# Modul 06 – float

Ziel: Korrektes Anwenden von float

Aufgabenstellung:

1. Verwenden Sie die Grundstruktur aus M-004-Lab-Listen-und-Tabellen (oder alternativ aus M-002-Lab-CSS-Basics).
2. Erstellen Sie einen weiteren *demo-container* innerhalb des *main-content-containers*.

HTML:

<div class="demo-container"></div>



Hinweise:

* Da wir im M-004-Lab schon Hintergrundfarbe und Padding für den *demo-container* festgelegt haben, sehen wir in der Vorschau schon etwas, obwohl unser neuer *demo-container* noch keinen Inhalt hat.
* Ohne Padding und ohne Inhalt hätte der *demo-container* eine Höhe von 0 und wäre daher unsichtbar. Im Moment sehen wir nur die Höhe des Paddings.
* Beachten Sie, dass wir keine Breite vergeben haben, der Container sich aber über die gesamte Breite erstreckt – das normale Verhalten eines Elementes mit *display: block*. Für ein Div ist standardmäßig *display: block* vordefiniert.

1. Erstellen Sie innerhalb des neuen *demo-container*s zwei Textblöcke mit Überschrift in jeweils einem neuen Containerelement (z.B. *article*) und etwas Dummytext.

HTML (z.B.):

