

LiaScript Layout Elemente

Generell werden LiaScript Elemente nur in eine Box gerendert. Alle weiteren Layouts können mit Inklusion unserer CSS-Datei und den dort definierten Klassen realisiert werden. Diese müssen in der Preamble eingebunden werden mit:

```
link: ./style.css
```



Bilder

Markdown Bilder können mit `![Alt Text/Beschreibung](bildadresse)` eingefügt werden.



Thomas Wolf, www.foto-tw.de, CC BY-SA 3.0 <https://creativecommons.org/licenses/by-sa/3.0>, via Wikimedia Commons

Der Untertitel ist in diesem Fall händisch unter das Bild geschrieben.

besser!

LiaScript hat die Möglichkeit noch Bilduntertext einzufügen. `![Alt Text](bildadresse "subtitle")`



Thomas Wolf, www.foto-tw.de, CC BY-SA 3.0 <https://creativecommons.org/licenses/by-sa/3.0>, via Wikimedia Commons

Größenanpassungen

Weiterhin kann die Größe des Bildes angepaßt werden. (Und weitere Aspekte die man mit CSS manipulieren kann.) `![Alt Text](URL "subtitle")<!-- style="styleelement" -->` Dies nutzt einen html Kommentar um das `style` tag dem Bild zuzuweisen. `class` geht ebenso. Bei Angabe von css styles ist zu beachten die einzelnen elemente mit `;` zu trennen. `style="max-width: 400px;"` Prozentangaben sind auch möglich.



Thomas Wolf, www.foto-tw.de, CC BY-SA 3.0 <https://creativecommons.org/licenses/by-sa/3.0>, via Wikimedia Commons

Wenn nur eines der Attribute `width` oder `height` angegeben wird, skaliert das Bild im Verhältnis. Diese Attribute setzen die Größe auf den angegebenen Wert, was in kleineren Anzeigen (zB Handy) seltsam wirken kann. Um eine maximale Größe anzugeben kann man `max-width` bzw. `max-height` nutzen. Diese skaliert das Bild automatisch kleiner, wenn weniger Platz zur Verfügung steht. Aber es braucht dazu eine entsprechende Relationale Größenangabe `width: 100%;` zB.



NASA/SpaceX Dragon Capsule

```
<!-- style="width:250px;"-->
```



Soyuz TMA-7

```
<!-- style="max-width:500px; width:100%;"-->
```

Position des Bildes auf der Seite

Eine Zentrierung ist über prozent semantik ebenso möglich. Die Prozente beziehen sich auf die Breite des Containers in dem das Bild aktuell existiert. `<!-- style="width: 50%; margin-left: 25%;" -->` z.B. schiebt dies das Bild um 25% nach rechts.

Der `margin` kontrolliert einen Bereich um einen Block (div) in dem keine anderen Objekte sein dürfen, also wie das weiße auf einer Buchseite bis zum textblock. Man kann mit `margin: 0 0 0 0;` das anpassen (dieser Fall löscht alle margins). Jedes dieser Zahlen steuert einen der Bereiche. In Reihenfolge "oben", "rechts", "unten", "link". Angaben sind zB in `px`, `%` oder auch `cm` möglich.

Z.B.

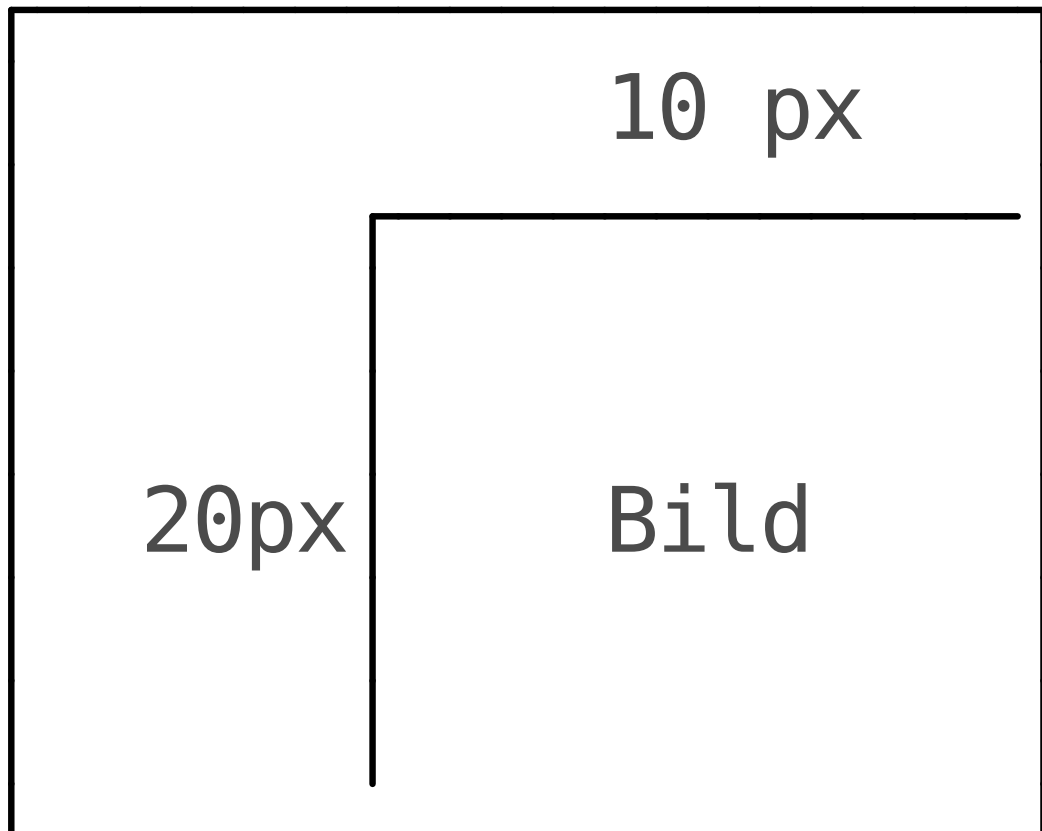
- `margin: 1cm 100px 0 20px;`
- `margin: 2% 3% 10% 2%;`

