vi

ex Modus

: Bereich s / reg. expression / subst. / mode

z.B. 1, $ s / a / A / g	ersetzt von Zeile 1 ab bis Dateiende ($) alle a durch A; g bedeutet alle a einer Zeile
z.B. %s / a / A	ersetzt in der ganzen Datei (%) jeweils das erste a einer Zeile durch A
z.B. 'a , 'b s / a / A / g	ersetzt von Zeile Marke 'a bis Marke 'b alle a durch A; g bedeutet alle a einer Zeile

: w [filename]

z.B. w	speichert Datei
z.B. w neuerName	speichert Datei unter neuem Namen (hier: neuerName)

:e filename	öffnet die genannte Datei
:n	springt zwischen offenen Dateien hin und her (n=next) (vorher mit **vi datei1 datei2 datei3 ..** öffnen)

/ text

z.B. / while.*number	sucht in der offenen Datei vom aktuellen Curserpunkt an vorwärts nach einem Abschnitt in dem das Wort "while" dann beliebig viele (*) Zeichen (.) dann "number" vorkommt
? Text	genau wie / text nur startet die Suche rückwärts
: zahl	springt in die "zahl"-te Zeile

command Modus

I text ESC	fuegt am Anfang der Zeile ein
i text ESC	fuegt an aktuelle Position ein
A text ESC	fuegt am Ende der Zeile ein
a text ESC	fuegt nach der akt. Position ein
R text ESC	ersetzt folgenden text
r char ESC	ersetzt folgendes Zeichen
O text ESC	fuegt Text eine Zeile über der aktuellen ein
o text ESC	fuegt Text eine Zeile unter der aktuellen ein
[zahl] S	ersetzzt erst [zahl] Zeichen und fügt dann ein
u	undo
^r	redo

[zahl] c	**[w,b, f, t, m, c]**	ersetzt [zahl] Zeichen
[zahl] d	**[w,b, f, t, m, d]**	löscht [zahl] Zeichen
[zahl] y	**[w,b, f, t, m, y]**	kopiert [zahl] Zeichen
	w	erstzt "Zeichen" durch "naechstes Wort" in obigen Befehlen
	b	erstzt "Zeichen" durch "vorheriges Wort" in obigen Befehlen
	f [char]	findet nächsten angegebenen Char (einschliesslich) und verbindet mit Befehl
	t [char]	findet nächsten angegebenen Char (ausschliesslich) und verbindet mit Befehl
	m	steht für Marke und führt genannten Befehl bis zur genannten Marke aus
	c, d, y	bezieht die Befehle auf die ganze Zeile

z.B. yfs findet nächstes "s" in der aktuellen Zeile und speiert sie im Puffer ohne Namen

Marken

m [char]	setzt an der aktuellen Position eine Marke mit Namen "char"
' char	ruft Marke mit Namen char auf

Marken können mit allen Befehlen kombiniert werden. Bsp. s.o.

Puffer

" char	ruft Puffer mit Namen char auf
z.B. 5dwa	löscht 5 Wörter und speichert sie in Puffer "a"
p [char]	fügt Pufferinhalt nach der aktuellen Position ein
P [char]	fügt Pufferinhalt vor der aktuellen Position ein

© Petra Hitzelberger 2000