# mPDF

## Example 52. New mPDF 4-2 features

```
<?php


ini_set("memory_limit","256M");


$html = '
<style>
body {
      font-family: sans-serif;
}
@page {
      margin-top: 2.0cm;
      margin-bottom: 2.0cm;
      margin-left: 2.3cm;
      margin-right: 1.7cm;
      margin-header: 8mm;
      margin-footer: 8mm;
      footer: html_myHTMLFooter;
      background-color:#ffffff;
}

@page :first {
      margin-top: 6.5cm;
      margin-bottom: 2cm;
      header: html_myHTMLHeader;
      footer: _blank;
      resetpagenum: 1;
      background-gradient: linear #FFFFFF #FFFF44 0 0.5 1 0.5;
      background: #ccffff url(bgbarcode.png) repeat-y fixed left top;
}
@page letterhead {
      margin-top: 2.0cm;
      margin-bottom: 2.0cm;
      margin-left: 2.3cm;
      margin-right: 1.7cm;
      margin-header: 8mm;
      margin-footer: 8mm;
      footer: html_myHTMLFooter;
      background-color:#ffffff;
}

@page letterhead :first {
      margin-top: 6.5cm;
      margin-bottom: 2cm;
      header: html_myHTMLHeader;
      footer: _blank;
      resetpagenum: 1;
}
.gradient {
      border:0.1mm solid #220044;
      background-color: #f0f2ff;
      background-gradient: linear #c7cdde #f0f2ff 0 1 0 0.5;
}
.rounded {
      border:0.1mm solid #220044;
      background-color: #f0f2ff;
      background-gradient: linear #c7cdde #f0f2ff 0 1 0 0.5;
      border-radius: 2mm;
      background-clip: border-box;
}
```

```
h4 {
      font-weight: bold;
      margin-top: 1em;
      margin-bottom: 0.3em;
}
div.text {
      padding:1em;
      margin-bottom: 0.25em;
      text-align:justify;
}
div.artificial {
      font-family: arialuni;       /* custom font using MS Arial Unicode  */
}
p { margin-top: 0; }
.code {
      font-family: mono;
      font-size: 9pt;
      background-color: #d5d5d5;
      margin: 1em 1cm;
      padding: 0 0.3cm;
}


</style>

<body>

<!--mpdf

<htmlpageheader name="myHTMLHeader">
<div style="float:right; width: 90; height: 90; text-align: right; vertical-align: bottom; border: 1mm
double #000088"><img src="tiger.png" width="90" /></div>

<div style="color:#0000BB;"><span style="font-weight: bold; font-size: 14pt;">mPDF Version
4.2</span><br />123 Anystreet<br />Your City<br />GD12 4LP<br /><span style="font-size:
15pt;">&#x260e;</span> 01777 123 567
</div>

<div style="clear: both; margin-top: 1cm; text-align: right;">{DATE jS F Y}</div>

</htmlpageheader>

<htmlpagefooter name="myHTMLFooter">
<table width="100%" style="border-top: 0.1mm solid #000000; vertical-align: top; font-size: 9pt;
color: #000055;"><tr>
<td width="25%"></td>
<td width="50%" align="center">See <a href="http://mpdf1.com/manual/index.php">documentation
manual</a> for further details</td>
<td width="25%" align="right">Page {PAGENO} of 9 pages</td>
</tr></table>
</htmlpagefooter>

mpdf-->

<h2>mPDF Version 4.2</h2>
<h2>New Features</h2>

<div class="gradient text">
<ul>
<li>image handling improved</li>
<li>table layout - additional control over resizing</li>
<li>vertical-alignment of images - better support for all CSS types</li>
<li>top and bottom margins collapse between block elements</li>
<li>improved support for CSS line-height</li>
<li>display progress bar whilst generating file</li>
<li>CSS @page selector can be specified when adding a pagebreak</li>
<li>CSS @page selector allows different margins, backgrounds, headers/footers on :first :left and
:right pages</li>
<li>PNG images with alpha channel fully supported</li>
<li>ability to generate italic and bold font variants from base font file</li>
```

```
<li>CJK fonts to embed as subsets</li>
<li>"double" border on block elements</li>
<li>character substitution for missing characters in UTF-8 fonts</li>
<li>direct passing of dynamically produced image data</li>
<li>background-gradient and background-image can now co-exist </li>
</ul>

Note: automatic top- and bottom-margin to accomodate varying header/footer size was introduced in v4.0
but was not highlighted cf. AutoHeaderMargin in the Manual.

