

## Generic Emacs Bindings

M-	Meta modifier (usually Alt or Esc key)
C-	Control modifier
S-	Shift modifier
C-x C-c	exit Emacs
C-g	abort current command
M-x command	call function <b>command</b>
C-h b	describe all key bindings

## File Manipulation

C-x C-f	open a new file
C-x C-s	save current file
C-x C-s	save to new file
C-x b	switch between opened files
C-x b TAB	list opened files and switch
C-x k	close current file

## Windows (= horizontal/vertical splitting)

C-x 2	split window horizontally
C-x 3	split window vertically
C-x o	switch focus between splitted windows
C-x 0	unsplit by deleting current window
C-x 1	unsplit by deleting all other windows

## Frames (= window-manager windows)

C-x 5 f	open file in new frame
C-x 5 b	switch to opened file in new frame
C-x 5 0	close current frame

## Navigation (text)

C-a / C-e	beginning/end of <i>line</i>
C-Home / C-End	beginning/end of <i>file</i>
C-← / C-→	backward/forward one <i>word</i>
C-↑ / C-↓	backward/forward one <i>paragraph</i>
M-g g	go to <i>line</i> number
M-x goto-char	go to <i>char</i> number (from file beginning)
C-l	center window on current position

## Editing

C-_	undo last operation
C-t	permute two <i>letters</i>
M-t	permute two <i>words</i>
C-k	cut end of line (as many times as needed)
C-DEL / M-d	cut end of word
C-backspace	cut beginning of word

## Selection (copy and paste)

C-Space	start selecting text
M-w	copy selection
C-w	cut selection
C-y	paste selection
M-y	circulate between selections (after C-y)

## Search and Replace

C-s	incremental forward search
C-r	incremental backward search
M-x replace-string	replace string
M-x replace-regexp	replace regexp
M-%	query before replacing string
C-M-%	query before replacing regexp

## Compiling

C-c C-c	compile
C-x ` / C-g p	go to next error
C-g p	go to previous error

## Usefull functions

M-x revert-buffer	re-read from file
M-x grep	find occurrences
M-x diff	find differences
M-x shell	start shell in a buffer

## Emacs TypeRex Configuration

Just run `cat emacs.append >> ~/.emacs` and edit the settings.

### Sample ~/.emacs file

```
;; Loading TypeRex mode for OCaml files
(add-to-list 'load-path "~/emacs.d")
(add-to-list 'auto-mode-alist
  '("\\.ml[iylp]?" . typerex-mode))
(autoload 'typerex-mode "typerex"
  "Major mode for editing Caml code" t)
(setq ocp-server-command "/home/me/bin/ocp-wizard")

;; TypeRex mode configuration
(setq typerex-in-indent 0)
(setq-default indent-tabs-mode nil)
(setq ocp-menu-trigger [mouse-3])
(add-to-list 'load-path "~/emacs.d/auto-complete-mode")
(setq ocp-auto-complete t)
(setq auto-complete-keys 'ac-keys-backquote-backslash)
(setq ac-auto-start 0)
```

### Sample .typerex file

```
# Sources are in ./src/
src
```

```
# Use libraries $OCAMLLIB/unix and...
I +unix /usr/local/lib/ocamlgraph
```

```
# Do not include these files/units in the project
- src/example.ml test
```

### Sample Makefile modification

```
# Call ocp-ocamlc instead of ocamlc
OCAMLC=ocp-ocamlc
```

## TypeRex Key Bindings

M-x typerex-help	show all bindings
C-o	show new C-o -prefixed commands

## Using type information

(code must be compiled with <code>-annot</code> for the last three)	
C-o c	show type and <b>comments</b> for current ident
C-c C-t	show type of current expr. or pattern
C-c C-l	show binding point of current ident
C-c C-f	show kind of current function call

## Interaction with the Toplevel

C-c C-s	typerex-run-caml
C-x C-e / C-c C-e	typerex-eval-phrase
C-c C-b	typerex-eval-buffer
C-c C-r	typerex-eval-region
C-c `	typerex-interactive-next-error-source
C-c C-k	typerex-kill-caml

## Navigation (OCaml)

C-M-↓ / C-M-n / C-c C-n / C-c C-↓	next phrase beginning
C-M-↑ / C-M-p / C-c C-p / C-c C-↑	previous phrase beginning
C-c home	beginning of enclosing module or class
C-c C-a	switch between .ml/.mli files
C-o a	cycle through connected definitions
C-o d	go to <b>definition</b> of current ident
C-o g	grep current ident
C-o t g	grep current toplevel module

## Inserting Constructions

C-c . l	let . in
C-c . m	match . with
C-c . t	try . with
C-c . f	for . do done
C-c . i	if . then else
C-c . w	while . do done
C-c . b	begin . end
C-c . c	class = object(self) .inherit as super end;;
C-x C-x	Switch between mark and cursor

## Edting

M-q / C-c C-q	typerex-indent-phrase
C-M-h	typerex-mark-phrase
C-M-\	indent-region
C-x n d	typerex-narrow-to-phrase

## Refactoring

C-o r	rename current ident
C-o t r	rename current <b>toplevel</b> module
C-o p	<b>p</b> run references in current module
C-o q	eliminate current open, <b>q</b> ualifying refs.
C-o u	<b>u</b> ndo last multiple-file modification