

Inhaltsverzeichnis

Syntax	3
1 Texte	3
1.1 Formatierungen	3
1.2 Absätze	3
1.3 Überschriften	4
1.4 unformatierte Abschnitte	4
1.5 Syntax-Hervorhebung	5
1.6 Zitat / Kommentar	5
1.7 Escapezeichen	6
1.8 Fußnoten	6
1.9 Linien	6
2 Links	7
2.1 Extern	7
2.2 Intern	7
2.3 Interwiki	8
2.4 Windows-Freigaben	8
2.5 Bilder als Links	8
3 Bilder und andere Dateien	9
3.1 Bilder im DokuWiki speichern	9
3.2 Größe von Bildern	9
3.3 Ausrichten von Bildern	10
3.4 Tooltips	10
3.5 Dateidownload	11
3.6 Automatische Verlinkung	11
4 Listen	11
5 Tabellen	12
5.1 Grundlegender Aufbau	12
5.2 Formatierung/Ausrichtung	12
6 sonstige Formatierungen	14
6.1 horizontale Linie	14
6.2 Smileys	14
6.3 Typografie	14
6.4 HTML und PHP einbinden	15
7 Kontroll-Makros	15
8 Syntax-Plugins	15
9 Color Plugin	16
10 Zellen Hintergrund Plugin	17
10.1 Syntax und Verwendung	17
10.2 Beispiele	17
10.3 Features	18
11 Table Width Plugin	18
11.1 Syntax	18
12 Wrap Plugin	18
12.1 One plugin to rule them all	18
12.2 Syntax	19
12.2.1 Examples	19
12.2.2 Classes	20

12.2.3 Widths	23
12.2.4 Languages and Text Directions	24
12.3 Demo	24
12.4 Configuration options	24
12.5 ODT Support	25
12.6 Toolbar picker	25
12.7 Extend with custom styles	26
12.7.1 Examples	26
12.7.2 Add former typography classes	26
12.8 How to use the helper	26
12.8.1 Example how to get just one kind of attribute	26
12.8.2 Example how to get all attributes	27

Syntax

[DokuWiki](#) zeichnet sich u. a. durch einfache Textformatierungen aus. Dadurch bleiben die Dateien, die den Inhalt enthalten, gut lesbar. In den nachfolgenden Abschnitten sind die möglichen Formatierungsarten aufgelistet, die innerhalb des Editors verwendet werden können. Einige der hier vorgestellten Formatierungen sind auch direkt über die [Quickbuttons](#) erreichbar. Ein erstes Beispiel, wie der Aufbau einer Seite aussehen kann, findest du, wenn du dir diese Seite im Quelltext-Modus anzeigen lässt. Klicke dazu auf den Button „Vorschau“.

Zum Ausprobieren oder Üben kann die [Spielwiese](#) verwendet werden.

1 Texte

1.1 Formatierungen

DokuWiki unterstützt **fetten**, *kursiven*, unterstrichenen und dicktengleichen ([D](#)**Dicke**) Text. Natürlich kann auch alles ***kombiniert*** werden.

```
DokuWiki unterstützt **fetten**, //kursiven//, __unterstrichenen__ und  
' 'dicktengleichen' ' Text.  
Natürlich kann auch alles **__/' 'kombiniert' '/'__** werden.
```

Ebenso gibt es _{tiefgestellten} und ^{hochgestellten} Text.

```
Ebenso gibt es <sub>tiefgestellten</sub> und <sup>hochgestellten</sup> Text.
```

Text kann als ~~gelöscht¹⁾~~ markiert werden.

```
Text kann als <del>gelöscht</del> markiert werden.
```

1.2 Absätze

Ein Absatz wird durch eine leere Zeile erzeugt. Eine neue Zeile (Zeilenumbruch) kann durch zwei „Backslashes“ (umgekehrte Schrägstriche: \) gefolgt von einem Leerzeichen erzeugt werden.

Dies ist ein Beispieltext mit einigen Zeilenumbrüchen.

Beachte: Die zwei umgekehrten Schrägstriche werden nur erkannt, wenn sie am Zeilenende stehen oder wenn ihnen ein

Leerzeichen folgt.

So\\sieht es ohne Leerzeichen aus.

```
Dies ist ein Beispieltext mit einigen Zeilenumbrüchen.\\  
Beachte: Die zwei  
umgekehrten Schrägstriche  
werden nur erkannt, wenn sie am Zeilenende stehen\\  
oder wenn ihnen ein\\  
Leerzeichen folgt.\\  
So\\sieht es ohne Leerzeichen  
aus.
```

Zeilenumbrüche durch zwei „Backslashes“ (umgekehrte Schrägstriche: \) sollten nur verwendet

werden, wenn sie unbedingt notwendig sind.

1.3 Überschriften

Mit fünf unterschiedlichen Ebenen, getrennt durch Überschriften, kann der Inhalt einer Seite strukturiert werden. Die Überschriften werden mit maximal sechs „=“ eingerahmt, je nach Größe. H1 ist die Überschrift auf höchster Ebene. Am einfachsten ist es zur Erzeugung der Überschriften, die Symbolleiste zu verwenden. Das Einfügen der Überschriften erfolgt über die Buttons, die ein „H“ anzeigen.

Es ist sinnvoll, wenn immer eine Level 1 Überschrift am Anfang eines Artikels steht.

