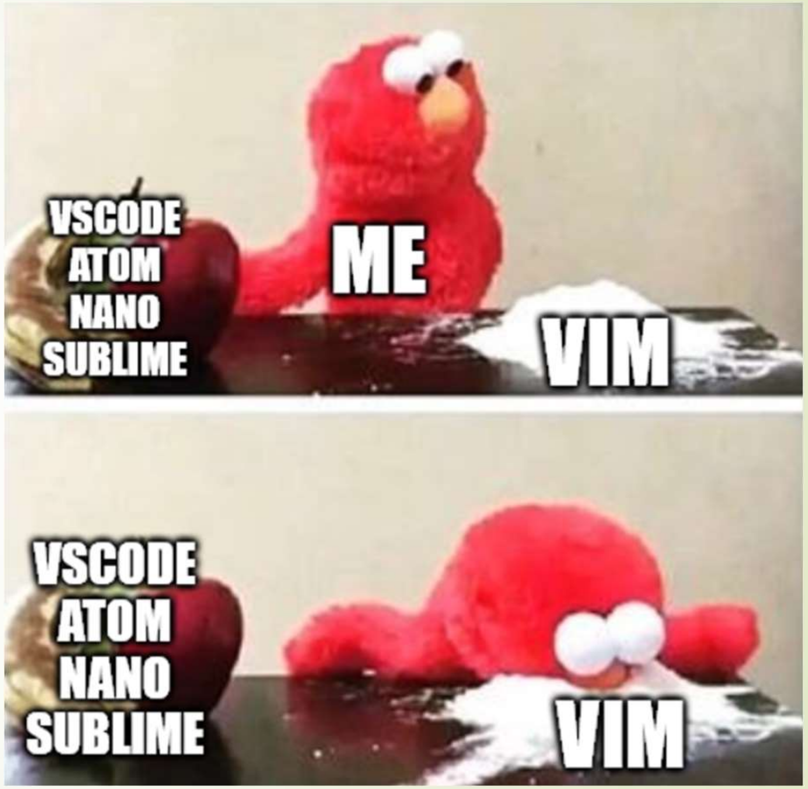
Vim go brrr

Victor Kuo



About me

Victor Kuo

Bitcoin for marks

- 5th Year
- Engineering (Space Mechatronics) + Science (Physics + Comp Sci)
- Email: victor.kuo1@sydney.edu.au
- Interned at Optiver (2019/20) and Atlassian (2020/21).
- Tutor INFO1110, ENGG1810, MTRX1705
- Languages: (In my order of preference!)
  - Python
  - HTML/CSS/Javascript/TypeScript (Web dev)
  - C++
  - C
  - Java
  - Assembly

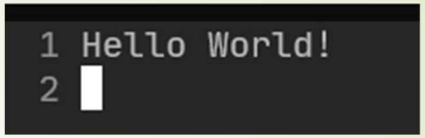
Feel free to ask me anything!

The schedule tonight…

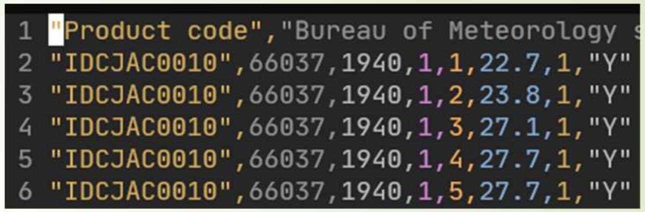
- What is Vim?
- Why use Vim?
- Vim crash course
- Tips to make Vim more useable!
  - Plugins
  - Computer settings
- Tmux
- Quality of life shell tweaks
- Caveats
- Closing

What is VIM?

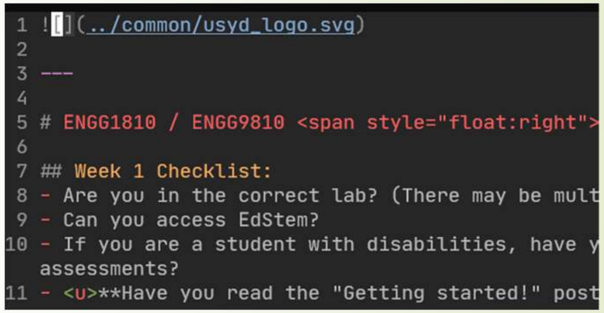
- A text editor!
  - Edits anything that is stored as raw text
  - E.g.
    - Text files
    - CSV files
    - Markdown files
    - Source code (no matter the language!)
    - Latex files
    - …
    - …
    - …



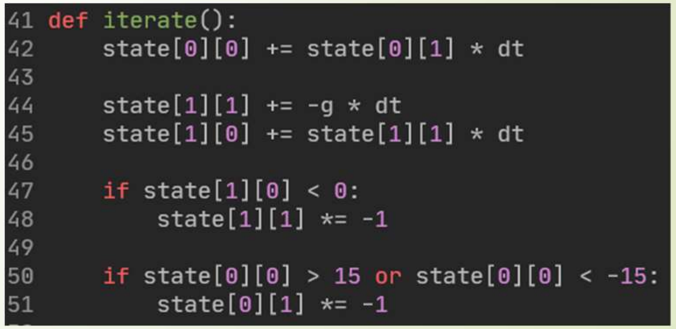
```
Hello World!

```



```
"Product code","Bureau of Meteorology s
"IDCJAC0010",66037,1940,1,1,22.7,1,"Y"
"IDCJAC0010",66037,1940,1,2,23.8,1,"Y"
"IDCJAC0010",66037,1940,1,3,27.1,1,"Y"
"IDCJAC0010",66037,1940,1,4,27.7,1,"Y"
"IDCJAC0010",66037,1940,1,5,27.7,1,"Y"
```



```
![](../common/usyd_logo.svg)

---

# ENGG1810 / ENGG9810 <span style="float:right">

## Week 1 Checklist:
- Are you in the correct lab? (There may be mult
- Can you access EdStem?
- If you are a student with disabilities, have y
assessments?
- <u>**Have you read the "Getting started!" post
```

