Title

[M-5] chdir() — Manipulate directories

Syntax

string scalar	`pwd()`
void	`chdir(`*string scalar dirpath*`)`
real scalar	`_chdir(`*string scalar dirpath*`)`
void	`mkdir(`*string scalar dirpath*`)`
void	`mkdir(`*string scalar dirpath*`,` *real scalar public*`)`
real scalar	`_mkdir(`*string scalar dirpath*`)`
real scalar	`_mkdir(`*string scalar dirpath*`,` *real scalar public*`)`
void	`rmdir(`*string scalar dirpath*`)`
real scalar	`_rmdir(`*string scalar dirpath*`)`

Description

`pwd()` returns the full name (path) of the current working directory.

`chdir(`*dirpath*`)` changes the current working directory to *dirpath*. `chdir()` aborts with error if the directory does not exist or the operating system cannot change to it.

`_chdir(`*dirpath*`)` does the same thing but returns 170 (a return code) when `chdir()` would abort. `_chdir()` returns 0 if it is successful.

`mkdir(`*dirpath*`)` and `mkdir(`*dirpath*`,` *public*`)` create directory *dirpath*. `mkdir()` aborts with error if the directory already exists or cannot be created. If *public* $\neq 0$ is specified, the directory is given permissions so that everyone can read it; otherwise, it is given the usual permissions.

`_mkdir(`*dirpath*`)` and `_mkdir(`*dirpath*`,` *public*`)` do the same thing but return 693 (a return code) when `mkdir()` would abort. `_mkdir()` returns 0 if it is successful.

`rmdir(`*dirpath*`)` removes directory *dirpath*. `rmdir()` aborts with error if the directory does not exist, is not empty, or the operating system refuses to remove it.

`_rmdir(`*dirpath*`)` does the same thing but returns 693 (a return code) when `rmdir()` would abort. `_rmdir()` returns 0 if it is successful.

Conformability

`pwd()`:

result:	1×1	

`chdir(`*dirpath*`)`:

dirpath:	1×1	
result:	*void*	

`_chdir(`*dirpath*`)`:

dirpath:	1×1	
result:	1×1	

`mkdir(`*dirpath*`,` *public*`)`:

dirpath:	1×1	
public:	1×1	(optional)
result:	*void*	

`_mkdir(`*dirpath*`,` *public*`)`:

dirpath:	1×1	
public:	1×1	(optional)
result:	1×1	

`rmdir(`*dirpath*`)`:

dirpath:	1×1	
result:	*void*	

`_rmdir(`*dirpath*`)`:

dirpath:	1×1	
result:	1×1	

Diagnostics

`pwd()` never aborts with error, but it can return `""` if the operating system does not know or does not have a name for the current directory (which happens when another process removes the directory in which you are working).

`chdir(`*dirpath*`)` aborts with error if the directory does not exist or the operating system cannot change to it.

`_chdir(`*dirpath*`)` never aborts with error; it returns 0 on success and 170 on failure.

`mkdir(`*dirpath*`)` and `mkdir(`*dirpath*`,` *public*`)` abort with error if the directory already exists or the operating system cannot change to it.

`_mkdir(`*dirpath*`)` and `_mkdir(`*dirpath*`,` *public*`)` never abort with error; they return 0 on success and 693 on failure.

`rmdir(`*dirpath*`)` aborts with error if the directory does not exist, is not empty, or the operating system cannot remove it.

`_rmdir(`*dirpath*`)` never aborts with error; it returns 0 on success and 639 on failure.

Also see

[M-4] **io** — I/O functions