

Help and Documentation Utilities

Version 5.3.3

February 13, 2013

This section describes utilities designed to connect to documentation and to support bug reports. See also [scribble/xref](#).

1 Searching in the Documentation

```
(require help/search)

(send-main-page [#:sub sub
                 #:notify notify
                 #:fragment fragment
                 #:query query]) → any
sub : path-string? = "index.html"
notify : (-> path? void) = void
fragment : (or/c #f string?) = #f
query : (or/c #f string?) = #f
```

Visits the documentation file *sub* in the user's browser.

This function builds a URL that points into the main collection documentation or into the user-specific documentation, depending on the *sub* argument. Once it finds the path, `send-main-page` passes the path to *notify*. The *fragment* and *query* arguments are passed to `send-url/file`, along with the URL.

```
(perform-search str [context]) → void?
str : string?
context : (or/c #f
           string?
           (list/c string? string?)) = #f
```

Searches for *str* in the documentation. The *context* argument supplies a context for the search or, if it is two strings, a context for the search and a label for that context.

2 Connecting to racket

```
(require help/help-utils)
```

The `help/help-utils` library is dynamically loaded by the `help` form that is available by default in racket.

```
(search-for str) → void?  
  str : (listof string?)
```

Calls `perform-search` after concatenating the elements of `str` and adding spaces between them.

```
(find-help/lib id lib) → void?  
  id : symbol?  
  lib : module-path?
```

Visits the documentation page for `id` as an export of `lib`.

```
(find-help id) → void?  
  id : identifier?
```

Visits the documentation for `id`.

```
(go-to-main-page) → void?
```

Visits the main entry page for the documentation.

3 Bug Reporting

```
(require help/bug-report)
```

```
(help-desk:report-bug [this-bug-id  
                      #:frame-mixin frame-mixin]) → void?  
this-bug-id : #f = (or/c #f exact-positive-integer?)  
frame-mixin : (make-mixin-contract frame%) = values
```

Opens a bug report window to edit the but report identified by *this-bug-id*. If *this-bug-id* is *#f*, then creates a new bug ID and uses that one.

The *frame-mixin* argument is passed the frame class before creating the window.

```
(saved-bug-report-titles/ids) → (listof brinfo?)
```

Returns a list of the saved bug reports.

```
(discard-all-saved-bug-reports) → void?
```

Deletes all of the saved bug reports, except those currently open in frames.

```
(struct brinfo (title id)  
              #:extra-constructor-name make-brinfo  
              #:transparent)  
title : label-string?  
id : number?
```

A record representing a saved bug report. The *id* field is suitable for use with `help-desk:report-bug`, and the *label* field is suitable for use in a GUI control.