

Pro Person eine HTML Einzelseite anlegen, auf der ihr mit Text, Bildern, Links ein Projekt im Themenbereich vorstellt. Das kann eine der Arbeiten aus der Präsentation am Montag sein, eine Arbeit aus dem Schaulager oder HEK oder auch eine selbstgewählte, andere Arbeit.

Notiere circa 2000 Zeichen (ohne Leerzeichen) zu:

Worum geht es im Projekt?

Mit welcher Motivation/Haltung wird gearbeitet?

Was wurde gemessen?

Welche Verfahren der Visualisierung wurden angewendet?

Git und Github

Wofür brauchen wir Git und Github?

- Version History
- Zusammenarbeit
- Deployment

Video zu Unterschied Git und Github:

<https://www.youtube.com/watch?v=BCQHnlnPusY>

Daily Project

Notiere als Vorbereitung einen Tag lang was du wann trinkst. Überlege, was du alles erfassen willst: Farbe, Temperatur, Volumen, mit wem du zusammen bist, etc. Setze auch Fotografie als Dokumentationsmittel ein.

<https://www.pinterest.com/pin/77546424806199639/>

<https://photoviz.tumblr.com/>

[Coffee Data Visualization](#)

<http://feltron.com/PhotoViz.html>

Grid Layouts: https://www.w3schools.com/css/css_grid_item.asp

JavaScript Filter Elements by Classname: https://www.w3schools.com/howto/howto_js_filter_elements.asp

Selektoren

CSS Kaskade und Spezifität

Aufbau von File Pfaden

Bilder einbinden

(Masseinheiten)

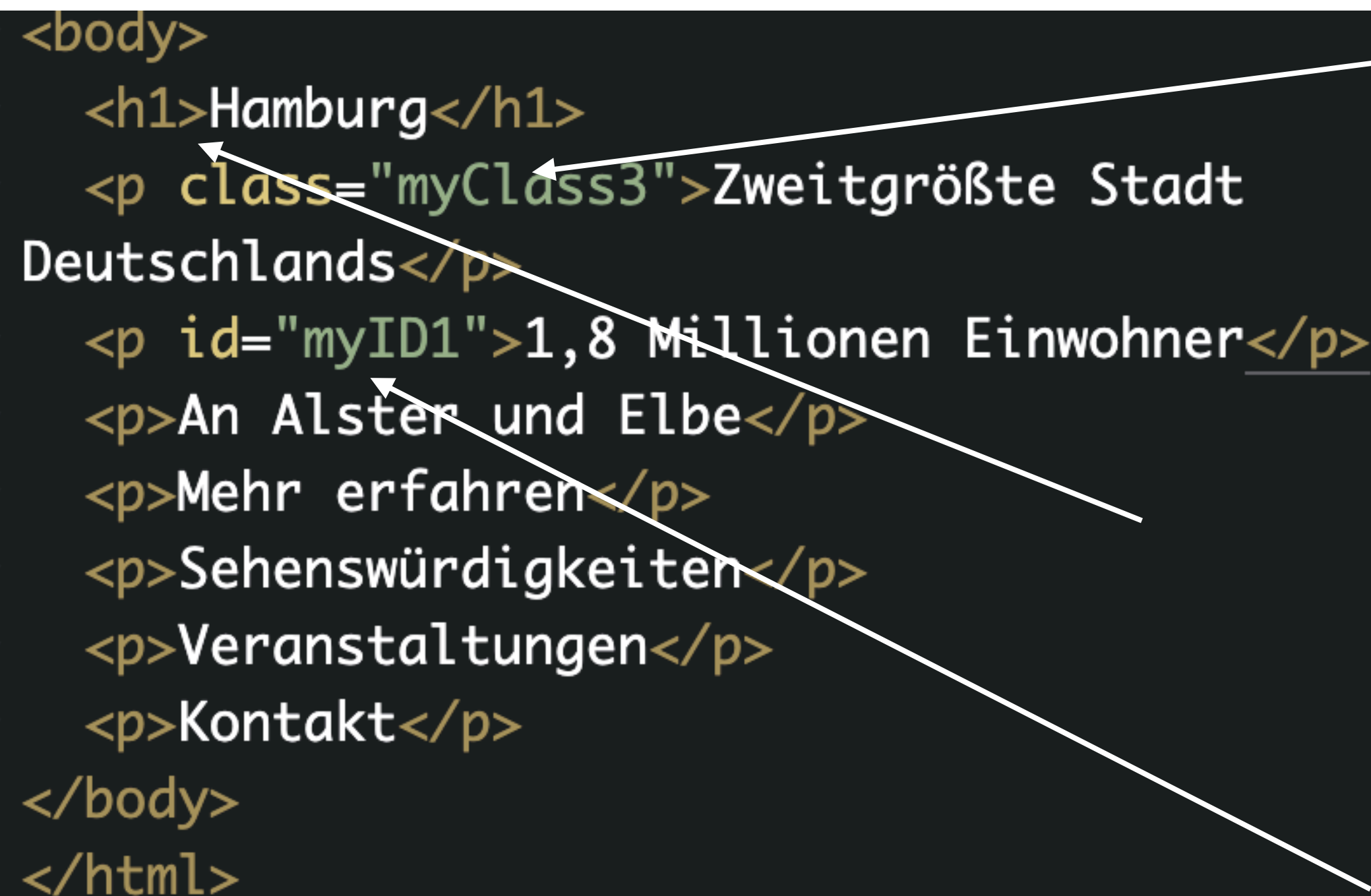
Box Model

Layoutfluss, display und (position)

