

MEHR VI IN 10 MINUTEN

über den besten Editor der Welt.

warum ist der vi so wie er ist?

```
$ cat ex-example.txt
eins
zwei
$ echo "1,$ s/eins/EINS/
i
Hallo
Welt
.
1,$ 1" | ex ex-example.txt
Hallo$
Welt$
EINS$
zwei$
$
```

warum ist der vi so wie er ist?

```
$ ex ex-example.txt
"ex-example.txt" 2L, 10C
Entering Ex mode.  Type "visual" to go to Normal mode.
:1,$ s/eins/EINS/
EINS
:i
Hallo
Welt
.
:1,$ l
Hallo$
Welt$
EINS$
zwei$
:q!
$
```

warum ist der vi so wie er ist?

- ▶ vi ist „nur“ ein visueller Modus von ex
- ▶ Wechsel mit
 - ▶ vi -> ex: Q
 - ▶ ex -> vi: visual (vi)
- ▶ Vorteil:
 - ▶ mächtige Kommandos, scriptbar

vi-Varianten

- ▶ Original: vi (Closed Source)
 - ▶ ex (ed -> sed, ed -> EDLIN.EXE)
- ▶ vim (vi improved)
 - ▶ view (vi -r)
 - ▶ easy-Mode (vi -y, Beenden mit Ctrl-L)
 - ▶ gvim / MacVim

Warum ist der vi (trotzdem) der beste Editor der Welt?

- ▶ Kommandos einfacher zu merken, da kürzer
- ▶ vollständig durch Tastatur bedienbar
- ▶ Dank ex scriptbar
- ▶ Wiederholungen durch „.“
- ▶ Syntax-Highlight für Alles
- ▶ Formatanweisungen in der Datei gespeichert (z.B. Tabs)

Suchen

- ▶ mit „/“ oder „?“ (rückwärts)
- ▶ „n“ Suche wiederholen
- ▶ „N“ rückwärts wiederholen
- ▶ Settings (\$HOME/.vimrc):
 - ▶ „set ignorecase“ - ignoriere Gross/Klein
 - ▶ „set incsearch“ - Search as you type
 - ▶ „set hlsearch“ - gefundene Stellen markieren

Aktion mehrfach ausführen

- ▶ 20i
 - ▶ -
 - ▶ Esc
- ▶ erzeugt: - - - - -

Puffer einfügen

- ▶ Zuletzt bearbeitete Zeichen/Zeilen sind in einem Puffer und können z.B. mit „p“ wieder eingefügt werden, mehrfach
 - ▶ „yy“ - kopiert Zeile
 - ▶ „p“
 - ▶ „p“
- ▶ verdreifacht aktuelle Zeile

markieren mit v / V

- ▶ text markieren
 - ▶ „v“ Zeichen
 - ▶ „V“ Zeilen
- ▶ Aktion ausführen, z.B.
 - ▶ „d“ Delete
 - ▶ „y“ Yank (Kopieren)

:wq

- ▶ Folien: <https://strukturpunkt.de/vi>
- ▶ Fragen?