

Tmux & More on Vim

And pretend that vscode doesn't exist

By: Amy Liu

A GPI extratation

This extratation is

••••

- A chill session
 - Show you some cool, time-saving tricks (commands) in vim ~~and convince you to love vim~~
 - Introduce tmux for the ultimate multi-tasking
-

What we have in store today

Vim

- Set mouse & .vimrc
- “I’m too lazy to type it out”
 - Abbreviations
 - Autocomplete
- Copy pasting ... but better
 - move & copy
 - Registers
- More magical stuff
 - Record-your-own commands!
 - Capture groups
 - Splitting windows
 - Vim as a calculator
- Running commands

Tmux

- Attaching & detaching
- Split screen
- Customization

What we have in store today

Vim

- Set mouse & .vimrc
- “I’m too lazy to type it out”
 - Abbreviations
 - Autocomplete
- Copy pasting ... but better
 - move & copy
 - Registers
- More magical stuff
 - Record-your-own commands!
 - Capture groups
 - Splitting windows
 - Vim as a calculator
- Running commands

Tmux

- Attaching & detaching
- Split screen
- Customization

Set Mouse

```
set mouse=a
```

Copy paste this into `~/.vimrc`

This will let your cursor be wherever your mouse clicks

What is `~/.vimrc` ?

It's a settings / configuration file. You can also change color themes!

<https://mediatemple.net/community/products/dv/204644480/enabling-vi-syntax-colors-and-highlighting>

What we have in store today

Vim

- Set mouse & .vimrc
- “I’m too lazy to type it out”
 - Abbreviations
 - Autocomplete
- Copy pasting ... but better
 - move & copy
 - Registers
- More magical stuff
 - Record-your-own commands!
 - Capture groups
 - Splitting windows
 - Vim as a calculator
- Running commands

Tmux

- Attaching & detaching
- Split screen
- Customization

Autocomplete

!! In insert mode ONLY!!

Ctrl-N to invoke **Ctrl-P** to select if
there are multiple choices

Abbreviations

:ab abbrev WhatYouWantToAbbrev

Ctrl-V to exempt

Works across split windows

What we have in store today

Vim

- Set mouse & .vimrc
- “I’m too lazy to type it out”
 - Abbreviations
 - Autocomplete
 - Copy pasting ... but better
 - move & copy
 - Registers
- More magical stuff
 - Record-your-own commands!
 - Capture groups
 - Splitting windows
 - Vim as a calculator
- Running commands

Tmux

- Attaching & detaching
- Split screen
- Customization

Copy Pasta ... but better

:3copy5 - to copy line **3** below line **5**

:2move7 - to move* line **7** with line **2**

:move +1 to move* the line **1** line ahead

:move -2 to move* the line **1** line back

Registers

- Kind of like named slots (<register character>) that you can store stuff
- In normal-mode/command-mode: accessed as “<register character>
 - + **y** or **p** to store or paste from
 - ex. “**wy** (**yank** what’s currently highlighted into **register w**)
- In insert-mode: **CTRL-R** <register character> to get the content of a register at your cursor
- **@**<register character> to invoke a recorded command (more on this later!)
- **:reg** to see what’s in the registers in one go

Special Registers

- “” unnamed register (any d, c, s, x, y, p commands’ text automatically associated with this register if you don’t specify one)
- “0 ~ “9 numbered registers
 - “0 has your last yank
 - “1 ~ “9 has your last 9 deleted text from newest to oldest
- “_ Black hole register
- Read only registers (can’t modify them)
 - “. last inserted text
 - “% current file path
 - “: last executed command (usually used as @:)
 - “# name of the last edited file

Special Registers

- And there's mooooooore, but tldr,
- “a to “z are your 26 named registers that you can freely store stuff into

What we have in store today

Vim

- Set mouse & .vimrc
- “I’m too lazy to type it out”
 - Abbreviations
 - Autocomplete
- Copy pasting ... but better
 - move & copy
 - Registers
- **More magical stuff**
 - Record-your-own commands!
 - Capture groups
 - Splitting windows
 - Vim as a calculator
- Running commands

Tmux

- Attaching & detaching
- Split screen
- Customization

Record-your-own Macros!

q then **<register character>** type your commands, **q** again to stop recording

@<register character> to invoke

How to access/view another file simultaneously? How to “split screen”?

Using your phone
to take a picture
of the other file

Opening a new
terminal/ ssh/
computer/ tab
to access the
other file

In vim,
:o
:O
:vs
:split

??????????????



H ar si H

-o to open files hamburgered
-O to open files hotdogged

:vs for vertical split
:split for horizontal split

Ctrl+w

h/j/k/l to travel
w to the next one

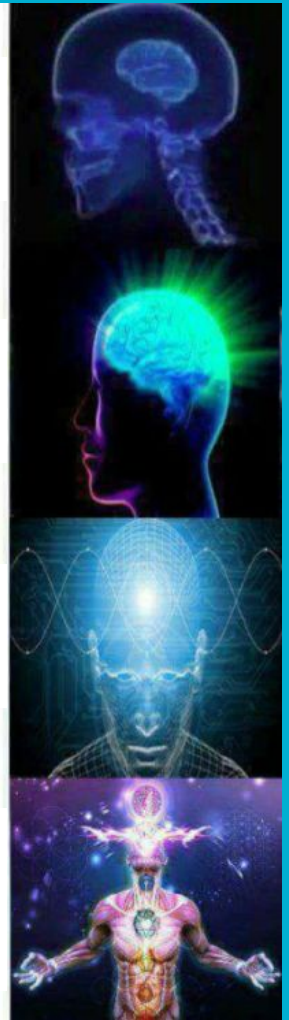
Using your phone
to take a picture
of the other file

Opening a new
terminal/ ssh/
computer/ tab
to access the
other file

In vim,

:o
:O
:vs
:split

??????????????



Calculator

!! In insert mode ONLY!! (“= is a special register!)

Ctrl-R then **=** then **your math expression**

More than just a Calculator

!! In insert mode ONLY!! (“= is a special register!)

Ctrl-R then **=** then **your math expression**
system(‘command’)

Run your code!

Type **:! <run command>**

Example: **:! echo "hello world"**

You will see the terminal and the output, press **ENTER** to come back to vim

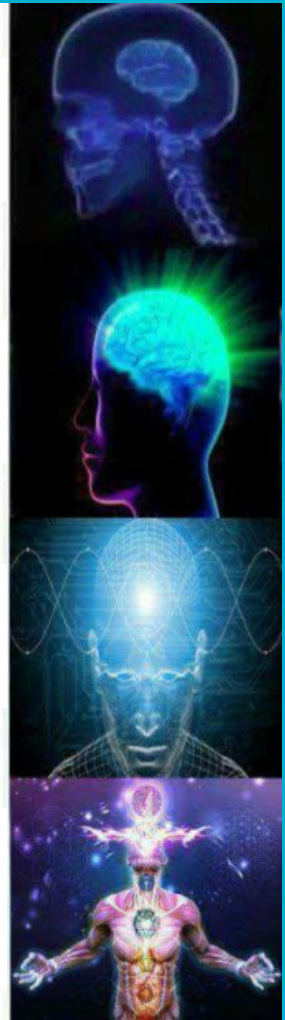
How to access/view another file simultaneously? How to “split screen”?

Using your phone
to take a picture
of the other file

Opening a new
terminal/ ssh/
computer/ tab
to access the
other file

In vim,
:o
:O
:vs
:split

??????????????



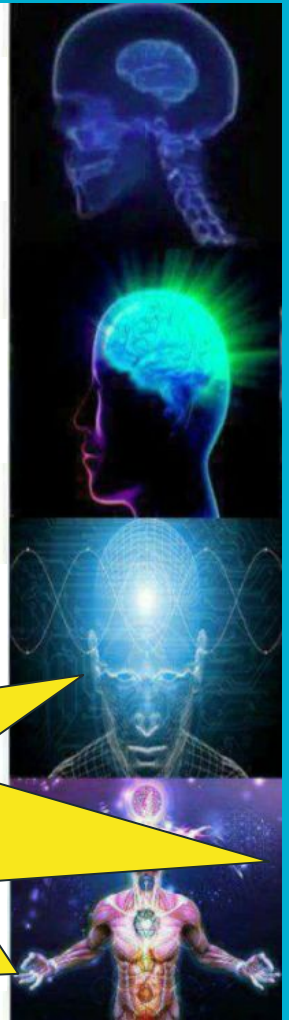
How to access/view
another file
simultaneously?
How to “split