Properties

Einfache CSS-Selektoren

```
<body>
  <h1>Hamburg</h1>
  <p class="myClass3">Zweitgrößte Stadt
Deutschlands</p>
  <p id="myID1">1,8 Millionen Einwohner</p>
  <p>An Alster und Elbe</p>
  <p>Mehr erfahren</p>
  <p>Sehenswürdigkeiten</p>
  <p>Veranstaltungen</p>
  <p>Kontakt</p>
</body>
</html>
```



```
.myClass3 {
  text-align: center;
  background-color: yellow;
  padding: 10px;
  margin: 5px;
  font-size: 14px;
  color: green;
  font-family: "Comic Sans MS", Arial;
}
h1 {
  width: 100%;
  font-weight: bold;
  text-align: center;
  font-size: 50px;
  color: indianred;
  font-family: Lato, "sans serif";
}
#myID1 {
  width: 100%;
}
```

Klasse

Tag

ID

Cascading Style Sheets

CSS Kaskade und Spezifität

Wie entscheidet der Browser, was angezeigt wird,
wenn mehrere Anweisungen auf dasselbe Element
zutreffen?

1. Reihenfolge (Kaskade)

Stylesheets kaskadieren - auf einer sehr einfachen Ebene bedeutet dies, dass die Reihenfolge der CSS-Regeln eine Rolle spielt; wenn zwei Regeln gelten, die die gleiche Spezifität haben, wird diejenige verwendet, die im CSS zuletzt steht.

Zuletzt bedeutet zugleich, am nächsten beim HTML Element.

https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance

Styleangaben mit !important gewinnen immer:

```
.better {  
    background-color: gray;  
    border: none !important;  
}
```

Zunahme Nähe

Variante 1: Extern im Head

```
<link href="css/style.css" rel="stylesheet">
```

Variante 2: als style im Head

```
<style>
  /* Basisgestaltung */
  html {
    font-family: sans-serif;
  }

  /* Abstände Logo und Slogan anpassen */
  .site-header {
    margin-bottom: 2em;
  }
  .site-logo {
    margin-bottom: 0;
  }
  .site-slogan {
    margin: 0;
  }
  .site-content {
    line-height: 1.5;
  }
</style>
```

Variante 3: als style beim Tag

```
<header class="site-header" style="font-color:red">
```


2. Spezifität

Die Spezifität ist die Art und Weise, wie der Browser entscheidet, welche Regel gilt, wenn mehrere Regeln unterschiedliche Selektoren haben, aber dennoch auf dasselbe Element zutreffen könnten. Sie ist im Grunde ein Maß dafür, wie spezifisch die Auswahl eines Selektors sein wird:

Ein Elementselektor ist weniger spezifisch - er wählt alle Elemente dieses Typs aus, die auf einer Seite erscheinen - und erhält daher eine niedrigere Punktzahl.

Ein Klassenselektor ist spezifischer - er wählt nur die Elemente auf einer Seite aus, die einen bestimmten Klassenattributwert haben - und erhält daher eine höhere Punktzahl.

https://www.w3schools.com/css/css_specificity.asp

```
.main-heading {  
    color: red;  
}  
  
h1 {  
    color: blue;  
}
```

```
<h1 class="main-heading">This is my heading.</h1>
```

Beispiel:

Das Element wird rot dargestellt, weil die Klasse spezifischer als der Tag ist.

Adressierung, File Paths

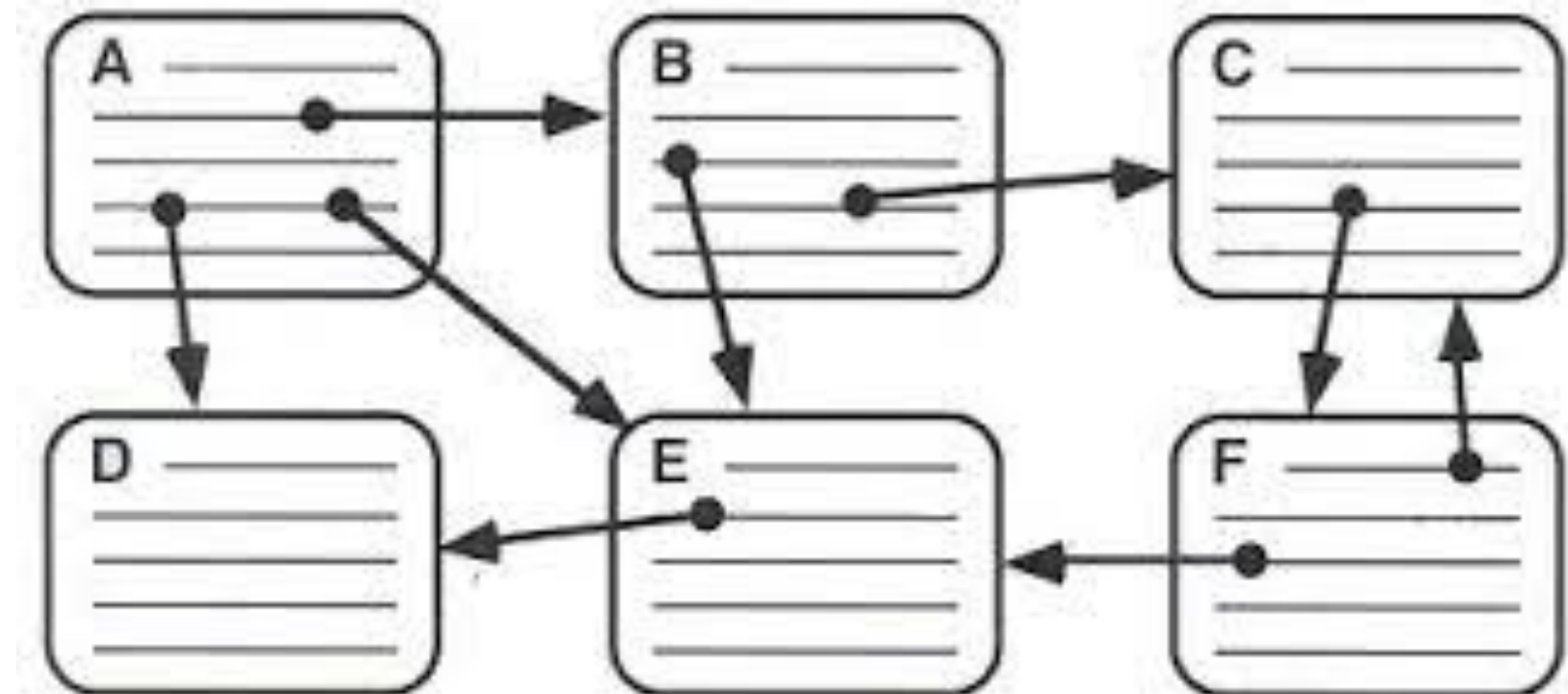
Ein Dateipfad beschreibt den Speicherort einer Datei in der Ordnerstruktur einer Website. Dateipfade werden verwendet, wenn auf externe Dateien verlinkt wird, wie z.B.:

Web-Seiten

Bilder

Stilvorlagen

JavaScripts



https://www.w3schools.com/html/html_filepaths.asp

Absolute File Paths

Ein absoluter Dateipfad ist die vollständige URL zu einer Datei, diese Art von Pfad wird benutzt, wenn die andere Datei auf einem anderen Server oder einer anderen Domain liegt.

Bsp. Einbinden eines Bildes von Wikimedia

```

```

Bsp. Link auf Google Map

```
<a href="https://goo.gl/maps/BTmcRRavwyqr3Uve7"> HSLU Viscosi auf Google Map </a>
```

Relative File Paths

Bei relativen File Paths bildet sich der Pfad aus dem hierarchischen Bezug der Datei, in der der Link liegt, zur Datei, die eingebunden wird. Ihr denkt immer von der Datei aus, in der der Link liegt.

https://www.w3schools.com/html/html_filepaths.asp

Beispiele

```
 The "picture.jpg" file is located in the same folder as the current page  
 The "picture.jpg" file is located in the images folder in the current folder  
 The "picture.jpg" file is located in the folder one level up from the current folder
```

Code along

Kaskade und Spezifität

01-repetiton-cascade.html —> beschrieben in assets/cascade.css

Pfade

Lege eine Datei an und binde zwei Bilder ein:

- eines von Wikimedia oder einem sonstigen Anbieter über absolute Pfade
- lege einen Folder an, den du ‚img‘ nennst, speichere dort ein Bild ab und binde es ein.
- lege zuoberst in deinem Repository ein index.html File an und erstelle Links zu den anderen Seiten, style die Links als Menü

HTML Bilder einbinden

HTML Tags, um Bilder einzubinden

<picture>

<figure>

<background>

HTML Bilder einbinden

** tag**

```

```

Das alt Attribute beschreibt den Inhalt für Screenreader.

Es ist notwendig, dieses Attribut zu setzen.

HTML Bilder einbinden

<picture> tag

```
<picture>
  <source media="(min-width: 700px)"
    srcset="uploads/20190825000042/geeks-221.png">

  <source media="(min-width: 450px)"
    srcset="uploads/20190802021607/geeks14.png">

  
</picture>
```

Das picture Element bietet die Möglichkeit, je nach Grösse des Ausgabemediums ein anderes Bild einzubinden.

HTML Bilder einbinden

<figure>

```
<figure>  
      
    <figcaption>Here is a image of laptop.</figcaption>  
</figure>
```

Der Unterschied zwischen dem figure- und dem image-Tag ist recht einfach. Der image-Tag wird verwendet, um das Bild in ein HTML-Dokument einzubetten, während der figure-Tag verwendet wird, um den Inhalt eines Bildes im HTML-Dokument semantisch zu organisieren. <figcaption> ist der semantisch korrekte Tag für die Bildlegende.

HTML Bilder einbinden

<background>

```
body {  
    background-image: url("paper.gif");  
}
```

Die Eigenschaft background-image gibt ein Bild an, das als Hintergrund für ein Element verwendet werden soll.

