

INTRODUÇÃO:

No vim temos vários "modos", que são estados do editor.

São eles:

Modo	Tecla	Rodapé	Descrição
de Inserção	i	-- INSERT --	Inserção de texto
de Comandos	<ESC>		Comandos de manipulação de texto
Linha de comando	:	:	Comandos de manipulação arquivo
Visual	v	-- VISUAL --	Seleção visual de texto
Busca	/	/	Busca de padrões no texto
Reposição	R	-- REPLACE --	Inserção sobreescrevendo

* Os modos foram colocados em ordem de importância de uso da grande maioria dos usuários.

* Para acessar qualquer um desses modos, na dúvida aperte <ESC> antes da sua tecla de acesso (2ª coluna da tabela)

* Para identificar em qual modo você está, olhe no rodapé da tela! (3ª coluna da tabela)

CHAMANDO O VIM:

```
vi                Abre o vim vazio, sem arquivo (muito pouco usado)
vi arquivo        Abre o arquivo "arquivo". Se ele não existir, o cria

vi arquivo +      Abre com cursor no fim do arquivo
vi arquivo +10    Abre com cursor na linha 10
vi arquivo +/linux Abre com cursor na 1ª ocorrência de "linux"
```

SALVANDO/SAINDO DO VIM:

<F12> Saída forçada, salva e sai (Se usar o .vimrc do aurélio &:))

Obs.: Antes de executar os comandos seguintes, aperte <ESC> para ir ao modo de comandos, e é claro, após, um <ENTER>

```
:w                Salva
:q                Sai
:wq ou :x ou ZZ  Salva e sai
:w!              Salva forçado
:q!              Sai forçado
:wq!             Salva e sai forçado
```

EDITANDO:

Ao entrar no vim, ele está no modo de comandos. Como saber? Olhe no rodapé da tela. Nada na última linha.

Para começar a Inserir um texto, aperte "i". Você verá que aparecerá um "-- INSERT --" no rodapé. Agora você pode digitar seu texto normalmente.

Quer parar para dar uma gravada? Veja o tópico acima. (<ESC>:w)
chega de editar e quer salvar e sair? Veja acima de novo. (<ESC>:wq)

COPIANDO e COLANDO:

*** Usando o mouse:

No modo de Inserção mesmo, note que o cursor do teclado (esse tracinho piscando na tela), está numa posição, e o cursor do mouse (mexa o mouse para que ele se mova) está em outra posição.

Selecione uma parte do texto com o mouse, segurando o botão esquerdo.

Ao colar, o texto selecionado com o MOUSE será colocado a partir do cursor do TECLADO. Experimente. Dependendo do mouse, a colagem se faz:

- Apertando o botão direito
- Segurando o botão esquerdo e apertando o direito ao mesmo tempo
- Apertando o botão do meio (mouse de 3 botões)

Tente, o seu será umas das 3 alternativas acima.

*** Usando o modo visual:

Entre no modo visual (<ESC>v), e simplesmente aperte as setas do teclado, movendo o cursor do TECLADO e selecionando o texto desejado. O comando para copiar o texto é o "y" de Yank.

Volte para o modo de comandos (<ESC>) e posicione o cursor do TECLADO no lugar onde você quer colar o texto selecionado. O comando de colagem é o "p" de Paste.

APAGANDO (DELETANDO):

Como no tópico anterior, use o modo visual (<ESC>v) para selecionar o texto desejado. Para apagá-lo, digite "d", de Delete.

Com o linux no modo texto, a tecla Delete do teclado também funciona para apagar texto. Já no X, isso é meio relativo... &:)

T U T O R I A L

<http://www.networkcomputing.com/unixworld/tutorial/009/009.part1.html>

SUBSTITUIÇÕES:

:s/velho/novo	substitua a 1ª ocorrência de velho por novo na linha corrente
:% s/velho/novo/	substitua em todo o arquivo (%) a 1ª ocorrência de velho por novo em cada linha
:% s/velho/novo/g	substitua em todo o arquivo (%), todas (g) as ocorrências de velho por novo
:% s/velho/novo/gc	igual ao anterior, mas pedindo confirmação para cada substituição
:% s/^[^String[0-9]]//gc	expressões regulares também funcionam, igual ao sed
:% s/./\u&/gc	converta para maiúsculas (\u) o 1º carácter (.) de cada linha (exclusiva vi/perl, no sed não funciona)

ABREVIações:

:ab	mostra todas as abbr
:abc[lear]	tira todos
:iab	apenas p/modo inserção
:iabc[lear]	tira todos de inserção
:cab	apenas p/modo de comando (:)
:cab[lear]	tira todos modo comando
:una vc	tira ab para vc

obs.: pontuação, espaço ou o ENTER, disparam a expansão de uma abreviação.
mas o Ctrl+] também pode ser usado, para expandir sem adicionar caracteres.

