



# Vi IMproved

Für Einsteiger und Fortgeschrittene

Michael Topal  
01.04.2017



1. whoami
2. Warum VIM?!
3. Basics
4. .vimrc
5. Movement
6. Text Manipulation
7. Advanced Usage



# Whoami

- Michael Topal  
@h4ckt1c
- FF-Hennef Mitglied seit: ...
- vim User seit ca. 4 Jahren



# Warum VIM?!

- Anpassungsfähiger Text-Editor
- Terminalanwendung
- Konfiguration sehr einfach übertragbar
- äußerst Umfangreich (Pro & Contra)
- große Community ([stackoverflow.com](https://stackoverflow.com))



# Warum VIM?!



`brew install vim`

`pacman -S vim`



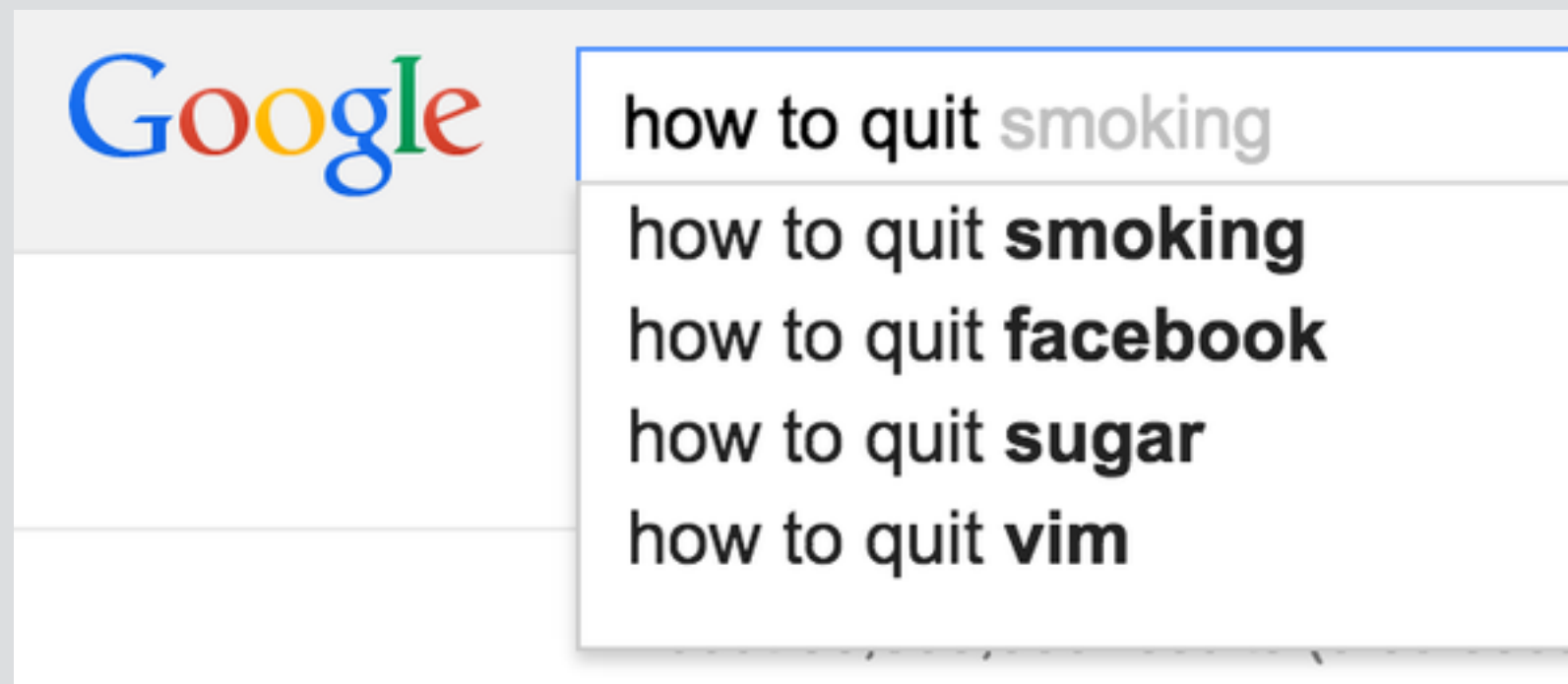
`apt-get install vim`

`yum install vim`





# Basic Usage





# Basic Usage

- Normal Mode
- Insert Mode
- Command Mode
- Visual Mode



# Basic Usage

Normal Mode:

- Navigation
- Dokumentbetrachtung
- Copy&Paste

No Escape





# Basic Usage

## Insert Mode:

- `i/I` insert
- `a/A` append
- `o/O` „open“ New Line

„Escape“ with **Esc**



# Basic Usage

Command Mode:

Die meisten Kommandos beginnen mit:

:

„Escape“ with **Esc-Esc**



# Basic Usage

## Command Mode:

- `:w`      Speichern (write)
- `:q`      Schließen (quit)
- `:wq`     Speichern & Schließen (write&quit)
- `:q!`     Schließen erzwingen (quit forced!)

„Escape“ with **Esc-Esc**



# Basic Usage

Command Mode:

```
:help
```

```
:help insert
```

```
:colorscheme solarized
```

```
:syntax on
```

„Escape“ with **Esc-Esc**



# Basic Usage

## Visual Mode:

- `V` Visual Line
- `v` Visual
- `<C-v>` Visual Block

„Escape“ with **Esc-Esc**



# Basic Usage

**u**      undo

**<C-r>**    redo



# Basic Usage

`u`    undo

`<C-r>`    redo

`:earlier`

`:later`



# .vimrc

- Konfigurationsdatei
- ~/.vimrc (Heimatverzeichnis)
- (sehr) personalisiert
- unzählige Beispiele verfügbar ([github.com](https://github.com))