Standardmäßig wird das Bild wiederholt, so dass es das gesamte Element abdeckt.

https://www.w3schools.com/css/css_background_image.asp

HTML Bilder einbinden

Bildformate

Gängige Formate sind:

- .jpeg, .jpg
- .gif
- .png
- .svg

https://developer.mozilla.org/en-US/docs/Web/Media/Formats/Image_types

HTML Bilder einbinden

jpeg und gif Vergleich

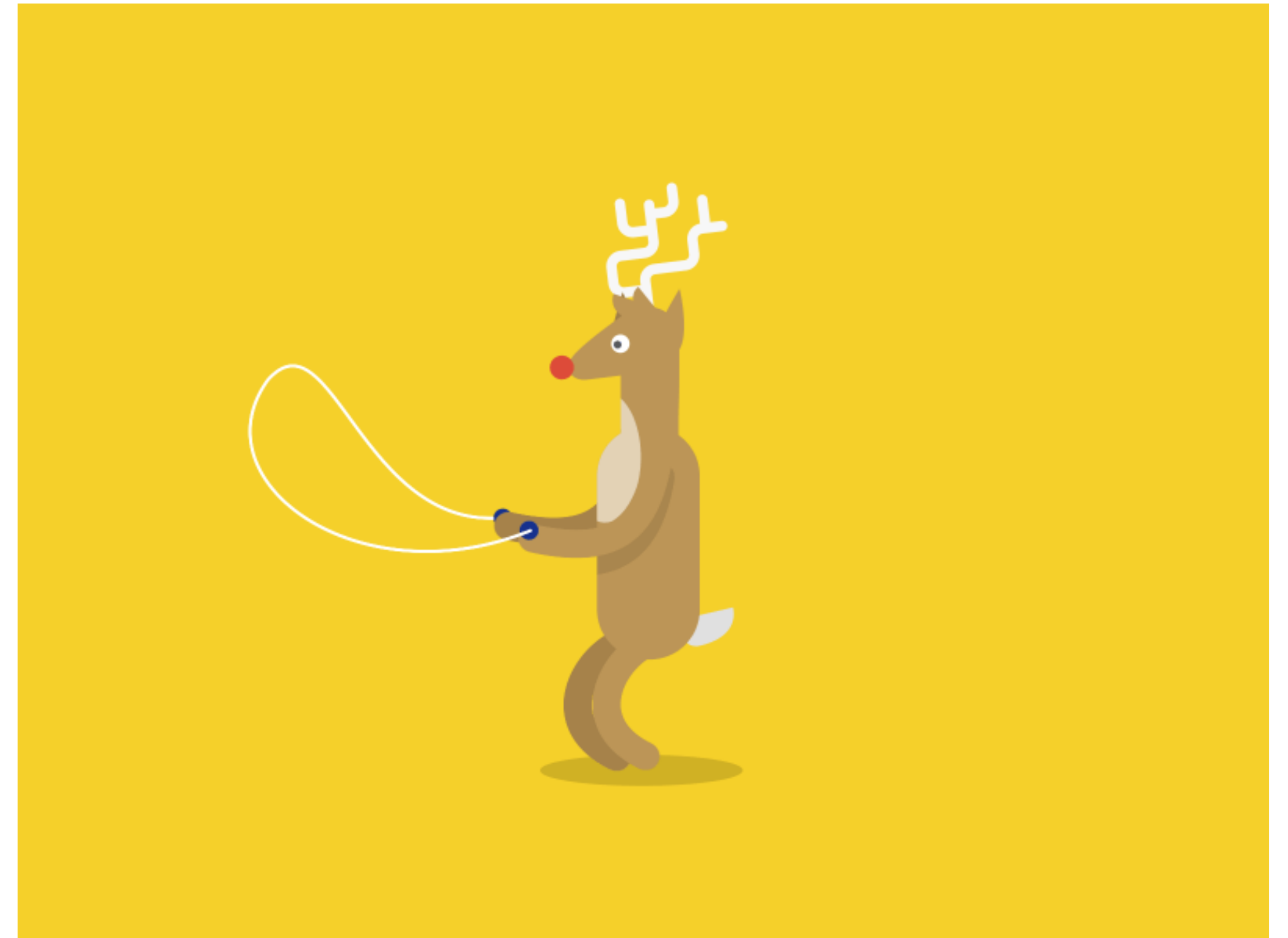
JPEG bietet eine verlustbehaftete Komprimierung von Standbildern (derzeit die beliebteste). **JPEG eignet sich viel besser für Fotos, während GIF für computergenerierte Bilder, Logos und Strichzeichnungen mit begrenzten Paletten geeignet ist.** Rechts der Vergleich zwischen einem stärker komprimierten JPG und demselben GIF. In diesem Bild unten ist die Dateigröße des JPG auf der linken Seite 18 Mal kleiner als die des GIF auf der rechten Seite.



HTML Bilder einbinden

jpeg und gif Vergleich

GIF verwendet einen verlustfreien Komprimierungsalgorithmus, so dass im GIF-Format kein Bilddatenverlust auftritt. GIF-Bilder unterstützen **Animation und Transparenz**.



HTML Bilder einbinden

gif und png Vergleich

- png unterstützt keine Animationen.
- Die Filegrösse von png ist generell grösser als die von gif
- Gif hat eine Farbtiefe von 256
- png hat eine Farbtiefe von über 16 Mio Farben
- Beide haben einen Transparenzkanal. Der von png ist smoother, der von gif hat bloss ein Bit, d.h. entweder transparent oder nicht.

HTML Bilder einbinden

svg

Vektorbildformat; ideal für Elemente der Benutzeroberfläche, Icons, Diagramme usw., die in **verschiedenen Größen genau gezeichnet** werden müssen. Kann z.Bsp. aus Illustrator exportiert werden.

<https://www.svgrepo.com/svg/4733/samples>

HTML Bilder einbinden

Bilder fürs Web optimieren!

