* [Welcome to CodeIgniter](http://docs.google.com/general/welcome.html)
* [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Downloading CodeIgniter](http://docs.google.com/installation/downloads.html)
  + [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Upgrading From a Previous Version](http://docs.google.com/installation/upgrading.html)
  + [Troubleshooting](http://docs.google.com/installation/troubleshooting.html)
* [CodeIgniter Overview](http://docs.google.com/overview/index.html)
  + [Getting Started](http://docs.google.com/overview/getting_started.html)
  + [CodeIgniter at a Glance](http://docs.google.com/overview/at_a_glance.html)
  + [Supported Features](http://docs.google.com/overview/features.html)
  + [Application Flow Chart](http://docs.google.com/overview/appflow.html)
  + [Model-View-Controller](http://docs.google.com/overview/mvc.html)
  + [Architectural Goals](http://docs.google.com/overview/goals.html)
* [Tutorial](http://docs.google.com/tutorial/index.html)
  + [Static pages](http://docs.google.com/tutorial/static_pages.html)
  + [News section](http://docs.google.com/tutorial/news_section.html)
  + [Create news items](http://docs.google.com/tutorial/create_news_items.html)
  + [Conclusion](http://docs.google.com/tutorial/conclusion.html)
* [Contributing to CodeIgniter](http://docs.google.com/contributing/index.html)
  + [Writing CodeIgniter Documentation](http://docs.google.com/documentation/index.html)
  + [Developer’s Certificate of Origin 1.1](http://docs.google.com/DCO.html)
* [General Topics](http://docs.google.com/general/index.html)
  + [CodeIgniter URLs](http://docs.google.com/general/urls.html)
  + [Controllers](http://docs.google.com/general/controllers.html)
  + [Reserved Names](http://docs.google.com/general/reserved_names.html)
  + [Views](http://docs.google.com/general/views.html)
  + [Models](http://docs.google.com/general/models.html)
  + [Helpers](http://docs.google.com/general/helpers.html)
  + [Using CodeIgniter Libraries](http://docs.google.com/general/libraries.html)
  + [Creating Libraries](http://docs.google.com/general/creating_libraries.html)
  + [Using CodeIgniter Drivers](http://docs.google.com/general/drivers.html)
  + [Creating Drivers](http://docs.google.com/general/creating_drivers.html)
  + [Creating Core System Classes](http://docs.google.com/general/core_classes.html)
  + [Creating Ancillary Classes](http://docs.google.com/general/ancillary_classes.html)
  + [Hooks - Extending the Framework Core](http://docs.google.com/general/hooks.html)
  + [Auto-loading Resources](http://docs.google.com/general/autoloader.html)
  + [Common Functions](http://docs.google.com/general/common_functions.html)
  + [Compatibility Functions](http://docs.google.com/general/compatibility_functions.html)
  + [URI Routing](http://docs.google.com/general/routing.html)
  + [Error Handling](http://docs.google.com/general/errors.html)
  + [Caching](http://docs.google.com/general/caching.html)
  + [Profiling Your Application](http://docs.google.com/general/profiling.html)
  + [Running via the CLI](http://docs.google.com/general/cli.html)
  + [Managing your Applications](http://docs.google.com/general/managing_apps.html)
  + [Handling Multiple Environments](http://docs.google.com/general/environments.html)
  + [Alternate PHP Syntax for View Files](http://docs.google.com/general/alternative_php.html)
  + [Security](http://docs.google.com/general/security.html)
  + [PHP Style Guide](http://docs.google.com/general/styleguide.html)
* [Libraries](http://docs.google.com/index.html)
  + [Benchmarking Class](http://docs.google.com/benchmark.html)
  + [Caching Driver](http://docs.google.com/caching.html)
  + [Calendaring Class](http://docs.google.com/calendar.html)
  + [Shopping Cart Class](http://docs.google.com/cart.html)
  + [Config Class](http://docs.google.com/config.html)
  + [Email Class](http://docs.google.com/email.html)
  + [Encrypt Class](http://docs.google.com/encrypt.html)
  + [Encryption Library](http://docs.google.com/encryption.html)
