NAME

Irccd.File - file I/O API

SYNOPSIS

Irccd.File.SeekCur
Irccd.File.SeekEnd
Irccd.File.SeekSet

Irccd.File.basename(*path*);

Irccd.File.dirname(*path*);

Irccd.File.exists(*path*);

Irccd.File.remove(*path*);

Irccd.File.stat(*path*);

Irccd.File(*path, mode*);

Irccd.File.prototype.basename();

Irccd.File.prototype.close();

Irccd.File.prototype.dirname();

Irccd.File.prototype.lines();

Irccd.File.prototype.read(*amount = undefined*);

Irccd.File.prototype.readline();

Irccd.File.prototype.remove();

Irccd.File.prototype.seek(*type, amount*);

Irccd.File.prototype.stat();

Irccd.File.prototype.tell();

Irccd.File.prototype.write(*data*);

DESCRIPTION

This module provides routines for opening and writing files on the disk.

For convenience, some functions are available as static functions and some as object methods.

CONSTANTS

The following constants properties are defined:

Irccd.File.SeekCur (int)
Seek from the current file position.

Irccd.File.SeekEnd (int)
Seek from end of the file.

Irccd.File.SeekSet (int)
Seek from beginning of the file.

METHODS

The **Irccd.File.basename**() method returns the file basename from *path* as specified in basename(3) POSIX function.

The **Irccd.File.dirname**() method returns the file directory name from *path* as specified in dirname(3) POSIX function.

The **Irccd.File.exists**() method checks if the file specified by *path* exists and returns true to indicate success. Warning: using this function is usually discouraged as it may introduce a possible race condition.

The **Irccd.File.remove**() method removes the file at the given *path*.

The optional **Irccd.File.stat**() method returns information from the file at the specified *path*. The returned object has the following properties. Not all properties are available and you must check its presence before using it.

atime (int)	The last access time.
blksize (int)	The block size.
blocks (int)	The number of blocks.
ctime (int)	The creation time.

dev (int)	The device.
gid (int)	The group numeric id.
ino (int)	The inode.
mode (int)	The mode.
mtime (int)	The modification time.
nlink (int)	The number of hard links.
rdev (int)	No description available.
size (int)	The file size.
uid (int)	The user numeric id.

The **Irccd.File**() function constructor opens a file specified by *path* with the specified *mode* similar to fopen(3) C standard function.

The **Irccd.File.prototype.basename**() method is the synonym of **Irccd.File.basename**() method but the path is taken from the file object.

The **Irccd.File.prototype.close**() method closes the file. It is usually not required as already closed in the destructor.

The **Irccd.File.prototype.dirname**() method is the synonym of **Irccd.File.dirname**() method but the path is taken from the file object.

The **Irccd.File.prototype.lines**() method reads all lines and return an array.

The **Irccd.File.prototype.read**() method reads the specified *amount* of characters or the whole file if undefined.

The **Irccd.File.prototype.readline**() method reads the next line available and returns it. If EOF was reached, returns undefined.

The **Irccd.File.prototype.remove**() method is the synonym of **Irccd.File.remove**() method but the path is taken from the file object.

The **Irccd.File.prototype.seek**() method sets the position in the file to the position *amount*. The type argument must be one of *Irccd.File.SeekCur*, *Irccd.File.SeekEnd* or *Irccd.File.SeekSet*.

The optional **Irccd.File.prototype.stat**() method is the synonym of **Irccd.File.stat**() method but the path is taken from the file object.

The **Irccd.File.prototype.tell**() methods returns the current position in the file.

The **Irccd.File.prototype.write**() method writes the string *data* to the file.

EXCEPTIONS

[Irccd.SystemError]	On I/O errors.
[Error]	On other errors.

SEE ALSO

irccd-api(3)

AUTHORS

The **irccd** daemon was written by David Demelier <*markand@malikania.fr*>.