Im Allgemeinen bedeutet Bildoptimierung die **Verringerung der Dateigröße** ohne Qualitätsverlust. Sie können Ihre Bilder in der Erstellungsphase optimieren (z. B. mithilfe der richtigen "Export"-Optionen in Photoshop). Ziel ist es, die **Datenmenge zu reduzieren**, die ein Nutzer herunterladen muss, damit er den gesuchten Inhalt schneller findet.

- **Bilder zuschneiden, auf die richtige Pixelgrösse reduzieren**
- **Den richtigen Filetype wählen**
- **Komprimieren**
- **Für responsive design verschiedene Bildgrößen wählen**

Code along

Deine Bilder von den Getränken aufbereiten (jpg ist für Fotografien meistens das richtige Format).

Speichern im Projekt in einem ‚img‘ Folder

Lege eine html Seite an und integriere die Bilder, einmal über img und über background Tag.

Integriere figure und figcaption, beschreibe bei figcaption, was relevant ist.

Masseinheiten

px Pixel

% Prozent relativ zum Elternelement

em Schriftgrösse des Elements selbst

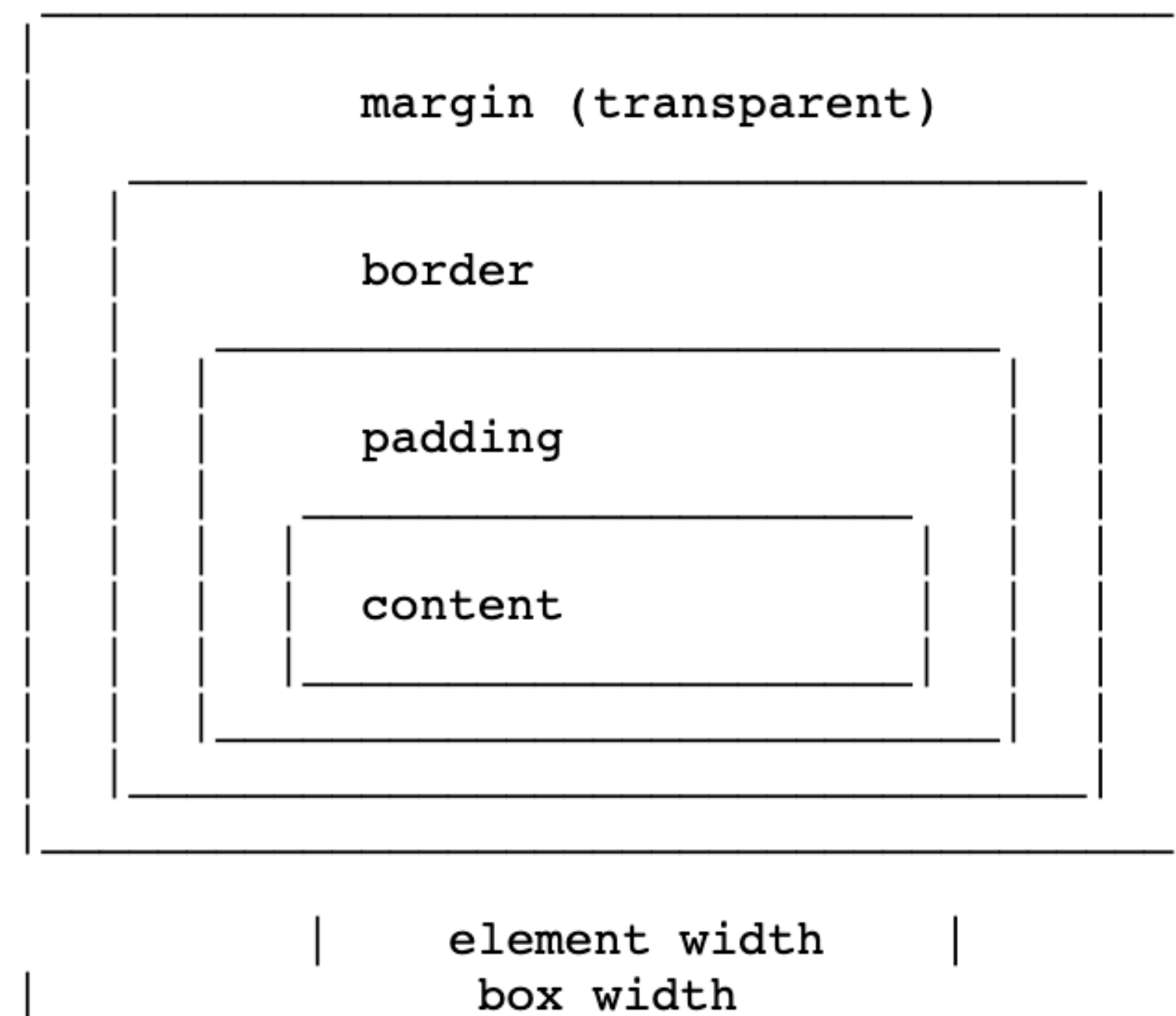
rem root em Schriftgrösse des Browsers

vw, vh viewport width und viewport height

https://www.w3schools.com/css/css_font_size.asp

Box Model – Layout

Gilt für alle Elemente mit `display:block;`

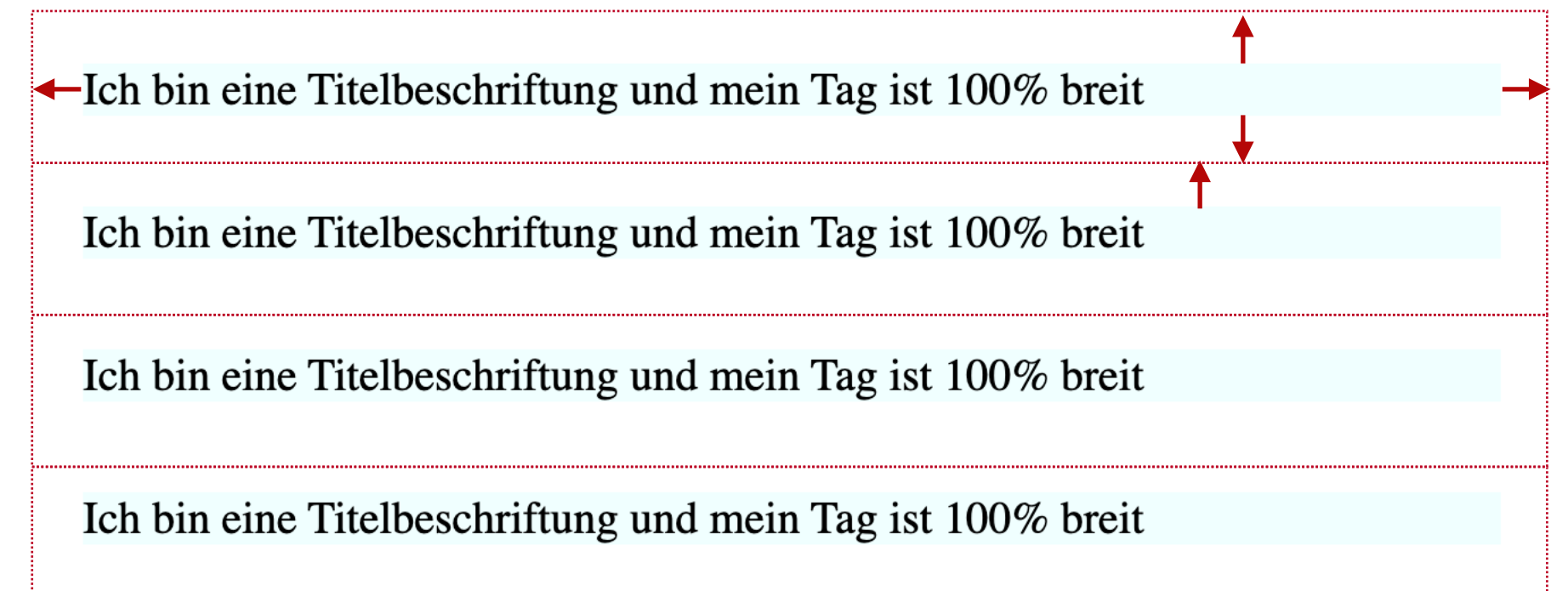


EINFÜHRUNG CSS

margin Property

```
p {  
  margin-top: 100px;  
  margin-bottom: 100px;  
  margin-right: 150px;  
