

## man egg

---

Access the CHICKEN manual from within the REPL  
Extension for Chicken Scheme  
Version 2.2

Mario Goulart

---

## Table of Contents

<b>1</b>	<b>About this egg .....</b>	<b>1</b>
1.1	Version history .....	1
1.2	Usage .....	1
<b>2</b>	<b>Documentation .....</b>	<b>2</b>
2.1	Procedures .....	2
2.2	Creating prepared documentation from arbitrary texinfo files ...	2
2.3	Using man from Emacs .....	2
<b>3</b>	<b>License .....</b>	<b>4</b>
	<b>Index .....</b>	<b>5</b>

# 1 About this egg

## 1.1 Version history

- 2.2 Adapted to SRFI-69 compatible hash-tables
- 2.1 Bug fixes and small enhancements
- 1.1 Added some information about the file format
- 1.0 Initial release

## 1.2 Usage

Load this egg like so:

```
(require-extension man)
```

## 2 Documentation

This extension searches and displays entries in the CHICKEN User's Manual

### 2.1 Procedures

`man:load` [procedure]

```
(man:load [FILENAME])
```

Loads `adoc` file that has been previously prepared with the `chicken-man` program. If no filename is given, it defaults to `chicken.man`. The optional argument for `man:load`, a file, contains a list of items representing the documentation of an object. Each item (also a list) has the following format: ("OBJECT NAME" "PROTOTYPE" "DESCRIPTION")

Example:

```
((("foo" "(foo FOO-ARG)" "My 'foo' procedure.")
 ("bar" "(bar)" "This is the 'bar' procedure."))
```

`man:help` [procedure]

```
(man:help SYMBOL)
```

Displays the description for the string or symbol `SYMBOL`.

`man:search` [procedure]

```
(man:search PATTERN)
```

Looks up any items that match the regular expression `PATTERN`

`man:search-desc` [procedure]

```
(man:search-desc PATTERN)
```

Looks up any items whose descriptions match the regular expression `PATTERN`

### 2.2 Creating prepared documentation from arbitrary texinfo files

You can use the `chicken-man` program to convert a `.texi` file into a document that can be searched with `man:help`:

```
$ chicken.man foo.texi
$ csi -quiet
#;1> (man:load "foo.doc")
#;2> (man:help ...)
```

### 2.3 Using man from Emacs

Put this code into your `.emacs` file to view the manual section for the current word (there must be a `*scheme*` buffer running somewhere):

```
(defun chicken-help ()
  (interactive)
  (let ((func (current-word)))
    (if func
        (process-send-string "*scheme*" (concat "(man:help \"" func "\"\n")))))
```

```
;; Binds F1 to chicken-help in the scheme-mode  
;; (add-hook 'scheme-mode-hook (lambda () (local-set-key [f1] 'chicken-help)))
```

## 3 License

Copyright (c) 2005, Mario Goulart  
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted under the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the author nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

# Index

man:help.....	2	man:search.....	2
man:load.....	2	man:search-desc.....	2