New Features in mPDF v5.7

Table of Contents

ToC Layout and styling

Table of Contents styling

When a Table of Contents is generated by mPDF using e.g. <tocpagebreak>, mPDF 5.7 will generate the ToC as HTML. This means that a CSS stylesheet can be used to format its appearance.

Example table of contents:

Section 1 5

3|16 .....

1

3|14 .....

2

3|17 .....

2

3|15 .....

3

total_paginas|3 .....

3

Chapter 1 6

Topic 1 7

This will result in the following HTML code generated (internally):

```
<div class="mpdf_toc" id="mpdf_toc_0">
 <div class="mpdf_toc_level_0">
  <a class="mpdf_toc_a" href="#__mpdfinternallink_1">
   <span class="mpdf_toc_t_level_0">Section 1</span>
  </a>
  <dottab outdent="2em" />
  <a class="mpdf_toc_a" href="#__mpdfinternallink_1">
   <span class="mpdf_toc_p_level_0">5</span>
  </a>
 </div>
 <div class="mpdf_toc_level_1">
  <a class="mpdf_toc_a" href="#__mpdfinternallink_2">
   <span class="mpdf_toc_t_level_1">Chapter 1</span>
  </a>
  <dottab outdent="2em" />
  <a class="mpdf_toc_a" href="#__mpdfinternallink_2">
   <span class="mpdf_toc_p_level_1">6</span>
  </a>
 </div>
 <div class="mpdf_toc_level_2">
  <a class="mpdf_toc_a" href="#__mpdfinternallink_3">
   <span class="mpdf_toc_t_level_2">Topic 1</span>
  </a>
  <dottab outdent="2em" />
  <a class="mpdf_toc_a" href="#__mpdfinternallink_3">
```

```
    <span class="mpdf_toc_p_level_2">7</span>
   </a>
  </div>
</div>
```

NB The id is "0" (mpdf_toc_0) for root/un-named ToC; otherwise it is lowercase of the name="" used for the ToC

Example Styling using CSS

The following CSSwill format the ToC as it appears in this document:

```
/* For Table of Contents */
div.mpdf_toc {
  font-family: sans-serif;
  font-size: 11pt;
}
a.mpdf_toc_a {
  text-decoration: none;
  color: black;
}

/* Whole line level 0 */
div.mpdf_toc_level_0 {
  line-height: 1.5;
  margin-left: 0;
  padding-right: 2em;
}

/* Title level 0 - may be inside <a> */
span.mpdf_toc_t_level_0 {
  font-weight: bold;
}

/* Page no. level 0 - may be inside <a> */
span.mpdf_toc_p_level_0 { }

/* Whole line level 1 */
div.mpdf_toc_level_1 {
  margin-left: 2em;
  padding-right: 2em;
}

/* Title level 1 */
span.mpdf_toc_t_level_1 {
  font-style: italic;
  font-weight: bold;
}

/* Page no. level 1 - may be inside <a> */
span.mpdf_toc_p_level_1 { }

/* Whole line level 2 */
div.mpdf_toc_level_2 {
  margin-left: 4em;
  padding-right: 2em;
}

/* Title level 2 */
span.mpdf_toc_t_level_2 { }

/* Page no. level 2 - may be inside <a> */
span.mpdf_toc_p_level_2 { }
```

NB padding-right should match <dottab> "outdent" (0 is default). See <dottab> for more details

The functions TOCpagebreakByArray() and TOCpagebreak() have a new final parameter, and HTML tags <TOC> and <TOCpagebreak> have a new attribute "tocoutdent". This should be blank or a valid CSS length e.g. "2em". See <dottab> for more details.

Text-align on decimal point

Text inside a table column can be aligned on a decimal point (or any other character) by using either HTML attribute or CSS.