  margin-left: 80px;  
}  
p {  
  margin: 100px; /* alle Richtungen */  
}
```

https://www.w3schools.com/css/css_margin.asp



EINFÜHRUNG CSS

padding Property

```
div {  
  padding-top: 50px;  
  padding-right: 30px;  
  padding-bottom: 50px;  
  padding-left: 80px;  
}  
p {  
  padding: 100px; /* alle Richtungen */  
}
```

https://www.w3schools.com/css/css_padding.asp

Ich bin eine Titelbeschriftung und mein Tag ist 100% breit

Ich bin eine Titelbeschriftung und mein Tag ist 100% breit

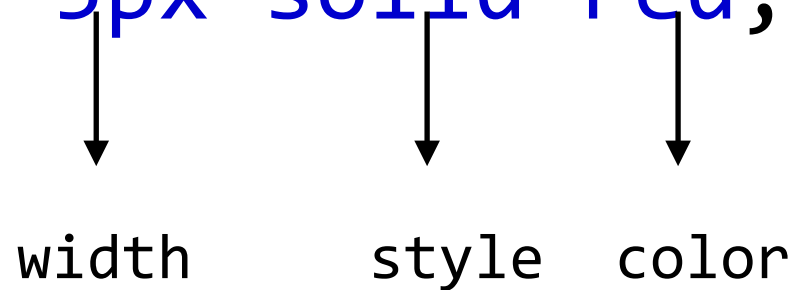
Ich bin eine Titelbeschriftung und mein Tag ist 100% breit

Ich bin eine Titelbeschriftung und mein Tag ist 100% breit

EINFÜHRUNG CSS

border Property

```
h1 {  
  border: 5px solid red;  
}
```



width style color

```
p {  
  border-radius: 50%; /* kreis */  
}
```

https://www.w3schools.com/cssref/pr_border.asp

EINFÜHRUNG CSS

box-shadow Property

```
#example1 {  
  border: 1px solid;  
  padding: 10px;  
  box-shadow: 5px 10px 8px 10px rgb(20,30,100);  
}
```

Diagram illustrating the components of the box-shadow property:

- 5px: h-offset
- 10px: w-offset
- 8px: blur
- 10px: spread
- rgb(20,30,100): color

https://www.w3schools.com/cssref/css3_pr_box-shadow.asp

EINFÜHRUNG CSS

Gradient Background

```
#grad {  
  background-image: linear-gradient(red, yellow); /*von oben nach unten*/  
}
```

