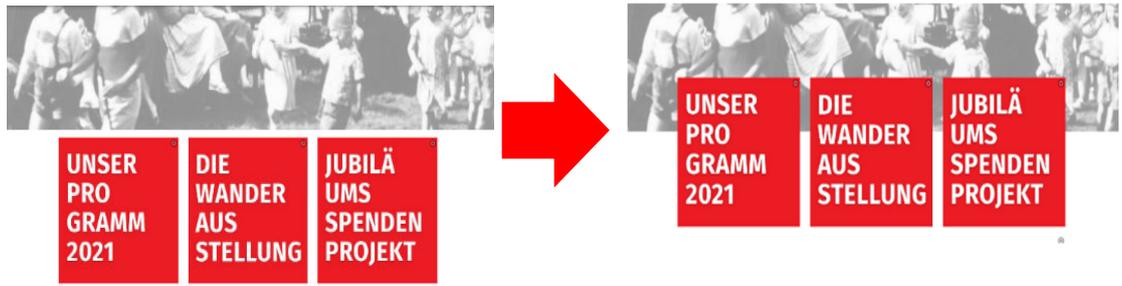


# CSS-Lösungen

in OpenCms



## Autor

Stabsstelle Öffentlichkeitsarbeit - Internet  
Karl Lober / Jürgen Eckert  
Domplatz 2  
96049 Bamberg  
Tel. 09 51 5 02 15 36  
Mobil 01 79 3 22 09 33  
E-Mail [eckert@erzbistum-bamberg.de](mailto:eckert@erzbistum-bamberg.de)

ERZBISTUM  
BAMBERG



## Inhalt

<b>1</b>	<b>Einfügen des Elements „Flexibler Inhalt“</b>	<b>3</b>
<b>2</b>	<b>CSS Grundlagen</b>	<b>4</b>
2.1	Grundaufbau	4
2.2	Überblick über mögliche Eigenschaften	4
2.2.1	margin	4
2.2.2	padding	5
2.2.3	height und width	5
2.2.4	Formatierung von Schrift	5
2.2.5	Media Queries	5
2.2.6	Selektoren	6
2.2.7	„!important“	6
2.2.8	Im Internet	6
<b>3</b>	<b>Elemente verschieben</b>	<b>7</b>
<b>4</b>	<b>Umformen</b>	<b>8</b>
<b>5</b>	<b>Kachelmenü</b>	<b>10</b>

## 1 Einfügen des Elements „Flexibler Inhalt!“

In OpenCms kann über das Element „Flexibler Inhalt“ eigener CSS Code in eine Seite eingefügt werden.

Klicken Sie auf den Zauberstab in der Menüleiste am oberen linken Bildschirmrand ...



und wählen Sie im Auswahlmenu unter „Typen“ „Fortgeschrittene Elemente“ aus.



Ziehen Sie dann ein Element vom Typ „Flexibler Inhalt“ an eine beliebige Stelle in Ihre Seite. Die Stelle, an der sich das Code-Element auf der Seite befindet, ist irrelevant. Allerdings nutzt das Element ein wenig Seitenhöhe, auch wenn der Inhalt nur aus CSS besteht.



# 2 CSS Grundlagen

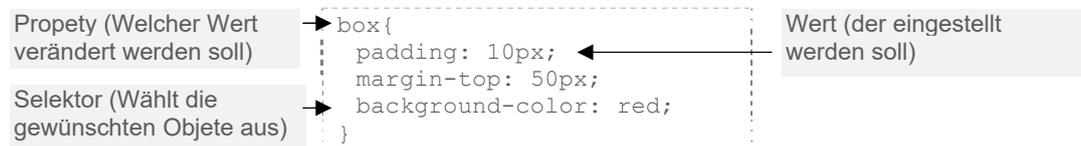
Um sicherzustellen, dass das CSS nur auf die beabsichtigten Elemente angewendet wird, verwenden Sie jeweils eine CSS-Anpassung. Diese sollte in keinem anderen Baustein auf der Seite verwendet werden.

Der CSS-Code muss dann zwischen zwei Style-Tags gesetzt werden.

```
<style>[CSS-Code]</style>
```

## 2.1 Grundaufbau

Der CSS-Code baut sich aus Bausteinen, wie folgt, auf:



## 2.2 Überblick über mögliche Eigenschaften

Nachfolgend ist ein kleiner Teil der möglichen Eigenschaften erklärt.