This example table uses the following CSS stylesheet:

```
<style>
.aBDP { text-align: "." center; }
.arBDP { text-align: "\66B" center; }
.middot { text-align: "\B7" center ; }
</style>
```

<TD> element	Column
align="left"	Left text
align="right"	Right text
align="center"	Center text
align="char"	1000.0001
align="char" char=","	1000,0001
align="char" char="·"	1000·0001
align="char" char="·"	1000·0001
style="text-align: '.' center"	100.001
style="text-align: '.' center"	DP aligned text
style="text-align: ',' center"	1.000,0001
class="aBDP"	10.01
class="aBDP"	1000
class="aBDP"	1000.0001
class="middot"	1 000·0001
class="aBDP"	1,000,000.00001
class="aBDP"	1.000000001
class="aBDP"	1.000.000.001
class="arBDP"	١،٥٦٧،٢٣٤,٢٣٤٧
class="aBDP"	(GBP) 1,000,000.00001 1,000,000.00001 (EUR) 1,000,000.00001

Automatic ToC and Bookmarks

A Table of Contents and/or Bookmarks can be generated automatically from any of the heading tags H1 - H6. This example will generate ToC and bookmarks from all <h3> tags (top level) and <h4> tags (next level)

```
$mpdf->h2toc = array('H3'=>0, 'H4'=>1);
$mpdf->h2bookmarks = array('H3'=>0, 'H4'=>1);
```

Improved line-breaking

mPDF will now avoid line-breaks in the middle of words even between <tags>

```
<b>Na<sub>2</sub>HCO<sub>3</sub></b>
```

Pellentesque purus feugiat semper. Donec nunc odio, et vitae pellentesque. Pellentesque **Na_2HCO_3** velit lacus.

CSS hyphens

The CSS property hyphens is now supported on all block elements

```
hyphens: manual | auto | none
```

In the following example, the word interdependent contains no soft hyphen or similar characters, and is moved to the next line.

Cum velit lacus pena sociis natoque penatibus et magnis disa montes, nascetur ridicuus interdependent (no characters suggesting line-break).

manual (default) Words are only broken at line breaks where there are characters inside the word that suggest line break opportunities. Characters can be explicit ("-" hard hyphen) or conditional (<wbr>).

Cum velit lacus pena sociis natoque penatibus et magnis disa montes, nascetur ridicuus inter-dependent (uses soft hyphen).

none - Words are not broken at line breaks, even if characters inside the word suggest line break points.

Cum velit lacus pena sociis natoque penatibus et magnis disa montes, nascetur ridicuus inter-dependent (hard hyphen).

auto Words can be broken at appropriate hyphenation points, as determined by characters inside the word.

Cum velit lacus pena sociis natoque penatibus et magnis disa montes, nascetur ridicuus interdepen-dent (No characters suggesting line-break).

SHY inside the word take priority over hyphenation points determined by other resources.

Cum velit lacus pena sociis natoque penatibus et magnis disa montes, nascetur ridicuus interdepen-dent (uses soft hyphen).

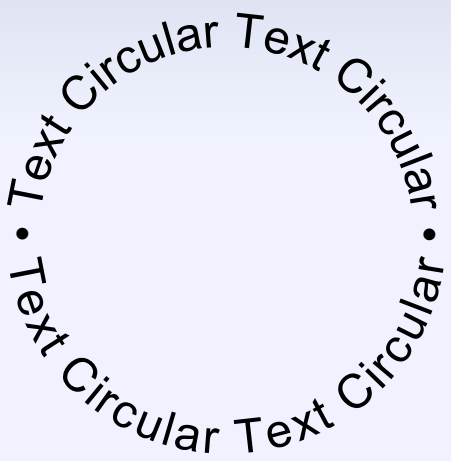
The configurable variables $this->hyphenate and $this->hyphenateTables are henceforth redundant and have no effect.

NB Support for <wbr> is new in mPDF 5.7

Text circle

Added in mPDF 5.6 but not included in New Features example - transparent background and Divider were new. Now also added support for font-size:auto

```
<textcircle r="30mm" top-text="Text Circular Text Circular" bottom-text="Text Circular Text Circular"
divider="&bull;" style="font-size: auto" />
```



List numbering

This list is set to start numbering at 5

```
<ol start="5">
```

5. List item number 1
6. List item number 2
7. List item number 3

<dottab> and outdent

<dottab> now supports a custom CSS property "outdent", which can also be used as an HTML attribute i.e. <dottab outdent="2em">

The first item uses <dottab outdent="4em"> whereas the following items have <dottab class="menu"> (with CSS dottab.menu{outdent: 4em;}) and all have padding-right="4em" on the element

