Examples for the Wrap Plugin

Basic syntax

An uppercase **<WRAP>** (or alternatively **<block>** or ** **) creates a **div** and should be used for **“big”** containers, **surrounding** paragraphs, lists, tables, etc.

```
<WRAP classes #id width :language>
"big" content
</WRAP>

or
<block classes #id width :language>
"big" content
</block>

or
<div classes #id width :language>
"big" content
</div>
```

A lowercase **<wrap>** (or alternatively **<inline>** or ****) creates a **span** and should be used for **“small”** containers, **inside** paragraphs, lists, tables, etc.

```
<wrap classes #id width :language>"small" content</wrap>

or
<inline classes #id width :language>"small" content</inline>

or
<span classes #id width :language>"small" content</span>
```

 Please note, some things **won't work with lowercase spans**:

- **alignments** (including alignments generated by changing the text direction)
- **multi-columns**
- and **widths**

if the according wrap isn't floated as well.

A shorthand of uppercase **<WRAP />** and lowercase **<wrap />** are available if the container is empty or unnecessary.

```
<WRAP classes #id />
or
<block classes #id />
or
```

```
<div classes #id />
```

and

```
<wrap classes #id />
or
<inline classes #id />
or
<span classes #id />
```

Classes and Styles

Columns and Floats

You can have columns easily by adding the class `column` and a width, e.g.

```
<WRAP column 30%>...content...</WRAP>
```

The example below uses the following structure:

```
<WRAP group>

<WRAP third column>...content...</WRAP>
<WRAP third column>...content...</WRAP>
<WRAP third column>...content...</WRAP>

</WRAP>
```

Floating Options

Normally you would only need the class `column`, but for more sophisticated uses (not only for columns, but for any other classes, like boxes and notes as well) you can have several kinds of “floats”:

- **`column`** is the same as `left` in LTR languages and the same as `right` in RTL languages
- **`left`** will let you float your wrap on the left
- **`right`** will let the wrap float right
- **`center`** will position the wrap in the horizontal

Widths

You can set any valid widths (but only on divs): `%, px, em, rem, ex, ch, vw, vh, pt, pc, cm, mm, in`, but most of the time you'd only want either

type	e.g.	note
%	30%	makes sense in most cases
px	420px	makes sense if your container contains images with a certain width

Width Keywords

With certain width keywords you can fit your columns automatically to fill the available horizontal space. Those columns will also react to the screen size, so will be responsive and wrap underneath each other on mobile devices.

There are four width keywords. These should not be combined with any other width, only `third` can be combined with `twothirds`.

- **`half`** fits two columns in a row

center of the page

type	e.g.	note
em	20em	makes sense if you like your wrap container to grow and shrink with the font size

- **third** fits three columns in a row
- **quarter** fits four columns in a row
- **twothirds** together with third fits a 2/3 and a 1/3 column in a row

Attention: In order to work properly, wraps with width keywords need an **additional <WRAP group> around a set** of them. If you mix several types in one group, you might need a <WRAP clear/> after a row.

You can use the same options with spans (as each element that floats is automatically a block level element), but it probably doesn't make too much sense. Widths on spans normally do not work (by design), but can make sense, when it is floating.

Attention: What is the difference between widths and width keywords and when is it best to use which? **Widths** can cause problems and will never fully add up, therefore will break the layout under some circumstances. (See box model for a technical explanation.) So, getting widths right will need some fiddling around and testing in various browsers. **Width keywords** on the other hand fit automatically and work better on mobile devices. But the drawback is that they need an extra wrap around them and don't work properly in older browsers (IE8 and under).

All of those options will also work in the boxes and notes wraps (see below).

Old Emulated Headline (deprecated)

When the emulatedHeadlines config option is enabled, every //**__text like this__**// or //**like that**// will create an “emulated headline” within a wrap. This feature is deprecated and will be removed at some point as standard headlines within wraps are supported for a while now.

If that config options is enabled and you need text that is bold and italic, simply use it the other way around: **//No Headline//**.

Multi-columns

Multi-columns work best in modern browsers (no IE9 and below) but should still be considered experimental as some browser behaviour is still inconsistent and buggy.

Just use **colsmall** for small width columns, **colmedium** for medium width columns and **collarge** for large width columns. Or you can use **col2** for 2 columns, **col3** for 3 columns, **col4** for 4 columns and **col5** for 5 columns. The former type of column is determined by its width, the latter by its amount. The width-based columns are ideal for different layouts and device widths.

Note: Multi-columns don't make sense for spans.

Don't use this for bigger columns containing more than just text. Use the columns and floats mentioned above instead.

Alignments

You can use these different text alignments:

- leftalign
- rightalign
- centeralign
- justify

Center aligned text ...

... and right aligned.

```
<WRAP centeralign>
Center aligned text ...
</WRAP>

<WRAP rightalign>
... and right aligned.
</WRAP>
```

You cannot add alignments to spans.

Boxes and Notes

round box 500px center

- box creates a box around the container and uses the colours from the template's style.ini as default colours (__background_alt__ and __text__)
- any of the classes info, tip, important, alert, help, download, todo will add a special note container with a corresponding icon

- the classes `danger`, `warning`, `caution`, `notice`, `safety` use safety colours (and no icons)
- `round` can be added to anything with a background colour or a border and will only work in modern browsers (no IE8 and under)



Info

<WRAP info></WRAP>



Tip

<WRAP tip></WRAP>



Important

<WRAP important></WRAP>



Alert

<WRAP alert></WRAP>



Help

<WRAP round help></WRAP>



Download

<WRAP download></WRAP>



Todo

<WRAP todo></WRAP>

Safety Notes:

Best only use simple markup in safety notes.