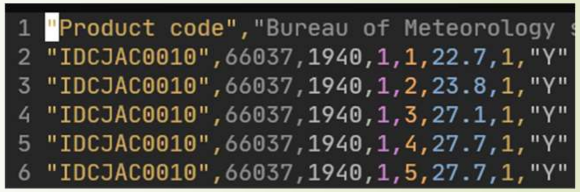


```python
def iterate():
    state[0][0] += state[0][1] * dt

    state[1][1] += -g * dt
    state[1][0] += state[1][1] * dt

    if state[1][0] < 0:
        state[1][1] *= -1

    if state[0][0] > 15 or state[0][0] < -15:
        state[0][1] *= -1
```

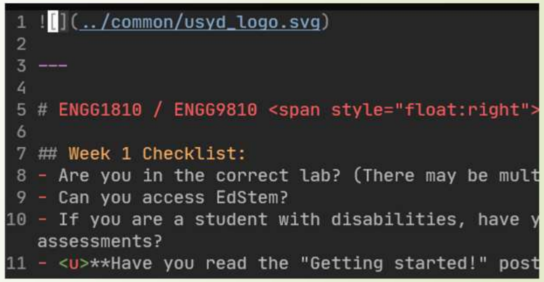
But what is a *text file*?

- Raw text.
  - Letters, numbers, symbols, etc.
  - No images, fancy colouring, groupings, etc.

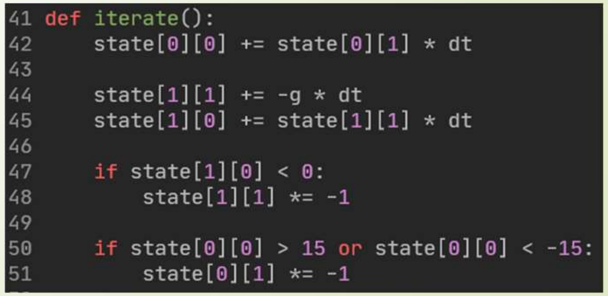
- What about binary files?



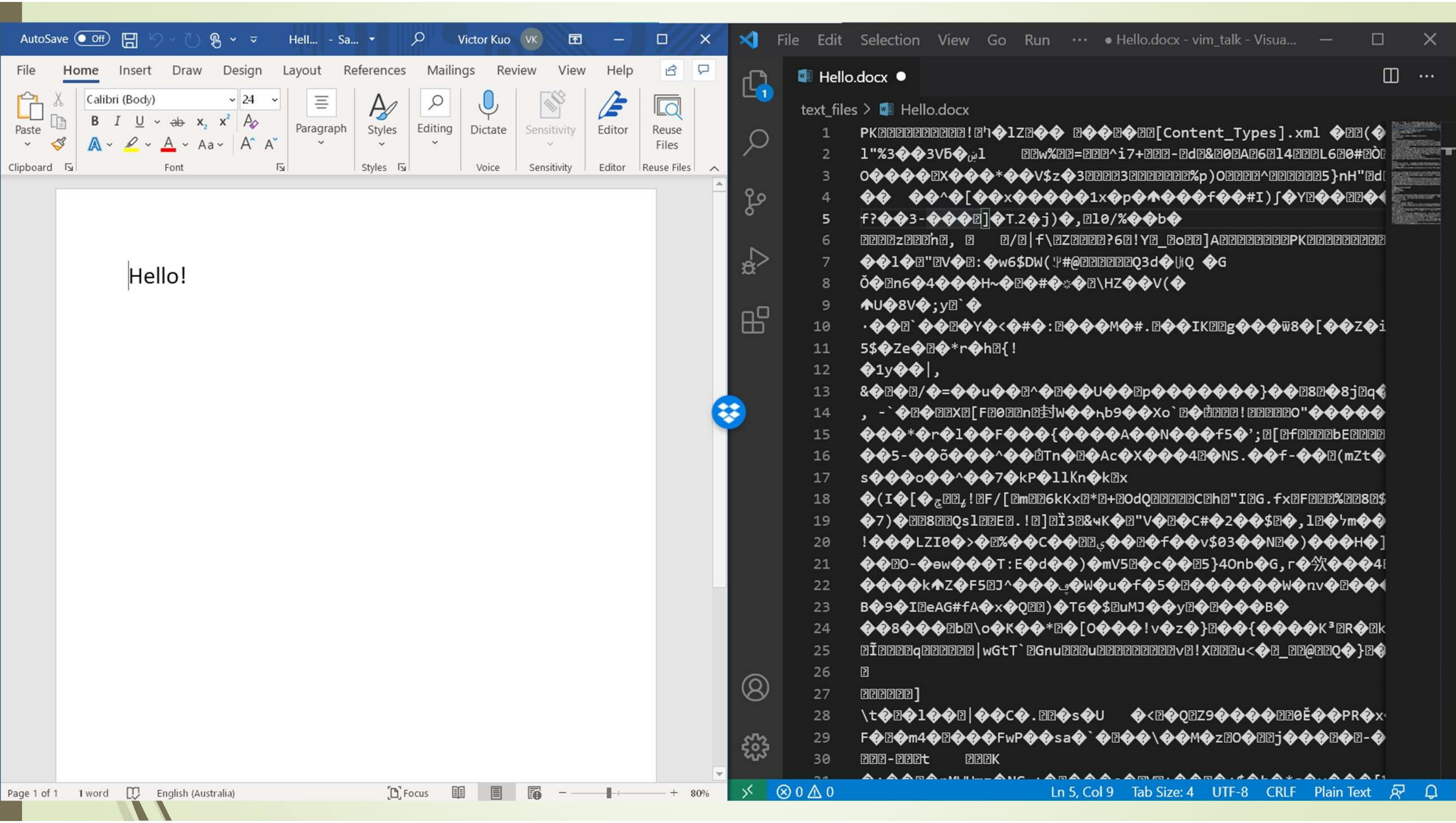
```
"Product code","Bureau of Meteorology s
"IDCJAC0010",66037,1940,1,1,22.7,1,"Y"
"IDCJAC0010",66037,1940,1,2,23.8,1,"Y"
"IDCJAC0010",66037,1940,1,3,27.1,1,"Y"
"IDCJAC0010",66037,1940,1,4,27.7,1,"Y"
"IDCJAC0010",66037,1940,1,5,27.7,1,"Y"
```



```
![](../common/usyd_logo.svg)

---

# ENGG1810 / ENGG9810 <span style="float:right">

## Week 1 Checklist:
- Are you in the correct lab? (There may be mult
- Can you access EdStem?
- If you are a student with disabilities, have y
assessments?
- <u>**Have you read the "Getting started!" post
```



```
def iterate():
    state[0][0] += state[0][1] * dt

    state[1][1] += -g * dt
    state[1][0] += state[1][1] * dt

    if state[1][0] < 0:
        state[1][1] *= -1

    if state[0][0] > 15 or state[0][0] < -15:
        state[0][1] *= -1
```