Ebenso ist es möglich nur die speziellen Werte anzugeben:

- `margin-bottom: 10px;`
- `margin-top: 50cm;`
- `margin-left: -5em;`
- `margin-right: -20cm;`

Um das Bild also „runterzuschieben“ muss der top margin angepasst werden.

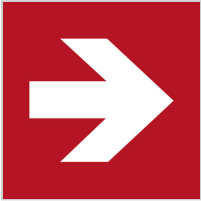
```
<!-- style="margin-left:20px; margin-top:10px;" -->
```



LiaScript Makros für Bilder

Das DiAgnostiK Projekt hat eine Reihe von Makrobibliotheken um Bilder einzubinden. Um Bilder zum Beispiel aus unserem ISO 7010 Repository zu importieren:

```
@Brandschutzzeichen.Richtungspfeil_Rechts(10)
```



Eine komplette Liste aller Symbole findet man in den entsprechenden Repos.

- [Textilpflegesymbole](#)
- [Bildersammlung](#)
- [Piktogramme](#)
- [Holzarten](#)

Mit `.src` hinter dem Befehl, anstelle der runden Klammern, kann der Pfad zu der Datei ausgegeben werden.

```
@Brandschutzzeichen.Richtungspfeil_Rechts.src
```

https://raw.githubusercontent.com/lfi-DiAgnostiK-Project/Piktogramme/refs/heads/main/img/Brandschutzzeichen/Richtungspfeil_Rechts.jpg

Layouts

Neben der Möglichkeit MarkdownElemente mit HTML Kommentaren style Informationen zu übergeben (`<!-- -style="something"-->`), können wir auch direkt HTML Elemente und deren Attribute spezifizieren. `<div class="a class" style="style infos">inhalt</div>` Dabei ist zu beachten die Elemente (`<div>` zB) wieder zu schließen (`</div>`).

Die Style Informationen in den Kommentaren können über einem Markdownelement stehen und gelten dann für alles unten drunter, oder hinter einem Markdownelement und gelten dann nur für dieses.

oranger Kommentar

```
<!-- style="background-color: white; color:orange;"-->  
> oranger Kommentar
```



oranger Kommentar

```
><!-- style="background-color: white; color:orange;"--> oranger Kommentar
```

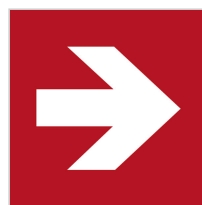
oranger Kommentar

```
> oranger Kommentar<!-- style="background-color: white; color:orange;"-->
```

Text Nebeneinander (Flex)

Wir nutzen die flex styles:

```
<section class="flex-container">  
<div class="flex-child" style="min-width: 200px; width: 50%;">  
@Brandschutzzeichen.Richtungspfeil_Rechts_unten(10)  
  
</div>  
<div class="flex-child" style="min-width: 200px; width: 50%;">  
  
@Brandschutzzeichen.Richtungspfeil_Rechts(10)  
  
</div>  
</section>
```



Aber Achtung wenn horizontal nicht genug Platz ist werden die Container umgebrochen! Wenn ich 3 Container a 300px breite reserviere, aber nur 750 px horizontaler Platz bereits steht sind nur die Container 1 und 2 in der selben Zeile und der dritte wird umgebrochen.

Umbrüche

Das HTML Element `
` sorgt für einen Zeilenumbruch, aber Achtung dass kann für Probleme bei anderen Bildschirmauflösungen sorgen. Lieber die style margin semantic nutzen wenn es um blosse Abstände geht.

Farben (Hinter- und Vordergrund)

Mit `style="background-color: farbe;"` kann die Hintergrundfarbe eines Elements eingestellt werden. Farbnamen und html Farbcodes werden dabei akzeptiert. Die Farbcodes sind Rot Grün Blau kodiert und in Hexadecimal.

Red, #FF0000

Green, #00FF00

Blue, #0000FF

#A090FF

Vordergrundfarben können auf die selbe Art mit `style="color: farbe;"` eingestellt werden.

Dies zum Beispiel für einen Markdown Kommentar `> text`



```
<!--style="background-color:cyan;"-->  
> text
```

also direkt darüber schreiben. Falls viele Änderungen notwendig sind, lieber eine Klasse im css file schreiben.

Dies funktioniert auch direkt für Texte.

Text der rot ist

```
<!-- style="font-weight: bolder; font-size: 20pt; color: #A00000;"-->  
Text der rot ist
```

Oder nur ein **rotes** Wort.

```
Oder nur ein rotes<!-- style="font-weight: bolder; font-size: 20pt; color:  
#A00000;"--> Wort.
```