Wenn in einer Seite mehr als drei Überschriften vorkommen, wird automatisch ein Inhaltsverzeichnis erstellt, welches alle Überschriften auflistet. Soll dieses Inhaltsverzeichnis nicht erstellt werden, muss der Befehl: `~~NOTOC~~` mit in die Seite aufgenommen werden.

```
===== Überschrift Ebene 1 =====
===== Überschrift Ebene 2 =====
===== Überschrift Ebene 3 =====
===== Überschrift Ebene 4 =====
===== Überschrift Ebene 5 =====
```

Innerhalb von Überschriften können standardmäßig keine weiteren Formatierungen vorgenommen werden! (s. [FAQ - Headerlinks](#))

1.4 unformatierte Abschnitte

Um Text so anzuzeigen, wie er geschrieben wurde, egal ob er DokuWiki Formatierungen oder andere Formatierungen enthält, kann man jede Zeile mit zwei Leerzeichen beginnen (so wie es in dieser Seite angewendet wurde) oder einen der Bezeichner „code“ bzw. „file“ benutzen:

```
Mit "code": In diesem Text werden keine überflüssigen Leerzeichen entfernt:
wie hier          <-- zwischen
```

```
<file plain Beispiel.txt>
Mit dem Bezeichner "file" kann ein bestimmter Abschnitt als einzelner
Bestandteil
auf einer Seite zum Download angeboten werden.
</file>
```

Innerhalb von „file“ ist auch die im Abschnitt weiter unten beschriebene [Syntax-Hervorhebung](#) aktivierbar.

[HelloWorld.pl](#)

```
#!/usr/bin/perl
print "Hello World!\n";
```

```
<file perl HelloWorld.pl>
```

```
#!/usr/bin/perl
print "Hello World!\n";
</file>
```

Um jegliche Umwandlung durch DokuWiki abzustellen, umschlieÙe den entsprechenden Bereich mit „nowiki“ Bezeichnen oder kürzer mit doppelten Prozentzeichen %%.

Dieser Text enthält Links wie: <http://www.splitbrain.org> und ****Formatierungen**** aber sie werden nicht umgesetzt und erscheinen als normaler Text.

Dieser Text ist nun mit %% markiert und hat die gleiche Wirkung wie "nowiki". "Dieser Text enthält links wie: <http://www.splitbrain.org> und ****Formatierungen**** aber sie werden nicht umgesetzt und erscheinen als normaler Text."

Einfach den Quelltext dieser Seite anschauen (den Knopf „Edit“ drücken) um zu sehen, wie diese Blöcke angewendet werden.

1.5 Syntax-Hervorhebung

DokuWiki kennt Syntax-Highlighting für Quellcode, um dessen Lesbarkeit zu erhöhen. Dafür wird GeSHi „**G**eneric **S**yntax **H**ighlighter“ benutzt. Somit wird jede Programmiersprache unterstützt, die GeSHi kennt. Es wird der gleiche Bezeichner wie im vorherigen Abschnitt benutzt, nur dass er nun auch den Namen der Sprache enthält: `<code java> Quelltext </code>`

```
/**
 * The HelloWorldApp class implements an application that
 * simply displays "Hello World!" to the standard output.
 */
class HelloWorldApp {
    public static void main(String[] args) {
        System.out.println("Hello World!"); //Display the string.
    }
}
```

Es werden folgende Sprachen unterstützt: *actionscript, actionscript-french, ada, apache, applescript, asm, asp, bash, caddcl, cadlisp, c, c_mac, cpp, csharp, css, **delphi**, diff, d, div, dos, eiffel, freebasic, gml, html4strict, ini, inno, **java**, javascript, lisp, lua, matlab, mpasm, nsis, objc, ocaml, ocaml-brief, oobas, oracle8, pascal, perl, php-brief, **php**, python, qbasic, scheme, sdlbasic, smarty, **sql**, ruby, vb, vbnet, vhd, visualfoxpro, **xml***

1.6 Zitat / Kommentar

Um einen Text als Zitat oder Kommentar zu kennzeichnen, wird das „größer als“ - Zeichen folgendermaßen verwendet:

```
Ich denke wir sollten es tun.

> Nein, sollten wir nicht.

>> Doch!
```

```
> Wirklich?

>> Ja!

>>> Dann lass es uns tun!

>>> Kommentare oder Zitate können auch über mehrere Zeilen gehen, wie dieser
völlig unsinnige Text zeigt. Um einen Text als Zitat oder Kommentar zu
kennzeichnen, wird das "größer als"-Zeichen folgendermaßen verwendet
```

Ich denke wir sollten es tun

Nein, sollten wir nicht.

Doch!

Wirklich?

Ja!

Dann lass es uns tun

Kommentare oder Zitate können auch über mehrere Zeilen gehen, wie dieser völlig unsinnige Text zeigt. Um einen Text als Zitat oder Kommentar zu kennzeichnen, wird das „größer als“-Zeichen folgendermaßen verwendet

1.7 Escapezeichen

Will man verhindern, dass DokuWiki irgendwelche Formatierungen am bzw. Umwandlungen im Text vornimmt so kann man die Prozentzeichen benutzen

```
%%[[wiki:testseite|Testseite]]%%
```

Die Ausgabe würde dann folgendermaßen aussehen [[wiki:testseite|Testseite]]

1.8 Fußnoten

Fußnoten ²⁾ werden durch eine doppelte runde Klammer dargestellt ³⁾. Der Text zwischen den Klammern wird an das Ende der Seite geschrieben. Die Reihenfolge und somit die Nummerierung übernimmt DokuWiki.

```
Fußnoten ((dies ist eine Fußnote)) werden durch eine doppelte runde Klammer
dargestellt ((eine zweite Fußnote))
```

1.9 Linien

Durch vier oder mehr Bindestriche bzw. Minuszeichen „-“ wird eine horizontale Linie erzeugt:

2 Links

Mit DokuWiki können fünf unterschiedliche Arten von Links benutzt werden:

2.1 Extern

Externe Links (Verweise auf andere Webseiten oder Dateien) werden automatisch erkannt: <http://www.google.com> oder einfach www.google.com. Verweise können auch hinter eigenen Begriffen versteckt werden: [hier ist google](#). E-Mail Adressen werden in spitze Klammern gesetzt: andi@splitbrain.org.