  + [File Uploading Class](http://docs.google.com/file_uploading.html)
  + [Form Validation](http://docs.google.com/form_validation.html)
  + [FTP Class](#gjdgxs)
  + [Image Manipulation Class](http://docs.google.com/image_lib.html)
  + [Input Class](http://docs.google.com/input.html)
  + [Javascript Class](http://docs.google.com/javascript.html)
  + [Language Class](http://docs.google.com/language.html)
  + [Loader Class](http://docs.google.com/loader.html)
  + [Migrations Class](http://docs.google.com/migration.html)
  + [Output Class](http://docs.google.com/output.html)
  + [Pagination Class](http://docs.google.com/pagination.html)
  + [Template Parser Class](http://docs.google.com/parser.html)
  + [Security Class](http://docs.google.com/security.html)
  + [Session Library](http://docs.google.com/sessions.html)
  + [HTML Table Class](http://docs.google.com/table.html)
  + [Trackback Class](http://docs.google.com/trackback.html)
  + [Typography Class](http://docs.google.com/typography.html)
  + [Unit Testing Class](http://docs.google.com/unit_testing.html)
  + [URI Class](http://docs.google.com/uri.html)
  + [User Agent Class](http://docs.google.com/user_agent.html)
  + [XML-RPC and XML-RPC Server Classes](http://docs.google.com/xmlrpc.html)
  + [Zip Encoding Class](http://docs.google.com/zip.html)
* [Database Reference](http://docs.google.com/database/index.html)
  + [Quick Start: Usage Examples](http://docs.google.com/database/examples.html)
  + [Database Configuration](http://docs.google.com/database/configuration.html)
  + [Connecting to a Database](http://docs.google.com/database/connecting.html)
  + [Running Queries](http://docs.google.com/database/queries.html)
  + [Generating Query Results](http://docs.google.com/database/results.html)
  + [Query Helper Functions](http://docs.google.com/database/helpers.html)
  + [Query Builder Class](http://docs.google.com/database/query_builder.html)
  + [Transactions](http://docs.google.com/database/transactions.html)
  + [Getting MetaData](http://docs.google.com/database/metadata.html)
  + [Custom Function Calls](http://docs.google.com/database/call_function.html)
  + [Query Caching](http://docs.google.com/database/caching.html)
  + [Database Manipulation with Database Forge](http://docs.google.com/database/forge.html)
  + [Database Utilities Class](http://docs.google.com/database/utilities.html)
  + [Database Driver Reference](http://docs.google.com/database/db_driver_reference.html)
* [Helpers](http://docs.google.com/helpers/index.html)
  + [Array Helper](http://docs.google.com/helpers/array_helper.html)
  + [CAPTCHA Helper](http://docs.google.com/helpers/captcha_helper.html)
  + [Cookie Helper](http://docs.google.com/helpers/cookie_helper.html)
  + [Date Helper](http://docs.google.com/helpers/date_helper.html)
  + [Directory Helper](http://docs.google.com/helpers/directory_helper.html)
  + [Download Helper](http://docs.google.com/helpers/download_helper.html)
  + [Email Helper](http://docs.google.com/helpers/email_helper.html)
  + [File Helper](http://docs.google.com/helpers/file_helper.html)
  + [Form Helper](http://docs.google.com/helpers/form_helper.html)
  + [HTML Helper](http://docs.google.com/helpers/html_helper.html)
  + [Inflector Helper](http://docs.google.com/helpers/inflector_helper.html)
  + [Language Helper](http://docs.google.com/helpers/language_helper.html)
  + [Number Helper](http://docs.google.com/helpers/number_helper.html)
  + [Path Helper](http://docs.google.com/helpers/path_helper.html)
  + [Security Helper](http://docs.google.com/helpers/security_helper.html)
  + [Smiley Helper](http://docs.google.com/helpers/smiley_helper.html)
  + [String Helper](http://docs.google.com/helpers/string_helper.html)
  + [Text Helper](http://docs.google.com/helpers/text_helper.html)
  + [Typography Helper](http://docs.google.com/helpers/typography_helper.html)
  + [URL Helper](http://docs.google.com/helpers/url_helper.html)
  + [XML Helper](http://docs.google.com/helpers/xml_helper.html)

