

libOftp 1.0 (stable)

unix Name	libftp
Project Name	libOftp
Version	1.0 stable
Project Admin	Omni
Coder	Omni
E-Mail	omnipresent@email.it
Download Link	http://sourceforge.net/projects/libftp
HomePage	http://libftp.sourceforge.net

libftp 1.0 (STABLE)

This is an FTP Library (libOftp) coded in C (under unix-like environment) to simplify the connections, the managements and the operations (e.g. GET a file and/or PUT a file) on a FTP servers, to the developers.

Functions Lists	
ftp_main.h	<ul style="list-style-type: none">✓ ftp_connect()✓ ftp_auth()✓ ftp_quit()✓ ftp_pasv()✓ ftp_list()✓ ftp_nlist()✓ ftp_put()✓ ftp_get()✓ ftp_login_handler()✓ ftp_dir_handler()✓ ftp_file_handler()✓ ftp_dir()✓ ftp_type()✓ ftp_noop()✓ ftp_system()✓ ftp_rmd()✓ ftp_mkd()✓ ftp_cwd()✓ ftp_rename()✓ strfnbytes()✓ find_pasv()✓ ftp_DataPort()✓ ftp_path()

❖ int ftp_connect ()

	Name	Informations
Function	ftp_connect()	Function used to connect to a Remote FTP Server.
Library	ftp_main.h	The library where the ftp_connect() is located.
Return Value	int	<ul style="list-style-type: none"> ✓ sock: Socket Descriptor is returned if nothing fails. ✓ HOST_NOT_FOUND: The specified host is unknown. ✓ TRY_AGAIN: Temporary error. ✓ NO_RECOVERY: Unexpected Server Error. ✓ NO_DATA: HostName has not IP Address. ✓ -1: Generic Error.
Argument 1	char *host	HostName or IP address of the FTP Server.
Argument 2	int port	Port number of the service (21).
Argument 3	struct sockaddr_in *ftp_server	Struct filled with arguments passed before. This Struct is important to make the 3-WAY HANDSHAKE and gain the connection with the FTP Server.
Argument 4	int verbose	Used to print on the screen the server reply.

Informations or Tips

- The developer doesn't have to create the socket with socket() because the function ftp_connect() made all; and the socket descriptor is what the developer has to pass to the other functions. (Too easy).
- The ftp_connect() function works also on Host Names and IP Addresses.
- You can also use the external variable struct sockaddr_in *ftp_server by calling it on your sources. (**extern int** struct sockaddr_in *ftp_server)

❖ int ftp_auth ()

	Name	Informations
Function	ftp_auth()	Function used to logging in the Remote FTP Server. (User e Password required)
Library	ftp_main.h	The library where the ftp_auth() is located.
Return Value	int	<ul style="list-style-type: none"> ✓ 0: Successfully connected and logged. ✓ NOT_CONNECTED: You are not connected to the FTP Server. ✓ USR_TOO_LONG: User Name too

		long (major than 50 bytes). ✓ PWD_TOO_LONG: Password too long (major than 50 bytes). ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	char *usr	The User Name (50 bytes).
Argument 3	char *pwd	The Password (50 bytes).
Argument 4	int verbose	Used to print on the screen the server reply.

❖ int ftp_quit ()

	Name	Informations
Function	ftp_quit()	Function used to log-off from the Remote FTP Server.
Library	ftp_main.h	The library where the ftp_quit() is located.
Return Value	int	✓ 0: Connection successfully closed. ✓ NOT_CONNECTED: You are not connected/logged. ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	int verbose	Used to print on the screen the server reply.

❖ int ftp_pasv ()

	Name	Informations
Function	ftp_pasv()	Function used to request the server-DTP to listen on a data port.
Library	ftp_main.h	The library where the ftp_pasv() is located.
Return Value	int	✓ 0: Server-DTP is now in "listen" mode on a data port. ✓ NOT_CONNECTED: You are not connected/logged. ✓ WRONG_DATA_P: Wrong Data Port. ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	int verbose	Used to print on the screen the server reply.

Information or Tips

- The global variable `data_port` is set to 0 if `ftp_pasv()` return an error.
- If 0 is returned from `ftp_pasv()` the data port is an integer containing the **data port**. If the developer has to make other data connection and the `data_port` is 0 has to call again the passive mode with `ftp_pasv()`.

