Legend

special keys are surrounded by **< >**	
e.g.: `<CT RL- c>` = control-c	
tokens are surrounded by **{ }**	
e.g.: {#} = some number, {id} = some id	
Follow me on twitter: @pasztech	

Navigation & Searching

next word	
`w`	
previous word	
`b`	
end of word	
`e`	
forward one screen	
`<CT RL- f>`	
back one screen	
`<CT RL- b>`	
forward half screen	
`<CT RL- d>`	
back half screen	
`<CT RL- u>`	
search for word under cursor	
`#`	

Files

Always set working directory to current file	
`set autochdir`	
File explorer	
`:e .`	
OS file explorer	
`:browse e`	
name of current file	
`:echo @%`	
Search for **file** in hierarchy and open it	
`:tabe **/{file}`	
Set working directory to current file	
`:lcd %:p:h`	
Save file with no eol	

Files (cont)

`:set binary :set noeol`	

Editing

open new line for insert above cursor	
`o`	
open new line for insert below cursor	
`O`	
change current word	
`cw`	
change entire line	
`cc`	
change rest of line	
`C`	
replace current character	
`r`	
replace all text until <ESC> is typed	
`R`	
swap characters	
`xp`	
delete forward	
`x`	
delete backward	
`X`	
delete to the end of the line	
`D`	
delete forward one word	
`dw`	
deletes **#** lines	
`{#}dd`	
toggle case	
`~`	
toggle case of word	
`g~iw`	
delete blank lines	

Editing (cont)

`:g/^$/d`	
delete trailing white space	
`:%s/\s \+$//`	
delete all extra white space between words	
`:s/\s\+/ /g`	
insert **chars #** times	
`<CT RL- o>{ #}{ cha rs} <ES C>`	

Selecting & Copying

match braces	
`%`	
select all	
`gg"+yG`	
copy forward one word	
`yw`	
copy from cursor to the end of line	
`y$`	
copy **#** lines	
`{#}yy`	

Tabs and Windows

split window and load another file	
`:split {file}`	
move cursor to another window	
`<CT RL- w> <CT RL- w>`	
maximize current window	
`<CT RL- w>_`	
make all equal size	
`<CT RL- w>=`	
close all other windows	
`<CT RL- w>o` **or** `:only`	
next tab | prev tab	
`gt` | `gT`	

🗋 Registers & Buffers

show name and contents of registers

`:reg`

paste what's in register **a**

`"{a}p`

yank selection into register **a**

`"{a}y`

record into register **a** (q again to stop)

`q{a}`

execute register **a**

`@{a}`

repeat last one

`@@`

List buffers

`:ls` or `:buffers`

Display buffer by **id**

`:b{id}`

Next and prev buffer

`:bnext` and `:bprev`

Delete buffer by **id**

`:bd{id}` (append ! to discard changes)

</> Surround.vim

change surroundings

`cs{fro m_c har }{t o_char}`

delete surroundings

`ds{char}`

surround **word**

`ysiw{char}`

surround line

`yss{char}`

An indispensable plugin: https://github.com/tpope/vim-surround