# .vimrc

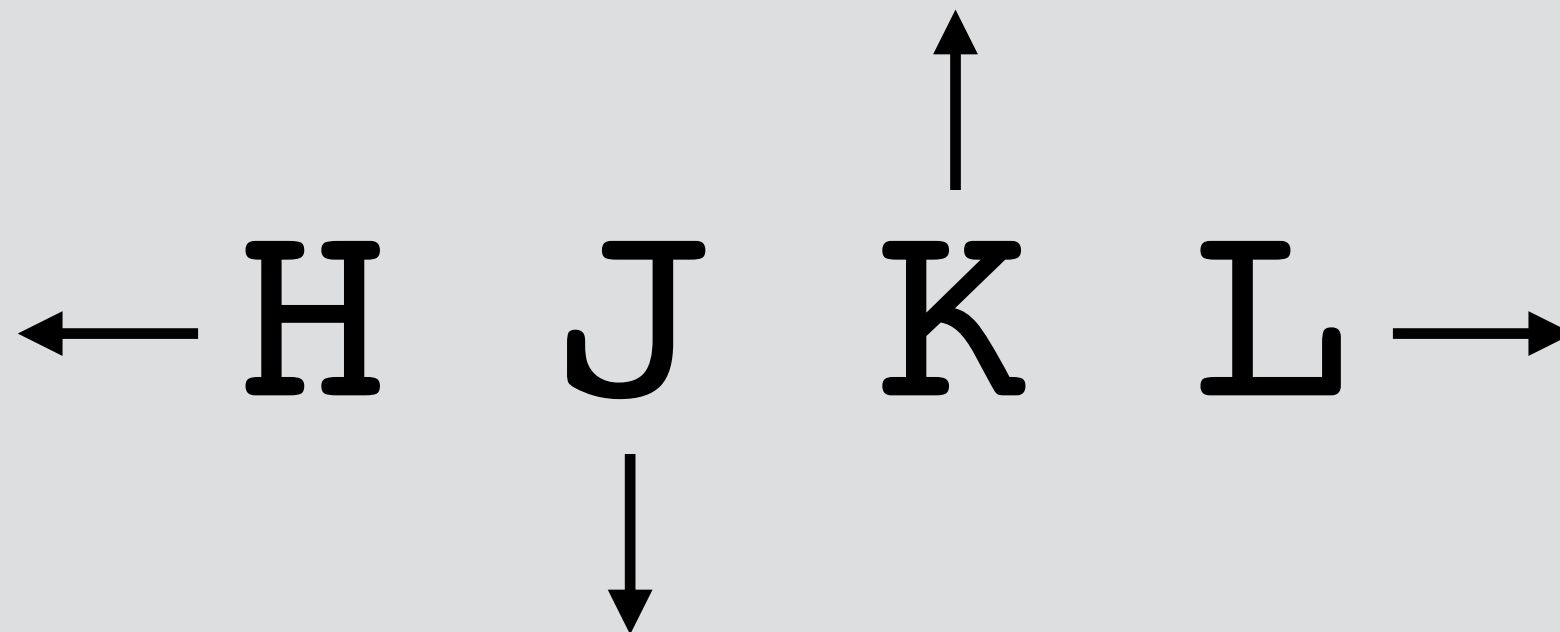
```
:colorscheme solarized
```

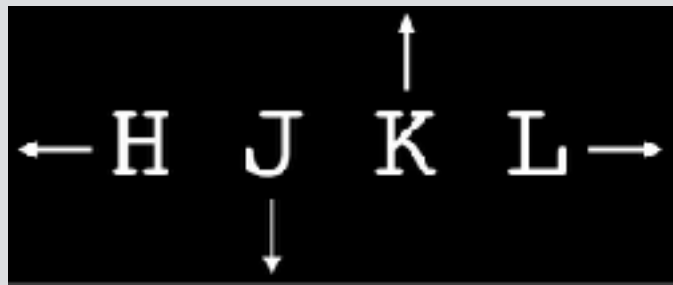
```
:syntax on
```

```
:set number
```



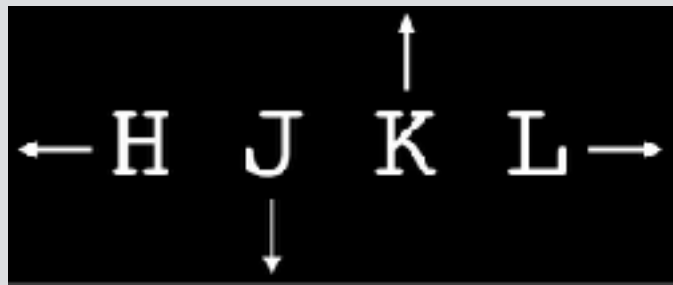
# Movement





# Movement

[ n ] W / w	Wort vorwärts
[ n ] B / b	Wort rückwärts
^	Zeilenanfang
\$	Zeilenende
gg	Dateianfang
G	Dateiende

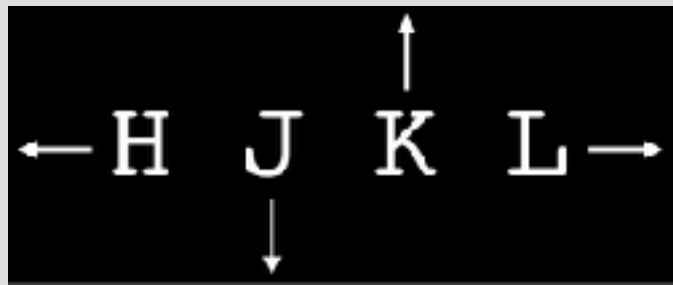


# Movement

Springe zu Zeile 6

:6

6G



# Movement

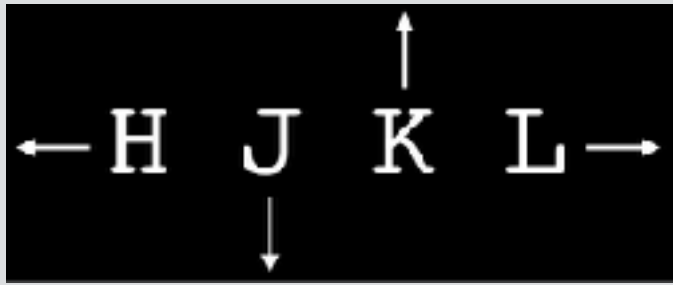
Let's scroll ;)

