OCaml Emacs Mode (Tuareg)

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 `command`
`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 b`	switch between opened files
`C-x b TAB`	list opened files and switch
`C-x k`	close current file

Compiling

`C-c C-c`	compile
`C-x '`	goto next error (also `C-g p`)
`C-g p`	goto preivous error

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
`C-x 1`	unsplit current window (if splitted)
`C-x 2`	split window horizontally
`C-x 3`	split window vertically
`C-x o`	switch focus between splitted windows

Navigation

`C-a`	go to beginning of line
`C-e`	go to end of line
`C-Home`	go to beginning of file
`C-End`	go to end of file
`C-Left`	go backward one word
`C-Right`	go forward one word
`C-Up`	go backward one paragraph
`C-Down`	go forward one paragraph
`M-g g`	goto line number
`M-x goto-char`	go to position
`C-l`	center window on current position

Editing

`C-_`	undo last operation
`C-t`	permute two letters
`C-k`	cut end of line
`M-d`	cut end of word
`C-backspace`	cut beginning of word
`M-u`	end of word to uppercase
`M-l`	end of word to lowercase
`C-d`	delete current character
`C-num letter`	print *num* times *letter*

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

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`)

Tuareg Key Bindings

Using type information

You must have compiled your code with the `-annot` option.

`C-mouse-2`	caml-types-explore
`C-c C-t`	caml-types-show-type
unbound	caml-types-show-ident

Edting

`C-c C-q`	tuareg-indent-phrase
`C-c home`	tuareg-move-inside-module-or-class-opening
`C-M-h`	tuareg-mark-phrase
`C-M-\`	indent-region
`C-x n d`	tuareg-narrow-to-phrase

Interaction with the Toplevel

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

Navigation

`C-c C-a`	tuareg-find-alternate-file (switch between .ml/.mli files)
`C-c C-down`	tuareg-next-phrase
`C-M-down`	tuareg-next-phrase
`C-c C-n`	tuareg-next-phrase
`C-M-n`	tuareg-next-phrase
`C-M-p`	tuareg-previous-phrase
`C-c C-up`	tuareg-previous-phrase
`C-c C-p`	tuareg-previous-phrase
`C-M-up`	tuareg-previous-phrase

Help and Completion

`C-c h`	caml-help
`C-c TAB`	tuareg-complete
`C-c [`	ocaml-open-module
`C-c ]`	ocaml-close-module
`C-c i`	ocaml-add-path

Inserting Constructions

`C-c . b`	tuareg-insert-begin-form
`C-c . c`	tuareg-insert-class-form
`C-c . f`	tuareg-insert-for-form
`C-c . i`	tuareg-insert-if-form
`C-c . l`	tuareg-insert-let-form
`C-c . m`	tuareg-insert-match-form
`C-c . t`	tuareg-insert-try-form
`C-c . w`	tuareg-insert-while-form
`C-x C-x`	Switch between mark and cursor

Emacs Configuration

Sample ~/.emacs file

```
(setq auto-mode-alist
    (cons '("\\.ml\\w?" . tuareg-mode)
    auto-mode-alist))
(autoload 'tuareg-mode "tuareg"
    "Major mode for editing Caml code" t)
(autoload 'camldebug "camldebug"
    "Run the Caml debugger" t)
(if (and (boundp 'window-system) window-system)
    (when (string-match "XEmacs" emacs-version)
        (if (not (and (boundp 'mule-x-win-initted)
                mule-x-win-initted))
            (require 'sym-lock))
        (require 'font-lock)))
(add-hook 'tuareg-mode-hook  '(lambda ()
    (define-key tuareg-mode-map "\M-q"
                    'tuareg-indent-phrase)
    (define-key tuareg-mode-map "\C-c \C-i"
                    'caml-types-show-ident)
    (define-key tuareg-mode-map [f4] 'goto-line)
    (define-key tuareg-mode-map [f5] 'compile)
    (define-key tuareg-mode-map [f6] 'recompile)
    (define-key tuareg-mode-map [f7] 'next-error)
    (auto-fill-mode 1)
    (setq tuareg-sym-lock-keywords nil)   ))
(setq column-number-mode t)
(setq mouse-wheel-follow-mouse t)
(setq standard-indent 2)
(transient-mark-mode 1)
(setq visible-bell t)
(setq show-paren-face 'modeline)
(setq global-font-lock-mode t)
(setq global-auto-revert-mode t)
```