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 £37.00

Fusce eleifend neque sit amet erat. Integer consectetuer nulla non orci. Morbi feugiat £3700.00

Cras odio. Donec mattis, nisi id euismod auctor, neque metus pellentesque risus, at eleifend lacus sapien et risus £27.00

Phasellus metus. Phasellus feugiat, lectus ac aliquam molestie, leo lacus tincidunt turpis, vel aliquam quam odio et sapien. Mauris ante pede, auctor ac, suscipit quis, malesuada sed, nulla. Integer sit amet odio sit amet lectus luctus euismod £7.00

Donec et nulla. Sed quis orci £1137.00

```
p.menu { text-align: justify; padding-right: 4em; }
dottab.menu { outdent: 4em; }
```

NB It is recommended to use   after the dottab if a space is required before the following content.

NB This (outdent) is also used in the Table of Contents (see earlier in this document).

Layers

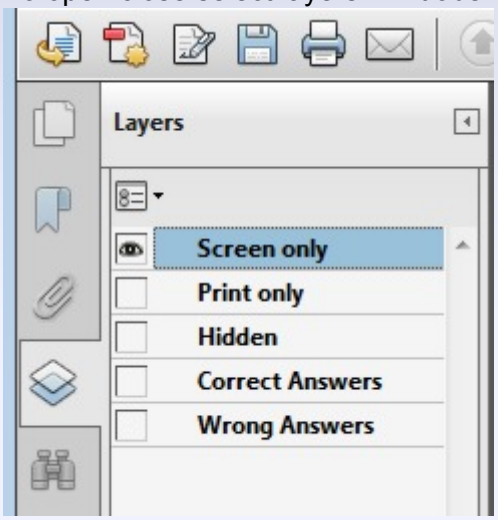
mPDF 5.7 will create layers in the document using the CSS property z-index. All layers are visible by default.

This demonstrates layers in a PDF document. Select the layers named "Correct Answers" or "Wrong Answers" in your PDF viewer (the layers pane should be open already in Adobe Acrobat)

What is the name of the Prime Minister of Britain?

What is the name of the David Beckham's bulldog?

To open/close/select layers in Adobe Reader (10):



The layer names and initial state can be set (optionally) e.g.

```
$mpdf->layerDetails[1]['state']='hidden'; // Set initial state of layer - "hidden" or ""
$mpdf->layerDetails[1]['name']='Correct Answers';
$mpdf->layerDetails[2]['state']='hidden';
$mpdf->layerDetails[2]['name']='Wrong Answers';
```

This is the code used in the example above:

```
<div style="z-index:1;color: green;float: left; width:30%;">Scarlet</div>
<div style="z-index:2;color: red;">Victoria</div>
```

To force the PDF reader to open with the layers tab open, set:

```
$mpdf->open_layer_pane = true;
```

CSS visibility on

CSS visibility:hidden is now supported on inline elements e.g.

This next bit of text is hidden - - and this isn't.

```
style="visibility:hidden;"
```

This next bit of text is only visible in print - - and this isn't.

```
style="visibility:printonly;"
```

This next bit of text is only visible on screen - Hidden text - and this isn't.

```
style="visibility:screenonly;"
```

You can show or hide these elements as for layers (above).

CSS "rem" unit

This line has the font-size set as 1rem

This line has the font-size set as 0.5rem

This line has the font-size set as 1.5rem

This line has the font-size set as 1em

This line has the font-size set as 0.5em

This line has the font-size set as 1.5em

CSS outline

This is text with an outline set by CSS

```
.outlined { text-outline: 0.1mm 0.1mm #FF0000; }
```

CSS background-clip, background-origin & background-size

CSS background-clip, background-origin & background-size are now supported for most block level elements. (Not supported in tables, nor on page/body backgrounds).

background-clip: border-box[default value]
background-origin: padding-box[default value]

background-clip: content-box;
background-origin: content-box

background-clip: padding-box;
background-origin: padding-box

background-clip: border-box;
background-origin: border-box

background-clip: content-box;
background-origin: border-box

background-clip: border-box;
background-origin: content-box