Externe Links werden automatisch erkannt: <http://www.google.com> oder einfach www.google.com.
Verweise können auch hinter eigenen Begriffen versteckt werden:
`[[http://www.google.com|hier ist google]]`.
E-Mail Adressen werden in spitze Klammern gesetzt: `<andi@splitbrain.org>`.

2.2 Intern

Interne Links, also Verweise auf Seiten innerhalb von DokuWiki, werden in doppelten eckigen Klammern gesetzt („Alt Gr+8“ bzw. Alt Gr+9 unter Windows bzw. Wahl taste+5 und Wahl taste+6 bei Apple-Betriebssystemen): [📄 pagename](#)

Ein Link kann auch hinter einem Begriff oder einer Beschreibung versteckt werden: [Beschreibung für einen Link](#).

Interne Links werden in doppelten eckigen Klammern gesetzt:
`[[pagename]]`

Ein Link kann auch hinter einem Begriff oder einer Beschreibung versteckt werden:
`[[pagename|Beschreibung für einen Link]]`.

Dateinamen für neue Seiten im Wiki werden automatisch in Kleinbuchstaben umgewandelt! Umlaute sind genauso wie Sonderzeichen nicht erlaubt!

Durch einen Doppelpunkt werden Namensbereiche im Wiki aufgelöst: [namespaces](#)

Durch einen Doppelpunkt werden Namensbereiche im Wiki aufgelöst: `[[de:namespaces]]`

Weitere Informationen zu Namensbereichen: [📄 de:namespaces](#).

Die Verlinkung zu einem speziellen Abschnitt innerhalb des Dokuments (Anker) ist auch möglich. Der Abschnittsnamen wird getrennt durch ein Doppelkreuz (#) an den Seitennamen gehängt.

Dieses Beispiel verlinkt zu [diesem Abschnitt](#).

Dieses Beispiel verlinkt zu `[[de:wiki:syntax#intern|diesem Abschnitt]]`.

Anmerkungen:

- Links zu [existierenden Seiten](#) werden anders angezeigt als Links zu [nicht-existierenden](#) Seiten.
- DokuWiki verwendet standardmäßig keine  [CamelCase](#) für Links. In der Einstellungsdatei:  [config](#) kann das aber geändert werden. **Hinweis:** wenn das Wort „DokuWiki“ als Link angezeigt wird, dann ist diese Option eingeschaltet.
- Ändert sich eine Überschrift im Zieldokument, führt der Anker ins Leere ohne dass der Link dabei als fehlerhaft angezeigt wird.
- Anker werden klein geschrieben, es werden Leerzeichen mit Unterstrichen und Umlaute mit ihren Umschreibungen (ä = ae) verwendet.

2.3 Interwiki

DokuWiki unterstützt auch Links in andere Wikis, sogenannte  [de:interwiki](#) Links. Ein Link zur Wikipedia Seite über Wikis:  [Wiki](#)

DokuWiki unterstützt auch Links in andere Wikis, sogenannte `[[doku>de:interwiki]]` Links.
Ein Link zur Wikipedia Seite über Wikis: `[[wpde>Wiki]]`

2.4 Windows-Freigaben

DokuWiki kennt auch Links auf Windows-Freigaben: [freigegebener-Ordner](#). Dies ist allerdings nur in einer homogenen Benutzergruppe sinnvoll (siehe  [Intranet](#)).

`[[\\server\freigabe|this]]`.

Hinweis:

- Aus Sicherheitsgründen funktioniert direktes Navigieren durch die Freigaben standardmäßig nur mit dem Internet Explorer (und das nur in der „lokalen Zone“)!
- Für Mozilla und Firefox kann es durch die Option [security.checkloaduri](#) eingeschaltet werden, was aber nicht empfohlen wird! (Weitere Informationen: [Links to local pages do not work](#))
- Opera ignoriert als Sicherheitsfeature seit [Version 7.54](#) alle Links auf *file:* in Webseiten.
- Für weitere Informationen siehe: 151

2.5 Bilder als Links

Bilder können für einen Link auf eine interne oder externe Datei verwendet werden. Dazu wird die Syntax zur Anzeige von Links einfach mit der zur Anzeige von Bildern (siehe unten) kombiniert:

`[[http://www.php.net|{{wiki:dokuwiki-128.png}}]]`



Bitte beachte: Bildformatierungen sind die einzigen Formatierungen, die in Linknamen benutzt werden können!

3 Bilder und andere Dateien

3.1 Bilder im DokuWiki speichern

Um Bilder im DokuWiki speichern zu dürfen, muss der Benutzer die entsprechende Berechtigung besitzen ([ACL](#)). Zum Hochladen ist mindestens die Berechtigung „Hochladen“ (engl. „Upload“), zum Löschen „Entfernen“ (engl. „Delete“) für den [Namensraum](#) erforderlich.

3.2 Größe von Bildern

Mit geschweiften Klammern können sowohl externe als auch interne [Bilder](#) eingebunden werden. Das Bild heißt in diesem Beispiel „dokuwiki-128.png“. Optional kann deren Größe angegeben werden:



Originalgröße:



Eine Breite vorgeben:



Breite und Höhe vorgeben:

Größe für externe Bilder vorgeben:

Originalgröße:	<code>{{wiki:dokuwiki-128.png}}</code>
Eine Breite vorgeben:	<code>{{wiki:dokuwiki-128.png?50}}</code>
Breite und Höhe vorgeben:	<code>{{wiki:dokuwiki-128.png?200x50}}</code>
Größe für externe Bilder vorgeben:	<code>{{http://de3.php.net/images/php.gif?200x50}}</code>

3.3 Ausrichten von Bildern

Mit Leerzeichen zur Rechten und/oder Linken innerhalb der Klammern kann ein Bild horizontal ausgerichtet werden (Links, Zentriert, Rechts).



```
{{wiki:dokuwiki-128.png }}  
{{ wiki:dokuwiki-128.png }}  
{{ wiki:dokuwiki-128.png}}
```

Bei Ausrichtung und der Verwendung eines Titels sollte man drauf achten, dass der Titel nach dem Leerzeichen kommt:

```
{{wiki:dokuwiki-128.png |Linksbündig}}  
{{ wiki:dokuwiki-128.png |Zentriert }}  
{{ wiki:dokuwiki-128.png|Rechtsbündig}}
```

3.4 Tooltips

Auch Titel, in den meisten Browsern als Tooltip angezeigt, können angegeben werden.



```
{{ wiki:dokuwiki-128.png |Dies ist ein Titel}}
```

3.5 Dateidownload

Wenn eine Datei angegeben wird, die kein Bild (gif, jpg, png) ist, so wird sie als Link angezeigt und so zum Download angeboten.

Beispiel

```
{{ wiki:dokuwiki-128.txt |Beispiel}}
```

3.6 Automatische Verlinkung

Bilder werden automatisch mit einer Verlinkung zur Ihren  [Metadaten](#) erstellt - Bsp.: 

Dieses Verhalten kann über die Option ?nolink⁴⁾ unterbunden werden⁵⁾:



```
{{wiki:dokuwiki-128.png?15&nolink}}
```

Außerdem kann mit ?direct das Bild, z.Bsp. als Vollansicht, direkt verlinkt werden:



```
{{wiki:dokuwiki-128.png?15&direct}}
```

4 Listen

DokuWiki kennt unsortierte und nummerierte Listen. Um einen Listeneintrag zu erzeugen, beginne eine Zeile mit zwei Leerzeichen. Diesen Leerzeichen folgt dann ein „*“ für unsortierte oder ein „1.“ für nummerierte Listen.

- Dies ist ein Eintrag einer unsortierten Liste
- Ein zweiter Eintrag
 - Unterschiedliche Ebenen einer Liste durch Hinzufügen von weiteren zwei Leerzeichen
- Noch ein Eintrag

1. Dies ist ein Eintrag einer nummerierte Liste
2. Ein zweiter Eintrag
 1. Unterschiedliche Ebenen einer Liste durch Hinzufügen von weiteren zwei Leerzeichen
3. Noch ein Eintrag

```
* Dies ist ein Eintrag einer unsortierten Liste
* Ein zweiter Eintrag
  * Unterschiedliche Ebenen einer Liste durch Hinzufügen von weiteren zwei
  Leerzeichen
* Noch ein Eintrag
```

- Dies ist ein Eintrag einer nummerierte Liste
- Ein zweiter Eintrag
 - Unterschiedliche Ebenen einer Liste durch Hinzufügen von weiteren zwei Leerzeichen
- Noch ein Eintrag

5 Tabellen

5.1 Grundlegender Aufbau

Mit DokuWiki können Tabellen ganz einfach erstellt werden:

hervorgehoben	zum Beispiel für	eine Titelzeile
Zeile 1 Spalte 1	Zeile 1 Spalte 2	Zeile 1 Spalte 3
Zeile 2 Spalte 1	doppelte Spaltenbreite (beachte den doppelten senkrechten Strich)	
Zeile 3 Spalte 1	Zeile 2 Spalte 2	Zeile 2 Spalte 3
Zeile 4 Spalte 1		Zeile 4 Spalte 3

Felder einer Tabelle starten mit „|“ für ein normales Feld oder mit einem „^“ für hervorgehobenes Feld.

```

^ hervorgehoben      ^ zum Beispiel für
^ eine Titelzeile      ^
| Zeile 1 Spalte 1    | Zeile 1 Spalte 2
| Zeile 1 Spalte 3    |
| Zeile 2 Spalte 1    | doppelte Spaltenbreite (beachte den doppelten
senkrechten Strich) ||
| Zeile 3 Spalte 1    | Zeile 2 Spalte 2
| Zeile 2 Spalte 3    |

```

5.2 Formatierung/Ausrichtung

Leerzellen

Zellen über mehrere Spalten werden erzeugt wenn man, wie oben gezeigt, am Ende einer solchen Spalte so viele „|“ oder „^“ ohne Leerzeichen dazwischen hintereinander setzt, wie Spalten überbrückt werden sollen. Kurz gesagt: Alle Zeilen einer Tabelle müssen die gleiche Anzahl an Feldtrennern „|“ oder „^“ haben.

Hervorgehobene Felder in Vertikaler Ausrichtung sind auch möglich.

	hervorgehoben 1	hervorgehoben 2
hervorgehoben 3	Zeile 1 Spalte 2	Zeile 1 Spalte 3
hervorgehoben 4	Zeile 2 Spalte 2	
hervorgehoben 5	Zeile 3 Spalte 2	Zeile 3 Spalte 3

Wie wir sehen, entscheidet der Feldtrenner („|“ oder „^“) vor einer Zelle über ihr Aussehen:

```

|          ^ hervorgehoben 1          ^ hervorgehoben 2          ^

```

^ hervorgehoben 3	Zeile 1 Spalte 2	Zeile 1 Spalte 3	
^ hervorgehoben 4	Zeile 2 Spalte 2		
^ hervorgehoben 5	Zeile 3 Spalte 2	Zeile 3 Spalte 3	

Zellen verbinden

Felder über mehrere Zeilen (**vertikal verbundene Zellen**) lassen sich durch hinzufügen von ::: in der zu verbindenden Zelle zusammenfügen.



Zum Stand der Release 2015-08-10a „Detritus“ können Zellen der ersten Zeile nicht in vertikal verbundene Zellen einbezogen werden.

Titelzeile 1	Titelzeile 2	Titelzeile 3
Zeile 1 Spalte 1		Zeile 1 Spalte 3
Zeile 2 Spalte 1	Diese Zelle ist vertikal verbunden	Zeile 2 Spalte 3
Zeile 3 Spalte 1		Zeile 3 Spalte 3

Außer den drei Doppelpunkten darf die zu verbindende Zelle keine weiteren Inhalte haben.

^ Titelzeile 1	^ Titelzeile 2	^ Titelzeile 3
^		
Zeile 1 Spalte 1	Diese Zelle ist vertikal verbunden	Zeile 1 Spalte 3
3		
Zeile 2 Spalte 1	:::	Zeile 2 Spalte 3
3		
Zeile 3 Spalte 1	:::	Zeile 3 Spalte 3
3		

Felder über mehrere Spalten (**horizontal verbundene Zellen**) lassen sich durch hinzufügen von ^ bzw. | in der zu verbindenden Zelle zusammenfügen.

Titelzeile 1	Titelzeile 2	Titelzeile 3
Zeile 1 Spalte 1	Diese Zelle ist horizontal verbunden	
Zeile 2 Spalte 1	Zeile 2 Spalte 2	Zeile 2 Spalte 3
Zeile 3 Spalte 1	Zeile 3 Spalte 2	Zeile 3 Spalte 3

^ Titelzeile 1	^ Titelzeile 2	^ Titelzeile 3
^		
Zeile 1 Spalte 1	Diese Zelle ist horizontal verbunden	
Zeile 2 Spalte 1	Zeile 2 Spalte 2	Zeile 2 Spalte 3
3		
Zeile 3 Spalte 1	Zeile 3 Spalte 2	Zeile 3 Spalte 3
3		

Textausrichtung

Der Inhalt einer Zelle kann horizontal innerhalb der Zelle ausgerichtet werden:

- Rechtsbündig - zwei Leerzeichen links vom Text
- Linksbündig - Leerzeichen rechts vom Text

- Zentriert - zwei Leerzeichen links und rechts vom Text

Tabelle mit Ausrichtung		
Rechts	Mitte	Links
Links	Rechts	Mitte
XXXXXXXXXXXXX	XXXXXXXXXXXXX	XXXXXXXXXXXXX

So sieht die Formatierung aus:

^ Tabelle mit Ausrichtung ^^^		
Rechts Mitte Links		
Links Rechts Mitte		
XXXXXXXXXXXXX XXXXXXXXXXXXX XXXXXXXXXXXXX		

Hinweis zu dem senkrechten Balken-Zeichen: dieses Zeichen ist auf einer PC-Tastatur erreichbar über die Taste „<“ bei zusätzlich gedrückter „AltGr“-Taste. Ist dort auch aufgedruckt. Apple-Nutzer erreichen das Zeichen mit der Wahl taste („Alt“) und der „7“.

6 sonstige Formatierungen

6.1 horizontale Linie

Mit vier aufeinander folgenden Minuszeichen (alleine in einer Zeile) kann eine horizontale Linie erzeugt werden:

6.2 Smileys

DokuWiki verwandelt oft genutzte  Emoticons zu kleinen Smiley-Grafiken. Weitere Smiley-Grafiken können im Ordner „smily“ abgelegt und dann in der Datei conf/smileys.conf bzw. für benutzereigene Smileys unter conf/smileys.local.conf eingetragen werden. Hier ist nun eine Übersicht der standardmäßig in DokuWiki enthaltenen Smileys:

																			
8-)	8-O	:-(	:-)	=)	:-/	:-\	:-?	:-D	:-P	:-O	:-X	:-	:-)	?:	!:	^_^	LOL	FIXME	DELETEME

6.3 Typografie

DokuWiki kann einfache Textzeichen in ihre typografischen Entsprechungen umwandeln. Hier eine Liste aller Zeichen, die DokuWiki kennt:

→ ← ↔ ⇒ ⇐ ⇔ » « – — 640x480 © ™ ® „He thought 'It's a man's world'...”

-> <- <-> => <= <=> >> << -- --- 640x480 (c) (tm) (r)
"He thought 'It's a man's world'..."

Bitte beachtet: Diese Umwandlungen können in der Konfigurationsdatei abgestellt werden

6.4 HTML und PHP einbinden

In einer Seite kann sowohl HTML als auch PHP direkt benutzt werden.

```
<html> This is some <font color=„red“ size=„+1“>HTML-Code</font> </html>
```

```
<html>
This is some <font color="red" size="+1">HTML-Code</font>
</html>
```

```
<php>
echo 'A logo generated by PHP: ';
echo '';
</php>
```

```
<php>
echo 'A logo generated by PHP: ';
echo '';
</php>
```

Achtung: Das Einbinden von HTML und PHP in einer Seite kann in der Konfigurationsdatei ausgeschaltet werden⁶⁾. Ist diese Option aktiviert, wird der Quellcode dieser Abschnitte als Text angezeigt und nicht ausgeführt.

Achtung: Das Einbinden von HTML und PHP auf einer Seite im Internet ist gewöhnlich ein schweres Sicherheitsrisiko! Diese Funktion ist nur für das Intranet gedacht!

7 Kontroll-Makros

Einige Syntax-Befehle beeinflussen wie Dokuwiki eine Seite beim Aufruf ausgibt. Die folgenden Kontroll-Makros sind verfügbar:

Makro	Beschreibung
~~NOTOC~~	Wenn dieses Makro in eine Seite eingebunden ist, wird das Table of Content nicht erstellt.
~~NOCACHE~~	Alle Seiten werden aus Performancegründen zwischengespeichert. Unter bestimmten Umständen ist dieses Verhalten aber nicht gewollt. Beim Einfügen dieses Makros wird DokuWiki angewiesen, den Inhalt der Seite neu aufzubauen.

8 Syntax-Plugins

Die Syntax von DokuWiki kann durch [Plugins](#) erweitert werden. Wie die einzelnen Plugins benutzt werden, steht auf den zugehörigen Beschreibungsseiten. Die folgenden Plugins sind auf dieser


```
<color rgb(80%,0%,0%)/rgb(100%,80%,100%)>text</color>
<color hsl(120,100%,30%)/hsl(180,50%,90%)>text</color>
```

folgendes Ergebnis sollte dann zu sehen sein:

text text text text text text

Dieses Plugin unterstützt die [X11 color names](#), Hex-Dreifache (wie #123 oder #123456 für RGB), Hex-Vierfach (#1234 oder #12345678 für RGBA), [rgb](#), [rgba](#), [hsl](#) und [hsla](#) Farbspezifikationen. Dieses Plugin kann keine Syntaxfehler in Ihren Farbspezifikationen erkennen. Wenn Sie diese Spezifikationen falsch verwenden, werden Sie nicht funktionieren. Beachten Sie, dass nicht alle Farbspezifikationen auf allen Browsern funktionieren. Überprüfen Sie im Zweifelsfall [Can I Use](#).

10 Zellen Hintergrund Plugin

10.1 Syntax und Verwendung

Eine Hintergrundfarbe wird durch „@“ eingeleitet und durch „:“ beendet. Sie können alle Namen und Farbcodes aus HTML verwenden. Wenn Sie keine Farbe angeben, wird Gelb verwendet. Siehe Beispiele unten:

10.2 Beispiele

Dieser Code:

```
^ Header 1      ^ Header 2  ^ Header 3      ^
| @lightgreen:**Span**      ||      |
| @#6495ed:Ligne 1 | Oui      | Oui, à valider |
| Ligne 2          | Oui      | @orange: Oui    |
| @:Ligne 3        | Oui      | @red:Non       |
```

erzeugt folgendes:

Header 1	Header 2	Header 3
Span		
Ligne 1	Oui	Oui, à valider
Ligne 2	Oui	Oui
Ligne 3	Oui	Non

Dieser Code:

```
| @palegreen:Validé | @orange:À valider | @red:Non fonctionnel |
```

dieses Ergebnis:

Validé À valider Non fonctionnel

10.3 Features

- Uses HTML color codes and names [HTML Color Code Chart](#)
- Works with other table formatting (alignment, bold, ...)

11 Table Width Plugin

11.1 Syntax

The width has to be specified at the start of a line before the table. The first value is used as width of the table, the rest is for columns. If you want to omit some value use a dash instead. The widths can be specified in any CSS units:

```
|< 100% 50px - 5em 10em 10% >|
^ A   ^ B   ^ C   ^ D   ^ E   ^
| 1   | 2   | 3   | 4   | 5   |
```

The relative units (percentage) for the table width refer to the width of the page, and when used for columns they refer for the table width. For example, the width specification `|< 50% 50% 50% >|` makes the table half width of the page and two equal columns quarter of the page width each.

If there are more columns in the table than there are values in the width specification, the width will be applied to the columns from left to right:

```
|< 50em 20% >|
^ 20%           ^ 80%           ^
| Specified width | The rest of the table width |
```

12 Wrap Plugin

Universal plugin which combines the functionality of many other plugins. Wrap wiki text inside containers (divs or spans) and give them a class (choose from a variety of preset classes), a width and/or a language with its associated text direction.

12.1 One plugin to rule them all

This plugin gives you the ability to wrap wiki text inside containers (divs or spans) and give them

1. a certain class (with loads of useful preset classes)
2. a width
3. a language with its associated text direction

It potentially replaces a lot of other plugins and is IMHO the better alternative for many.

It fully replaces: [class](#), [clearfloat](#), [div_span_shorthand](#), [divalign2](#), [divalign](#), [emphasis](#), [hide](#), [important_paragraph](#), [importanttext](#), [lang](#), [ltr](#), [noprint](#), [pagebreak](#), [side_note](#), [tip](#), [wpre](#)

It partly replaces: [box](#), [button](#), [color](#), [columns](#), [fontcolor](#), [fontfamily](#), [fontsize2](#), [fontsize](#), [highlight](#),

[layout](#), [note](#), [styler](#), [tab](#), [tablewidth](#), [typography](#)

12.2 Syntax

Basic Syntax:

```
<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>
```

An uppercase **<WRAP>** (or alternatively **<block>** or **<div>**) creates a **div** and should be used for „**big**“ containers, **surrounding** 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>
```

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

Since version 2013-06-13 there is also a shorthand syntax (for wraps without content):

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

and

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



Please note, some things **won't work with spans**: **alignments** (including alignments generated by changing the text direction), **multi-columns** and **widths** if the according wrap isn't floated as well.

12.2.1 Examples

The plugin comes with an example page, which should explain a lot and looks like this in the default

template (see below).

12.2.2 Classes

The following classes are currently available:

class name	description/notes
columns - similar to columns , side_note , styler , tip	
column	same as left in LTR languages and same as right in RTL languages
left	same as column, will let you float your container on the left
right	will let the container float right
center	will position the container in the horizontal center of the page
col2..col5	will show the text in multiple columns determined by their amount (2, 3, 4 or 5), only works in modern browsers (no IE9 and below)
colsmall, colmedium, collarge	will also show the text in multiple columns but determined by their width (small, medium or large), only works in modern browsers (no IE9 and below)
widths -  experimental, might not work as expected, includes mobile support	
half	fits two columns in a row, should be used in pairs
third	fits three or two columns in a row, should be used in triplets or together with twothirds
twothirds	fits two columns in a row when used together with third, one 1/3 wide and another 2/3 wide
quarter	fits four columns in a row, should be used in quads
alignments - similar to divalign , columns , styler -  don't work with spans!	
leftalign	aligns text on the left
rightalign	aligns text on the right
centeralign	centers the text
justify	justifies the text
boxes and notes - similar to box , note , tip	
box	creates a box around the container (uses colours from style.ini)
info (was information in first version)	creates a blue box with an info icon
important	creates an orange box with an important icon
alert ( was warning in previous versions)	creates a red box with an alert icon
tip	creates a yellow box with a tip icon
help	creates a violet box with a help icon
todo	creates a cyan box with an todo icon
download	creates a green box with a download icon
round	adds rounded corners to any container with a background colour or a border (only works in modern browsers, i.e. no IE)
danger	creates a red danger safety note
warning	creates an orange warning safety note
caution	creates a yellow caution safety note
notice	creates a blue notice safety note
safety	creates a green safety note
marks - similar to emphasis , important_paragraph , importanttext	
hi	marks text as highlighted
lo	marks text as less significant

class name	description/notes
em	marks text as especially emphasised
miscellaneous	
clear	similar to clearfloat , should preferably be used with divs, i.e. uppercase <WRAP>s
tabs	if wrapped around a list of links, will show those as tabs
hide	hides the text per CSS (the text will still appear in the source code, in non-modern browsers and is searchable)
noprint	displays text on the screen, but not in print, similar to noprint
onlyprint	displays text only in print, but not on the screen
pagebreak	forces a new page in printouts (not visible on the screen), similar to pagebreak
nopagebreak	tries to avoid a pagebreak in printouts (not visible on the screen)
spoiler	shows white text on a white background, only to be revealed by highlighting it; similar to hide
button	when wrapped around a link, styles it like a button
tablewidth	sets widths of tables inside to whichever width the wrap gets, partly replaces tablewidth
indent	indents the text, could be used instead of tab
outdent	„outdents“ the text, could partly be used instead of outdent
prewrap	wraps text inside pre-formatted code blocks, similar to wpre

12.2.2.1 Known restrictions

- WRAPs export to ODT format but not everything works 100%
- Round corners only work in modern browsers (no IE8 and below).
- Multiple columns only work in modern browsers (no IE9 and below).
- Width classes are experimental and only work in modern browsers (no IE8 and below).
- Normal DokuWiki Headlines used to not work and a work-around was added. Now that headlines do work, the work-around is not needed anymore but kept for backwards-compatibility. It was deprecated in version 2018-04-22 and disabled by default. They can be enabled by using the emulatedHeadlines [config option](#). The following syntax would then produce two different kinds of emulated headlines inside any wrap:
 - `/**_Big Underlined Headline_**/` (They will look a bit different in safety notes.)
 - `/**Small Headline**/`

You might need to adjust a few of the classes to your template's needs, especially `hi`, `lo` and `em`. If you have a dark or otherwise heavily coloured theme, please use the `darkTpl` [config option](#).

The classes are easily adjustable and extensible. Any wishes are welcome.

12.2.3 Widths

You can set any valid widths on any uppercase <WRAP> container: %, px, em, rem, ex, ch, vw, vh, pt, pc, cm, mm, in. Just set the width before or after or with the classes, e.g.

```
<WRAP someclass 50% anotherclass>...
```

All except percentages will be reduced to have the maximum width available on smaller screens.

You can also use the width keywords `half`, `third`, `twothirds` and `quarter`. To work correctly they need another wrap around them. E.g.

```
<WRAP group>
  <WRAP half column>...</WRAP>
  <WRAP half column>...</WRAP>
</WRAP>
```

will result in two columns next to each other, which will wrap underneath each other on smaller screens and mobile devices.

12.2.4 Languages and Text Directions

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

```
<wrap :en>This text is explicitly marked as English.</wrap>
```

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 like to mark a text with a different text direction than the default one, you should use `div`s, i.e. uppercase `<WRAP>`s. Otherwise the text alignment won't change as well.

This makes it a better replacement of `ltr` (and `lang`).

12.3 Demo

You can see a demo of the plugin on demo.selfthinker.org.

“Examples” (demo) in Russian (for v2011-05-15). [Source](#).

12.4 Configuration options

Option	Description	Default value
<code>noPrefix</code>	Which (comma separated) class names should be excluded from being prefixed with „wrap_“ (* and ? wildcards allowed)	<code>tabs, group</code>
<code>restrictedClasses</code>	Restrict usage of plugin to these (comma separated) classes (* and ? wildcards allowed)	<i>(empty)</i>
<code>restrictionType</code>	Restriction type, specifies if classes above shall be included or excluded	<code>0</code>
<code>syntaxDiv</code>	Which syntax should be used in the toolbar picker for block wraps?	<code>WRAP</code> (other choices: <code>div</code> , <code>block</code>)
<code>syntaxSpan</code>	Which syntax should be used in the toolbar picker for inline wraps?	<code>wrap</code> (other choices: <code>span</code> , <code>inline</code>)
<code>darkTpl</code>	Optimise colours for dark templates?	<code>0</code>
<code>emulatedHeadlines</code>	Use emulated headings? (deprecated)	<code>0</code>

12.5 ODT Support



There have been more updates to the Wrap as well as the ODT plugin, so more stuff works. The below should be updated with what works and what doesn't.

Since version 2015-06-13 the Wrap plugin supports exporting most of its functionality/styling to ODT when using at least version 2015-06-29 of the [ODT Plugin](#). By default, Wrap syntax will be exported to ODT using 'print' CSS styles. This means the exported Wrap elements will look the same when printing a wiki page. If you want to have the ODT exported Wrap elements look like displayed in the browser (i.e. with 'screen' CSS styles), then use the following ODT plugin configuration settings:

- add wrap to the 'usestyles' config setting
- set the 'media_sel' setting to 'screen'

If you prefer a user defined CSS style for the Wrap ODT export, then simply place a file 'odt.css' into the Wrap plugin folder with your own CSS code (and set config setting 'media_sel' to 'print').

Here is what is currently **not** supported:

- columns: left/right/center/column is partly supported; they are positioned correctly, but content is not floating around them
- widths are not supported except % and half/third/quarter
- boxes and notes: hardly any formatting inside them is supported, therefore emulated headings also don't work
- tabs will just show a list of links
- noprint
- nopagebreak
- onlyprint only works on boxes
- languages are set correctly but do not seem to affect text alignment
- shorthand syntax
- Not supported because not relevant in ODT: clear, prewrap

12.6 Toolbar picker



The wrap picker in the editing toolbar adds the most common wrap syntaxes.

- „columns“ creates a set of half columns
- „simple centered box“ creates a standard box (60% wide, centered)
- „info, tip, important, alert, help, download, todo box“ creates specifically themed boxes (also 60% wide, centered)
- „clear floats“ creates a `<WRAP clear/>`
- „especially emphasised, highlighted, less significant“ creates the respective marks

12.7 Extend with custom styles

If you like to add your own classes and styles to the plugin, you can simply add the styles for your class preceded by „wrap_“ to your [user styles](#). Please note, any classes need to be **lower case**.

E.g. if you need a <WRAP myclass>, you edit (or create if it doesn't exist) your conf/userstyle.css and add your .wrap_myclass{} with its style definitions to it. (If necessary, edit conf/userprint.css⁷⁾ for the print view, conf/userrtl.css⁸⁾ for RTL languages and conf/userall.css⁹⁾ for all styles as well.)

User permissions for every file used must be similar to original DokuWiki files.

Since version 2010-03-14 you have the possibility to exclude certain class names from being prefixed with „wrap_“. Just add a comma separated list of class names into the config option „noPrefix“ in the configuration manager.

12.7.1 Examples

in style.css

```
.dokuwiki div.wrap_note{ /* added */
    background-color: #eee;
    color: #000;
    padding: .5em .5em .5em .5em;
    margin-bottom: 1em;
    overflow: hidden;
}
```

call in DW-page:

```
<WRAP note>...</WRAP>
```

Here are some [useful Wrap extensions](#) created by users of this plugin.

12.7.2 Add former typography classes

The old typography classes were removed in version 2011-05-15. If you need something similar, use the [Block](#) plugin instead. Or you can use your own and copy them to your own user styles (see [above](#)).

12.8 How to use the helper

From version 2011-05-15 on the plugin includes a helper plugin which you can use to add classes, width and lang/dir to any other plugin.

12.8.1 Example how to get just one kind of attribute

```
// get lang from wrap helper plugin
$lang = '';
if(!plugin_isdisabled('wrap')) {
    $wrap = plugin_load('helper', 'wrap');
```

```

    $attr = $wrap->getAttributes($data);
    if($attr['dir']) $lang = ' lang="'. $attr['lang']. '"
xml:lang="'. $attr['lang']. '" dir="'. $attr['dir']. '"';
}

// add lang to your plugin's output
$renderer->doc .= '<span ' . $lang . ' class="foo">';

```

getAttributes() expects the string with „classes #id width :language“. It returns an array with

- \$attr['class']: CSS class(es)
- \$attr['id']: CSS ID
- \$attr['width']: width
- \$attr['lang'] and \$attr['dir']: language and text direction

12.8.2 Example how to get all attributes

```

// get attributes from wrap helper plugin
$attr = '';
if(!plugin_isdisabled('wrap')) {
    $wrap = plugin_load('helper', 'wrap');
    $attr = $wrap->buildAttributes($data, 'additionalClass');
}

// add those attributes to your plugin's output
$renderer->doc .= '<div ' . $attr . ">";

```

buildAttributes() expects the same string as above („classes #id width :language“) and an optional string for additional classes, in case your plugin has CSS classes of its own which it needs to reuse. It returns a string with all the attributes prepared for HTML.

1)

wer das Tag „eingefügt“ (<ins>) benötigt. kann dies über Plugins realisieren. Siehe ganz unten!

2)

dies ist eine Fußnote

3)

eine zweite Fußnote

4)

ggf. mit einem „&“ mit anderen Optionen wie der Größenangabe kombinieren

5)

Standardmäßig lässt sich [dieses, oft unerwünschte, Verhalten](#) momentan (13.03.2009) nicht für das komplette DokuWiki deaktivieren

6)

Konfigurationsoptionen [htmllok](#) bzw. [phpok](#)

7)

conf/printstyle.css in Anteater

8)

conf/rtlstyle.css in Anteater

9)

conf/allstyle.css in Anteater