<C-e> Up

<C-y> Down

<C-d> Half-Page down

<C-u> Half-Page up



# Movement

## Suche

/foobar

Vorwärts

?foobar

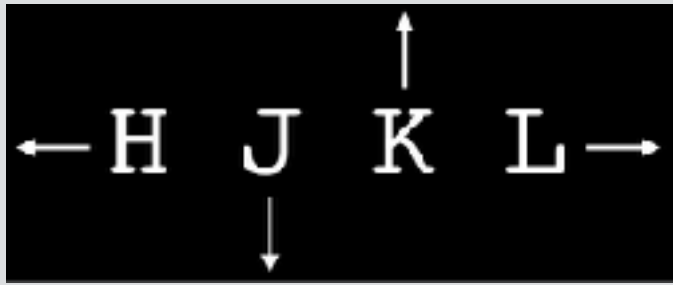
Rückwärts

n (ext)

Nächster Treffer

N (ext)

Vorheriger Treffer



# Movement

## Simple Search/Jump

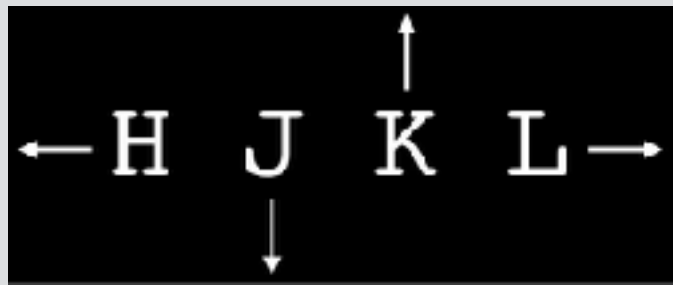
(Nur in aktueller Zeile)

**F / f**      **T / t**

Gedankenstütze:

find

'til

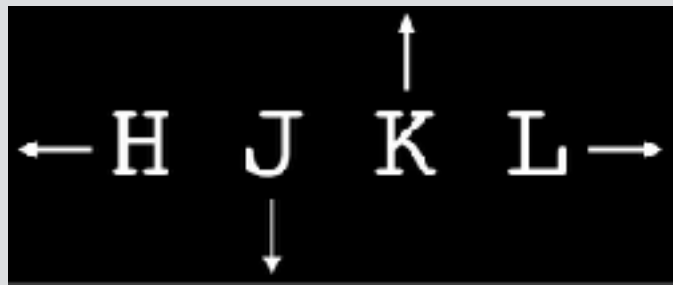


# Movement

Springe zur schließenden Klammer

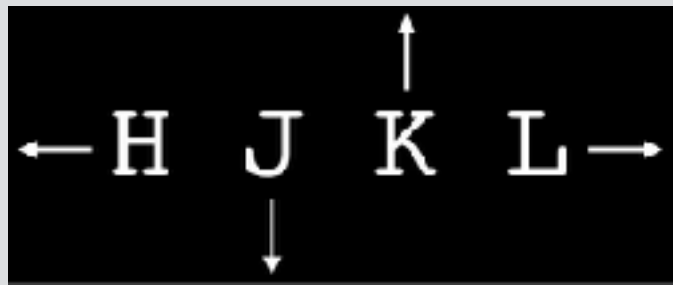
%





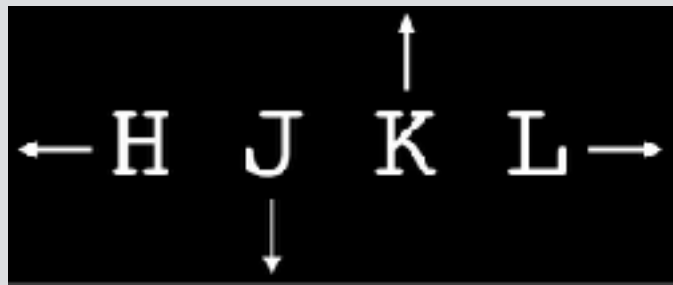
# Text Manipulation

- **dd** Delete Line
- **D** Delete vom Cursor bis zum Ende
- **cc** Cut Line (and start insert)
- **C** Cut ab Cursor bis zum End (and start insert)
- **x** Ein Zeichen löschen (Backspace?!)
- **s** Ein Zeichen löschen (and start insert)



