User's Guide » What Else Can I Do?

Fixed position blocks

mPDF >= 4.0

Position

The CSS property "position" is **partially** supported in mPDF, allowing a block element (div etc.) to be placed at a fixed position on the page.

```
position: fixed|absolute
```

absolute - treats the whole physical page as the containing element
fixed - treats the 'printable page' (inside the margins) as the containing element

```
top|left|bottom|right: LENGTH (any valid length unit: em, mm, px, % etc.) DEFAULT = auto
```

rotate - (introduced mPDF 5.0) rotates the block element 90 degrees clockwise or anticlockwise

top|left|bottom|right|width|height are all used to set the size of the block element BEFORE rotation i.e. if the width is not specified, the lft and right margins will be used to calculate the width. After rotation, left|top are used to position the top left corner of the (rotated) element. There is an exception if bottom or right are specified without the respective top or left values; in this case these values are used to position the bottom or right edge of the (rotated) element.

```
90|-90 DEFAULT = BLANK
```

Limitations:

- There is no equivalent concept of the viewport as in the browser (in CSS)
- All positioning is relative to the current page of the document being written
- Position is overridden if it would be off the page, so that the element displays within the containing element
- Overflow:auto causes text to autofit within the block size (additionally constrained if necessary to page edges).
- Fixed-position or floating elements nested inside other fixed-position or floating elements are not supported
- Probably INCOMPATIBLE with keep-with-table, columns etc.
- Annotations were disabled prior to mPDF 5.0

Note: width (including the value 'auto'), height, margin-left -right -top -bottom, padding-left -right -top -bottom, are all supported.

Note: Using overflow:auto can cause mPDF to be very slow, as it attempts to write the same HTML reiteratively until it finds a reasonable fit to the available size.

Overflow
The CSS property "overflow" determines how text is displayed if the block element size is too small for the text.

```
overflow: visible|hidden|auto  DEFAULT = visible
```

visible - all text will show, even if it lies outside the defined block element

hidden - text will be 'clipped' so that overflowing text is not visible

auto - text will be automatically reduced in size if required to fit inside the block element

Example

Example #1

```
<div style="position: absolute; top: 50mm; left: 50mm; width: 100mm;">
This is text in a fixed position block element.
</div>
```

Example #2 - Centres a block in the middle of the page

```
<style>
.myfixed {
    position: absolute;
    overflow: visible;
    left: 0;
    right: 0;
    width: 100mm;   /* you must specify a width */
    margin-top: auto;
    margin-bottom: auto;
    margin-left: auto;
    margin-right: auto;
    border: 1px solid #000088;
    background-color: #EEDDFF;
}
</style>
<div class="myfixed">Cum sociis natoque penatibus et magnis dis parturient montes, nascetur
ridiculus mus. Donec mattis lacus ac purus feugiat semper. Donec aliquet nunc odio, vitae
pellentesque diam. Pellentesque sed velit lacus. Duis quis dui quis sem consectetur
sollicitudin. Cras dolor quam, dapibus et pretium sit amet, elementum vel arcu.</div>
```

Example #3 - Rotated barcode at the bottom right corner of the page

```
<div style="position: fixed; right: 0mm; bottom: 0mm; rotate: -90;">
<barcode code="978-0-9542246-0" class="barcode" />
</div>
```

Printed on Wed 05 Aug 2015 12:08:28 GMT +0100 (DST)