Hello!

```
PK□□□□□□□□□□□!□ๅ�1Z□�� □���□�□□[Content_Types].xml �□□(�
1"%3��3VБ�ش1    □□w%□□=□□□^i7+□□□-□d□&□0□A□6□14□□□L6□0#□Ò
O�����□X����*��V$z�3□□□□3□□□□□□□□%p)O□□□□^□□□□□□□5}nH"□d
��  ��^�[��x������1x�p�🡅����f��#I)ʃ�Y□��□□�
f?��3-����□]�T.2�j)�,□10/%��b�
□□□□z□□□n□, □   □/□|f\□Z□□□□?6□!Y□_□o□□]A□□□□□□□□□PK□□□□□□□□□
��1�□"□V�□:�w6$DW(ய#@□□□□□□□Q3d�ህQ �G
Ŏ�□n6�4���H~�□�#�҉�□\HZ��V(�
🡅U�8V�;y□`�
·���□`���□�Y�<�#�:□����M�#.□���IK□□g����ѿ8�[��Z�i
5$�Ze�□�*r�h□{!
�1y��|,
&�□�□/�=��u���□^�□��U��□p���������}���□8□�8j□q�
, -`�□�□□X□[F□0□□n□封W��ɦb9��Xo`□�□□□□!□□□□□O"������
����*�r�1��F����{�����A��N����f5�';□[□f□□□□bE□□□□
��5-��õ����^���□Tn�□�Ac�X����4□�NS.��f-���□(mZt�
s����o��^���7�kP�11Kn�k□x
�(I�[�ݹ□□ɫ!□F/[□m□□6kKx□*□+□OdQ□□□□□□C□h□"I□G.fx□F□□□%□□8□$
�7)�□□8□□Qs1□□E□.!□]□Ï3□&ԿK�□"V�□�C#�2��$□�,1□�ᢣm��
!����LZI0�>�□%��C��□□ﻱ���□�f��v$03��N□�)����H�]
��□O-�ɵw����T:E�d��)�mV5□�c��□5}4Onb�G,r�欸����4
�����k🡅Z�F5□J^����ڀ�W�u�f�5�□��������W�ոv�□���
B�9�I□eAG#fA�x�Q□□)�T6�$□uMJ��y□�□����B�
��8����□b□\o�Ꝁ��*□�[O����!v�z�}□���{�����K³□R�□k
□Ĩ□□□□q□□□□□□□|wGtT`□Gnu□□□u□□□□□□□□□□v□!X□□□u<�□_□□@□□Q�}□�
□
□□□□□□□]
\t�□�1���□|��C�.□□�s�U   �<□�Q□Z9������□□0Ě��PR�x
F�□�m4�□����FwP��sa�`�□��\��M�z□O�□□j�����□�□-�
□□□-□□□t    □□□K
```

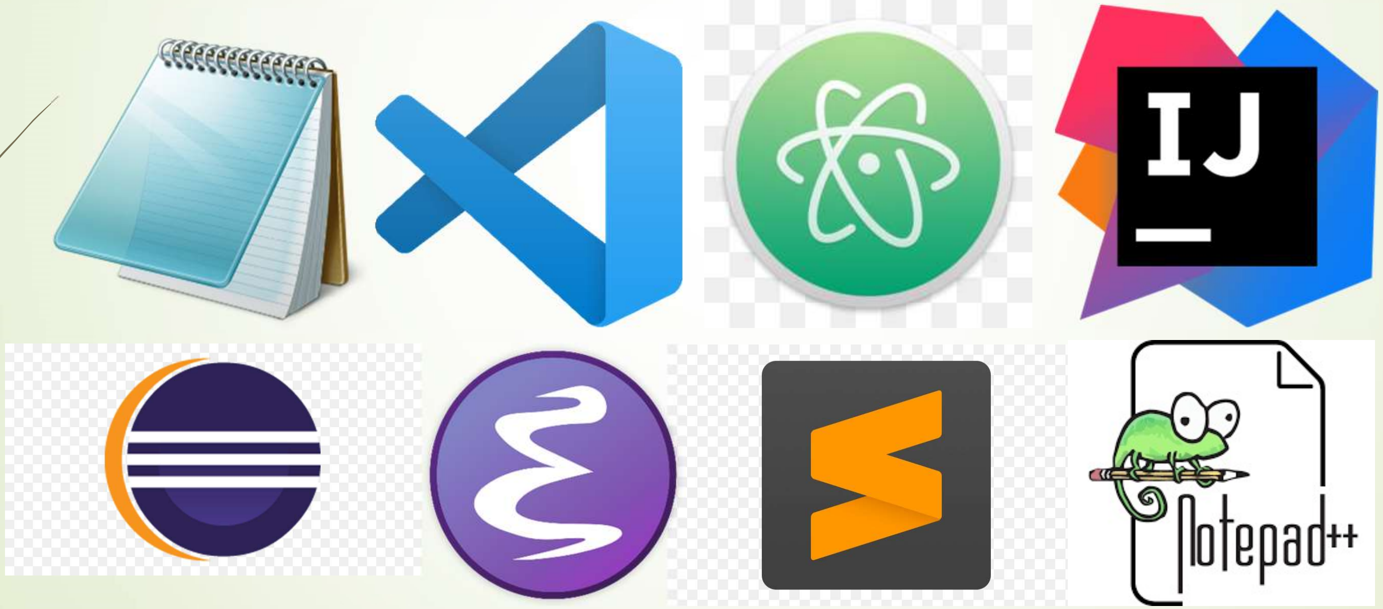
Why learn to edit *text* files when we have “better” software?