❖ `int ftp_list ()`

	Name	Informations
Function	<code>ftp_list()</code>	Function used to get the list of the files in the current directory.
Library	<code>ftp_main.h</code>	The library where the <code>ftp_list()</code> is located.
Return Value	int	<ul style="list-style-type: none">✓ 0: Files successfully listed.✓ NOT_CONNECTED: You are not connected/logged.✓ WRONG_DATA_P: Wrong Data Port.✓ DATA_CONNECTION_ERROR: Data connection (on server-DTP) error.✓ -1: Generic Error.
Argument 1	int <code>sck</code>	The Socket Descriptor.
Argument 2	int <code>verbose</code>	Used to print on the screen the server reply.

Information or Tips

- When the return value is "DATA_CONNECTION_ERROR" the global variable `data_port` is set to 0. If the developer has to make other data connection has to call again the passive mode with `ftp_pasv()`.
- `ftp_list()` also save a file `LIST.txt`, in the directory where the code is running, with the list of the files.

❖ `int ftp_nlist ()`

	Name	Informations
Function	<code>ftp_nlist()</code>	Function used to get the name list of the files in the current directory.
Library	<code>ftp_main.h</code>	The library where the <code>ftp_nlist()</code> is located.
Return Value	int	<ul style="list-style-type: none">✓ 0: Files successfully listed.✓ NOT_CONNECTED: You are not connected/logged.✓ WRONG_DATA_P: Wrong Data Port.✓ DATA_CONNECTION_ERROR: Data

		connection (on server-DTP) error. ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	int verbose	Used to print on the screen the server reply.

Information or Tips

- When the return value is "DATA_CONNECTION_ERROR" the global variable data_port is set to 0. If the developer has to make other data connection has to call again the passive mode with ftp_pasv().
- ftp_list() also save a file NLIST.txt, in the directory where the code is running, with the list of the files.

❖ int ftp_put ()

	Name	Informations
Function	ftp_put ()	Function used to put a file on the FTP Server.
Library	ftp_main.h	The library where the ftp_put() is located.
Return Value	int	<ul style="list-style-type: none"> ✓ 0: Files successfully sent and stored. ✓ NOT_CONNECTED: You are not connected/logged. ✓ WRONG_DATA_P: Wrong Data Port. ✓ DATA_CONNECTION_ERROR: Data connection (on server-DTP) error. ✓ FLN_TOO_LONG: File Name too long (major than 1024 bytes). ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	char *FileName	Local file name to upload.
Argument 3	int verbose	Used to print on the screen the server reply.

❖ int ftp_get ()

	Name	Informations
Function	ftp_get ()	Function used to get a file from the FTP Server.
Library	ftp_main.h	The library where the ftp_get() is located.
Return Value	int	<ul style="list-style-type: none"> ✓ 0: Files successfully sent and stored. ✓ NOT_CONNECTED: You are not connected/logged. ✓ WRONG_DATA_P: Wrong Data

		Port. ✓ DATA_CONNECTION_ERROR: Data connection (on server-DTP) error. ✓ FLN_TOO_LONG: File Name too long (major than 1024 bytes). ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	char *FileName	Local file name to upload.
Argument 3	int verbose	Used to print on the screen the server reply.

❖ int ftp_dir ()

	Name	Informations
Function	ftp_dir ()	Function used to get the current working directory.
Library	ftp_main.h	The library where the ftp_dir() is located.
Return Value	int	✓ 0: Remote working path found. Global variable path is now setted. ✓ NOT_CONNECTED: You are not connected/logged. ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	int verbose	Used to print on the screen the server reply.

Information or Tips

- Global variable "path" (**extern char** path[1050]) contains the remote working path.
- ftp_dir() calls the function (in **ftp_str.h**) ftp_path that return 0 if the remote working directory is found and path is setted otherwise -1 if an error occur.
- Declare the global variable path in your source code if you want to work on it.

❖ int ftp_type ()

	Name	Informations
Function	ftp_type ()	Function used to change the representation type.
Library	ftp_main.h	The library where the ftp_type() is located.
Return Value	int	✓ 0: Type changed (in this version only IMAGE type is allowed). ✓ NOT_CONNECTED: You are not

		connected/logged. ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	int type_cmd	Choose your representation type: ✓ ASCII Type: 1 ✓ Non-Print Type: 2 ✓ Image Type: 3 ✓ EBCDIC Type: 4 ✓ Telnet Type: 5
Argument 3	int verbose	Used to print on the screen the server reply.

