## Notes

| | |
|---|---|
| 🔌 | Command derived from plugin |
| 👤 | Command derived from preferences |

🔗 `.tmux.conf` for this cheatsheet available on Github 🎧 brennanmeadowcroft/dotfiles

🔗 `.bash_profile` for this cheatsheet available on Github 🎧 brennanmeadowcroft/dotfiles

## 🔌 TMUX

| | |
|---|---|
| `<Tmux Leader >+ x` | Close current pane |
| `<Tmux Leader >+ -` | New horizontal pane |
| `<Tmux Leader >+ |` | New vertical pane |
| `<Tmux Leader>+z` | Toggle current pane to full screen |
| `<Tmux Leader >+ [` | Set cursor to move up screen |
| `Alt+<a rro w>` | Move to pane in direction of arrow ↑ / ↓ / ← / → |

ℹ️ `<Tmux Leader>` is `Ctrl+a`

## Navigating Directories

| | |
|---|---|
| `ls` | List contents of current directory |
| `tree` | Display current directory and one level below as tree |
| `tree <dir>` | Display `<di r>` and one level below as tree 🔌 |
| `cd <di r>` | Change directory to `<di - r>` |
| `cd ..` | Move up to parent of current directory |
| `cd -` | Change to previous working directory |
| `pushd <dir>` | Create mark for `<di r>` in a stack |
| `popd` | Go to marked directory at top of stack |

## Working With Lines

| | |
|---|---|
| `Ctrl+a` | Move cursor to *BEGINNING* of current line |
| `Ctrl+e` | Move cursor to *END* of current line |
| `Alt+b` | Move cursor *BACK* one word |
| `Alt+f` | Move cursor *FORWARD* one word |
| `Ctrl+k` | Delete from cursor to *END* of line |
| `Ctrl+u` | Delete from cursor to *BEGINNING* of line |
| `Alt+d` | Delete word *IN FRONT* of cursor |
| `Ctrl+w` | Delete word *BEHIND* cursor |

## Navigating Processes

| | |
|---|---|
| `ps` | List running processes in Bash |
| `ctrl-z` | Move current process to background in *suspended* state |
| `bg` | Start most recent suspended process & run in background |
| `fg` | Bring most recent background process to foreground |
| `kill <pid>` | Force quit `<pi d>` from `ps` |

ℹ️ The output of `ps` can be piped to another command such as `grep` or `kill`

## Working With Files

| | |
|---|---|
| `vim <fi le>` | Open a file in vim |
| `vim .` | Open vim at current working directory |
| `open .` | Open current directory in Finder (OSX) |
| `cat <fi le>` | Read and output current file into bash |
| `less <fi le>` | Display a paged version of file in terminal |
| `findfile <file>` | Search current directory and children for `<fi le >` 👤 |

ℹ️ Using `less` instead of `cat` keeps bash history and screen clean as `less` opens another window for viewing