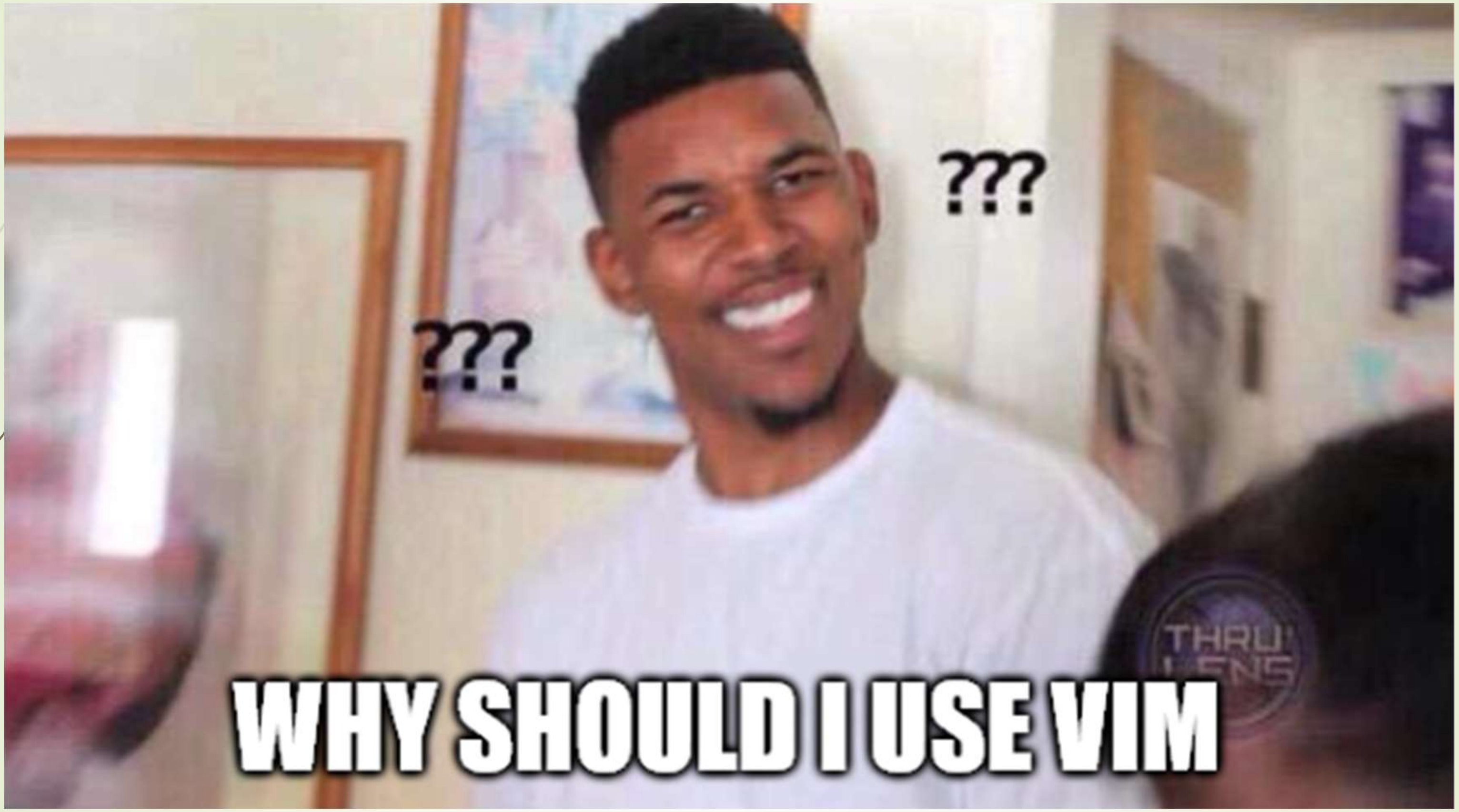
- Text files:
  - Will never disappear.
  - Fully standardised across the world.
  - Viewable and editable by any format abiding editor.

- Binary files:
  - Regularly change versions, format, styles, contents.
  - Hide away your actual contents in binary form.
  - You’re at the mercy of the software provider.
  - (This is not to say that binary files are bad, or that proprietary software is bad)

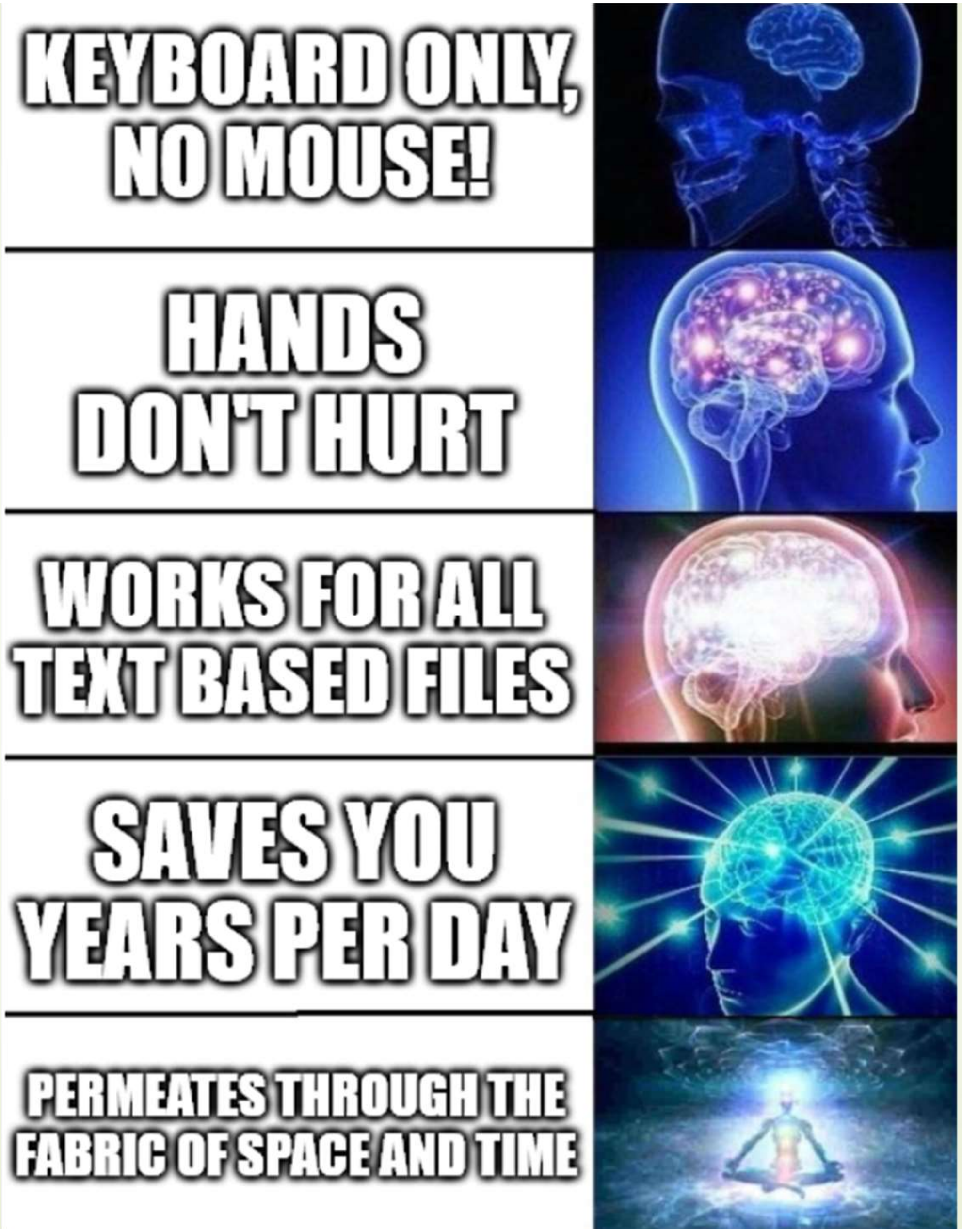
What is VIM?