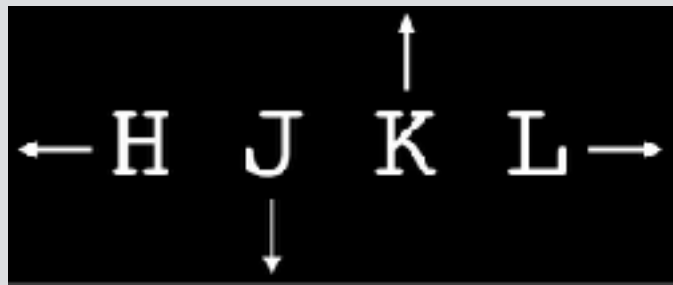
# Text Manipulation

- *ndd* Delete *n* Lines
- *ncc* Cut *n* Lines (and start insert)
- *nx* *n* Zeichen löschen (Backspace?!)
- *ns* *n* Zeichen löschen (and start insert)



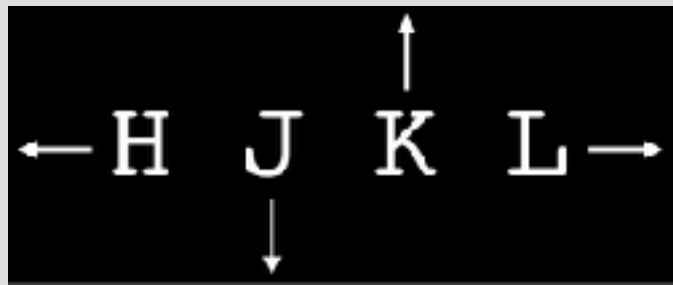
# Text Manipulation

- *dj* Delete Line and above
- *cj* Delete Line and above (and start insert)



