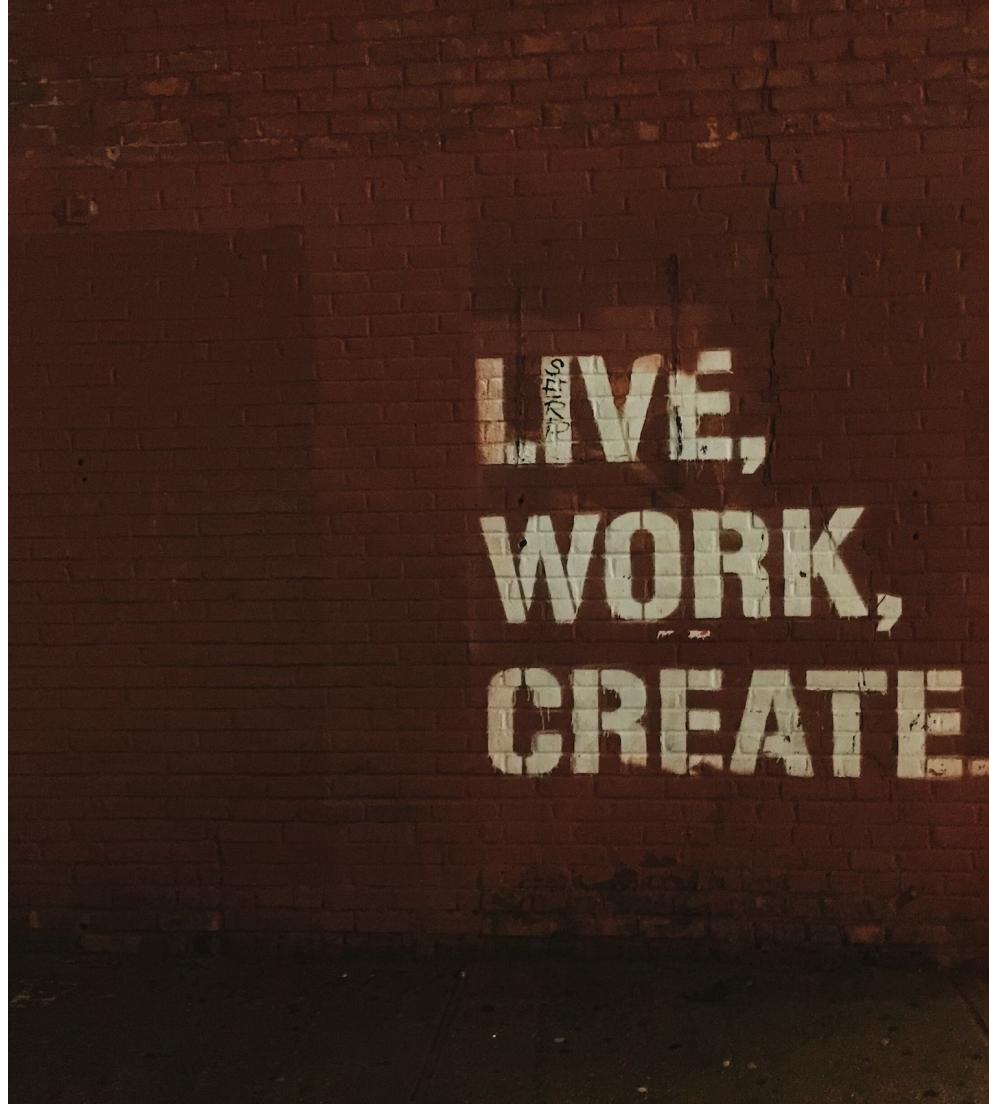
- ~~XXXXXXXXXX~~: Vim/Emacs □□

- **X X X X X** CapsLock and Ctrl



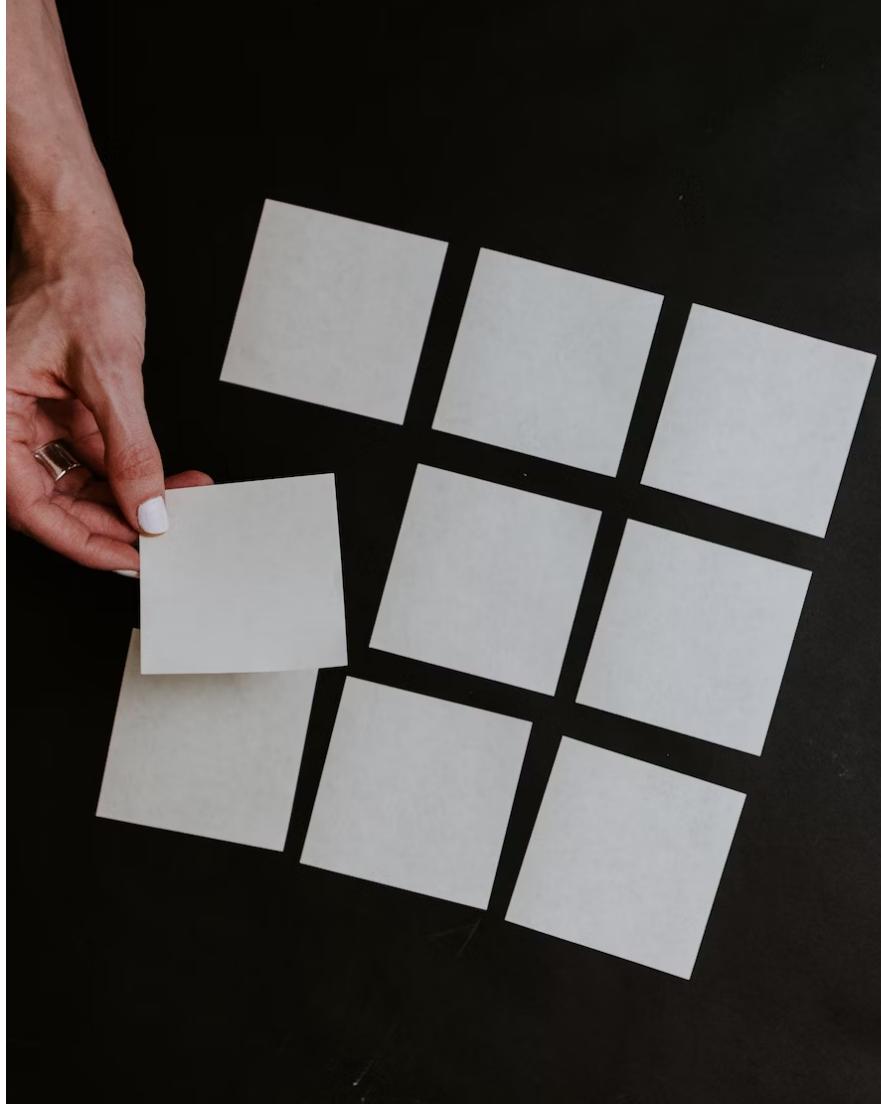
Applications & WebSites

- 1. Extensions
- 2. Learning
 - 1. Nodejs & JavaScript
 - 2. CSS
- 3. Tools
- 4. Others



Extensions

- OneTab `Cmd + e`
- GoogleTranslate/Youdao
- The great suspender ->
-
- OctoTree ->
- XXX Tab □□
- vimium
 - jk: □□□
 - gg G: □□□
 - HL: □□□
 - JK: □□□□



工具类

- ReRes
- AppMock for 安卓
- Charles
- whistle for Web

The screenshot shows the whistle for Web application running in a browser. The main window displays a table of network requests and responses. On the right side, there are two detailed panes: one for the selected request (https://www.google.com) showing Headers, Body, and Cookies; and another for the Response showing Headers, Preview, Body, and Cookies.

Network Requests Table:

#	Result	Method	Protocol	ServerIP	Host	URL	Type	Time
1	200	GET	H2	66.220.148.145	www.google.com	/complete/search?cl...	text/javascript	346ms
2	200	GET	H2	108.160.162.109	www.google.com	/complete/search?cl...	text/javascript	293ms
3	200	GET	H2	20.205.243.166	github.com	/	text/html	847ms
4	200	GET	H2	108.160.162.109	www.google.com	/complete/search?cl...	text/javascript	69ms
