

Input encoding

mPDF accepts UTF-8 encoded text by default for all functions.

You can use the following to allow you to write html code encoded in other than utf-8 (in functions like WriteHTML()):

```
$mpdf->allow_charset_conversion=true; // Set by default to TRUE
$mpdf->charset_in='windows-1252';
```

Note: *charset_in* requires codes recognised by the PHP function iconv i.e. windows-1252 not win-1252

If *allow_charset_conversion* is **TRUE** mPDF will also read the charset from the HTML header if present e.g.

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

Alternatively, you could convert the html to utf-8 encoding before passing it to mPDF, using any one of the PHP functions:

- `utf8_encode($ansi_encoded_html)` // only converts ISO-8859-1 to UTF-8
- `iconv('windows-1252', 'UTF-8', $ansi_encoded_html)`
- `mb_convert_encoding($ansi_encoded_html, 'UTF-8', 'windows-1252')`

Note the different order of the parameters, and the different codepage names used by the different functions. The codepage names recognised vary from platform to platform, and your PHP configuration.

A list of codepages recognised by iconv() can be found at <http://www.gnu.org/software/libiconv/>

In PHP5 you can list the codepages recognised by mb_ functions using `mb_list_encodings()`.

Also note that each function has different ways of failing if illegal characters are encountered.

Printed on Wed 05 Aug 2015 12:05:13 GMT +0100 (DST)