### 2.2.1 margin

Die „margin“ Eigenschaft legt den Abstand vom Rand eines Objekts nach außen fest. Um den Abstand in eine bestimmte Richtung zu verändern, auch die folgenden Einstellungen verwendet werden.:

- margin-top
- margin-right
- margin-bottom
- margin-left

Die Eigenschaft „margin“ kann ein bis vier Werte erhalten:

**margin: 1px 2px 3px 4px**

- margin-top ist 1px.
- margin-right ist 2px.
- margin-bottom ist 3px.
- margin-left ist 4px.

**margin: 1px 2px 3px**

- margin-top ist 1px.
- margin-right und margin left sind 2px.
- margin-bottom ist 3px.

### margin: 1px 2px

- margin-top und margin-bottom sind 1px.
- margin-right und margin left sind 2px.

### margin: 1px

- margin in alle Richtungen ist 1px.

## 2.2.2 padding

Die „padding“ Eigenschaft legt den Abstand vom Rand eines Elements zu den inneren Elementen fest. Um den Abstand in eine bestimmte Richtung zu verändern, stehen auch hier wieder vier weitere Eigenschaften zur Verfügung:

- Padding-top
- padding-right
- padding-bottom
- padding-left

Wie beim „margin“ können hier ein bis vier Werte angegeben werden. Die Auswirkungsrichtung verhält sich jeweils wie bei margin.

## 2.2.3 height und width

Mit der Eigenschaft „height“ kann die Höhe eines Elements in Pixeln oder prozentual festgelegt werden, mit „width“ die Breite.

## 2.2.4 Formatierung von Schrift

Auch um Schrift zu formatieren gibt es einige Möglichkeiten.

- „color“ legt die Schriftfarbe fest („color: red“, „color: #FFFFFF“).
- Die Schriftgröße wird über „font-size“ bestimmt und kann z.B. über Pixel oder em (em ist eine relative Angabe) angegeben werden.
- Mit „font-weight“ wird die Dicke der Schrift angegeben („font-weight:bold“, „font-weight:200“).
- Die Hintergrundfarbe eines Objekts wird mit „background-color“ bestimmt.
- „z-index“ legt die Ebene eines Objekts fest und so, ob ein Element ein anderes überlappt.

## 2.2.5 Media Queries

Mit sogenannten „Media Queries“ kann man bestimmte CSS Teile auch erst ab einer bestimmten Bildschirmbreite ausführen lassen. Im Beispiel ab einer Breite von unter 1020 Pixel.

```
@media only screen and (max-width: 1013px) {  
  .caption{  
    padding: 20px;  
  }  
}
```

### 2.2.6 Selektoren

Es können auch mehrere Selektoren hintereinander verwendet werden, um so einen Filter-Effekt zu erreichen.

```
.box .body{...}
```

Außerdem kann man mit Selektoren eine Verknüpfung erstellen.

```
.body.default{...}
```

### 2.2.7 „!important“

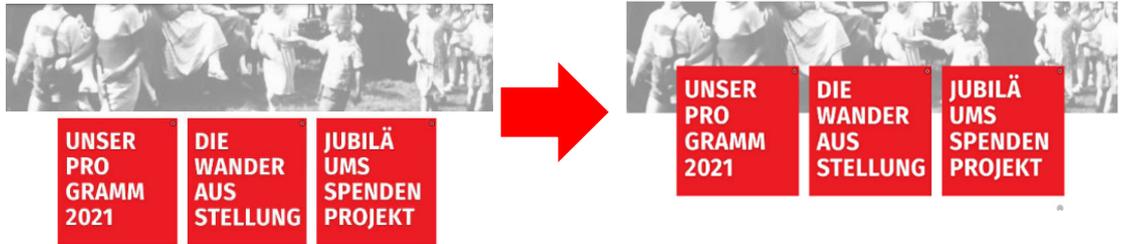
Um zu verhindern, dass Properties überschrieben werden, kann „!important“ hinter den Wert gesetzt werden.

### 2.2.8 Im Internet

Weiter Informationen können Sie unter <https://www.w3schools.com/css/> nachschlagen.

# 3 Elemente verschieben

Es ist mittels CSS möglich, Elemente zu Verschieben. In diesem Beispiel werden Inhalts-Abschnitte verschoben, die jeweils nur aus einem Bild bestehen.