<div class="demo-container">

        <article class="text-container">

            <h2>Tolle Überschrift</h2>

            <p>lorem20</p>

            <p>lorem10</p>

            <p>lorem15</p>

        </article>

<article class="text-container">

            <h2>Tolle Überschrift</h2>

            <p>lorem20</p>

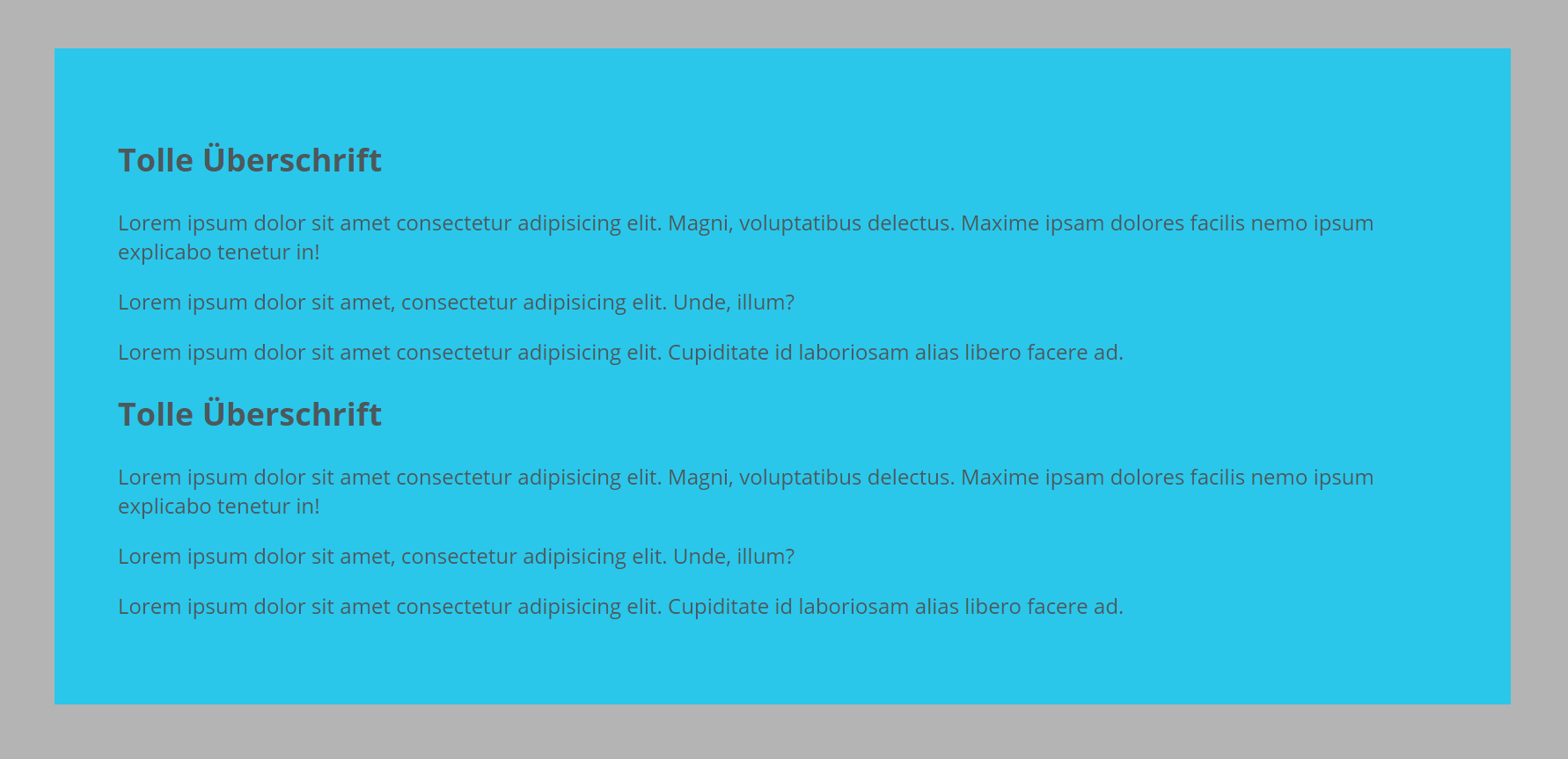
            <p>lorem10</p>

            <p>lorem15</p>

        </article>

     </div>

Hinweis: Mit *lorem+Anzahl Wörter +Enter* können Sie sich von VS Code einen Dummytext generieren lassen.



1. Geben Sie dem *text-container* eine Hintergrundfarbe und einen Rahmen zum Testen, damit wir gut sehen, wo diese Elemente liegen, wo wie viel Abstand ist usw. – beide Properties werden später wieder entfernt.

CSS:

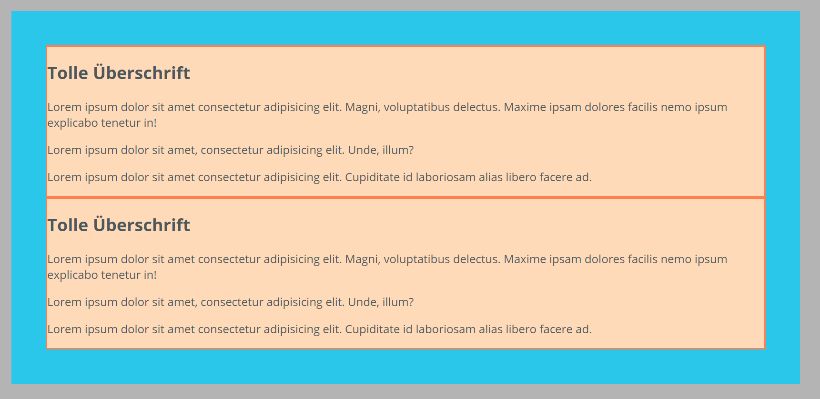
.text-container{

            background-color: peachpuff;

            border: 2px solid coral;

        }

Hinweis: Es ist hilfreich, zum Testen Farben zu verwenden, von denen man sicher weiß, dass sie auf der eigentlichen Webseite nicht vorkommen. Besonders Farbkombinationen, die einem unangenehm auffallen, sind zum Testen hilfreich – sie fallen einem sofort ins Auge, und man vergisst später nicht so leicht darauf, die Testfarbe wieder zu entfernen.



Au! Diese Farben tun ein bisschen weh – und sind somit hervorragend zum Testen geeignet 😉

1. Bringen Sie die beiden Textcontainer mittels *float*-Anweisung nebeneinander.

CSS:

width: 50%;

            float: left;

            box-sizing: border-box;

Hinweis: Ohne *box-sizing: border-box* passen die beiden Textcontainer nicht nebeneinander, auch wenn wir 50% Breite angeben – denn die Breite der Rahmen wird noch außen hinzuaddiert.

Optisch ergibt sich dadurch folgende Katastrophe:

Hinweis: Der übergeordnete Container (*demo-container*) erkennt nicht, dass er einen Inhalt hat.