❖ int ftp_system ()

	Name	Informations
Function	ftp_system ()	Function used to find out the operating system at the server.
Library	ftp_main.h	The library where the ftp_system() is located.
Return Value	int	✓ 0: Operating System found and printed on the video. ✓ NOT_CONNECTED: You are not connected/logged. ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	int verbose	Used to print on the screen the server reply.

Information or Tips

- Global variable "system_name" (**extern char** system_name[1050]) contains the remote Operating System Name.

❖ int ftp_rmd ()

	Name	Informations
Function	ftp_rmd ()	Function used to delete a directory.
Library	ftp_main.h	The library where the ftp_rmd() is located.
Return Value	int	✓ 0: Directory deleted ✓ NOT_CONNECTED: You are not connected/logged. ✓ DIR_TOO_LONG: Directory name is too long (major than 1015 bytes). ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	char *Directory	Directory Name.
Argument 3	int verbose	Used to print on the screen the server reply.

❖ `int ftp_del ()`

	Name	Informations
Function	<code>ftp_del ()</code>	Function used to delete a file.
Library	<code>ftp_main.h</code>	The library where the <code>ftp_del()</code> is located.
Return Value	<code>int</code>	<ul style="list-style-type: none"> ✓ 0: File deleted ✓ NOT_CONNECTED: You are not connected/logged. ✓ DIR_TOO_LONG: Directory name is too long (major than 1015 bytes). ✓ -1: Generic Error.
Argument 1	<code>int sck</code>	The Socket Descriptor.
Argument 2	<code>char *FileName</code>	File Name.
Argument 3	<code>int verbose</code>	Used to print on the screen the server reply.

❖ `int ftp_mkd ()`

	Name	Informations
Function	<code>ftp_mkd ()</code>	Function used to make a directory.
Library	<code>ftp_main.h</code>	The library where the <code>ftp_mkd()</code> is located.
Return Value	<code>int</code>	<ul style="list-style-type: none"> ✓ 0: Directory created. ✓ NOT_CONNECTED: You are not connected/logged. ✓ DIR_TOO_LONG: Directory name too long (major than 1015 bytes). ✓ -1: Generic Error.
Argument 1	<code>int sck</code>	The Socket Descriptor.
Argument 2	<code>char *Directory</code>	Directory name that has to be created.
Argument 3	<code>int verbose</code>	Used to print on the screen the server reply.

❖ `int ftp_cwd ()`

	Name	Informations
Function	<code>ftp_cwd ()</code>	Function used to change the current working directory.
Library	<code>ftp_main.h</code>	The library where the <code>ftp_cwd()</code> is located.

Return Value	int	<ul style="list-style-type: none"> ✓ 0: Directory changed. ✓ NOT_CONNECTED: You are not connected/logged. ✓ DIR_TOO_LONG: Directory name too long (major than 1015 bytes). ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	char *Directory	Directory name that will be the new current working directory.
Argument 3	int verbose	Used to print on the screen the server reply.

❖ int ftp_rename ()

	Name	Informations
Function	ftp_rename ()	Function used to rename a file or directory.
Library	ftp_main.h	The library where the ftp_rename() is located.
Return Value	int	<ul style="list-style-type: none"> ✓ 0: Directory renamed. ✓ NOT_CONNECTED: You are not connected/logged. ✓ NAME_TOO_LONG: Old/New rirectory name too long (major than 1015 bytes). ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	char *old_name	Old File/Directory name that has to be changed.
Argument 3	char *new_name	The new File/Directory name that will be changed.
Argument 4	int verbose	Used to print on the screen the server reply.

❖ int ftp_noop ()

	Name	Informations
Function	ftp_noop ()	No Operation Function.
Library	ftp_main.h	The library where the ftp_noop() is located.
Return Value	int	<ul style="list-style-type: none"> ✓ 0: Function success. ✓ -1: Generic Error.
Argument 1	int sck	The Socket Descriptor.
Argument 2	int verbose	Used to print on the screen the server reply.

ftplib

Information

- RFC959: <http://www.faqs.org/rfcs/rfc959.html>
- For any comments, suggestions, information about bugs or whatever please do not hesitate to contact me. Forum/E-mail messages are welcome.
- If you find errors in this document please contact me.