Toggle Table of Contents

[CodeIgniter](http://docs.google.com/index.html)

* [Welcome to CodeIgniter](http://docs.google.com/general/welcome.html)
* [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Downloading CodeIgniter](http://docs.google.com/installation/downloads.html)
  + [Installation Instructions](http://docs.google.com/installation/index.html)
  + [Upgrading From a Previous Version](http://docs.google.com/installation/upgrading.html)
  + [Troubleshooting](http://docs.google.com/installation/troubleshooting.html)
* [CodeIgniter Overview](http://docs.google.com/overview/index.html)
  + [Getting Started](http://docs.google.com/overview/getting_started.html)
  + [CodeIgniter at a Glance](http://docs.google.com/overview/at_a_glance.html)
  + [Supported Features](http://docs.google.com/overview/features.html)
  + [Application Flow Chart](http://docs.google.com/overview/appflow.html)
  + [Model-View-Controller](http://docs.google.com/overview/mvc.html)
  + [Architectural Goals](http://docs.google.com/overview/goals.html)
* [Tutorial](http://docs.google.com/tutorial/index.html)
  + [Static pages](http://docs.google.com/tutorial/static_pages.html)
  + [News section](http://docs.google.com/tutorial/news_section.html)
  + [Create news items](http://docs.google.com/tutorial/create_news_items.html)
  + [Conclusion](http://docs.google.com/tutorial/conclusion.html)
* [Contributing to CodeIgniter](http://docs.google.com/contributing/index.html)
  + [Writing CodeIgniter Documentation](http://docs.google.com/documentation/index.html)
  + [Developer’s Certificate of Origin 1.1](http://docs.google.com/DCO.html)
* [General Topics](http://docs.google.com/general/index.html)
  + [CodeIgniter URLs](http://docs.google.com/general/urls.html)
  + [Controllers](http://docs.google.com/general/controllers.html)
  + [Reserved Names](http://docs.google.com/general/reserved_names.html)
  + [Views](http://docs.google.com/general/views.html)
  + [Models](http://docs.google.com/general/models.html)
  + [Helpers](http://docs.google.com/general/helpers.html)
  + [Using CodeIgniter Libraries](http://docs.google.com/general/libraries.html)
  + [Creating Libraries](http://docs.google.com/general/creating_libraries.html)
  + [Using CodeIgniter Drivers](http://docs.google.com/general/drivers.html)
  + [Creating Drivers](http://docs.google.com/general/creating_drivers.html)
  + [Creating Core System Classes](http://docs.google.com/general/core_classes.html)
  + [Creating Ancillary Classes](http://docs.google.com/general/ancillary_classes.html)
  + [Hooks - Extending the Framework Core](http://docs.google.com/general/hooks.html)
  + [Auto-loading Resources](http://docs.google.com/general/autoloader.html)
  + [Common Functions](http://docs.google.com/general/common_functions.html)
  + [Compatibility Functions](http://docs.google.com/general/compatibility_functions.html)
  + [URI Routing](http://docs.google.com/general/routing.html)
  + [Error Handling](http://docs.google.com/general/errors.html)
  + [Caching](http://docs.google.com/general/caching.html)
  + [Profiling Your Application](http://docs.google.com/general/profiling.html)
  + [Running via the CLI](http://docs.google.com/general/cli.html)
  + [Managing your Applications](http://docs.google.com/general/managing_apps.html)
  + [Handling Multiple Environments](http://docs.google.com/general/environments.html)
  + [Alternate PHP Syntax for View Files](http://docs.google.com/general/alternative_php.html)
  + [Security](http://docs.google.com/general/security.html)
  + [PHP Style Guide](http://docs.google.com/general/styleguide.html)
* [Libraries](http://docs.google.com/index.html)
  + [Benchmarking Class](http://docs.google.com/benchmark.html)
  + [Caching Driver](http://docs.google.com/caching.html)
  + [Calendaring Class](http://docs.google.com/calendar.html)
  + [Shopping Cart Class](http://docs.google.com/cart.html)
  + [Config Class](http://docs.google.com/config.html)
  + [Email Class](http://docs.google.com/email.html)
  + [Encrypt Class](http://docs.google.com/encrypt.html)