</div>
<br />

<div class="gradient text">
<h4>Page backgrounds</h4>
Background images, gradients and/or colours can be used together on the same page. On this page, the
bars on the left hand side are created using a background-image, whilst a background-gradient sets the
background to the whole page.
</div>
<br />

<div class="gradient text" style="background-color: #d9def0; border-style: double; border-
color:#444444; border-width:1mm;">
<h4>CSS "double" border</h4>
Block elements can now use the CSS property: border(style) = double. See also the tiger logo in the
header of this page.
</div>
<br />

<div class="gradient text">
<h4>CJK fonts to embed as subsets</h4>
When writing documents with Chinese, Japanese or Korean characters, mPDF has previously required the
end-user to download Adobe\'s free CJK font pack.
The ability to embed font subsets now makes it feasible to use open license CJK fonts. 2 fonts are now
available to download as an additional font-pack:
<ul>
<li>zn_hannom_a -  contains all characters in the SJIS, BIG-5, and GBK codepages; original file was
Han Nom A font (Hi-res version) from http://vietunicode.sourceforge.net/fonts/fonts_hannom.html</li>
<li>unbatang_0613 - contains all the (Korean) characters in the UHC codepage; original file from from
http://kldp.net/projects/unfonts/download</li>
</ul>
The following characters only added an extra 15kB to the size of this PDF file, and approximately 0.15
seconds extra to compile:<br />
Chinese (traditional) <span style="font-family:zn_hannom_a">'."\xe6\x86\x82\xe9\xac\xb1".'</span> ;
chinese (simplified) <span style="font-family:zn_hannom_a">'."\xe6\x9d\xa5\xe8\x87\xaa".'</span> ;
japanese <span style="font-family:zn_hannom_a">'."\xe3\x81\x9f\xe3\x82\x90".'</span> ; korean <span
style="font-family:unBatang_0613">'."\xed\x82\xa4\xec\x8a\xa4".'</span>
</div>
<br />

<div class="artificial gradient text">
<h4>Artificial Bold and Italic</h4>
The text in this block is in ArialUnicodeMS font. Using embedded subsets it covers most characters you
want to print - BUT it does not have bold, italic, or bold-italic forms.<br />
From version 4.2, mPDF will create "artificial" font styles if they are not available as separate font
files:<br />
<p style="font-weight: bold">The quick brown fox jumps over a lazy dog</p>
<p style="font-style: italic">The quick brown fox jumps over a lazy dog</p>
<p style="font-weight: bold; font-style: italic">The quick brown fox jumps over a lazy dog</p>
</div>
<br />

<div class="gradient text" style="font-family: \'Trebuchet MS\'">
<h4>Character substitution in UTF-8 files</h4>
This paragraph has the font-family set to Trebuchet MS, and the document has the default font set as
DejaVuSansCondensed.
The following characters are not present in the Trebuchet font, and are substituted from the core
Adobe Zapfdingbats font:<br />

&#x2710; &#x2711; &#x2712; &#x2713; &#x2714; &#x2715; &#x2716; &#x2717; &#x2718; &#x2719; &#x271a;
```

```
&#x271b; &#x271c; &#x271d; &#x271e; &#x271f;<br />
The characters are not present in the Trebuchet font, and are substituted from the (default)
DejaVuSansCondensed font:<br />
&#x280; &#x281; &#x282; &#x283; &#x284; &#x285; &#x286; &#x287; &#x288; &#x289; &#x28a; &#x28b;
&#x28c; &#x28d; &#x28e; &#x28f;<br />
Character substitution in UTF-8 files is enabled by setting:
<p class="code">
$mpdf->useSubstitutionsMB = true;
</p>
<div style="color:red; padding:0; margin:0;">NB In mPDF 5.0 this has changed to
<p class="code" style="padding:0; margin:0;">
$mpdf->useSubstitutions = true;
</p>
</div>
It is not recommended to enable this for regular use, as it will add to the processing time.
