

## Navigation

- [index](#)
- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »
- [The Python Standard Library](#) »
- [19. Internet Data Handling](#) »

## 19.5. `mhlib` — Access to MH mailboxes¶

Deprecated since version 2.6: The `mhlib` module has been removed in Python 3.0. Use the [mailbox](#) instead.

The `mhlib` module provides a Python interface to MH folders and their contents.

The module contains three basic classes, [MH](#), which represents a particular collection of folders, [Folder](#), which represents a single folder, and [Message](#), which represents a single message.

```
class mhlib.MH([path[, profile]])¶
```

[MH](#) represents a collection of MH folders.

```
class mhlib.Folder(mh, name)¶
```

The [Folder](#) class represents a single folder and its messages.

```
class mhlib.Message(folder, number[, name])¶
```

[Message](#) objects represent individual messages in a folder. The `Message` class is derived from [mimetools.Message](#).

### 19.5.1. MH Objects¶

[MH](#) instances have the following methods:

```
MH.error(format[, ...])¶
```

Print an error message – can be overridden.

```
MH.getprofile(key)¶
```

Return a profile entry (`None` if not set).

```
MH.getpath()¶
```

Return the mailbox pathname.

```
MH.getcontext()¶
```

Return the current folder name.

```
MH.setcontext(name)¶
```

Set the current folder name.

```
MH.listfolders()¶
```

Return a list of top-level folders.

```
MH.listallfolders()¶
```

Return a list of all folders.

```
MH.listsubfolders(name)¶
```

Return a list of direct subfolders of the given folder.

```
MH.listallsubfolders(name)¶
```

Return a list of all subfolders of the given folder.

```
MH.makefolder(name)¶
```

Create a new folder.

```
MH.deletefolder(name)¶
```

Delete a folder – must have no subfolders.

```
MH.openfolder(name)¶
```

Return a new open folder object.

### 19.5.2. Folder Objects¶

[Folder](#) instances represent open folders and have the following methods:

```
Folder.error(format[, ...])¶
```

Print an error message – can be overridden.

```
Folder.getfullname()¶
```

Return the folder's full pathname.

`Folder.getsequencesfilename()`[¶](#)

Return the full pathname of the folder's sequences file.

`Folder.getmessagefilename(n)`[¶](#)

Return the full pathname of message *n* of the folder.

`Folder.listmessages()`[¶](#)

Return a list of messages in the folder (as numbers).

`Folder.getcurrent()`[¶](#)

Return the current message number.

`Folder.setcurrent(n)`[¶](#)

Set the current message number to *n*.

`Folder.parsesequence(seq)`[¶](#)

Parse msgs syntax into list of messages.

`Folder.getlast()`[¶](#)

Get last message, or 0 if no messages are in the folder.

`Folder.setlast(n)`[¶](#)

Set last message (internal use only).

`Folder.getsequences()`[¶](#)

Return dictionary of sequences in folder. The sequence names are used as keys, and the values are the lists of message numbers in the sequences.

`Folder.putsequences(dict)`[¶](#)

Return dictionary of sequences in folder name: list.

`Folder.removemessages(list)`[¶](#)

Remove messages in list from folder.

`Folder.refilemessages(list, tofolder)`[¶](#)

Move messages in list to other folder.

`Folder.movemessage(n, tofolder, ton)`[¶](#)

Move one message to a given destination in another folder.

`Folder.copymessage(n, tofolder, ton)`[¶](#)

Copy one message to a given destination in another folder.

### 19.5.3. Message Objects [¶](#)

The [Message](#) class adds one method to those of [mimertools.Message](#):

`Message.openmessage(n)`[¶](#)

Return a new open message object (costs a file descriptor).

## [Table Of Contents](#)

[19.5. mhlb — Access to MH mailboxes](#)

- [19.5.1. MH Objects](#)
- [19.5.2. Folder Objects](#)
- [19.5.3. Message Objects](#)

## Previous topic

[19.4. mailbox — Manipulate mailboxes in various formats](#)

## Next topic

[19.6. mimertools — Tools for parsing MIME messages](#)

## This Page

- [Show Source](#)

## Navigation

- [index](#)
- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »

- [The Python Standard Library](#) »
- [19. Internet Data Handling](#) »

© [Copyright](#) 1990-2010, Python Software Foundation.

The Python Software Foundation is a non-profit corporation. [Please donate.](#)

Last updated on Feb 26, 2010. Created using [Sphinx](#) 0.6.3.