  + [Encryption Library](http://docs.google.com/encryption.html)
  + [File Uploading Class](http://docs.google.com/file_uploading.html)
  + [Form Validation](http://docs.google.com/form_validation.html)
  + [FTP Class](#gjdgxs)
  + [Image Manipulation Class](http://docs.google.com/image_lib.html)
  + [Input Class](http://docs.google.com/input.html)
  + [Javascript Class](http://docs.google.com/javascript.html)
  + [Language Class](http://docs.google.com/language.html)
  + [Loader Class](http://docs.google.com/loader.html)
  + [Migrations Class](http://docs.google.com/migration.html)
  + [Output Class](http://docs.google.com/output.html)
  + [Pagination Class](http://docs.google.com/pagination.html)
  + [Template Parser Class](http://docs.google.com/parser.html)
  + [Security Class](http://docs.google.com/security.html)
  + [Session Library](http://docs.google.com/sessions.html)
  + [HTML Table Class](http://docs.google.com/table.html)
  + [Trackback Class](http://docs.google.com/trackback.html)
  + [Typography Class](http://docs.google.com/typography.html)
  + [Unit Testing Class](http://docs.google.com/unit_testing.html)
  + [URI Class](http://docs.google.com/uri.html)
  + [User Agent Class](http://docs.google.com/user_agent.html)
  + [XML-RPC and XML-RPC Server Classes](http://docs.google.com/xmlrpc.html)
  + [Zip Encoding Class](http://docs.google.com/zip.html)
* [Database Reference](http://docs.google.com/database/index.html)
  + [Quick Start: Usage Examples](http://docs.google.com/database/examples.html)
  + [Database Configuration](http://docs.google.com/database/configuration.html)
  + [Connecting to a Database](http://docs.google.com/database/connecting.html)
  + [Running Queries](http://docs.google.com/database/queries.html)
  + [Generating Query Results](http://docs.google.com/database/results.html)
  + [Query Helper Functions](http://docs.google.com/database/helpers.html)
  + [Query Builder Class](http://docs.google.com/database/query_builder.html)
  + [Transactions](http://docs.google.com/database/transactions.html)
  + [Getting MetaData](http://docs.google.com/database/metadata.html)
  + [Custom Function Calls](http://docs.google.com/database/call_function.html)
  + [Query Caching](http://docs.google.com/database/caching.html)
  + [Database Manipulation with Database Forge](http://docs.google.com/database/forge.html)
  + [Database Utilities Class](http://docs.google.com/database/utilities.html)
  + [Database Driver Reference](http://docs.google.com/database/db_driver_reference.html)
* [Helpers](http://docs.google.com/helpers/index.html)
  + [Array Helper](http://docs.google.com/helpers/array_helper.html)
  + [CAPTCHA Helper](http://docs.google.com/helpers/captcha_helper.html)
  + [Cookie Helper](http://docs.google.com/helpers/cookie_helper.html)
  + [Date Helper](http://docs.google.com/helpers/date_helper.html)
  + [Directory Helper](http://docs.google.com/helpers/directory_helper.html)
  + [Download Helper](http://docs.google.com/helpers/download_helper.html)
  + [Email Helper](http://docs.google.com/helpers/email_helper.html)
  + [File Helper](http://docs.google.com/helpers/file_helper.html)
  + [Form Helper](http://docs.google.com/helpers/form_helper.html)
  + [HTML Helper](http://docs.google.com/helpers/html_helper.html)
  + [Inflector Helper](http://docs.google.com/helpers/inflector_helper.html)
  + [Language Helper](http://docs.google.com/helpers/language_helper.html)
  + [Number Helper](http://docs.google.com/helpers/number_helper.html)
  + [Path Helper](http://docs.google.com/helpers/path_helper.html)
  + [Security Helper](http://docs.google.com/helpers/security_helper.html)
  + [Smiley Helper](http://docs.google.com/helpers/smiley_helper.html)
  + [String Helper](http://docs.google.com/helpers/string_helper.html)
  + [Text Helper](http://docs.google.com/helpers/text_helper.html)
  + [Typography Helper](http://docs.google.com/helpers/typography_helper.html)
  + [URL Helper](http://docs.google.com/helpers/url_helper.html)
  + [XML Helper](http://docs.google.com/helpers/xml_helper.html)