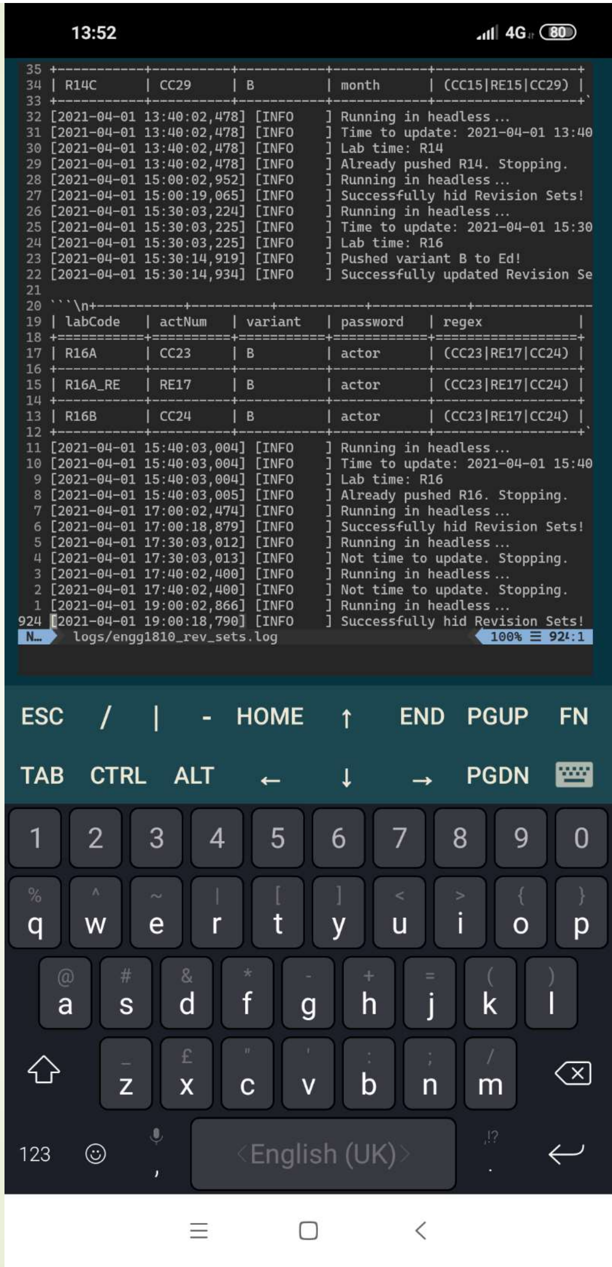
- A text editor!
- Essentially, the same function as:




???
???
THRU LENS
WHY SHOULD I USE VIM

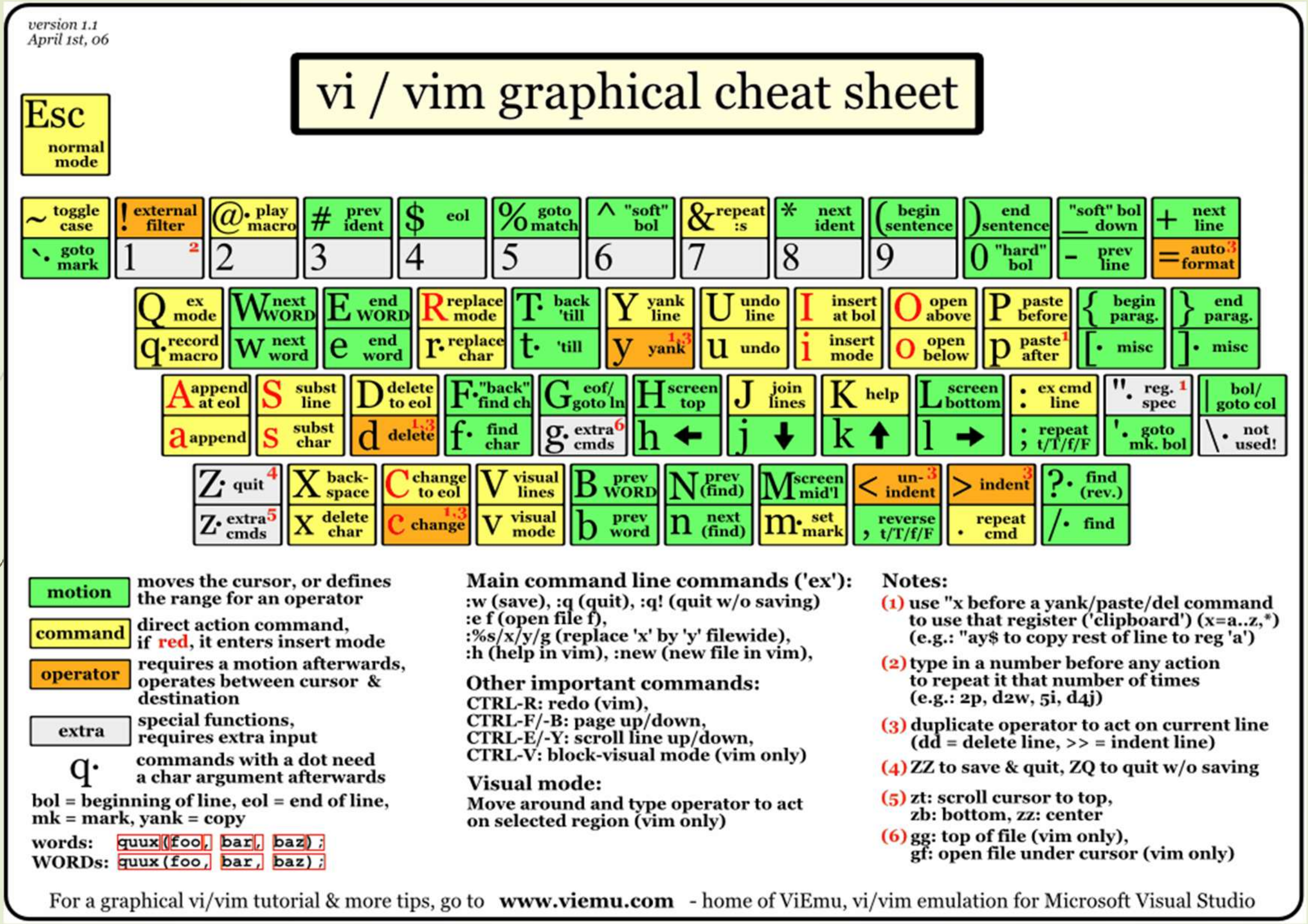
This one sparks joy.
Vim
This one does not spark joy.
Notepad++
IJ


KEYBOARD ONLY, NO MOUSE!
HANDS DON'T HURT
WORKS FOR ALL TEXT BASED FILES
SAVES YOU YEARS PER DAY
PERMEATES THROUGH THE FABRIC OF SPACE AND TIME



~~~
+------------+----------+------------+-------------+------------------+
| R14C       | CC29     | B          | month       | (CC15|RE15|CC29) |
+------------+----------+------------+-------------+------------------+`
[2021-04-01 13:40:02,478] [INFO    ] Running in headless...
[2021-04-01 13:40:02,478] [INFO    ] Time to update: 2021-04-01 13:40
[2021-04-01 13:40:02,478] [INFO    ] Lab time: R14
[2021-04-01 13:40:02,478] [INFO    ] Already pushed R14. Stopping.
[2021-04-01 15:00:02,952] [INFO    ] Running in headless...
[2021-04-01 15:00:19,065] [INFO    ] Successfully hid Revision Sets!
[2021-04-01 15:30:03,224] [INFO    ] Running in headless...
[2021-04-01 15:30:03,225] [INFO    ] Time to update: 2021-04-01 15:30
[2021-04-01 15:30:03,225] [INFO    ] Lab time: R16
[2021-04-01 15:30:14,919] [INFO    ] Pushed variant B to Ed!
[2021-04-01 15:30:14,934] [INFO    ] Successfully updated Revision Se