Danger

<WRAP danger></WRAP>

Warning

<WRAP warning></WRAP>

Caution

<WRAP caution></WRAP>

Notice

<WRAP round notice></WRAP>

Safety

<WRAP round safety></WRAP>

You can use notes and boxes also inside text with spans like this: info, help, alert, important, tip, download, todo and round box and danger, warning, caution, notice, safety.

```
<wrap info>info</wrap>, <wrap help>help</wrap>, ...
```

Marks

You can mark text as highlighted, less significant and **especially emphasised**.

```
You can mark text as <wrap hi>highlighted</wrap>, <wrap lo>less
significant</wrap> and <wrap em>especially emphasised</wrap>.
```

This might look ugly in some templates and should be adjusted accordingly.

Tabs

You can create a row of tabs by simply wrapping a list of links in <WRAP tabs></WRAP>.

- Some page
- This page

- Another page

```
<WRAP tabs>
  * [[Some page]]
  * [[example|This page]]
  * [[Another page]]
</WRAP>
```


Please note, the styling of these tabs depend on the template you are using and not on the wrap plugin. If you only see a list of links and no tabs, please make sure to add “tabs” to the `noPrefix` config option and that your template supports at least the 2012-01-25 “Angua” DokuWiki release.

Miscellaneous

Clear float

After using any of the float classes, you might come across following text protruding into the space where only the floating containers should be. To prevent that, you should simply add this after your last column:

```
<WRAP clear />
```

Table width

You can set the width of a table via `tablewidth` as every table inside that wrap will always be 100% wide. This makes it possible to give tables any width by adding an additional width to the wrap (or none for 100%).

Table	is
80%	wide

```
<WRAP tablewidth 80%>
^ Table ^ is ^
| 80% | wide |
</WRAP>
```

Indent

This text will appear indented.

```
<wrap indent>This text will appear indented.</wrap>
```

Outdent

This text will appear “outdented”.

```
<wrap outdent>This text will appear "outdented".</wrap>
```

Prewrap

```
Inside this code block the
words will wrap to a new
line although they are all
in one line.
```

```
<WRAP prewrap 250px>
<code>
Inside this code block the words will wrap to a new line although they are
all in one line.
</code>
</WRAP>
```

Spoiler

Here follows a spoiler:

```
Here follows a spoiler: <wrap spoiler>Darth Vader is Luke's father.</wrap>
```

Just select the text in the spoiler box to be able to read its content.

Button links

A link that looks like a button: Syntax

```
A link that looks like a button: <wrap button>[[wiki:Syntax]]</wrap>
```

Hide

The following text is hidden:

```
The following text is hidden: <wrap hide>John, please revise that
sentence.</wrap>
```

 Warning: The text will still appear in the source code, in non-modern browsers and is searchable. Do not hide any security risky secrets with it!

Pagebreak

The following will add a pagebreak:

```
The following will add a pagebreak: <WRAP pagebreak />
```

This has no effect on the browser screen. A pagebreak will force a new page in printouts.

Nopagebreak

The following will try to avoid a pagebreak:

much content, belonging together (like a long table)

```
The following will try to avoid a pagebreak: <WRAP nopagebreak>much content,
belonging together (like a long table)</WRAP>
```

This also has no effect on the browser screen. It will try to avoid a page break in printouts.

Noprint

```
<wrap noprint>This text appears on the screen, but not in print.</wrap>
```

Onlyprint

This text does not appear on the screen, but only in print.

```
<wrap onlyprint>This text does not appear on the screen, but only in
print.</wrap>
```

Combining and Nesting

You can combine and nest all classes and types of boxes, e.g.

```
<WRAP box 350px right :en>
=== Outer box floats right ===

<WRAP 165px left>
Inner nested box floats left and
is partly <wrap em
hi>__em__phasized and
__hi__ghlighted with a nested
<wrap notice>__notice__</wrap>
inside</wrap>.
</WRAP>

Text inside outer right box, but
```

Outer box floats right

Inner nested box floats left and is partly **emphasized and highlighted with a nested notice inside**.

Text inside outer right box, but beneath inner left box.

Round tip box underneath, after a

```
beneath inner left box.

<WRAP clear />

<WRAP round tip>
Round tip box underneath, after a
''clear''.
</WRAP>

</WRAP>
```



clear.

Language and Text Direction

You can change the language and the reading direction of a wrap container by simply adding a colon followed by the language code, like this:

```
<WRAP :he>
זה עברית. ((<wrap :en>This means "This is Hebrew.", at least according to
[[http://translate.google.com/|Google Translate]].</wrap>))
</WRAP>
```

זה עברית. [1]

The text direction (`rtl`, right to left or `ltr`, left to right) will get inserted automatically and is solely dependent on the language. The list of currently supported languages is taken from: http://meta.wikimedia.org/wiki/Template:List_of_language_names_ordered_by_code (If you specify a language not listed there, it simply won't do anything.)

1)
This means “This is Hebrew.”, at least according to Google Translate.

From:
https://www.ousia.it/fildoku/ - **LIBRO FILOSOFIA**

Permanent link:
https://www.ousia.it/fildoku/doku.php?id=wrap_plugin_example

Last update: **2024/11/21 19:19**