

LectureDoc2

Cheat Sheet for DHBW Corporate Design

Information

Author: Michael Eichberg
Version: 2024-02-26

Vertical Titles

Add the class `vertical-title` to rotate the title and to change the layout of the slide to a column-based layout. To get back to a row based layout add a container with the class `width-100`.

Example

```
.. class:: vertical-title

<Slide Title>

.. container:: width-100

<Row based content layout.>
```

Slide Tweaks

Slide without Title

To hide the title of a slide assign the class `no-title`.

Smaller Slide Titles

Use the class `smaller-slide-title`.

Vertically Centered Content

Use the class `center-child-elements`.

Example

```
.. class:: center-child-elements
no-title
smaller-slide-title

Hidden on the slide!

<Slide Content>
```

Sections and Subsections

Create a slide that marks the beginning of a new section or subsection by adding the class `new-section` or `new-subsection` to the slide.

Exercises and Solutions

Use `integrated-exercise` for slides with exercises. Solutions can be added by using the custom directive `protected-exercise-solution`

Example (Solution in supplemental information)

```
.. admonition:: Solution
: class: supplemental
exercise-solution
```

Example (Solution requires password.)

```
.. class:: integrated-exercise
```

Exercise: XYZ

<Exercise Description>

```
.. protected-exercise-solution:: A
password
```

<Protected Solution>

Table of Contents

A navigable table of contents can be created using standard `rst` techniques.

Example

Table of Contents

```
- 'Section 1 Title'_
- 'Subsection 1.1 Title'_
```

Footnotes

.. [#] and [#]_ create footnotes.

Test\ [#]_

```
.. [#] 'test.org'
```

Explicit Footers

A container with the class `footer-left`, `footer-right` or `block-footer`.

References

Use standard `rst` references.

Example

```
...
Like described in [Eic24]_ ...
...
```

References

```
.. [Eic24] LectureDoc2; 2024
```

Copy to Clipboard

Add `copy-to-clipboard` to a code block to enable copying code to the clipboard.

Example

```
.. code:: java
: class: copy-to-clipboard

public static void main(...)
```

Fade-out Content

Add the class `faded-to-white` to a container to whiten the content.

Boxes with Supplemental Information on the Slide

Example

```
.. admonition:: TBD
: class: note

Some text in a box.
```

Supplemental Information

Use the directive `supplemental` for respective information.

Example

```
.. supplemental::
```

<Text>

Text Alignment

Control text alignment:
`text-align-[left|center|right]`

Images

Adding a drop-shadow and rounded corners: `picture`.

Tables

The layout can be adapted using:

`compact`, `compact-cells`,
`no-table-borders`,
`no-inner-borders`,
`no-column-borders`,
`fake-header[-2nd]-row` and
`fake-header[-2nd]-column`.

Animation

`incremental` (and `wobble`).

`highlight-line-on-hover` (always usable), `highlight-on-hover` (explicit column or row headers are not supported) or
`highlight-identical-cells`

Lists

■ `list-with-explanations` renders text paragraphs of list items less pronounced.
(As shown here.)

■ Use `impressive` to make the list more impressive:

Add `negative-list` to use `"!"` for bullet points.

Add `positive-list` to use `"✓"` for bullet points.

Example

```
- Point 1
- Point 2
.. class:: negative-list list-with-explanations
Some on-slide explanation.

.. class:: positive-list
- Point 3
```

Decorations

`line-above` draws a horizontal lines.

`box-shadow` adds a shadow.

`rounded-corners` the corners will be rounded.

Example

```
.. container:: margin-top-1em
line-above
padding-top-1em
box-shadow

Text
```

Font Styling

***rem* based relative sizes:** `xxl`, `huge`, `large`, `small`, `footnotesize`, `scriptsize`, `tiny`, `x-tiny`, `xx-tiny`

***em* based relative sizes:** `larger`, `smaller`, `much-smaller`

Font weight: `bold`, `light`, `thin`

Font family: `monospaced`, `serif`

Font style: `italic`

Slide Transitions

Available slide transitions:
`transition-move-left`,
`transition-scale`,
`transition-fade`,
`transition-move-to-top`

Example

```
.. class:: transition-move-left

<Slide Title>
```

Revealing Slide Content

All elements with the class `incremental` are revealed `incrementally`.

Example

```
.. class:: incremental

- Item 1 - Part 1
: incremental: 'Item 1 - Part 2'
- Item 2
```

Column-based Layouts

Use `two-columns` and `three-columns` for respective layouts.

Example

```
.. container:: two-columns

.. container:: column no-separator

<Column 1>

.. container:: column

<Column 2>
```

Add `no-default-width` to the root container for content based column widths. Use class `no-separator` on the left column to remove the separator.

Stacked Layouts

Stacked layouts are based on nested `rst` containers for each layer. Each layer - except of the first one - needs to have the class `incremental` and/or the class `overlay` for transparent layers. (Up to 10 layers are supported.)

Images in Stacked Layouts

To avoid that a parent element of a floating element is collapsed add the class `clearfix` to the parent element; i.e., when a layer just contains a floating image.

Example

```
.. stack::

.. layer:: clearfix

.. image:: <p1.svg>
: align: left

.. layer:: overlay

.. image:: <p2.svg>

.. layer:: incremental

<Content>
```

Semantic-based Text Markup

`minor`: for less important text.

`obsolete`: for obsolete statements.

`ger`: to markup German Words. `eng`: to markup English words. `ger-quote`: Uses German quotation marks.

Box sizes

Use `width-100%` and `width-75%` to control the width of a container.

Colors (roles)

Font Colors

DHBW Colors: `dhbw-red`, `dhbw-gra`, `dhbw-light-gray`

DHBW Compatible Colors: `the-blue`, `the-green`, `the-orange`

Other: `black`, `shiny-green`, `shiny-red`, `dark-red`

Background Colors

DHBW Colors: `dhbw-red-background`,

`dhbw-gray-background`,

`dhbw-light-gray-background`

DHBW Compatible Colors:

`the-blue-background`,

`the-green-background`,

`the-yellow-background`

Other: `light-green-background`, `white-background`

Example

```
:dhbw-red: 'Red Text.'
```