# Text Manipulation

- **r** replace
- **R** Replace-Mode

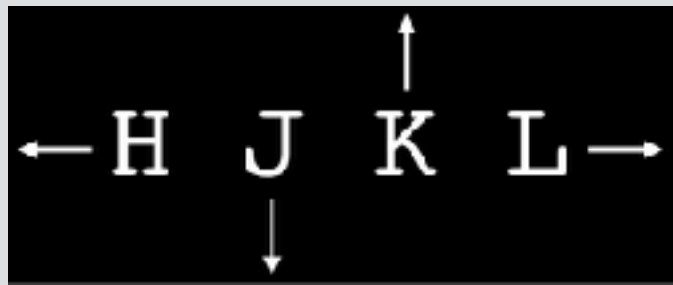


# Text Manipulation

„Spaß mit Zahlen“

<C-a>      Inkrement

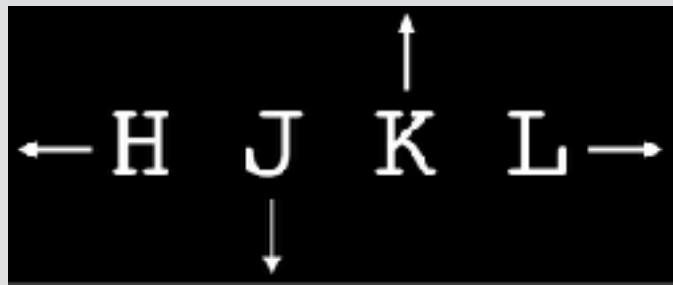
<C-x>      Dekrement



# Text Manipulation

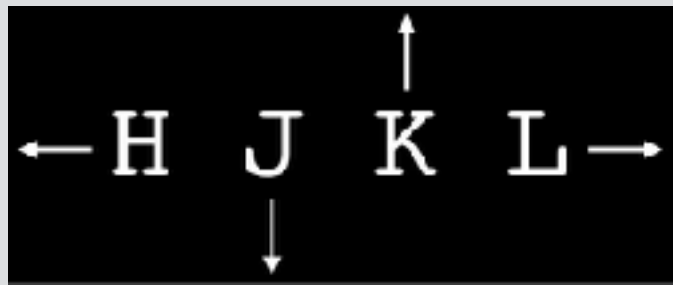
## Toggle Case

~	Wechsel
U	Groß
u	klein



# Advanced Usage

Auto-Vervollständigung:

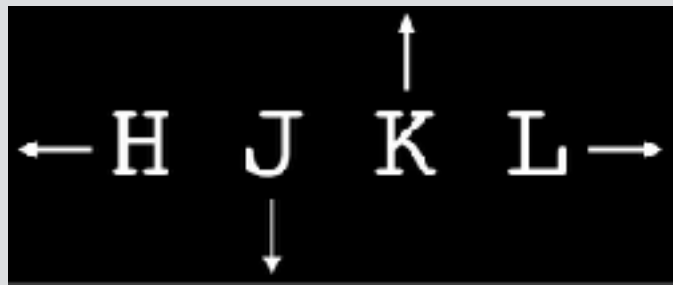


# Advanced Usage

Auto-Vervollständigung:

Im Insert-Mode





# Advanced Usage

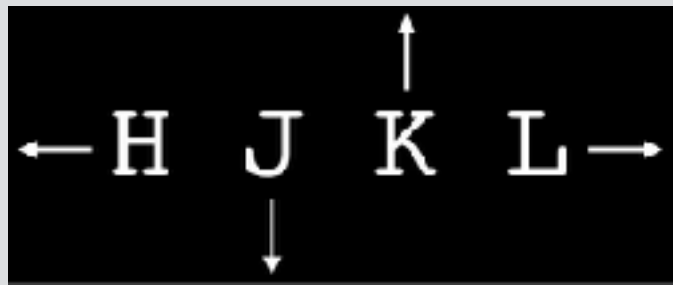
Auto-Vervollständigung:

Im Insert-Mode

von Wörtern:

`<C-x><C-n>` (next)

`<C-x><C-p>` (previous)



# Advanced Usage

Auto-Vervollständigung:

Im Insert-Mode

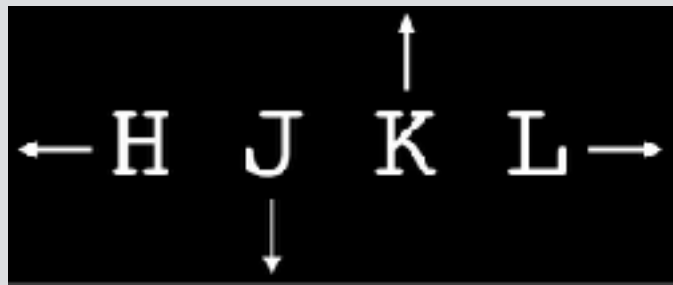
von Wörtern:

`<C-x><C-n>` (next)

`<C-x><C-p>` (previous)

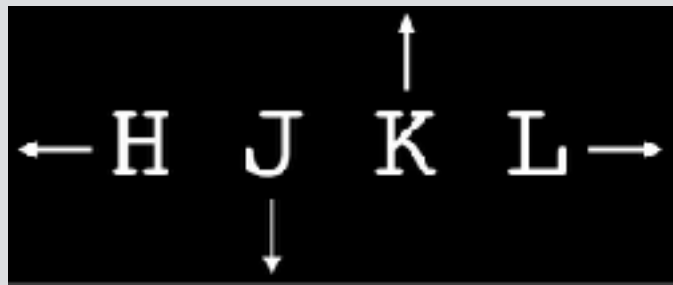
von Dateien / Pfaden:

`<C-x><C-f>`



# Motions

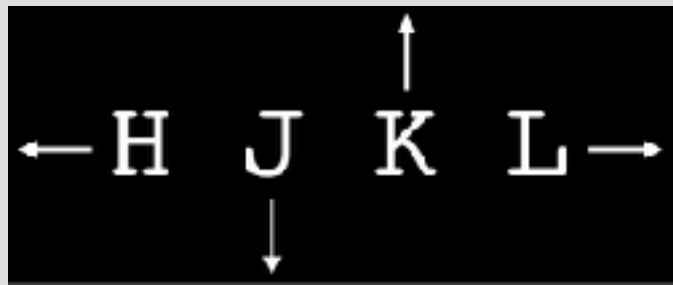
Bildlich gesprochen  
Was möchte ich machen?



# Motions

Bildlich gesprochen  
Was möchte ich machen?

delete-two-words  
delete-in-quotes (“”)  
visual-in-tag



# Motions

delete

yank

cut

visual

in

around

2

3

word

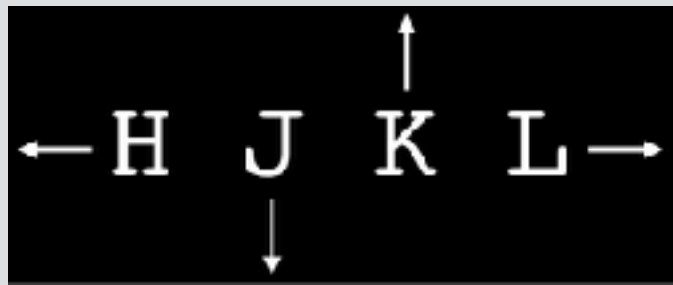
paragraph

tag

quotes

single-quotes

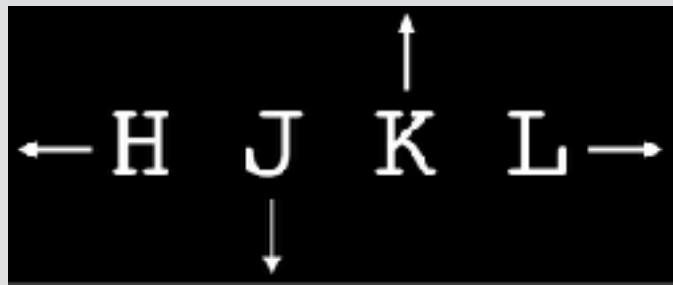
brackets



# Motions

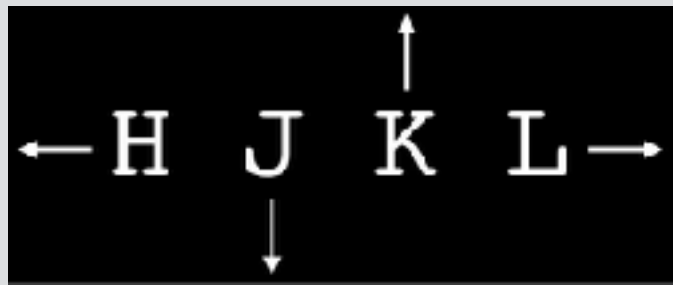
## Praktische Beispiele:

ciw	cut in word
d2w	delete 2 words
dib	delete in brackets ()
diB	delete in Brackets {}
yi"	yank in quotes (double)
yi'	yank in quotes (single)
vat	visual-mode around tag <>



# Verschlüsselung

- :X      Passphrase wird abgefragt
- :w      Speichern nicht vergessen



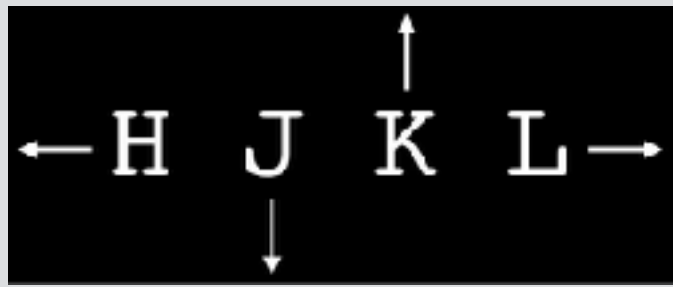
# Verschlüsselung

- :X      Passphrase wird abgefragt
- :w      Speichern nicht vergessen

```
:set cryptmethod=blowfish2
```