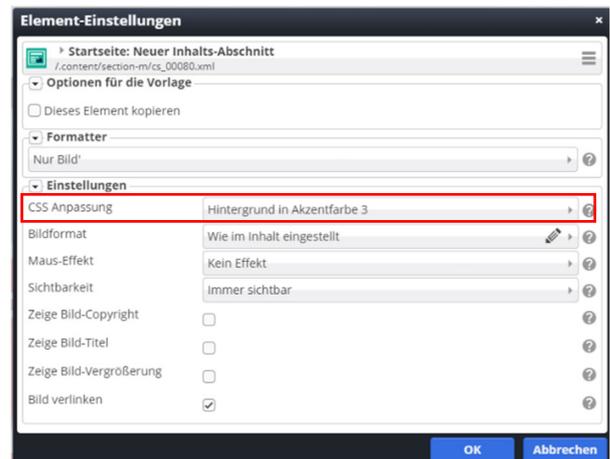
Der folgende Code verschiebt Bilder, die die CSS-Anpassung „Hintergrund in Akzentfarbe 3“ haben, 150 Pixel nach oben. Bei einem schmaleren Bildschirm wird die Anpassung rückgängig gemacht.

```
.box-akzent3{  
  padding: 0px;  
  margin-top: -150px;  
  position: sticky;  
  z-index:5;  
  background-color: transparent;  
}  
@media only screen and (max-width: 1013px) {  
  .box-akzent3{  
    margin-top: 0px;  
  }  
}
```

Hier werden alle Elemente mit der Klasse „box-akzent3 ausgewählt

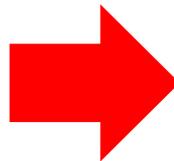
Damit das Elementen im Vordergrund bleibt, muss die Position auf „Sticky“ gesetzt werden

Nun geht man in die Einstellungen des Bildes (des Inhalts-Abschnittes) und wählt die CSS-Anpassung „Hintergrund in Akzentfarbe 3“. Das Element wird verschoben.



# 4 Umformen

Es ist auch möglich, Element umzuformatieren, so kann zum Beispiel folgende Umformung erreicht werden:



Jetzt verwenden wir die CSS-Anpassung „Hintergrund in Akzentfarbe 2“. Der Code schaut dabei so aus:

```
.box-akzent2{
  padding: 0px;
  background-color: #e40013;
}
.box-akzent2 .body.default{
  padding: 0px 15px;
  height: 160px;
}
.box-akzent2 .body.default .heading .piece-heading{
  font-size: 1em;
  font-weight: 200;
  margin-bottom: 7px;
}
.box-akzent2 .body.default .text p{
  font-weight: bold;
  font-size: 1.5em;
}
```

Ein Problem hierbei ist die gewünschte Verlinkung, egal wo man auf die Komponente klickt. Hierfür wird der Linkbutton über die ganze Breite eingefügt und auf die gesamte Höhe ausgedehnt und transparent gemacht.

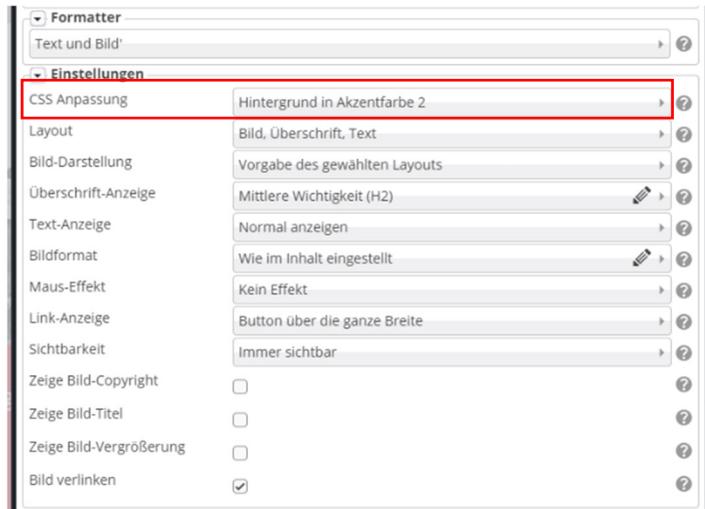
```
.box-akzent2 .link a{
  position: absolute;
  top: 0px;
  bottom: 0px;
  z-index: 20;
  background-color: transparent;
  color: transparent;
}
```

← Hier wird das Festlegen einer fixen Position für das Element ermöglicht, wie es in den nächsten zwei Zeilen geschieht.