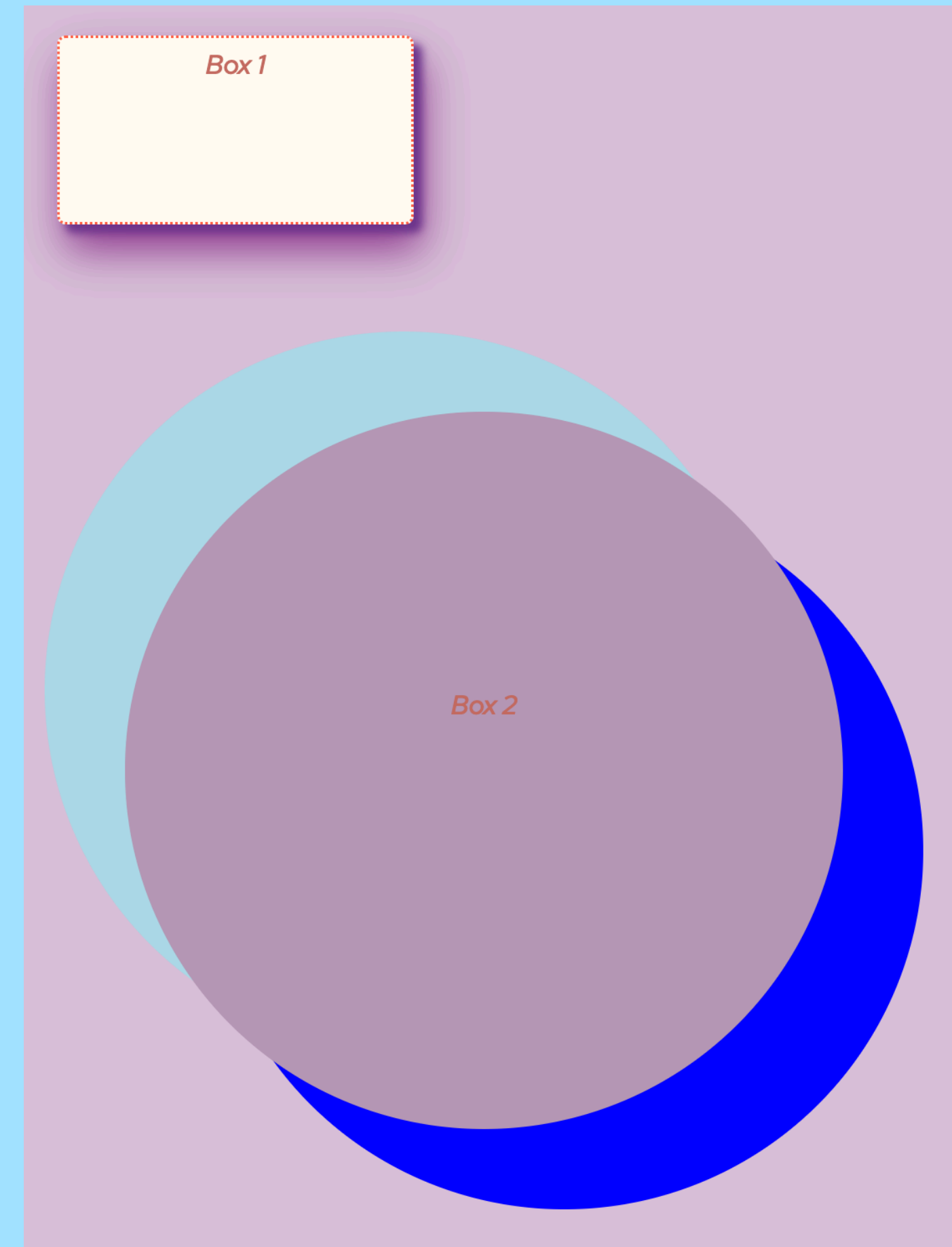
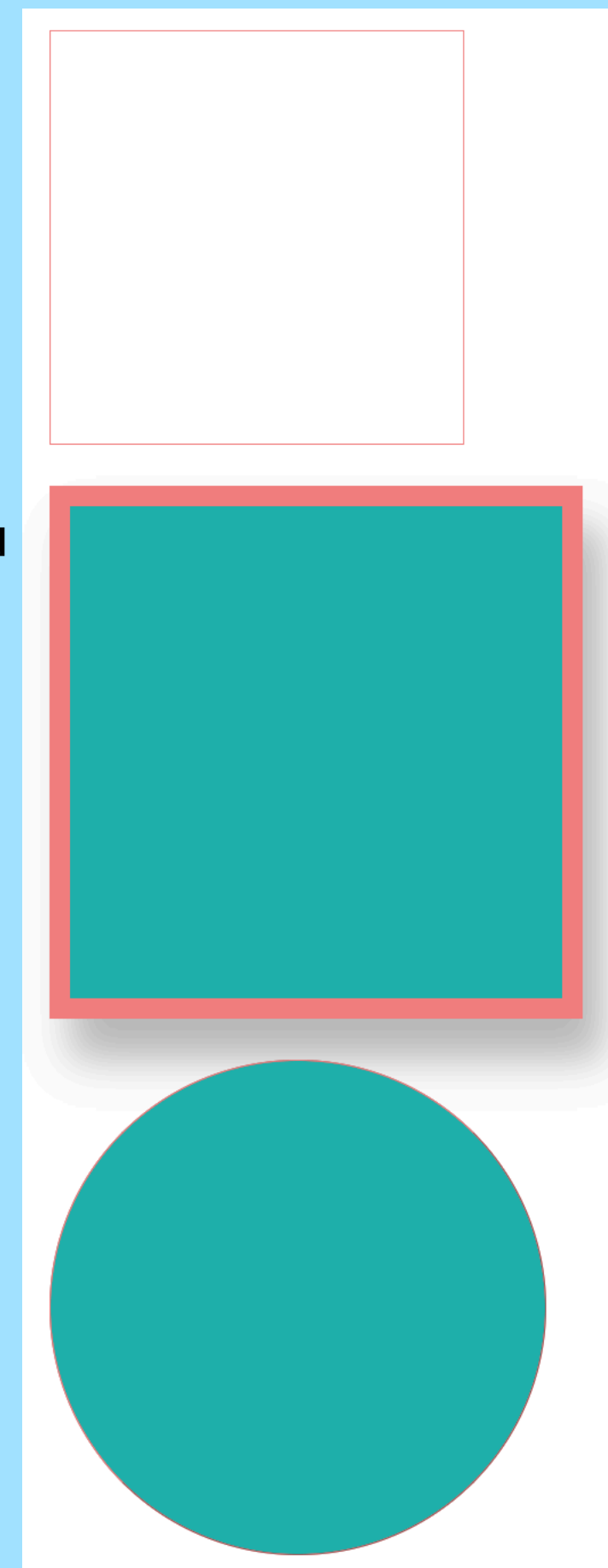