(ab vim 7.4.399, sonst blowfish)

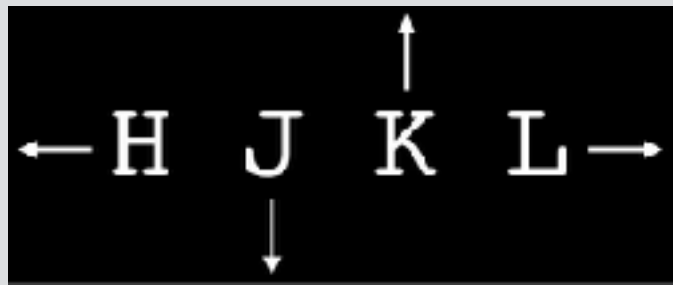




# Mehrere Dateien

`vim -p *.php`      Öffnet alle .php Dateien in Tabs

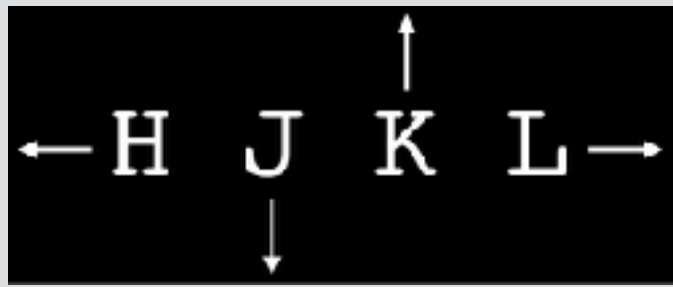
<code>gt</code>	Nächstes Tab
<code>gT</code>	Vorheriges Tab
<code>2gt</code>	Zweites Tab
<code>:q</code>	Tab schließen



# Datei-Explorer

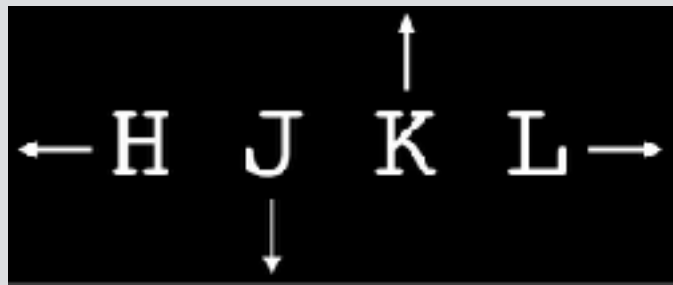
Built-in (!)  
Navigation mit hjkl

**:Texlore**



# Suchen/Ersetzen

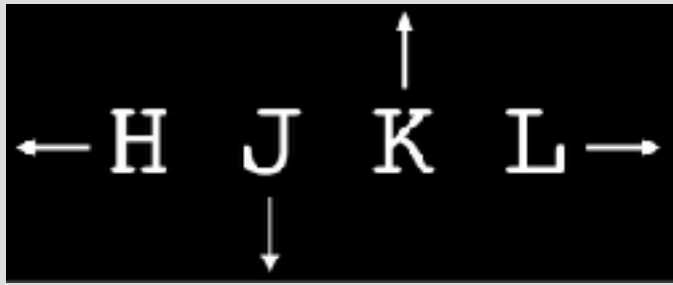
```
:%s/foo/bar/g
```



# Suchen/Ersetzen

```
: %s /foo /bar /g
```

:	Command Mode
%	In der ganzen Datei
s	Substitute
foo	Suche „foo“
bar	Ersetze durch „bar“
g	global-Flag



## VIM - Workshop



```
$ alias nano=vim
```

# Fragen?