## 4. Umformen

---

Die Einstellungen für die Komponente schauen dabei so aus:



Formatter	
Text und Bild	
<b>Einstellungen</b>	
CSS Anpassung	Hintergrund in Akzentfarbe 2
Layout	Bild, Überschrift, Text
Bild-Darstellung	Vorgabe des gewählten Layouts
Überschrift-Anzeige	Mittlere Wichtigkeit (H2)
Text-Anzeige	Normal anzeigen
Bildformat	Wie im Inhalt eingestellt
Maus-Effekt	Kein Effekt
Link-Anzeige	Button über die ganze Breite
Sichtbarkeit	Immer sichtbar
Zeige Bild-Copyright	<input type="checkbox"/>
Zeige Bild-Titel	<input type="checkbox"/>
Zeige Bild-Vergrößerung	<input type="checkbox"/>
Bild verlinken	<input checked="" type="checkbox"/>

## 5 Kachelmenü

Es werden Sliderelemente so verändert, dass mehrere zusammen ein Kachelmenü bilden.



Hier werden die Slider direkt verändert und in den Einstellungen keine CSS-Anpassung verwendet. Alle Sliderelemente auf der Seite sollen die gleiche Veränderung erhalten.

```
.type-slick-slider .slide-wrapper.custom .caption{
  width: 100%;
}
.type-slick-slider .slide-wrapper.custom .caption strong{
  font-size: 2em;
  font-weight: bold;
  height: 400px;
}
.type-slick-slider .slide-wrapper.custom .caption small{
  font-size: 1.5em;
  width: 300px;
  position: absolute;
  left: 50%;
  -webkit-transform: translateX(-50%);
  transform: translateX(-50%);
  top: 200px;
}
@media only screen and (max-width: 1200px) {
  .type-slick-slider .slide-wrapper.custom .caption.center strong{
    font-size: 1.7em;
    height: 500px;
  }
  .type-slick-slider .slide-wrapper.custom .caption.center small{
    font-size: 1.2em;
    top: 150px;
    width: 714px;
  }
}
@media only screen and (max-width: 1013px) {
  .type-slick-slider .slide-wrapper.custom .caption{
    top: 200px !important;
  }
  .type-slick-slider .slide-wrapper.custom .caption.center strong{
    font-size: 3.8em;
    height: 500px;
  }
  .type-slick-slider .slide-wrapper.custom .caption.center small{
    font-size: 2.2em;
    top: 300px;
  }
}
```

Mit diesen vier Zeilen wird das Objekt zentriert.  
Setzt die Entfernung, abhängig von der Oberen Kante des Eltern Objekts (möglich durch „position: absolute“)

Über die Media Queries wird die Schriftgröße und Position an kleinere Bildschirme angepasst.

```
    }  
  }  
  @media only screen and (max-width: 763px) {  
    .type-slick-slider .slide-wrapper.custom .caption{  
      top: 50px !important;  
    }  
    .type-slick-slider .slide-wrapper.custom .caption.center strong{  
      font-size: 2.4em;  
    }  
    .type-slick-slider .slide-wrapper.custom .caption.center small{  
      font-size: 1.3em;  
      top: 300px;  
    }  
  }  
  @media only screen and (max-width: 600px) {  
    .type-slick-slider .slide-wrapper.custom .caption.center strong{  
      font-size: 2em;  
    }  
    .type-slick-slider .slide-wrapper.custom .caption.center small{  
      font-size: 1.1em;  
    }  
  }  
  @media only screen and (max-width: 420px) {  
    .type-slick-slider .slide-wrapper.custom .caption{  
      top: 80px !important;  
    }  
    .type-slick-slider .slide-wrapper.custom .caption.center strong{  
      font-size: 1.4em;  
    }  
    .type-slick-slider .slide-wrapper.custom .caption.center small{  
      font-size: 0.8em;  
      top: 170px;  
    }  
  }  
  @media only screen and (max-width: 350px) {  
    .type-slick-slider .slide-wrapper.custom .caption.center small{  
      top: 130px;  
      font-size: 0.6em;  
    }  
  }  
}
```

Nun werden die Element-Einstellungen des Sliders wie folgt eingestellt:

