Oh My Zsh Zsh config framework

START HERE

`omz update`
Update Oh My Zsh

`alias`
List all aliases

`omz reload`
Apply .zshrc changes

`mkcd <dir>`
Make dir and enter it

`plugins=(git z)`
Enable plugins

`gst`
git status (git plugin)

`ZSH_THEME="robbyrussell"`
Pick a theme

`uninstall_oh_my_zsh`
Remove it again

~/.zshrc plugins=() and ZSH_THEME → `source oh-my-zsh.sh` → **$ZSH** plugins/ and themes/ → `overlaid by` → **$ZSH/custom** your plugins, themes, *.zsh → `loads into` → **Your shell** aliases, completion, prompt

CONCEPTS

Oh My Zsh is a framework: it only reads ~/.zshrc, loads the plugins and theme you list, and adds aliases and functions.

Set plugins, theme and zstyle options before the line that sources oh-my-zsh.sh, or they are ignored.

Separate plugin names with spaces or newlines, never commas. Every plugin adds startup time, so enable only what you use.

INSTALL AND UPDATE

`sh -c "$(curl -fsSL https://raw.githubusercontent.com/ohmyzsh/ohmyzsh/master/tools/install.sh)"`
Install with curl
The installer renames an existing .zshrc to .zshrc.pre-oh-my-zsh.

`omz update`
Update Oh My Zsh now

`omz reload`
Reload the shell config
Runs exec zsh. Prefer it over source ~/.zshrc.

`uninstall_oh_my_zsh`
Uninstall and restore old shell config

`omz help`
List omz commands

`zstyle ':omz:update' mode auto`
Update without asking
Modes: auto, reminder, disabled. Put it in .zshrc before Oh My Zsh loads.

`zstyle ':omz:update' frequency <days>`
Days between update checks

`zstyle ':omz:update' cooldown <days>`
Only apply updates this old

PLUGINS

`plugins=(git z extract)`
Enable plugins in .zshrc

`omz plugin list`
List available plugins

`omz plugin enable <name>`
Add a plugin to .zshrc

`omz plugin disable <name>`
Remove a plugin from .zshrc

`aliases`
List aliases of enabled plugins
Needs the aliases plugin.

PLUGINS (cont.)

`sudo plugin: Esc Esc`
Prefix the command line with sudo

`extract <file>`
Unpack most archive formats
Needs the extract plugin.

`copypath [<path>]`
Copy absolute path to clipboard

`copybuffer plugin: Ctrl+O`
Copy the current command line

`z <part-of-dir>`
Jump to a frequently used dir
Needs the z plugin.

`git clone <repo> ${ZSH_CUSTOM:-~/.oh-my-zsh/custom}/plugins/<name>`
Install an external plugin
Then add <name> to plugins=(). Autosuggestions and syntax-highlighting are installed this way.

THEMES

`ZSH_THEME="robbyrussell"`
Set the theme in .zshrc

`ZSH_THEME="random"`
New random theme each shell

`omz theme list`
List installed themes

`omz theme use <name>`
Try a theme in this shell only

`omz theme set <name>`
Save a theme to .zshrc

Some themes (agnoster, for example) need a Powerline or Nerd Font in your terminal. Prefer starship if you want a prompt independent of the framework.

NAVIGATION

`..` / `...` / `....`
Go up 1, 2, 3 directories
Also ..... for four levels.

`-`
Back to the previous directory

`d`
List recent directories, numbered
Alias of dirs -v.

`1 ... 9`
Jump to the nth recent directory
Runs cd -<n>.

`mkcd <dir>`
Make dir (with parents) and enter

`take <dir|url>`
mkcd, or fetch/clone a .git or tar URL

NAVIGATION (cont.)

`md <dir>`
mkdir -p

`zsh_stats`
20 most used commands

`alias`
List all defined aliases

GIT PLUGIN ALIASES

`gst`
git status

`gss`
git status --short

`ga <file>` / `gaa`
git add / add --all

`gcmsg "<msg>"`
git commit --message

`gc`
git commit --verbose

`gcam "<msg>"`
commit --all --message

`gco <branch>`
git checkout

`gcb <branch>`
git checkout -b

`gsw <branch>` / `gswc <branch>`
git switch / switch -c

`gcm`
Check out the main branch

`gb`
git branch

`gd`
git diff

`gdca`
git diff --cached

`gl`
git pull

`gp`
git push

`gpf`
Force push with lease

`gpsup`
Push and set upstream to origin

`gfa`
Fetch all, with tags, pruned

`glog`
Oneline decorated graph log

`gm <branch>`
git merge

`grb` / `grbi`
git rebase / interactive

`gpr`
git pull --rebase

`gsta` / `gstp`
Stash push / stash pop

GIT PLUGIN ALIASES (cont.)

`grst <file>`
git restore --staged

`grhh`
git reset --hard
Discards local changes.

`gpristine`
Hard reset plus clean -dfx
Destructive: removes untracked and ignored files.

`gwip` / `gunwip`
Commit all as WIP / undo that commit

`grt`
cd to the repository root

DOCKER PLUGIN ALIASES

`dps` / `dpsa`
docker ps / ps -a

`dr <image>`
docker container run

`drit <image>`
container run -it

`dxcit <ctr> <cmd>`
container exec -it

`dlo <ctr>`
docker container logs

`dils`
docker image ls

CUSTOMIZE

`plugins=(git z)`
Plugin list, before sourcing

`$ZSH/custom/*.zsh`
Your own config, auto-loaded

`custom/plugins/<n>/<n>.plugin.zsh`
Layout of a custom plugin

RECIPES

Typical .zshrc

```
export ZSH="$HOME/.oh-my-zsh"
ZSH_THEME="robbyrussell"
zstyle ':omz:update' mode reminder
plugins=(git z extract sudo)
source $ZSH/oh-my-zsh.sh
```

Add an external plugin

```
git clone https://github.com/zsh-users/zsh-autosuggestions \
  ${ZSH_CUSTOM:-~/.oh-my-zsh/custom}/plugins/zsh-autosuggestions
# then in .zshrc: plugins=(git zsh-autosuggestions)
omz reload
```

Daily git loop with aliases

```
gswc fix-typo
gaa && gcmsg "Fix typo"
gpsup
gst
```