</div>



<pagebreak />

<h2>Margin-collapse</h2>
<p>mPDF has always allowed margins to be collapsed at the top and bottom of pages. This is specified
by the custom CSS property "margin-collapse: collapse"</p>

<p>mPDF 4.2 also allows margins to collapse between block elements on the page. This is the default
behaviour in browsers, and has been enabled in mPDF 4.2 by default.</p>

<p>In the next 2 paragraphs, the first one has the margin-bottom set to 3em, and the second has the
margin-top set to 0em. So the vertical-space between paragraphs is 3em:</p>

<p class="gradient" style="font-size: 10pt; padding: 0 0.3em; margin-bottom: 3em;">The quick brown fox
jumps over a lazy dog</p>
<p class="gradient" style="font-size: 10pt; padding: 0 0.3em; margin-top: 0em;">The quick brown fox
jumps over a lazy dog</p>

<p>In the next 2 paragraphs, the first one has the margin-bottom set to 2em, and the second has the
margin-top set to 1em. The margins collapse to the larger of the adjoining margins i.e. 2em:</p>

<p class="gradient" style="font-size: 10pt; padding: 0 0.3em; margin-bottom: 2em;">The quick brown fox
jumps over a lazy dog</p>
<p class="gradient" style="font-size: 10pt; padding: 0 0.3em; margin-top: 1em;">The quick brown fox
jumps over a lazy dog</p>


<pagebreak />

<h2>Images</h2>

<h4>PNG Alpha channel</h4>
PNG alpha channel transparency is now fully supported, and works against solid backgrounds, gradients
or background images:
<table>
<tr>
<td><img style="vertical-align: top" src="alpha.png" width="90" /></td>
<td style="background-color:#FFCCFF; "><img style="vertical-align: top" src="alpha.png" width="90"
/></td>
<td style="background-color:#CCFFFF;"><img style="vertical-align: top" src="alpha.png" width="90"
/></td>
<td style="background-color:#CCFFFF; background-gradient: linear #88FFFF #FFFF44 0 0.5 1 0.5; "><img
style="vertical-align: top" src="alpha.png" width="90" /></td>
<td style="background-color:#CCFFFF; background: transparent url(\'bgrock.jpg\') repeat scroll right
top;"><img style="vertical-align: top" src="alpha.png" width="90" /></td>
</tr>
</table>

<br />

<h4>Image Border and padding</h4>
Image padding is now supported as well as border and margin:
```

```
<img src="sunset.jpg" width="100" style="border:3px solid #44FF44; padding: 1em; vertical-align: text-
top; " />
<br />

<h4>Vertical alignment</h4>
<div>From mPDF version 4.2 onwards, most of the values for "vertical-align" are supported: top,
bottom, middle, baseline, text-top, and text-bottom.<br />
<b>Note:</b> The default value for vertical alignment has been changed to baseline, and the default
padding to 0, consistent with most browsers.
</div>
<br />
<div class="gradient" style="font-size: 80%;">
baseline: <img src="sunset.jpg" width="50" style="vertical-align: baseline;" />
text-bottom: <img src="sunset.jpg" width="30" style="vertical-align: text-bottom;" />
middle: <img src="sunset.jpg" width="30" style="vertical-align: middle;" />
bottom: <img src="sunset.jpg" width="80" style="vertical-align: bottom;" />
text-top: <img src="sunset.jpg" width="50" style="vertical-align: text-top;" />
top: <img src="sunset.jpg" width="100" style="vertical-align: top;" />
</div>


<pagebreak />
<h4>Image Alignment</h4>
<div>From mPDF version 4.2 onwards, in-line images can be individually aligned (vertically).