5	304	GET	H2	185.199.108.133	avatars.githubusercontent.c...	/u/11450939?s=40&...		70ms
6	200	GET	H2	142.250.204.110	encrypted-tbn0.gstatic.c...	/images?q=tbn:ANd...	image/png	220ms
7	200	GET	H2	108.160.162.109	www.google.com	/	text/html	354ms
8	200	GET	H2	108.160.162.109	www.google.com	/xjs/_/js/k=xjs.s.zh.B...	text/javascript	652ms
9	200	GET	H2	108.160.162.109	www.google.com	/images/branding/g...	image/png	37ms
10	200	GET	H2	203.208.40.66	www.gstatic.com	/og/_/js/k=og.qtmr....	text/javascript	620ms
11	204	POST	H2	108.160.162.109	www.google.com	/gen_204?i=webhp...	text/html	62ms
12	200	POST	H2	142.250.204.46	clients4.google.com	/chrome-sync/com...	application/vnd.goo...	249ms
13	200	POST	H2	113.96.237.96	doocr.weixin.qq.com	/report/node/	application/json	168ms
14	200	POST	H2	142.251.43.14	aa.google.com	/u/0/_/gog/get?rt=j&...	application/json	360ms
15	429	GET	H2	172.217.31.10	content-autofill.google...	/v1/pages/CHRDaHJ...	application/x-proto...	142ms
16	200	GET	H2	108.160.162.109	www.google.com	/complete/search?cl...	application/json	61ms
17	200	GET	H2	108.160.162.109	www.google.com	/xjs/_/js/k=xjs.s.zh.B...	text/javascript	470ms
18	204	GET	H2	108.160.162.109	www.google.com	/client_204?&tvp=...	text/html	43ms
19	200	GET	H2	108.160.162.109	www.google.com	/is/bp/NHksFvwOA...	text/javascript	31ms