Using your phone
to take a picture
of the other file

Opening a new
terminal/ ssh/
computer/ tab
to access the
other file

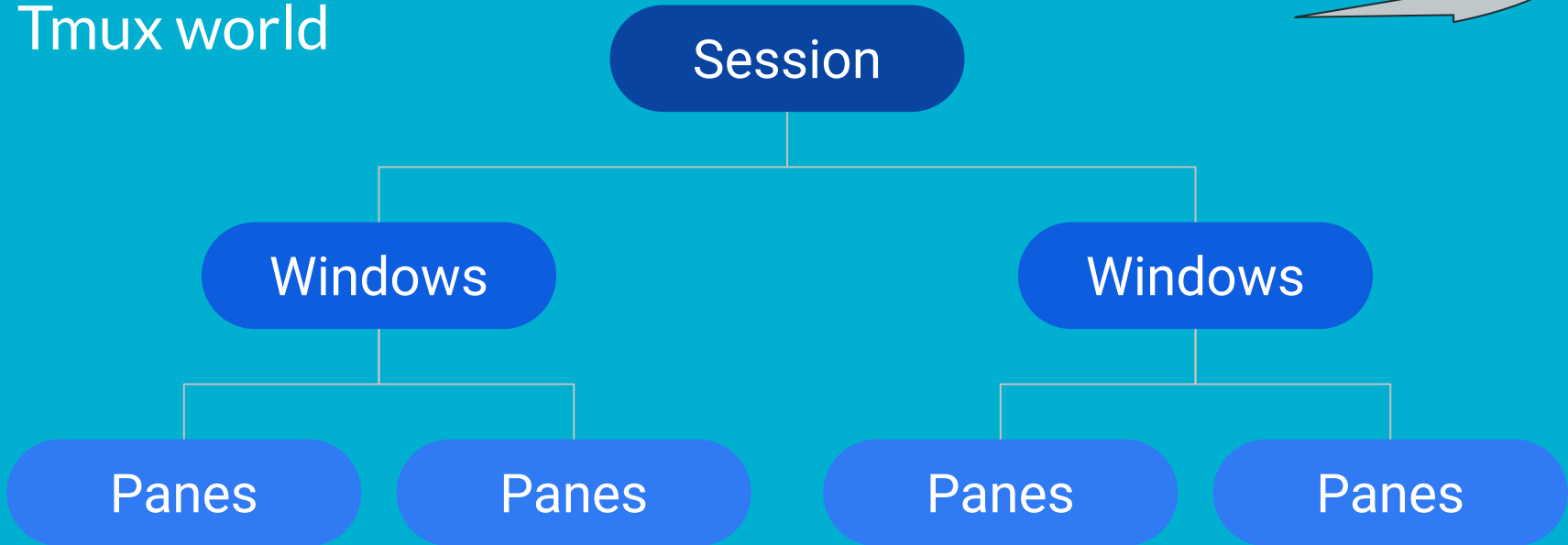
In vim.

Tmux



Terminal

Tmux world



First, how to enter and exit tmux

tmux - creating a brand new session

tmux detach - detaching from session

tmux a - attaching on to previous session

tmux kill-server - (permanently) kill all session

Handling windows & panes

prefix + c - creating a new window

prefix + n - move to the next window

prefix + & - kill current window

prefix + w - list all the window

Prefix is
default
CTRL-b

prefix + %(vertical) or " (horizontal)
- “split screen” / create new pane

prefix + o - move to the next pane

prefix + x - kill the current pane

prefix + space - change layout

prefix then type
:setw synchronize-panes

prefix + z (zoom) make
current pane full screen

For more shortcuts:

<https://gist.github.com/MohamedAlaa/2961058>

There is `~ /.vimrc`, there's also
`~ /.tmux.conf`

You can change your prefix and shortcuts of commands (for example, the pane splitting commands). You can also customize how tmux looks!

<https://www.hamvocke.com/blog/a-guide-to-customizing-your-tmux-conf/>

Thanks for listening

(and pretending that you didn't know anything about vscode)

Have a good weekend :D