</div>

<div class="gradient" style="margin: 0.5em 0;">
These images <img src="img1.png" style="vertical-align: top;" />
are <img src="img2.png" style="vertical-align: top;" />
<b>top</b> <img src="img3.png" style="vertical-align: top;" />
aligned <img src="img4.png" style="vertical-align: middle;" />
</div>

<div class="gradient" style="margin: 0.5em 0;">
These images <img src="img1.png" style="vertical-align: text-top;" />
are <img src="img2.png" style="vertical-align: text-top;" />
<b>text-top</b> <img src="img3.png" style="vertical-align: text-top;" />
aligned <img src="img4.png" style="vertical-align: middle;" />
</div>

<div class="gradient" style="margin: 0.5em 0;">
These images <img src="img1.png" style="vertical-align: bottom;" />
are <img src="img2.png" style="vertical-align: bottom;" />
<b>bottom</b> <img src="img3.png" style="vertical-align: bottom;" />
aligned <img src="img4.png" style="vertical-align: middle;" />
</div>

<div class="gradient" style="margin: 0.5em 0;">
These images <img src="img1.png" style="vertical-align: text-bottom;" />
are <img src="img2.png" style="vertical-align: text-bottom;" />
<b>text-bottom</b> <img src="img3.png" style="vertical-align: text-bottom;" />
aligned <img src="img4.png" style="vertical-align: middle;" />
</div>

<div class="gradient" style="margin: 0.5em 0;">
These images <img src="img1.png" style="vertical-align: baseline;" />
are <img src="img2.png" style="vertical-align: baseline;" />
<b>baseline</b> <img src="img3.png" style="vertical-align: baseline;" />
aligned <img src="img4.png" style="vertical-align: middle;" />
</div>

<div class="gradient" style="margin: 0.5em 0;">
These images <img src="img1.png" style="vertical-align: middle;" />
are <img src="img2.png" style="vertical-align: middle;" />
<b>middle</b> <img src="img3.png" style="vertical-align: middle;" />
aligned <img src="img5.png" style="vertical-align: bottom;" />
</div>


<pagebreak />
```

```
<h4>Images from PHP</h4>

<br />
<img src="var:smileyface" />
<br />
This image was created with the following code:

<p class="code">
      $img = imagecreatetruecolor(200, 200);<br />
      $white = imagecolorallocate($img, 255, 255, 255);<br />
      $red   = imagecolorallocate($img, 255,   0,   0);<br />
      $green = imagecolorallocate($img,   0, 255,   0);<br />
      $blue  = imagecolorallocate($img,   0,   0, 255);<br />
      imagearc($img, 100, 100, 200, 200,  0, 360, $white);<br />
      imagearc($img, 100, 100, 150, 150, 25, 155, $red);<br />
      imagearc($img,  60,  75,  50,  50,  0, 360, $green);<br />
      imagearc($img, 140,  75,  50,  50,  0, 360, $blue);<br />
      ob_start();<br />
      imagejpeg($img);<br />
      $mpdf->smileyface = ob_get_clean(); <br />
      imagedestroy($img);<br />
</p>
and written to the document using:
<p class="code">
&lt;img src="var:smileyface" /&gt;
</p>


<pagebreak>
<h4>Line-height inheritance</h4>
Line-height inheritance has been altered to follow the CSS2 recommendation:
<ul>
<li>normal is inherited as "normal"</li>
<li>1.2 is inherited as a factor</li>
<li>120% is converted to an actual value and then inherited as the computed value</li>
<li>em is converted to an actual value and then inherited as the computed value</li>
<li>px pt mm are inherited as fixed values</li>
</ul>

<div>Relative values (e.g. 1.3, normal)</div>
<div style="font-size: 12pt; line-height: 2.0; border: 0.2mm solid #880000; background-color: #FFEECC;
padding: 0.3em;">
This DIV has the line-height set as "2.0" and font-size as 12pt. The line-height is therefore 24pt,
but the factor of 2 is inherited...<br />
Nulla felis erat, imperdiet eu, ullamcorper non, nonummy quis, elit. Suspendisse...
<div style="font-size: 8pt; border: 0.2mm solid #880000; background-color: #FFEECC; padding: 0.3em;">
This DIV has the font-size set as 8pt. The line-height of 2 is inherited...<br />
Nulla felis erat, imperdiet eu, ullamcorper non, nonummy quis, elit. Suspendisse potenti. Ut a eros at
ligula vehicula pretium. Maecenas feugiat pede vel risus. Nulla et lectus. Fusce eleifend neque sit
amet erat. Integer consectetuer nulla non orci.
<div style="font-size: 18pt; border: 0.2mm solid #880000; background-color: #FFEECC; padding: 0.3em;">
This DIV has the font-size set as 18pt. The line-height of 2 is inherited...<br />
Nulla felis erat, imperdiet eu, ullamcorper non...
</div>
</div>
</div>
<br />
<div>Absolute values (e.g. 130%, 1.3em, 18pt)</div>
<div style="font-size: 12pt; line-height: 200%; border: 0.2mm solid #880000; background-color:
#FFEECC; padding: 0.3em;">
This DIV has the line-height set as "200%" and font-size as 12pt. The computed line-height of 24pt is
inherited...<br />
Nulla felis erat, imperdiet eu, ullamcorper non, nonummy quis, elit. Suspendisse...
<div style="font-size: 8pt; border: 0.2mm solid #880000; background-color: #FFEECC; padding: 0.3em;">
This DIV has the font-size set as 8pt. The computed line-height of 24pt is inherited...<br />
Nulla felis erat, imperdiet eu, ullamcorper non, nonummy quis, elit. Suspendisse potenti. Ut a eros at
ligula vehicula pretium. Maecenas feugiat pede vel risus. Nulla et lectus. Fusce eleifend neque sit
amet erat. Integer consectetuer nulla non orci.
<div style="font-size: 18pt; border: 0.2mm solid #880000; background-color: #FFEECC; padding: 0.3em;">
This DIV has the font-size set as 18pt. The computed line-height of 24pt is inherited...<br />
```

```
Nulla felis erat, imperdiet eu, ullamcorper non...
</div>
</div>
</div>


<pagebreak />
<h4>Line-height & vertical alignment</h4>
<div>In these examples, top and bottom padding are set to 0, so the block height = line height.</div>
<div>The inline text (set to a larger font-size) inherits the line-height as a factor of the largest
font-size i.e. the line height will expand to reflect the largest font on the line.<br />
Line-height: "normal" (set in mPDF by default as 1.33).</div>
<p class="gradient" style="font-size: 10pt; line-height: normal; padding: 0 0.3em;">Normal text <span
style="font-size: 16pt;">16pt font-size &#194;</span> and normal again</p>

<div>Line-height: 2.0 When using relative line-heights, the text is aligned vertically so that the
centre-line of the line goes throught the middle of the largest font.</div>
<p class="gradient" style="font-size: 10pt; line-height: 2.0; padding: 0 0.3em;">Normal text <span
style="font-size: 16pt;">16pt font-size &#194;</span> and normal again</p>

<div>Line-heights set as a percentages are computed on the base font-size, and are then inherited and
treated the same as absolute lengths. This is also true for "em" values. The line-height of this line
is set as 200% of the paragraph font-size (10pt).<br />
When using absolute line-heights, the text is aligned vertically so that the centre-line of the line
goes throught the middle of the base font.<br />
This means that as far as possible, multiple lines will remain equally spaced<br />
Line-height: 200% </div>
<p class="gradient" style="font-size: 10pt; line-height: 200%; padding: 0 0.3em;">Normal text <span
style="font-size: 16pt;">16pt font-size &#194;</span> and normal again</p>

<div>If the line includes a font-size greater than 1.6 times the computed line-height, then the text
baseline is dropped so that the text will approximately fit within the line-height.
<br />Line-height: 2em</div>
<p class="gradient" style="font-size: 10pt; line-height: 2em; padding: 0 0.3em;">Normal text <span
style="font-size: 18pt;">18pt font-size &#194;</span> and normal again</p>

<div>If the line includes a font-size greater than 2 times the computed line-height, then the line-
height is increased to accommodate the larger fontsize.<br />
Line-height: 2em</div>
<p class="gradient" style="font-size: 10pt; line-height: 2em; padding: 0 0.3em;">Normal text <span
style="font-size: 24pt;">24pt font-size &#194;</span> and normal again</p>

<br />
This broadly reflects the behaviour of IE and Firefox. Note that tall characters such as &#194; may
fall outside the computed line-heights. See the same in an <a href="example52_lineheight.htm">HTML
page</a>.


<pagebreak page-selector="letterhead" />
<h2>Extended use of CSS @page selectors</h2>
The CSS @page selector, together with the pseudo-selectors :first :left :right have increased support
in mPDF 4.2<br />
A named @page can be selected when forcing a new page, e.g. this page was started with:<br />
<span style="font-family: mono; font-size: 9pt;">&lt;pagebreak page-selector="letterhead" /&gt;</span>
<br />
The header and background on this page (and page 1 of the document) are set by the CSS selector: @page
letterhead :first {} whilst subsequent pages have no header, a footer, and no background.
<br />
CSS @page selectors allow different margins, backgrounds, headers/footers to be set on :first :left
and :right pages. Only fixed or mirrored left- and right-margins are supported (i.e. cannot specify
different margins for :left and :right).
<br />
This layout can be used to produce company letters with only the first page on letterheaded paper.

<pagebreak />
<h2>Table Layout control</h2>
<p>mPDF attempts to layout tables according to HTML and CSS specifications. However, because of the
difference between screen and paged media, mPDF resizes tables when necessary to make them fit the
page. This will happen if the minimum table-width is greater than the page-width. Minimum table-width
is defined as the minimum width to accomodate the longest word in each column i.e. words will never be
```

```
split.
</p>
<p>This resizing (minimum-width) can be disabled using a custom CSS property "overflow" on the TABLE
tag. There are 4 options:</p>
&lt;table style="overflow: auto"&gt; (this is the default, using resizing)
<table border="1" style="overflow: auto; border-collapse: collapse; padding: 0.1em; background-color:
#DDFFFF"><tr>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
</tr></table>
<br />
&lt;table style="overflow: visible"&gt; (disables resizing, but allows overflow to show)
<table border="1" style="overflow: visible; border-collapse: collapse; padding: 0.1em; background-
color: #DDFFFF"><tr>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
</tr></table>
<br />

&lt;table style="overflow: hidden"&gt; (disables resizing, and hides/clips any overflow)
<table border="1" style="overflow: hidden; border-collapse: collapse; padding: 0.1em; background-
color: #DDFFFF"><tr>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
</tr></table>
<br />

&lt;table style="overflow: wrap"&gt; (forces words to break as necessary)
<table border="1" style="overflow: wrap; border-collapse: collapse; padding: 0.1em; background-color:
#DDFFFF"><tr>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
<td>Verylongwordwithnospacesinitatall</td>
</tr></table>
<br />



';
if ($_REQUEST['html']) { echo $html; exit; }


//==============================================================
//==============================================================
//==============================================================
define('_MPDF_URI','../');    // required for the progress bar

include("../mpdf.php");

$mpdf=new mPDF('','A4','','',15,15,20,20,5,5);

$mpdf->StartProgressBarOutput(2);      // 2 => advanced mode

$mpdf->SetDisplayMode('fullpage');

$mpdf->useSubstitutions = true;

// Dynamically create image in var:smileyface
      $img = imagecreatetruecolor(200, 200);
      $white = imagecolorallocate($img, 255, 255, 255);
      $red   = imagecolorallocate($img, 255,   0,   0);
      $green = imagecolorallocate($img,   0, 255,   0);
      $blue  = imagecolorallocate($img,   0,   0, 255);
      imagearc($img, 100, 100, 200, 200,  0, 360, $white);
      imagearc($img, 100, 100, 150, 150, 25, 155, $red);
      imagearc($img,  60,  75,  50,  50,  0, 360, $green);
      imagearc($img, 140,  75,  50,  50,  0, 360, $blue);
```

```
      ob_start();
      imagejpeg($img);
      $mpdf->smileyface = ob_get_clean();
      imagedestroy($img);


$mpdf->WriteHTML($html);

$mpdf->Output();
exit;

//==============================================================
//==============================================================
//==============================================================
//==============================================================


?>
```