Selected Request Headers:

host	www.google.com
sec-ch-dpr	2
sec-ch-viewport-width	1728
sec-ch-ua	"Not A;Brand";v="99", "Chromium";v="101", "Google Chrome";v="101"
sec-ch-ua-mobile	?0
sec-ch-ua-full-version	"101.0.4951.64"
sec-ch-ua-arch	"arm"
sec-ch-ua-platform	"macOS"

Selected Response Headers:

date	Fri, 20 May 2022 11:43:45 GMT
expires	-1
cache-control	private, max-age=0
content-type	text/html; charset=UTF-8
strict-transport-security	max-age=31536000
accept-ch	Sec-CH-Viewport-Width, Sec-CH-Viewport-Height, Sec-CH-DPR, Sec-CH-UA-Platform, Sec-CH-UA-Platform-Version, Sec-CH-UA-Full-Version, Sec-CH-UA-

Learning

- [CodeCademy](#)
- [Coursera](#)
- [FreeCodeCamp](#)
- [Udemy](#)
- [Khan Academy](#)
- [edX](#)
- [MIT The Missing Semester of Your CS Education](#)



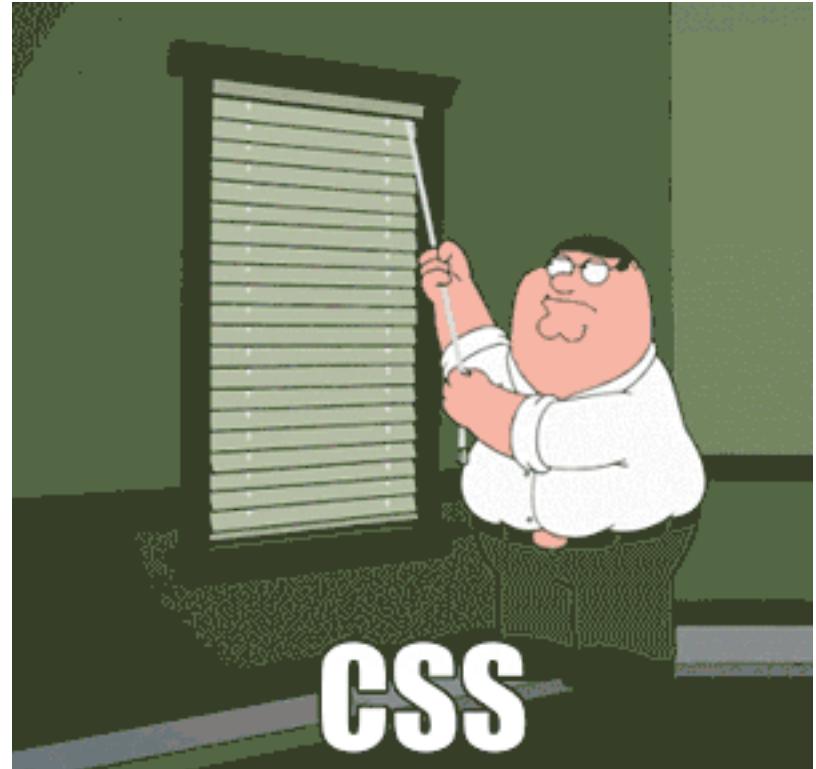
Nodejs & JavaScript

- <https://zh.javascript.info/>
 - <https://nodeschool.io/>
 - <https://pluralsight.com/paths/javascript-core-language>
 - Dr. Axel Rauschmayer <https://2ality.com/>
 - Dan Abramov <https://overreacted.io/>
 - Eric Elliott
https://medium.com/@_ericelliott
 - Wes Bos <https://wesbos.com>

```
Successed | 12/4/20 at 11:01 PM 67 + ↵
TaskController.swift
174 }
175
176 }
177 }
178 }
179 func requestBoards(){
180     TrelloServiceGet.sharedInstance.getBoardsRef { (res,
181         data) in
182     if res == nil {
183         let dataEx = data!
184         for (i, elem) in dataEx.enumerated(){
185             let initi = TrelloBoardModel(dict: elemData)
186             print("initi")
187             self.boardsArr.append(initi)
188             DispatchQueue.main.async {
189                 self.boardsCollectionView.delegate = self
