test

kowodo

September 20, 2015

Contents

1 Blbnuti

$$x = \sqrt{b} \tag{1}$$

If $a^2 = b$ and $b = 2$, then the solution must be either

$$a = +\sqrt{2}$$

or

$$a = -\sqrt{2}$$

.

β Ω

Ahoj$_{\text{dole}}$ Ahoj$_{\text{dole}}$ Ahoj$^{\text{nahore}}$ Ahoj$^{\text{nahore}}$

Best site I have ever seen

Neljepsi stranka co znam

2 .emacs

```
(require 'package)
(add-to-list 'package-archives
             '("melpa" . "http://melpa.org/packages/") t)
(when (< emacs-major-version 24)
  ;; For important compatibility libraries like cl-lib
  (add-to-list 'package-archives '("gnu" . "http://elpa.gnu.org/packages/")))
(package-initialize)


;;
;; helm mode
;;
(require 'helm)
(require 'helm-config)

;; The default "C-x c" is quite close to "C-x C-c", which quits Emacs.
;; Changed to "C-c h". Note: We must set "C-c h" globally, because we
;; cannot change 'helm-command-prefix-key' once 'helm-config' is loaded.
(global-set-key (kbd "C-c h") 'helm-command-prefix)
(global-unset-key (kbd "C-x c"))

(define-key helm-map (kbd "<tab>") 'helm-execute-persistent-action) ; rebind tab to run
(define-key helm-map (kbd "C-i") 'helm-execute-persistent-action) ; make TAB works in t
(define-key helm-map (kbd "C-z")  'helm-select-action) ; list actions using C-z

(when (executable-find "curl")
  (setq helm-google-suggest-use-curl-p t))

(setq helm-split-window-in-side-p         t ; open helm buffer inside current window
      helm-move-to-line-cycle-in-source   t ; move to end or beginning of source whe
      helm-ff-search-library-in-sexp      t ; search for library in 'require' and 'de
      helm-scroll-amount                  8 ; scroll 8 lines other window using M-<ne
      helm-ff-file-name-history-use-recentf t)

(helm-mode 1)

;;
;; ace jump mode major function
```

```
;;

(eval-after-load "org"
        '(define-key org-mode-map "\C-c " 'nil)) ; unmap key, was org-table-blank-field

(add-to-list 'load-path "~/.emacs.d/elpa/ace-jump-mode-20140616.115")
(autoload
  'ace-jump-mode
  "ace-jump-mode"
  "Emacs quick move minor mode"
  t)
;; you can select the key you prefer to
(define-key global-map (kbd "C-c SPC") 'ace-jump-mode)


;;
;; enable a more powerful jump back function from ace jump mode
;;
(autoload
  'ace-jump-mode-pop-mark
  "ace-jump-mode"
  "Ace jump back:-)"
  t)
(eval-after-load "ace-jump-mode"
  '(ace-jump-mode-enable-mark-sync))
(define-key global-map (kbd "C-x SPC") 'ace-jump-mode-pop-mark)
```

3 nìjaký pøeklad

1. Jedu vlakem.

2. Ztratil jsem vlak.

3. Ztratil jsem se ve vlaku.

4. Ztratil jsem se v autì.

5. Ztratil jsem se s autem.

6. Ztratil jsem auto.

7. Ztratil jsem letadlo.

8. Ztratil jsem se v letadle.

9. Ztratil jsem se s letadlem.

10. Ztracen cestou do autobusu.

11. Ztracen s autobusem.

12. Ztracen v autobusu.

13. Ztracený autobus.

14. Zatracený kamarád.

15. Zatracený pas dej kamarádovi.

16. Budiž zatracen i s kamarádem.

17. Zatracen pro kamaráda.

18. Ztracen kamarádovi.

19. Zaklínadlo o kamarádovi.

20. Ztracen bez kamaráda.

21. Ztracená kamarádka.

22. Zaklínadlo o kamarádce.

23. Obìtování bez kamarádky.

24. Rituál s kamarádkou.

25. Drogy dejte ke kamarádce.

26. Jehla v paži s kamarádkou.

4 ditaa