



Rocky Enterprise Linux 9.2 Manual Pages on command 'Mail::Send.3pm'

C:\>man Mail::Send.3pm

Mail::Send(3pm) User Contributed Perl Documentation Mail::Send(3pm)

NAME

Mail::Send - Simple electronic mail interface

SYNOPSIS

```
require Mail::Send;

$msg = Mail::Send->new;
$msg = Mail::Send->new(Subject => 'example', To => 'timbo');

$msg->to('user@host');
$msg->to('user@host', 'user2@example.com');
$msg->subject('example subject');
$msg->cc('user@host');
$msg->bcc('someone@else');

$msg->set($header, @values);
$msg->add($header, @values);
$msg->delete($header);
```

Launch mailer and set headers. The filehandle returned

```
# by open() is an instance of the Mail::Mailer class.  
# Arguments to the open() method are passed to the Mail::Mailer  
# constructor.  
  
$fh = $msg->open; # some default mailer  
$fh = $msg->open('sendmail'); # explicit  
print $fh "Body of message";  
$fh->close      # complete the message and send it  
                or die "couldn't send whole message: $!\n";
```

DESCRIPTION

Mail::Send creates e-mail messages without using the Mail::Header knowledge, which means that all escaping and folding must be done by you! Also: do not forget to escape leading dots. Simplicity has its price.

When you have time, take a look at Mail::Transport which is part of the MailBox suite.

METHODS

Constructors

```
Mail::Send->new(PAIRS)
```

A list of header fields (provided as key-value PAIRS) can be used to initialize the object, limited to the few provided as method: "to", "subject", "cc", and "bcc". For other header fields, use add().

Header fields

```
$obj->add($fieldname, @values)
```

Add values to the list of defined values for the \$fieldname.

```
$obj->bcc(@values)
```

```
$obj->cc(@values)
```

```
$obj->delete($fieldname)
```

```
$obj->set($fieldname, @values)
```

The @values will replace the old values for the \$fieldname. Returned is the LIST of values after modification.

\$obj->subject(@values)

\$obj->to(@values)

Sending

\$obj->open(%options)

The %options are used to initiate a mailer object via Mail::Mailer::new().

Then Mail::Mailer::open() is called with the knowledge collected in this

"Mail::Send" object.

Be warned: this module implements raw smtp, which means that you have to escape lines which start with a dot, by adding one in front.

SEE ALSO

This module is part of the MailTools distribution,

<http://perl.overmeer.net/mailtools/>.

AUTHORS

The MailTools bundle was developed by Graham Barr. Later, Mark Overmeer took over maintenance without commitment to further development.

Mail::Cap by Gisle Aas <aas@oslonett.no>. Mail::Field::AddrList by Peter Orbaek <poe@cit.dk>. Mail::Mailer and Mail::Send by Tim Bunce <Tim.Bunce@ig.co.uk>. For other contributors see ChangeLog.

LICENSE

Copyrights 1995-2000 Graham Barr <gbarr@pobox.com> and 2001-2017 Mark Overmeer <perl@overmeer.net>.

This program is free software; you can redistribute it and/or modify it under the same terms as Perl itself. See <http://www.perl.com/perl/misc/Artistic.html>