190                 self.boardsCollectionView.dataSource =
191                     self
192                 self.boardsCollectionView.reloadData()
193             }
194         }
195     }
196 }
197 func requestListsFromBoards(id: String){
198     TrelloServiceGet.sharedInstance.getBoardRef(boardId: id)
199     { (res, data) in
200     if res == nil {
201         let dataEx = data!
202         for (i, elem) in dataEx.enumerated(){
203             DispatchQueue.main.async {
204                 self.listsCollectionView.delegate = self
205                 self.listsCollectionView.dataSource =
206                     self
207                 self.listsCollectionView.reloadData()
208             }
209             let initi = TrelloListModel(dict: elemData)
210             self.listsArr.append(initi)
211             print("list")
212             self.requestAllCardsFromList(id: initi.id)
213         }
214     }
215     var totalCards = 0
216     for list in self.listsArr{
217         totalCards += list.cardsCount
218     }
219 }
220 //
```

CSS

- [grid](#)
- [flexbox](#)
- [CSS Grid by codepen](#)
- <https://codepen.io/trending>
- <https://css-tricks.com/>
- CSS ~~UX~~ by codepen~~UX~~
 - [grid](#)
 - [flexbox](#)
- Design: csszengarden/dribbble/
- [tailwindcss](#)



Tools

- Warp: ~~AI~~ ~~AI~~ ~~AI~~
- ssh-copy-id: connect to hosts without any password
- ~~XXX/UML/XXX~~:
 - <https://www.processon.com>
 - <https://excalidraw.com>
 - <https://app.diagrams.net>
 - <https://www.omnigroup.com/omnigraffle>
 - ~~XXX~~ OmniGroup ~~AI~~ ~~AI~~ OmniFocus ~~AI~~ OmniPlan ~~AI~~
- ~~XXX~~
 - ~~AI~~ <https://naotu.baidu.com/>
 - xMind
 - MindMaster



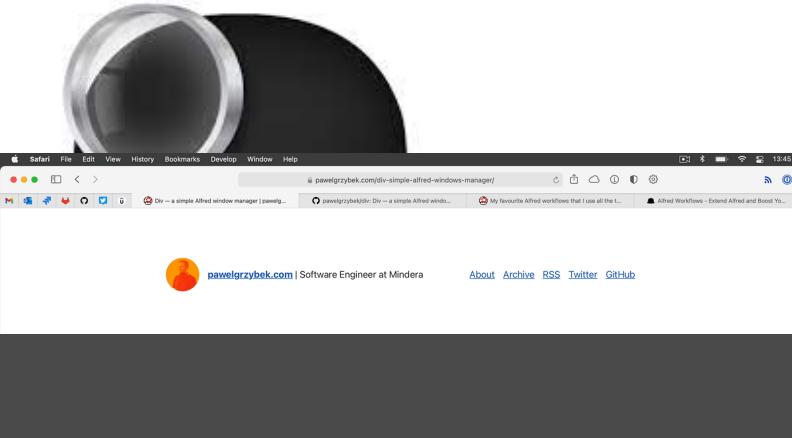
- Alfred

- Features:

- Quick Search
- Clipboard History

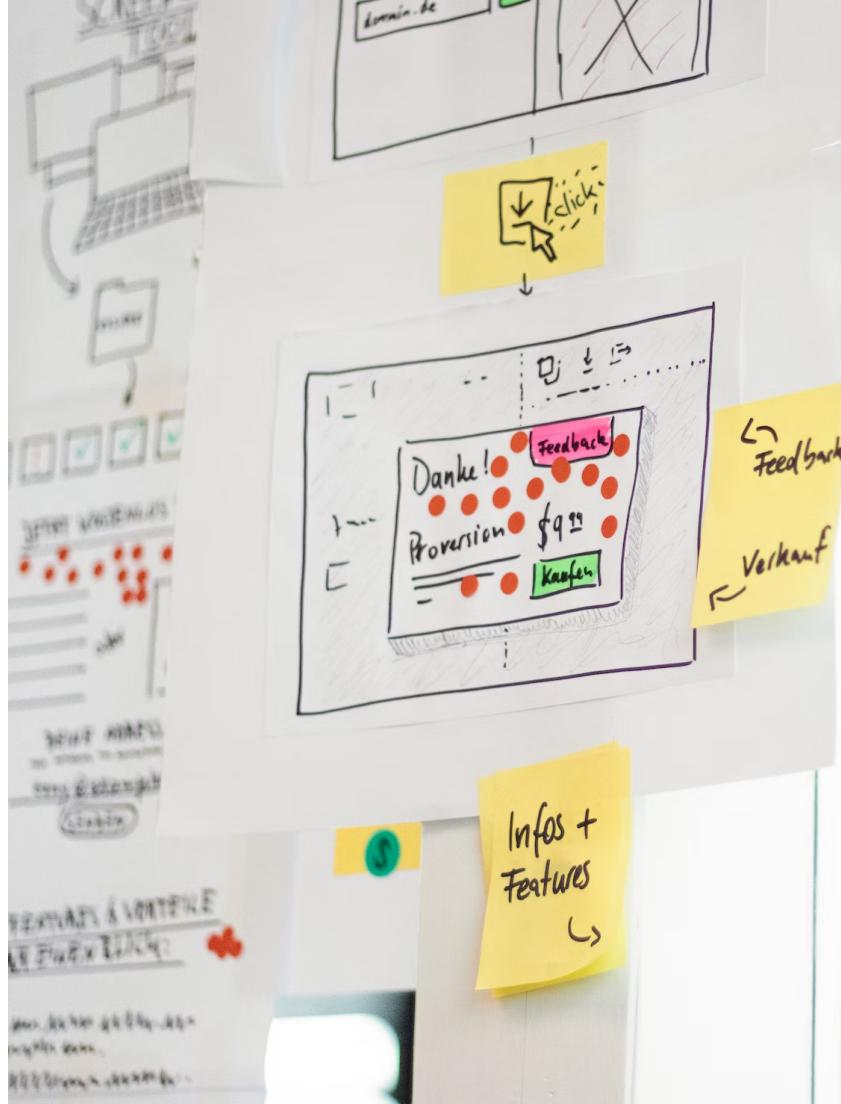
- Workflows:

- ⌘+SpaceFinder ↔ Iterm, div
- API⌘+Space mdn, httpcode
- ⌘+Shift+SpaceColors, ip
- More:
 - [awesome-alfred-workflows](#)
 - [alfred-workflows-that-i-cant-live-without](#)



- Tools

- The Programmer's Notebook: Quiver
- A note-taking tool for networked thought: Roam
 - https://roamresearch.com
- Your wiki, docs & projects. Together:
Notion
- ~~XXXXXX~~+FaaS ~~Github~~
Pages/Netlify/Vercel/Heroku
- Bookmarklet:
 - inject scripts ~~X~~ add\$, add_, addMoment,
angularWatchCount
 - my bookmarklets



Others

- [MagicWormhole](#)
- Hosts [SwitchHosts](#)
- [free ruler](#)
- [utools](#)
- [pap.er](#)
- [listen1](#)
- [Paste](#)
- [ccleaner](#)
- [snipaste](#)
- [sizeup/Magnet](#)
- [gifLICEcap](#)
- [diffKaleidoscope](#)
- [markdowntypora](#)
- [Jira/Lean/OminiFocus/BaseCamp](#)
- <http://jiumodiary.com>
- <http://xclient.info>
- [+](#)

More

- Blog☒
 - CSDN\yuque\medium\juejin
 - github pages
 - wordpress
 - hugo/hexo/astro
- PPT☒
 - keynote
 - slides.com
 - reveal.js
 - slidev

{ Markdown & MDX



- 代码审查
- 代码审查流程
- 三行
- 代码审查工具
- XXXXXX Commit XXX

Follow me on Twitter:



chaofeis
@chaofeis · [Follow](#)

回复 10 转发 1 收藏 1
<https://chafel.github.io/tools4developer>

6:54 AM · Jun 9, 2023

[Read 1 reply](#)