[CodeIgniter](http://docs.google.com/index.html)

* [Docs](http://docs.google.com/index.html) »
* [Libraries](http://docs.google.com/index.html) »
* FTP Class

classic layout

FTP Class[**¶**](#30j0zll)

CodeIgniter’s FTP Class permits files to be transferred to a remote server. Remote files can also be moved, renamed, and deleted. The FTP class also includes a “mirroring” function that permits an entire local directory to be recreated remotely via FTP.

Note

SFTP and SSL FTP protocols are not supported, only standard FTP.

* [Working with the FTP Class](#1fob9te)
  + [Initializing the Class](#3znysh7)
  + [Usage Examples](#2et92p0)
* [Class Reference](#tyjcwt)

[**Working with the FTP Class**](#3dy6vkm)[**¶**](#1fob9te)

[**Initializing the Class**](#1t3h5sf)[**¶**](#3znysh7)

Like most other classes in CodeIgniter, the FTP class is initialized in your controller using the $this->load->library function:

$this->load->library('ftp');

Once loaded, the FTP object will be available using: $this->ftp

[**Usage Examples**](#4d34og8)[**¶**](#2et92p0)

In this example a connection is opened to the FTP server, and a local file is read and uploaded in ASCII mode. The file permissions are set to 755.

$this->load->library('ftp');  
  
$config['hostname'] = 'ftp.example.com';  
$config['username'] = 'your-username';  
$config['password'] = 'your-password';  
$config['debug'] = TRUE;  
  
$this->ftp->connect($config);  
  
$this->ftp->upload('/local/path/to/myfile.html', '/public\_html/myfile.html', 'ascii', 0775);  
  
$this->ftp->close();

In this example a list of files is retrieved from the server.

$this->load->library('ftp');  
  
$config['hostname'] = 'ftp.example.com';  
$config['username'] = 'your-username';  
$config['password'] = 'your-password';  
$config['debug'] = TRUE;  
  
$this->ftp->connect($config);  
  
$list = $this->ftp->list\_files('/public\_html/');  
  
print\_r($list);  
  
$this->ftp->close();

In this example a local directory is mirrored on the server.

