NAME

Mail::Message::Convert::MimeEntity − translate Mail::Message to MIME::Entity vv

INHERITANCE

```
 Mail::Message::Convert::MimeEntity
   is a Mail::Message::Convert
   is a Mail::Reporter
```

SYNOPSIS

```
 use Mail::Message::Convert::MimeEntity;
 my $convert = Mail::Message::Convert::MimeEntity−>new;

 my Mail::Message $msg    = Mail::Message−>new;
 my MIME::Entity  $entity = $convert−>export($msg);

 my MIME::Entity  $entity = MIME::Entity−>new;
 my Mail::Message $msg    = $convert−>from($entity);

 use Mail::Box::Manager;
 my $mgr     = Mail::Box::Manager−>new;
 my $folder  = $mgr−>open(folder => 'Outbox');
 $folder−>addMessage($entity);
```

DESCRIPTION

The MIME::Entity extends Mail::Internet message with multiparts and more methods. The Mail::Message objects are more flexible in how the message parts are stored, and uses separate header and body objects.

Extends "DESCRIPTION" in Mail::Message::Convert.

METHODS

Extends "METHODS" in Mail::Message::Convert.

Constructors

Extends "Constructors" in Mail::Message::Convert.

Mail::Message::Convert::MimeEntity−>**new**(%options)
Inherited, see "METHODS" in Mail::Message::Convert

Converting

Extends "Converting" in Mail::Message::Convert.

`$obj`−>**export**(`$message`, [$parser])
Returns a new MIME::Entity message object based on the information from the `$message`, which is a Mail::Message object.

You may want to supply your own `$parser`, which is a MIME::Parser object, to change the parser flags. Without a `$parser` object, one is created for you, with all the default settings.

If `undef` is passed, in place of a `$message`, then an empty list is returned. When the parsing failes, then MIME::Parser throws an exception.

example:

```
 my $convert = Mail::Message::Convert::MimeEntity−>new;
 my Mail::Message $msg  = Mail::Message−>new;
 my manual MIME::Entity  $copy = $convert−>export($msg);
```

`$obj`−>**from**($mime_object)
Returns a new Mail::Message object based on the information from the specified MIME::Entity. If the conversion fails, the `undef` is returned. If `undef` is passed in place of an OBJECT, then an empty list is returned.

example:

```
  my $convert = Mail::Message::Convert::MimeEntity−>new;
  my MIME::Entity  $msg  = MIME::Entity−>new;
  my Mail::Message $copy = $convert−>from($msg);
```

$obj−>**selectedFields**($head)
Inherited, see "Converting" in Mail::Message::Convert

Error handling

Extends "Error handling" in Mail::Message::Convert.

$obj−>**AUTOLOAD**()
Inherited, see "Error handling" in Mail::Reporter

$obj−>**addReport**($object)
Inherited, see "Error handling" in Mail::Reporter

$obj−>**defaultTrace**([$level]|[$loglevel, $tracelevel]|[$level, $callback])
Mail::Message::Convert::MimeEntity−>**defaultTrace**([$level]|[$loglevel, $tracelevel]|[$level, $callback])
Inherited, see "Error handling" in Mail::Reporter

$obj−>**errors**()
Inherited, see "Error handling" in Mail::Reporter

$obj−>**log**([$level, [$strings]])
Mail::Message::Convert::MimeEntity−>**log**([$level, [$strings]])
Inherited, see "Error handling" in Mail::Reporter

$obj−>**logPriority**($level)
Mail::Message::Convert::MimeEntity−>**logPriority**($level)
Inherited, see "Error handling" in Mail::Reporter

$obj−>**logSettings**()
Inherited, see "Error handling" in Mail::Reporter

$obj−>**notImplemented**()
Inherited, see "Error handling" in Mail::Reporter

$obj−>**report**([$level])
Inherited, see "Error handling" in Mail::Reporter

$obj−>**reportAll**([$level])
Inherited, see "Error handling" in Mail::Reporter

$obj−>**trace**([$level])
Inherited, see "Error handling" in Mail::Reporter

$obj−>**warnings**()
Inherited, see "Error handling" in Mail::Reporter

Cleanup

Extends "Cleanup" in Mail::Message::Convert.

$obj−>**DESTROY**()
Inherited, see "Cleanup" in Mail::Reporter

DIAGNOSTICS

Error: Converting from MIME::Entity but got a $type, return
Error: Package $package does not implement $method.
Fatal error: the specific package (or one of its superclasses) does not implement this method where it should. This message means that some other related classes do implement this method however the class at hand does not. Probably you should investigate this and probably inform the author of the package.

SEE ALSO

This module is part of Mail-Message distribution version 3.012, built on February 11, 2022. Website: *http://perl.overmeer.net/CPAN/*

LICENSE