# Voorbereiding

```
git clone github.com/nedap/vim-workshop.git
of
github.com/nedap/vim-workshop/archive/master.zip
(bit.ly/115jxMA)

   vim-workshop/
   |-files/ (oefenbestanden)
   |-presentation/ (deze presentatie en handouts.pdf)
   |-vim-config/
   |-startvim <= hiermee starten we vim vandaag
```

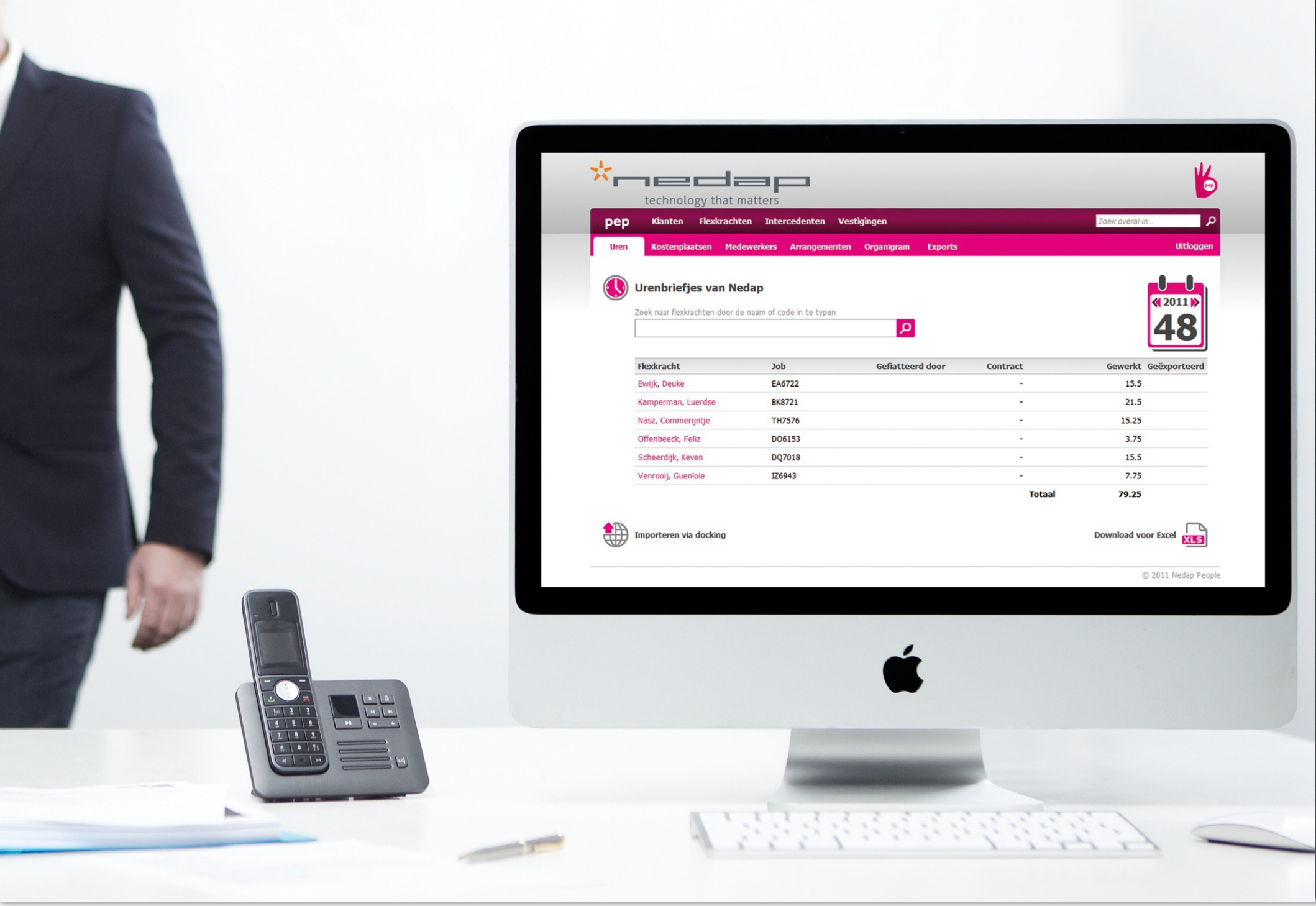
Nog geen vim geïnstalleerd => www.vim.org => download
Geen inet? Vraag de usb stick.