$this->load->library('ftp');  
  
$config['hostname'] = 'ftp.example.com';  
$config['username'] = 'your-username';  
$config['password'] = 'your-password';  
$config['debug'] = TRUE;  
  
$this->ftp->connect($config);  
  
$this->ftp->mirror('/path/to/myfolder/', '/public\_html/myfolder/');  
  
$this->ftp->close();

[**Class Reference**](#2s8eyo1)[**¶**](#tyjcwt)

*class* CI\_FTP[¶](#17dp8vu) connect([*$config = array()*])[¶](#3rdcrjn)

|  |  |
| --- | --- |
| Parameters: | * **$config** (*array*) – Connection values |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Connects and logs into to the FTP server. Connection preferences are set by passing an array to the function, or you can store them in a config file.

Here is an example showing how you set preferences manually:

$this->load->library('ftp');  
  
$config['hostname'] = 'ftp.example.com';  
$config['username'] = 'your-username';  
$config['password'] = 'your-password';  
$config['port'] = 21;  
$config['passive'] = FALSE;  
$config['debug'] = TRUE;  
  
$this->ftp->connect($config);

**Setting FTP Preferences in a Config File**

If you prefer you can store your FTP preferences in a config file. Simply create a new file called the ftp.php, add the $config array in that file. Then save the file at *application/config/ftp.php* and it will be used automatically.

**Available connection options**

|  |  |  |
| --- | --- | --- |
| Option name | Default value | Description |
| **hostname** | n/a | FTP hostname (usually something like: ftp.example.com) |
| **username** | n/a | FTP username |
| **password** | n/a | FTP password |
| **port** | 21 | FTP server port number |
| **debug** | FALSE | TRUE/FALSE (boolean): Whether to enable debugging to display error messages |
| **passive** | TRUE | TRUE/FALSE (boolean): Whether to use passive mode |

upload(*$locpath*, *$rempath*[, *$mode = 'auto'*[, *$permissions = NULL*]])[¶](#26in1rg)

|  |  |
| --- | --- |
| Parameters: | * **$locpath** (*string*) – Local file path * **$rempath** (*string*) – Remote file path * **$mode** (*string*) – FTP mode, defaults to ‘auto’ (options are: ‘auto’, ‘binary’, ‘ascii’) * **$permissions** (*int*) – File permissions (octal) |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Uploads a file to your server. You must supply the local path and the remote path, and you can optionally set the mode and permissions. Example:

$this->ftp->upload('/local/path/to/myfile.html', '/public\_html/myfile.html', 'ascii', 0775);

If ‘auto’ mode is used it will base the mode on the file extension of the source file.

If set, permissions have to be passed as an octal value.

download(*$rempath*, *$locpath*[, *$mode = 'auto'*])[¶](#lnxbz9)

|  |  |
| --- | --- |
| Parameters: | * **$rempath** (*string*) – Remote file path * **$locpath** (*string*) – Local file path * **$mode** (*string*) – FTP mode, defaults to ‘auto’ (options are: ‘auto’, ‘binary’, ‘ascii’) |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Downloads a file from your server. You must supply the remote path and the local path, and you can optionally set the mode. Example:

$this->ftp->download('/public\_html/myfile.html', '/local/path/to/myfile.html', 'ascii');

If ‘auto’ mode is used it will base the mode on the file extension of the source file.

Returns FALSE if the download does not execute successfully (including if PHP does not have permission to write the local file).

rename(*$old\_file*, *$new\_file*[, *$move = FALSE*])[¶](#35nkun2)

|  |  |
| --- | --- |
| Parameters: | * **$old\_file** (*string*) – Old file name * **$new\_file** (*string*) – New file name * **$move** (*bool*) – Whether a move is being performed |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Permits you to rename a file. Supply the source file name/path and the new file name/path.

// Renames green.html to blue.html  
$this->ftp->rename('/public\_html/foo/green.html', '/public\_html/foo/blue.html');

move(*$old\_file*, *$new\_file*)[¶](#1ksv4uv)

|  |  |
| --- | --- |
| Parameters: | * **$old\_file** (*string*) – Old file name * **$new\_file** (*string*) – New file name |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Lets you move a file. Supply the source and destination paths:

// Moves blog.html from "joe" to "fred"  
$this->ftp->move('/public\_html/joe/blog.html', '/public\_html/fred/blog.html');

Note

If the destination file name is different the file will be renamed.

delete\_file(*$filepath*)[¶](#44sinio)

|  |  |
| --- | --- |
| Parameters: | * **$filepath** (*string*) – Path to file to delete |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Lets you delete a file. Supply the source path with the file name.

$this->ftp->delete\_file('/public\_html/joe/blog.html');

delete\_dir(*$filepath*)[¶](#2jxsxqh)

|  |  |
| --- | --- |
| Parameters: | * **$filepath** (*string*) – Path to directory to delete |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Lets you delete a directory and everything it contains. Supply the source path to the directory with a trailing slash.

Important

Be VERY careful with this method! It will recursively delete **everything** within the supplied path, including sub-folders and all files. Make absolutely sure your path is correct. Try using list\_files() first to verify that your path is correct.

$this->ftp->delete\_dir('/public\_html/path/to/folder/');

list\_files([*$path = '.'*])[¶](#z337ya)

|  |  |
| --- | --- |
| Parameters: | * **$path** (*string*) – Directory path |
| Returns: | An array list of files or FALSE on failure |
| Return type: | array |

Permits you to retrieve a list of files on your server returned as an array. You must supply the path to the desired directory.

$list = $this->ftp->list\_files('/public\_html/');  
print\_r($list);

mirror(*$locpath*, *$rempath*)[¶](#3j2qqm3)

|  |  |
| --- | --- |
| Parameters: | * **$locpath** (*string*) – Local path * **$rempath** (*string*) – Remote path |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Recursively reads a local folder and everything it contains (including sub-folders) and creates a mirror via FTP based on it. Whatever the directory structure of the original file path will be recreated on the server. You must supply a source path and a destination path:

$this->ftp->mirror('/path/to/myfolder/', '/public\_html/myfolder/');

mkdir(*$path*[, *$permissions = NULL*])[¶](#1y810tw)

|  |  |
| --- | --- |
| Parameters: | * **$path** (*string*) – Path to directory to create * **$permissions** (*int*) – Permissions (octal) |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Lets you create a directory on your server. Supply the path ending in the folder name you wish to create, with a trailing slash.

Permissions can be set by passing an octal value in the second parameter.

// Creates a folder named "bar"  
$this->ftp->mkdir('/public\_html/foo/bar/', 0755);

chmod(*$path*, *$perm*)[¶](#4i7ojhp)

|  |  |
| --- | --- |
| Parameters: | * **$path** (*string*) – Path to alter permissions for * **$perm** (*int*) – Permissions (octal) |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Permits you to set file permissions. Supply the path to the file or directory you wish to alter permissions on:

// Chmod "bar" to 755  
$this->ftp->chmod('/public\_html/foo/bar/', 0755);

changedir(*$path*[, *$suppress\_debug = FALSE*])[¶](#2xcytpi)

|  |  |
| --- | --- |
| Parameters: | * **$path** (*string*) – Directory path * **$suppress\_debug** (*bool*) – Whether to turn off debug messages for this command |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Changes the current working directory to the specified path.

The $suppress\_debug parameter is useful in case you want to use this method as an is\_dir() alternative for FTP.

close()[¶](#1ci93xb)

|  |  |
| --- | --- |
| Returns: | TRUE on success, FALSE on failure |
| Return type: | bool |

Closes the connection to your server. It’s recommended that you use this when you are finished uploading.

[Next](http://docs.google.com/image_lib.html)   [Previous](http://docs.google.com/form_validation.html)

© Copyright 2014 - 2018, British Columbia Institute of Technology. Last updated on Jun 12, 2018.

Built with [Sphinx](http://sphinx-doc.org/) using a [theme](https://github.com/snide/sphinx_rtd_theme) provided by [Read the Docs](https://readthedocs.org).