1. Korrigieren Sie die „optische Katastrophe“, indem Sie Ihrem CSS folgende Anweisung hinzufügen:

.demo-container::after{

            display: block;

            content: '';

            clear: both;

        }

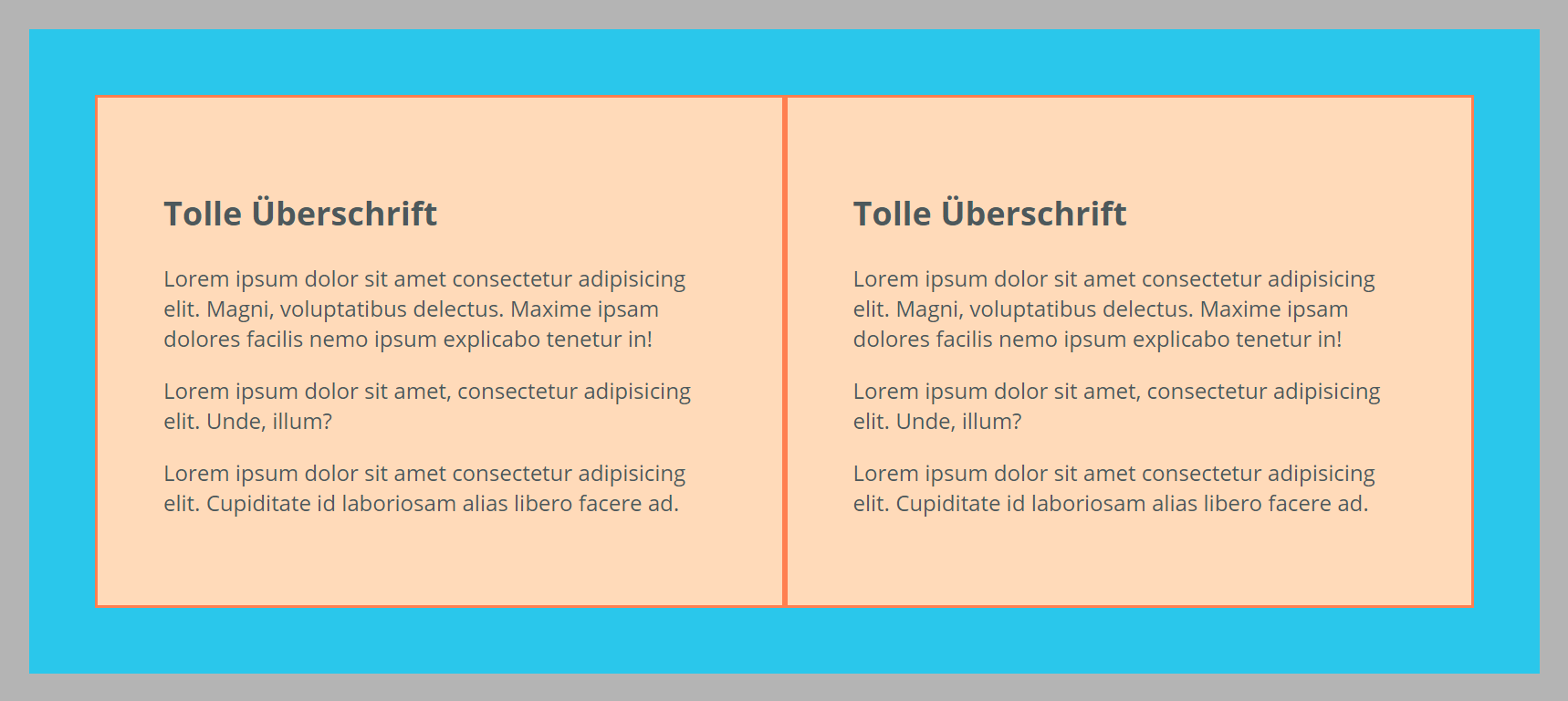
Hinweis: Dabei handelt es sich um ein Pseudoelement – diese brauchen wir sonst erst im fortgeschrittenen Bereich und sind nicht Teil dieses Kurses; wir verwenden nur diese eine Anweisung, um das *float*-Problem in den Griff zu bekommen.

Die Textcontainer sind jetzt da, wo sie hingehören:

1. Geben Sie den Textcontainern ein Padding.

CSS:

 padding: 3em;



1. Wenn Sie mit Ihren Textcontainern zufrieden sind, entfernen Sie die Testhintergrundfarbe und den Rahmen. Manchmal wirken Textblöcke auch besser im Blocksatz:

CSS:

text-align: justify;