nedap
technology that matters

nedap | Steppingstone

nedap | Steppingstone

nedap | Steppingstone


nedap
technology that matters
pep
Klanten
Flexkrachten
Intercedenten
Vestigingen
Zoek overal in...
Uren
Kostenplaatsen
Medewerkers
Arrangementen
Organigram
Exports
Uitloggen
Urenbriefjes van Nedap
Zoek naar flexkrachten door de naam of code in te typen
2011
48
Flexkracht
Job
Gefiatteerd door
Contract
Gewerkt
Geëxporteerd
Ewijk, Deuke
EA6722
-
15.5
Kamperman, Luerdse
BK8721
-
21.5
Nasz, Commerijntje
TH7576
-
15.25
Offenbeeck, Feliz
DO6153
-
3.75
Scheerdijk, Keven
DQ7018
-
15.5
Venrooij, Guenloie
IZ6943
-
7.75
Totaal
79.25
Importeren via docking
Download voor Excel
XLS
© 2011 Nedap People

nedap | Steppingstone

# STEPPING STONE


BC


BC

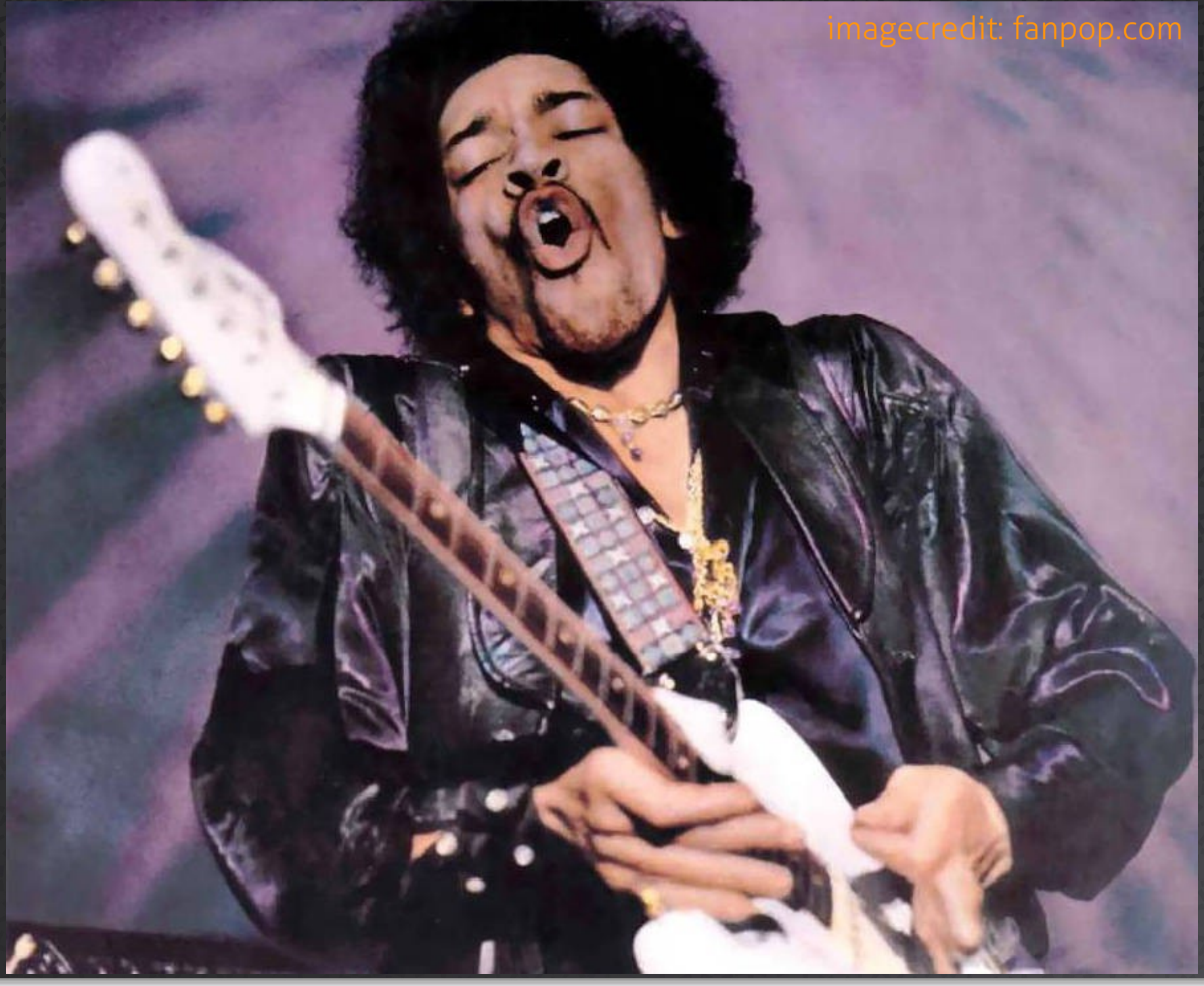

nedap | Steppingstone


Vim

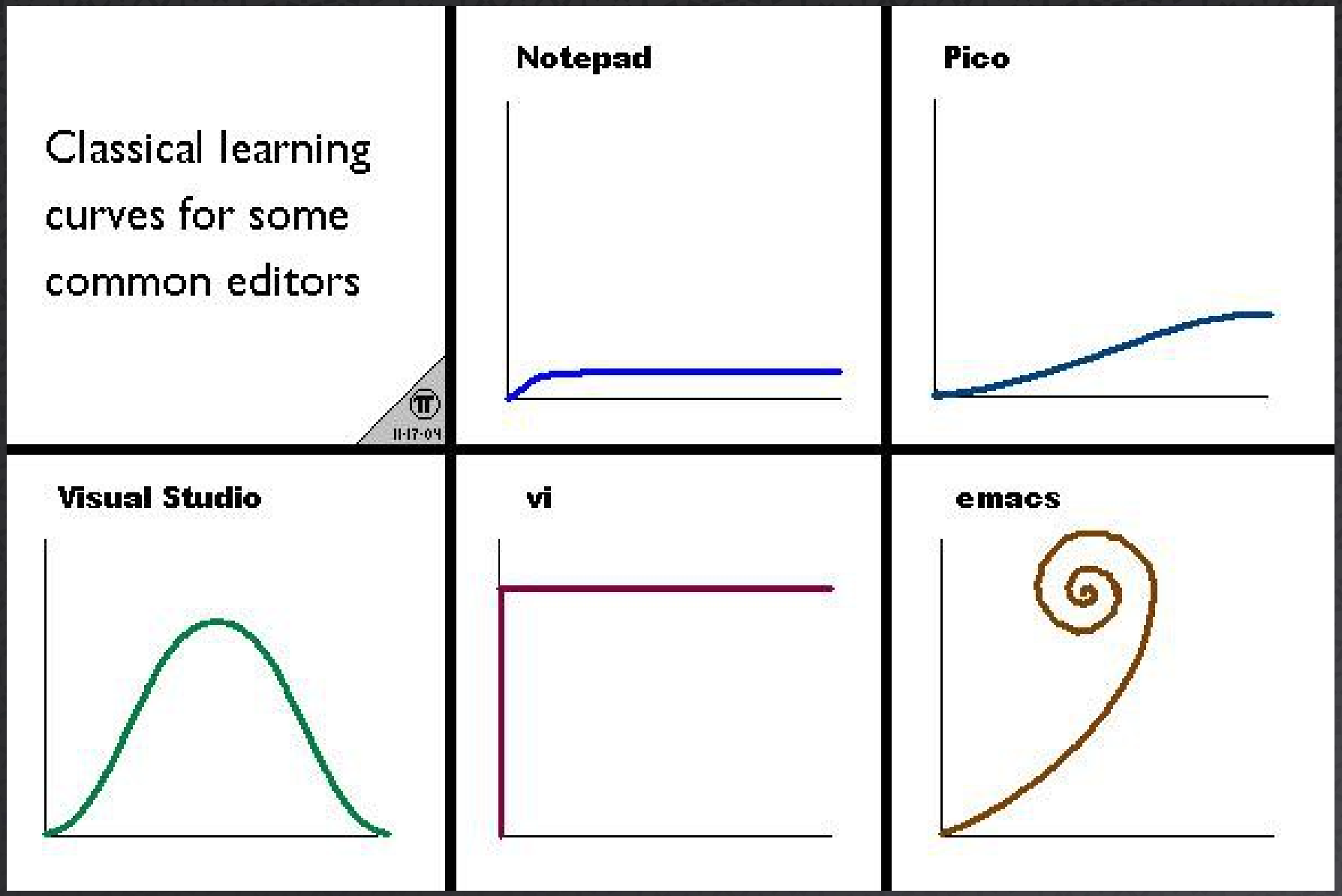
```
Example header
==============

This is header 2
```

```
Example header
==============

This is header 2
================
```

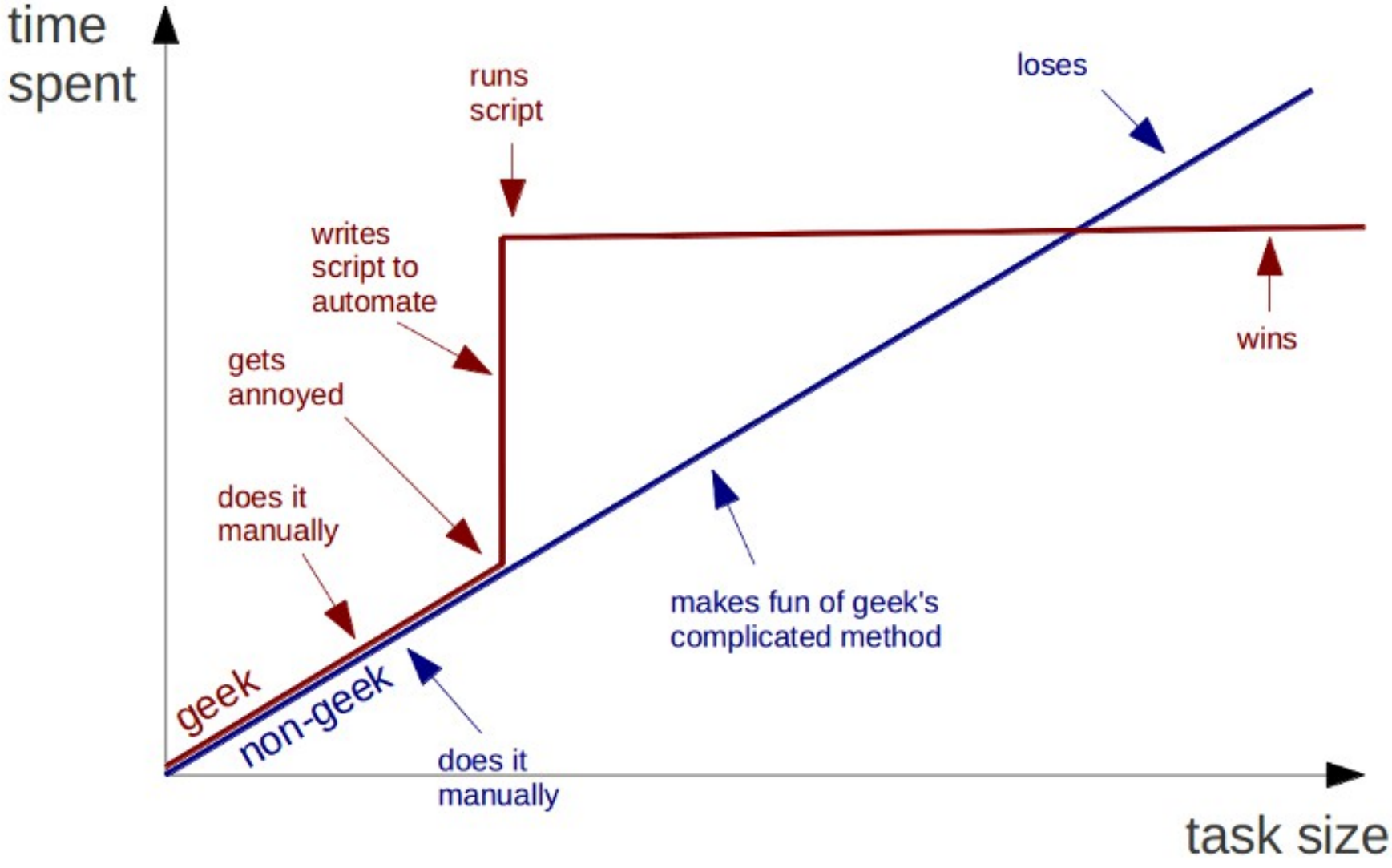
```
Example header
==============

This is header 2
================

This is a somewhat longer header
```


imagecredit: fanpop.com

imagecredit: mlkshk.com



# Geeks and repetitive tasks



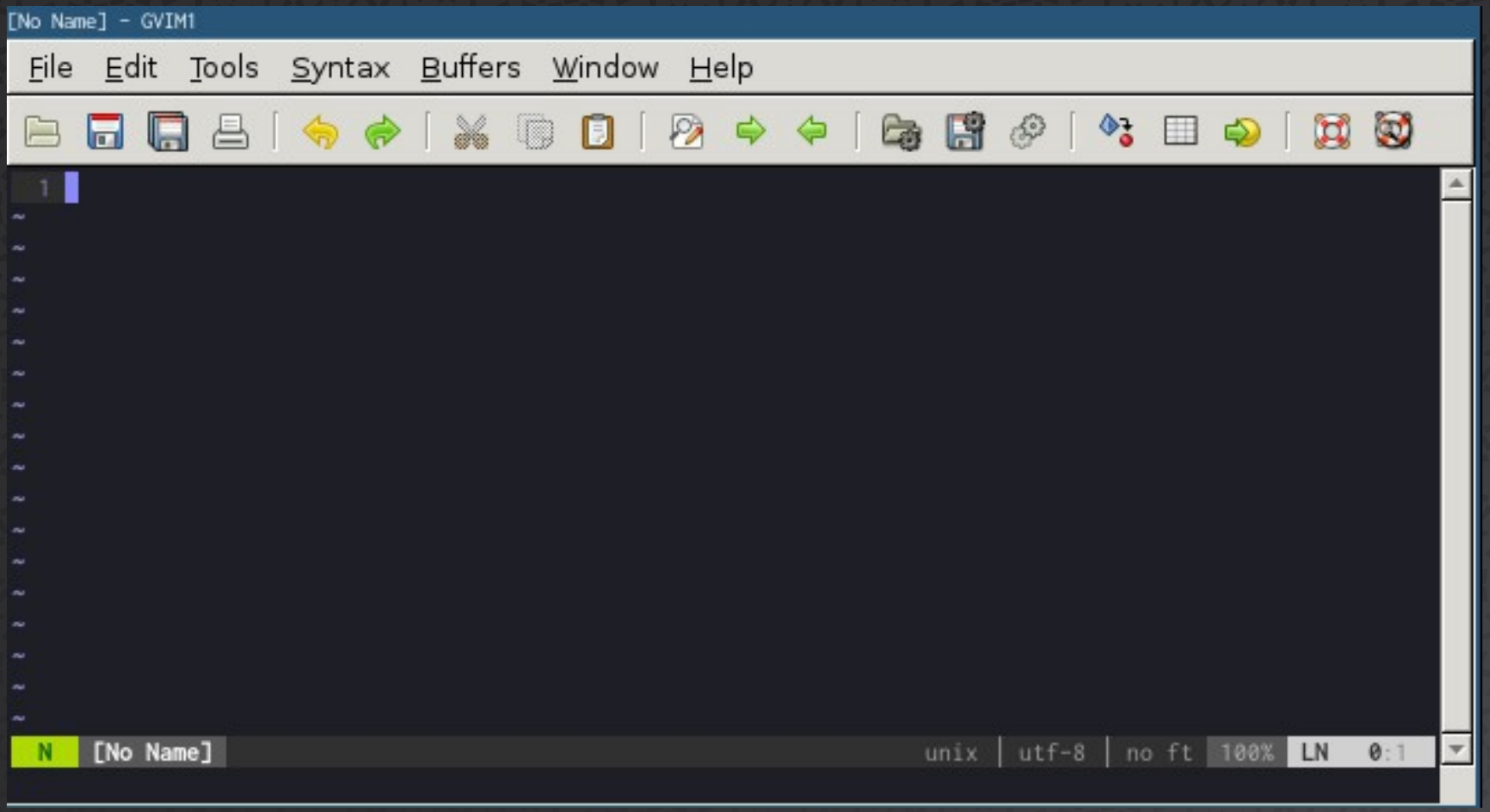
# Vim History


Image credit: http://www.catonmat.net/blog/why-vim-uses-hjkl-as-arrow-keys/

# Don't panic

| | |
|---|---|
| `<esc>` | back to normal mode |
| `:q!` | quit discarding changes |
| `:wq` | write changes & quit (or ZZ, or :x) |
| `:e [file]` | edit [file] (:e! <enter> reload file) |
| `:h` | vim help |

# Vim display


[No Name] - GVIM1
File Edit Tools Syntax Buffers Window Help
1
N [No Name] unix | utf-8 | no ft 100% LN 0:1

# Vim Modes

# Vim modes

* Normal mode (esc) – move cursor/view
  – Text manipulation
* Insert mode (i) – Input of text
* Visual mode (v) – Text selection

# Vim modes

* NORMAL mode <esc>
* INSERT mode
  - i **i**nsert
  - I **i**nsert bol (begin of line)
  - a **a**ppend
  - A **a**ppend eol (end of line)
  - o **o**pen line
  - O **o**pen line above
* VISUAL mode v, V or <ctrl>+v

# Undo / Redo

| | |
|---|---|
| `u` | Undo |
| `<ctrl>+r` | Redo |

# Repeat actions

# repeat action n times

4<action> executes the action 4 times
(example: 4l moves 4 characters to the right)

# Motions

# Motions

h j k l

← ↓ ↑ →

# Exercise

Startup vim (`./startvim`)

- `i f f b r <esc>`
- `a a <esc>`
- `a e <esc>`
- `3 h`
- `i e <esc> .` *(an actual dot)*
- `A k <esc>`
- `I c o <esc>`
- `/a <enter>`
- `xp`

| | reference |
|---|---|
| `i` | insert |
| `a` | append |
| `h` | To left |
| `A` | Append eol |
| `.` | Repeat cmd |
| `I` | Insert bol |
| `/a` | Search “a” |
| `xp` | Delete/paste |

# coffeebreak

# Motions

| | | |
|---|---|---|
| <home> | ^ | begin of line |
| <end> | $ | end of line |
| <ctrl>+<pgup> | gg | begin of file |
| <ctrl>+<pgdn> | G | end of file |

# Motions

| | | |
|---|---|---|
| <pgup> | <ctrl>+f | Page forward |
| <pgdown> | <ctrl>+b | Page backward |
| | { } | Paragraph up/down |
| | zz | Center on cursor |

# Motions

| | |
|---|---|
| w | next **w**ord |
| e | **e**nd of word |
| b | **b**eginning of word |
| :22 | goto line 22 |

# Operators

| | |
|---|---|
| c | **c**hange (has to be followed by motion) |
| C | **C**hange line from cursor to end of line |
| cc | **c**hange entire line |
| d | **d**elete (has to be followed by motion) |
| D | **D**elete from cursor to end of line |
| dd | **d**elete entire line |

# Operator + Motion

# operator + motion

| | |
|---|---|
| `c w` | **c**hange **w**ord |
| `d b` | **d**elete to **b**egin of word |
| `c ^` | **c**hange to bol |
| `d G` | **d**elete to eof |

# Exercise

Startup vim (./startvim)

- open files/1-lorem.txt
- go to 3rd paragraph
- move down 1 line
- change 3rd word to “bla”
- change the following 4 words to “blabla”
- delete the 6th line
- go to the end of the 7th word of 4 paragraphs down and change from there to the beginning of that line.

reference

| | |
|---|---|
| :e [file] | Open [file] |
| { / } | Prev. / next paragraph |
| 4cw | Change 4 words |
| c | change |
| c^ | Change to bol |
| dd | Delete line |
| 7} | Move down 7 paragraphs |
| d G | Delete from cursor to eof. |

# More motions

`f {char}` Find character

`t {char}` to/till character

# operator + motion

| | |
|---|---|
| `c f {char}` | **c**hange till character (incl that char) |
| `d t {char}` | **d**elete to character |
| `v f {char}` | Select till character (incl that char) |

# exercise

Open files/4-boilerplate.html

Line 16: select navigation/extensions and swap them

– line 18: deck.hashes.css should be deck.hash.css

– line 82: same as 18

reference

| | |
|---|---|
| `c f {char}` | **c**hange till character (incl that char) |
| `d t {char}` | **d**elete to character |
| `v f {char}` | Select till character (incl that char) |

# Searching

# Searching

| | |
|---|---|
| `/{string}` | Search string |
| `n` | next occurrence |
| `<shift>+n` `(N)` | previous occurrence (backwards) |

# operator + search

`c / {string}` Change to string

`d / {string}` Delete to string

`v / {string}` Select till 1st char of string!