```\n+------------+----------+------------+-------------+-------------------
| labCode    | actNum   | variant    | password    | regex            |
+============+==========+============+=============+==================+
| R16A       | CC23     | B          | actor       | (CC23|RE17|CC24) |
+------------+----------+------------+-------------+------------------+
| R16A_RE    | RE17     | B          | actor       | (CC23|RE17|CC24) |
+------------+----------+------------+-------------+------------------+
| R16B       | CC24     | B          | actor       | (CC23|RE17|CC24) |
+------------+----------+------------+-------------+------------------+`
[2021-04-01 15:40:03,004] [INFO    ] Running in headless...
[2021-04-01 15:40:03,004] [INFO    ] Time to update: 2021-04-01 15:40
[2021-04-01 15:40:03,004] [INFO    ] Lab time: R16
[2021-04-01 15:40:03,005] [INFO    ] Already pushed R16. Stopping.
[2021-04-01 17:00:02,474] [INFO    ] Running in headless...
[2021-04-01 17:00:18,879] [INFO    ] Successfully hid Revision Sets!
[2021-04-01 17:30:03,012] [INFO    ] Running in headless...
[2021-04-01 17:30:03,013] [INFO    ] Not time to update. Stopping.
[2021-04-01 17:40:02,400] [INFO    ] Running in headless...
[2021-04-01 17:40:02,400] [INFO    ] Not time to update. Stopping.
[2021-04-01 19:00:02,866] [INFO    ] Running in headless...
[2021-04-01 19:00:18,790] [INFO    ] Successfully hid Revision Sets!
~~~

Let’s get started!



vi / vim graphical cheat sheet

🟩 motion	moves the cursor, or defines the range for an operator
🟨 command	direct action command, if red, it enters insert mode
🟧 operator	requires a motion afterwards, operates between cursor & destination
⬜ extra	special functions, requires extra input
q·	commands with a dot need a char argument afterwards

bol = beginning of line, eol = end of line,
mk = mark, yank = copy

words: quux(foo, bar, baz);
WORDs: quux(foo, bar, baz);

Main command line commands ('ex'):
:w (save), :q (quit), :q! (quit w/o saving)
:e f (open file f),
:%s/x/y/g (replace 'x' by 'y' filewide),
:h (help in vim), :new (new file in vim),

Other important commands:
CTRL-R: redo (vim),
CTRL-F/-B: page up/down,
CTRL-E/-Y: scroll line up/down,
CTRL-V: block-visual mode (vim only)

Visual mode:
Move around and type operator to act on selected region (vim only)

Notes:

(1) use "x before a yank/paste/del command to use that register ('clipboard') (x=a..z,*) (e.g.: "ay$ to copy rest of line to reg 'a')

(2) type in a number before any action to repeat it that number of times (e.g.: 2p, d2w, 5i, d4j)

(3) duplicate operator to act on current line (dd = delete line, >> = indent line)

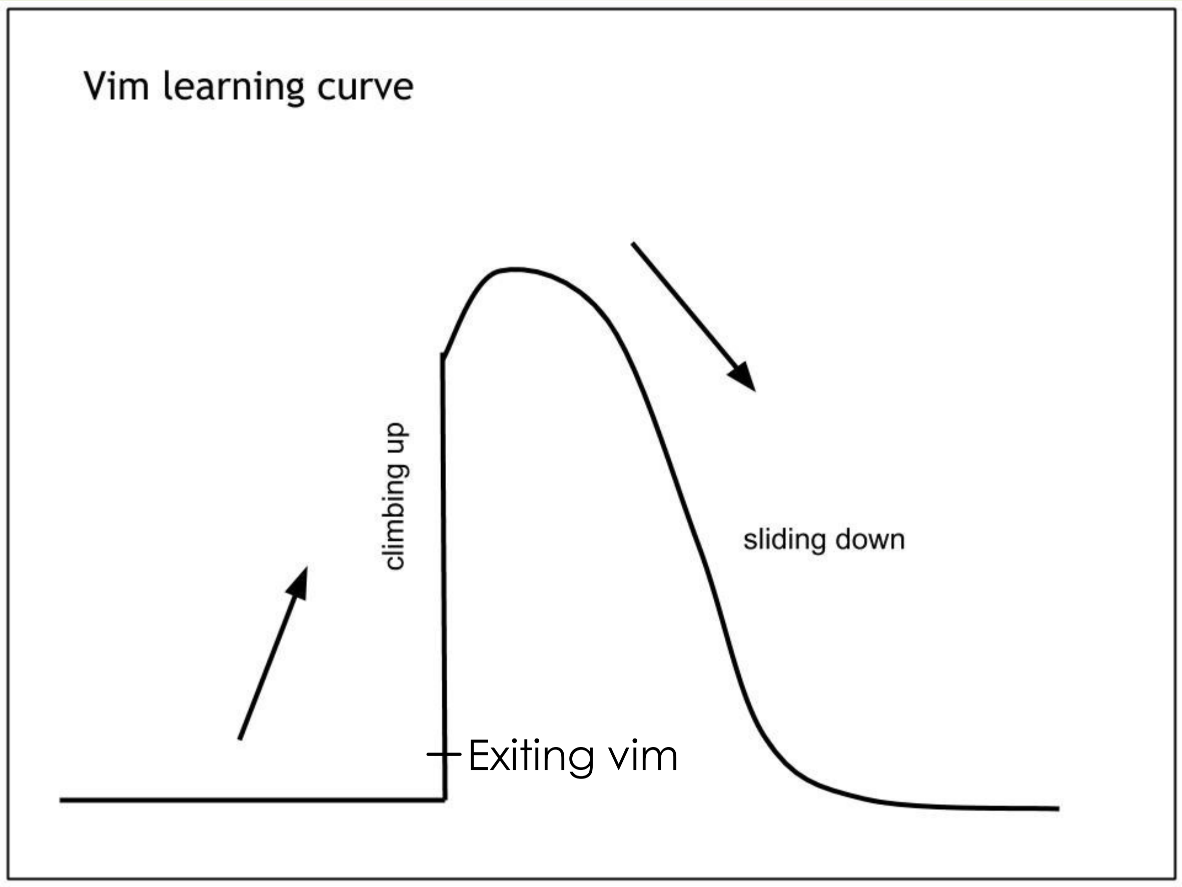
(4) ZZ to save & quit, ZQ to quit w/o saving

(5) zt: scroll cursor to top, zb: bottom, zz: center

(6) gg: top of file (vim only), gf: open file under cursor (vim only)

For a graphical vi/vim tutorial & more tips, go to **www.viemu.com** - home of ViEmu, vi/vim emulation for Microsoft Visual Studio

http://www.viemu.com/vi-vim-cheat-sheet.gif

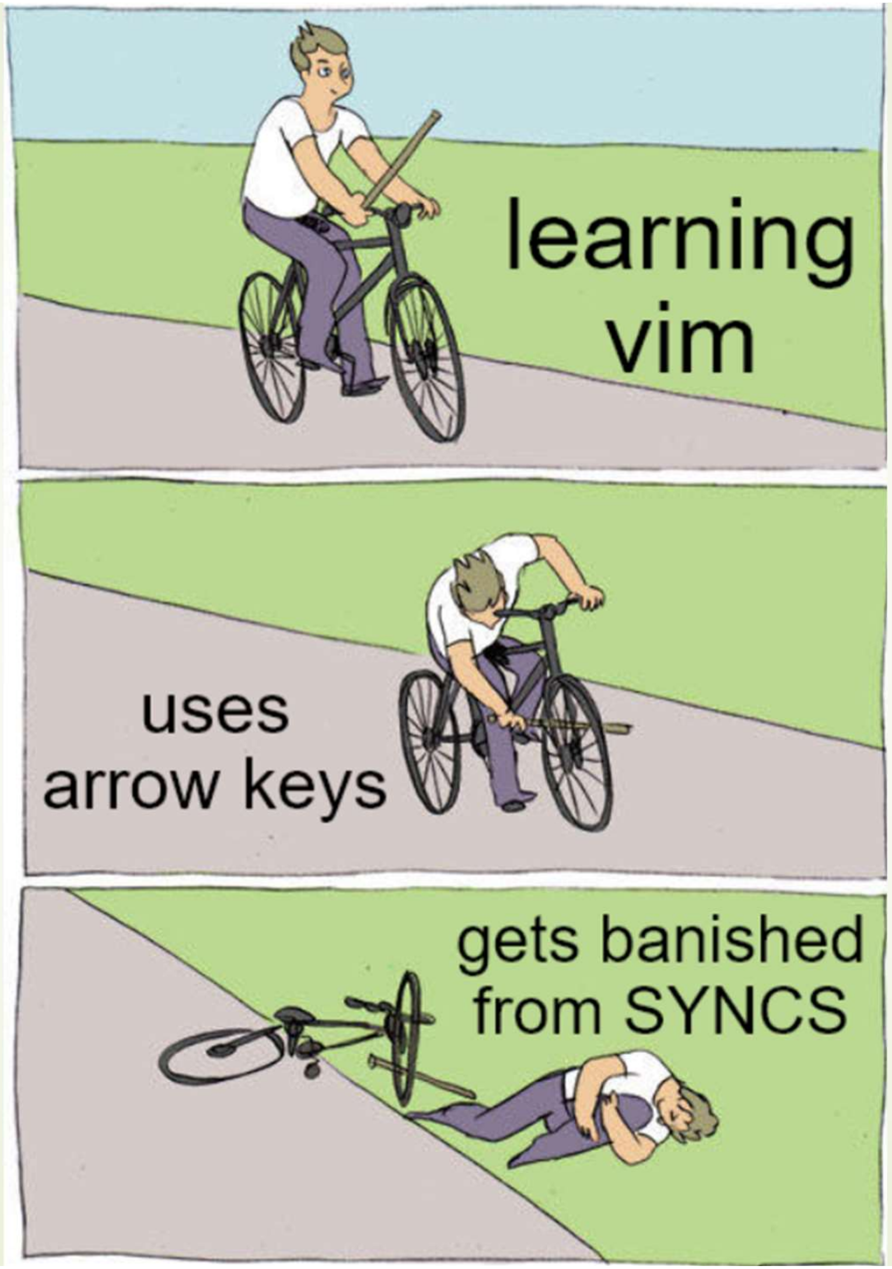

Vim learning curve
climbing up
sliding down
Exiting vim

The goal tonight

How to get the most out of this talk…

- Sit back, relax, and enjoy!
- The goal is to give you guys exposure, to let you know what is out there.
- Feel free to download these slides afterwards and try it out on your own ☺.
- Taking notes or following along will drain you too quickly and you'll miss what is really important! (imo at least)

Let’s actually get started…!


learning vim
uses arrow keys
gets banished from SYNCS

A great way to practise… Vim-be-good

5 minute break!

- First 5 people can try VIM-BE-GOOD!

More complex Vim…

5 minute break!

- Next 5 people can try VIM-BE-GOOD!

Tips before learning Vim yourself

1. Remap caps lock to be ctrl
   - Your fingers will thank you for this!
2. Remap leader to space
   - Leader is the prefix to a lot of your user defined commands.
3. Increase the key repeat rate
   - You don’t want to be here for years, right?
4. Decrease the key repeat delay (to start the repeat)
   - Same as above… lol.
5. Don’t get banished from SYNCS (unbind the arrow keys if you’re tempted!

Plugins

- Recommended:
  - Plug 'junegunn/fzf', { 'do': { -> fzf#install() } } " fuzzy find
  - Plug 'junegunn/fzf.vim'
  - Plug 'morhetz/gruvbox'
  - Plug 'tpope/vim-commentary'
  - Plug 'preservim/nerdtree'
  - Plug 'tpope/vim-surround'
  - Plug 'jiangmiao/auto-pairs'

  - <your language syntax highlighter>
  - <Intellisense recommend COC>

Other niceties…

Vim & Tmux

EFFICIENCY
VIM
TMUX

Zsh

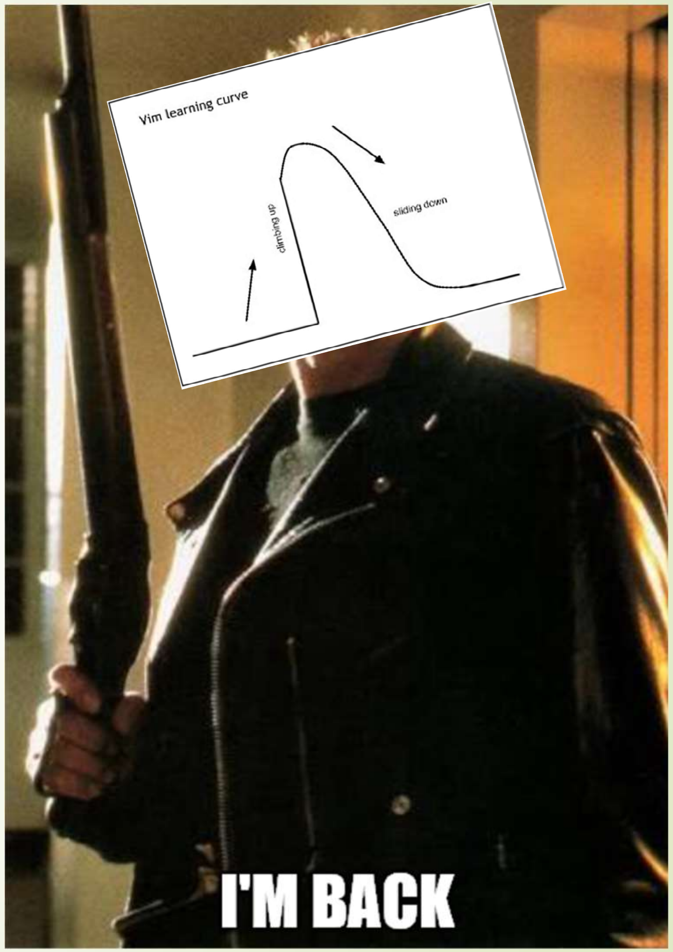
- A better shell.
- Works on Linux, Mac, Windows (WSL)…!

- Autocomplete
- Git shortcuts

Vim in the browser!?!? (Vimium)

Caveats to using Vim

- Live debugging is hard… but there are other ways around it…
- Installing your vim plugins and config on a new machine takes a bit longer than installing a single VSCode installer.
- Massive productivity drop at the start.
  - Learning all the basic shortcuts takes time…
  - Learning all the plugins takes time…
  - Configuring your new keymaps takes time…



Closing

- Vim is HARD to get started with.
  - But it's a long term investment on your hands (and girlfriend 😉).
  - Always look for easier ways to do things.
- Once you see the light, once you come out of the other side of the tunnel, you will emerge a new, woke coder.

Further study

- `vimtutor`
- This talk’s recording
  - Repo: https://github.com/vck3000/vim_talk_v1
- The Primeagen
- https://vim-adventures.com/
- Vim talk v2 ??

Thank you for listening!

- Q&A time