OPções DO SET:

:set		
autowrite	aw	salva a cada alteração
backspace	bs	comportamento backspace(1 ou 2)
errorbell	eb	campainha de erro
expandtab	et	troca tab por espaços
fileformat=dos	ff	converte o arquivo para DOS
hidden	hid	preserva buffer
hlsearch	hls	ilumina a última procura
ignorecase	ic	case insensitive na busca
incsearch	is	ilumina procura enquanto digita
laststatus=2		mostra linha de estado
lazyredraw	lz	não redesenha em macros
lines=N		número de linhas na tela
magic		usar mágicas na procura de padrões
number	nu	mostra núm da linha
report=N		mostra aviso quando N linhas mudaram (0=sempre)
showcmd		mostra o comando que se está fazendo
showmatch	sm	mostra o casamento de {},[],()
smartcase	scs	assume "noic" quando tiver maiúsculas
textwidth=N		quebra de linha do texto
undolevels	ul=N	guarde os N últimos comandos p/desfazer (padrão=1000)
vb t_vb=		retira o "beep" de erro

MAIÚSCULAS/MINÚSCULAS:

5~ inverte 5 próximos
g~\$ inverte até o fim da linha
seleciona, u converte p/minúsculas
seleciona, U converte p/maiúsculas
seleciona, ~ inverte
*este seleciona, é com o modo visual

DEFININDO COLUNA DE QUEBRA DE LINHA:

:set textwidth=N

mas, já estando num arquivo pronto:

:set wm=30 (esse 30 é o número de colunas a cortar, 80-30=50 colunas)
gq<ate aonde vc quer ir>

ex.
gqG (ate o fim do arquivo)
gq\$ (ate o fim da linha)
ou seleciona com o modo visual e gq

Ctrl+:

modo COMANDO:

A = incrementa um número	(Add)
X = decrementa um número	
S = ScrollLock	
L = redesenha tela	
V = modo visual	(Visual Vertical)
G = status do arquivo	
M = início da próxima linha	
E = linha abaixo sem mover cursor	
Y = linha acima sem mover cursor	
N = próxima linha	(Next)
P = linha anterior	(Previous)
F = PageDown	(Forward)
B = PageUp	(Backyard)
U = PageUp / 2	(Up)
D = PageDown / 2	(Down)

modo INSERÇÃO:

A = insere o último texto inserido	
I = TAB	
S = ScrollLock	
H = BackSpace	
T = 2 tab's no início da linha	(Two Tabs)
V = anula expansão do próx. caractere	
J = Enter - quebra de linha	
M = Enter - quebra de linha	
L = redesenha tela	
R = insere conteúdo do registrador [a-z]	*ver abaixo

K = insere um dígrafo	*ver abaixo
N = procura palavra no texto atual	(Next)
P = procura palavra no texto atual	(Previous)
Y = copia caractere que está acima	(Yank)

CARACTERES ESPECIAIS (DÍGRAFOS):

ga	mostra o código da letra sobre o cursor
:dig	mostra todos os dígrafos disponíveis
ex.: para botar um $\frac{1}{2}$, Ctrl+K,1,2	

ARQUIVOS E JANELAS MÚLTIPLOS:

```
:wn          grava o atual e vai ao próximo
:wN         grava o atual e vai ao anterior
:args       mostra todos os arquivos atuais
:qa         sai fora de todo mundo de uma vez
:all        abre todos os arquivos em janelas individuais
```

Tecla chave das janelas	= Ctrl+W	
j, seta abaixo	move para janela abaixo	
k, seta acima	move para janela acima	
o	apenas esta janela, fecha todas as outras	(Only)
+, -	muda o tamanho da janela	
=	deixa todas as janelas com tamanhos iguais	

DICAS DIVERSAS:

```
:xit          igual :wq, mas só grava se tiver sido alterado algo
```

```
imprime no arquivo o próprio nome do arquivo editado qdo N_ARQ é digitado
:map N ARQ ^R=expand("%:t:r")^M
```

dar dois TABs no início de cada linha:
seleciona várias linhas, 2>

```
macro para criar contagem
---> il <ESC> qa Y p Ctrl+A q 5@a
```

alinhamento, sim! alinhamento

```
:left
:right
:center
```

```
procurar 2 palavras ao mesmo tempo
/palavra1\|palavra2
```

REGISTRADORES:

"[a-z] use o registrador [a-z] para o próximo delete, cópia ou cola
:reg mostra o conteúdo de todos os registradores
:reg [a-z] mostra o conteúdo do registradores [a-z]
obs.: aqui este [a-z] na verdade pode ser: 0-9a-z%#:.-="

MARCAS:

m[a-z] marca em [a-z] a posição corrente do cursor
\[a-z] vai até a marca [a-z]
`` vai até a posição anterior ao último pulo (alterna)
:marks mostras as marcas ativas

GRAVAÇÃO DE SEQUÊNCIAS DE COMANDOS:

q[a-z] inicia a gravação de uma sequência no registrador [a-z]
q[A-Z] inicia a gravação, adicionando no registrador [a-z]
q pára a gravação
@[a-z] executa a sequência do registrador [a-z] (5 vezes? 5@a)

dica: pode-se colocar o @[a-z] dentro da própria gravação do q[a-z]!
 assim ele é executado recursivamente. muito útil quando há uma
 procura de padrões na gravação. faz para todas as ocorrências.

MAPEAMENTOS:

:map <F9> <ESC>:r!date	mapeamento em modo de comando
:imap <F9> <ESC>:r!date	mapeamento em modo de inserção
:cmap <F9> r!date	mapeamento em modo linha de comando
:vmap <F9> :r!date	mapeamento em modo visual

ex.:

"html: negrito no trecho selecionado
:vmap <F9> d`<i><esc>pa # html: negrito no trecho selecionado

"liga/desliga autoIndent

:map ,si :set ai!<cr>:echo "autoIndent="&ai<cr>

"mostrar os espaços em branco no fim das linhas

:map <F9> / *\$^M

S I N T A X E:

Comment = ciano
Constant = roxo
Identifier = ciano

PreProc = azul escuro
Special = vermelho
Statement = amarelo
String = roxo
Type = verde

Todo = preto, fundo marrom
Error = branco, fundo vermelho
Ignore = preto, fundo preto! - esconde