# exercise

Open files/4-boilerplate.html

Line 16: select navigation/extensions and swap them

– line 18: deck.hashes.css should be deck.hash.css

– line 82: same as 18

– line 71: “Permalink” => uppercase

reference

| | |
|---|---|
| /{char} | Search {char} |
| n | Next occurance |

# Cut/Copy/Paste & visual mode

# Cut/Yank(copy)/Paste

| | |
|---|---|
| y w | **y**ank **w**ord |
| y y | yank line |
| d w | **d**elete **w**ord |
| p | **p**aste after |
| shift+p (P) | **p**aste before |

# Visual mode

| | |
|---|---|
| `v` | Visual mode |
| `shift+v` (`V`) | visual line mode |
| `ctrl+v` | visual block mode |

# Visual mode + motions

| | |
|---|---|
| `v w` | **v**isual select **w**ord |
| `3 shift+v (V)` | **V**isual select 3 lines |
| `ctrl+v G` | **v**isual select a block to eof |

# Visual mode + operators

| | |
|---|---|
| c | **c**hange selection |
| d | **d**elete selection |
| y | **y**ank (copy) selection |
| r | **r**eplace selection |
| p | **p**aste over selection |
| : | Modifier command (example: s/search/replace/) |

# Excercise

Open files/6-digits.txt

– Remove lines 1, 3 and last
– Make a csv file

| | reference |
|---|---|
| v | Visual mode |
| <shift>+v | Visual line mode |
| <ctrl>+v | Visual block mode |
| c | **c**hange selection |
| d | **d**elete selection |
| y | **y**ank (copy) selection |
| r | **r**eplace selection |
| p | **p**aste over selection |
| : | Modifier command (e.g.: s/search/replace/) |

# Search/replace

| | |
|---|---|
| `:s/before/after/` | Replaces the first occurance of the word “before” with “after” on this line. |
| `:s/before/after/g` | Replaces each occurance of the word “before” with after on this line or selection (global) |
| `:%s?/usr/lib?/usr/lib64?g` | Entire file, ? is allowed as well as a delimiter |
| `:24,27:s<ctrl<shift<g` | On lines 24-27, < is also a valid delimiter |

# Excercise

Open files/6-digits.txt

– Remove lines 1, 3 and last
– Make a csv file using search/replace

reference

:s/search/replace/

:%s?search?replace/g

# operator + text object

# Text objects

| | |
|---|---|
| `i` | in/inner |
| `a` | all/around |

# Text objects

| | |
|---|---|
| “ ’ ` ( [ { } ] ) | Delimiters (ci“) |
| w | **w**ord |
| s | **s**entence |
| p | **p**aragraph |
| t | **t**ag (bla) |

# Operator + Text objects

| | |
|---|---|
| `c i "` | **c**hange **i**nner string |
| `d i t` | **d**elete **i**nner **t**ag |
| `y a p` | **y**ank **a**ll **p**aragraph |
| `v a s` | **v**isual select **a**ll **s**entence |
| `c a w` | **c**hange **a**ll **w**ord |

# .vimrc

| | |
|---|---|
| `:so %` | Reload vimrc while editing |
| `:so $MYVIMRC` | General reload |
| `repo:/files/(g)vimrc-alex` | My vimrc... bit messy.<br>(g => gvimrc => if/else thing<br>works only half afaik) |

Tip: store your vimrc in a git repo like so:
https://github.com/aswen/dotvim

# Buffers & Windows

# buffers: bufexplorer

| | |
|---|---|
| `\ b` | open buffer explorer |
| `enter` | switch to selected buffer |
| `d` | delete buffer (close file) |

# Windows

| | |
|---|---|
| `ctrl + w s` | **w**indow **s**plit |
| `ctrl + w v` | **w**indow **v**ertical split |
| `ctrl + w q` | **w**indow **q**uit |
| `ctrl + w h j k l` | **w**indow **f**ocus |

# plugins

Pathogen (github.com/tpope)

/bundle

Tool to make it easy to manage plugins

Git clone > bundle

```
git submodule add ...
git submodule init
git submodule update
```

Easier to use git submodules.
Easier updates.

www.vim.org
(vim website, vim scripts)

vimcasts.org

vimgolf.com
vim-adventures.com


Nedap events:
bubbleconf.com
nedapatlowlands.com


Nedap hires: werkenbijnedap.com