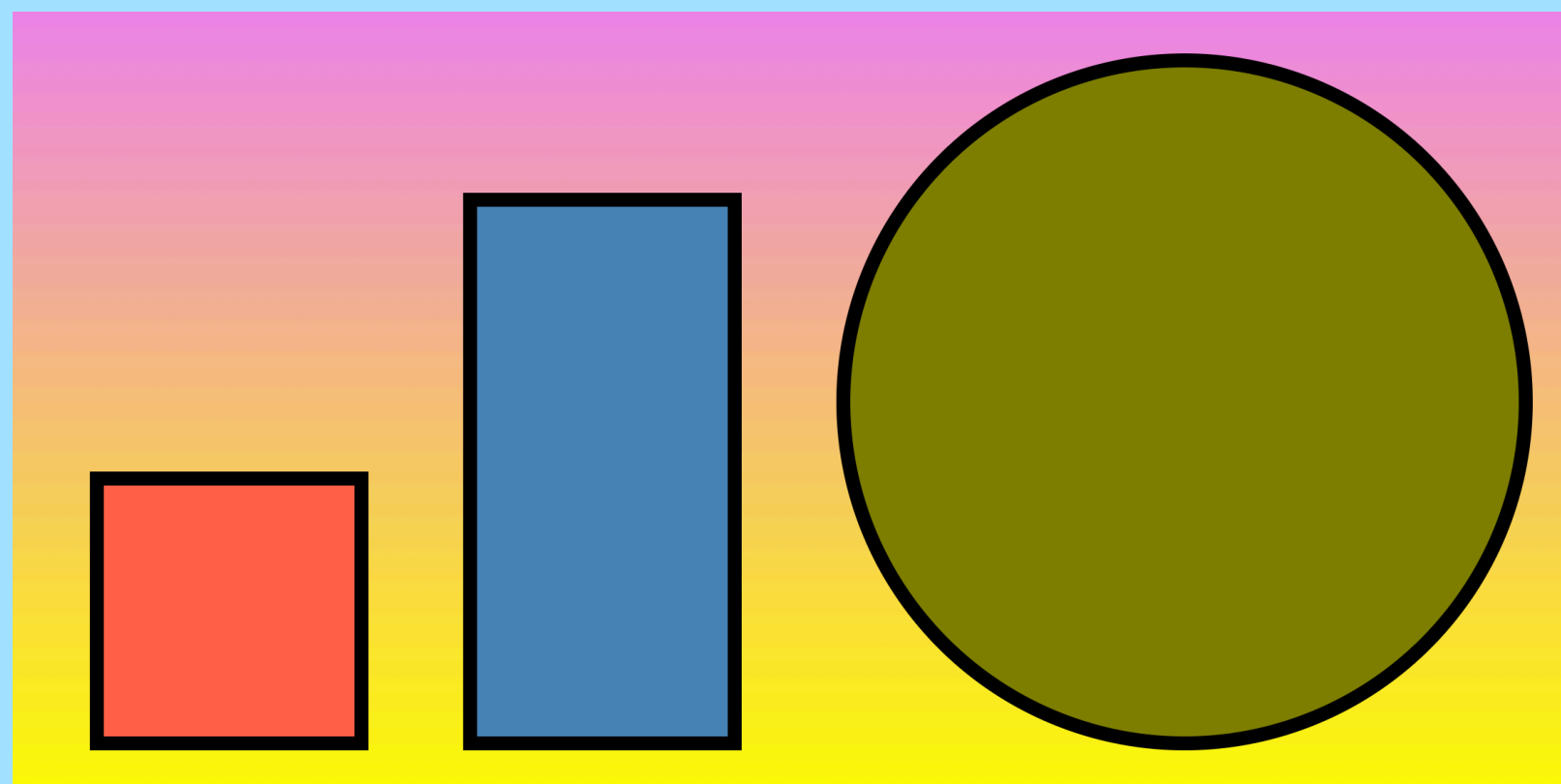
https://www.w3schools.com/css/css3_gradients.asp

Code along

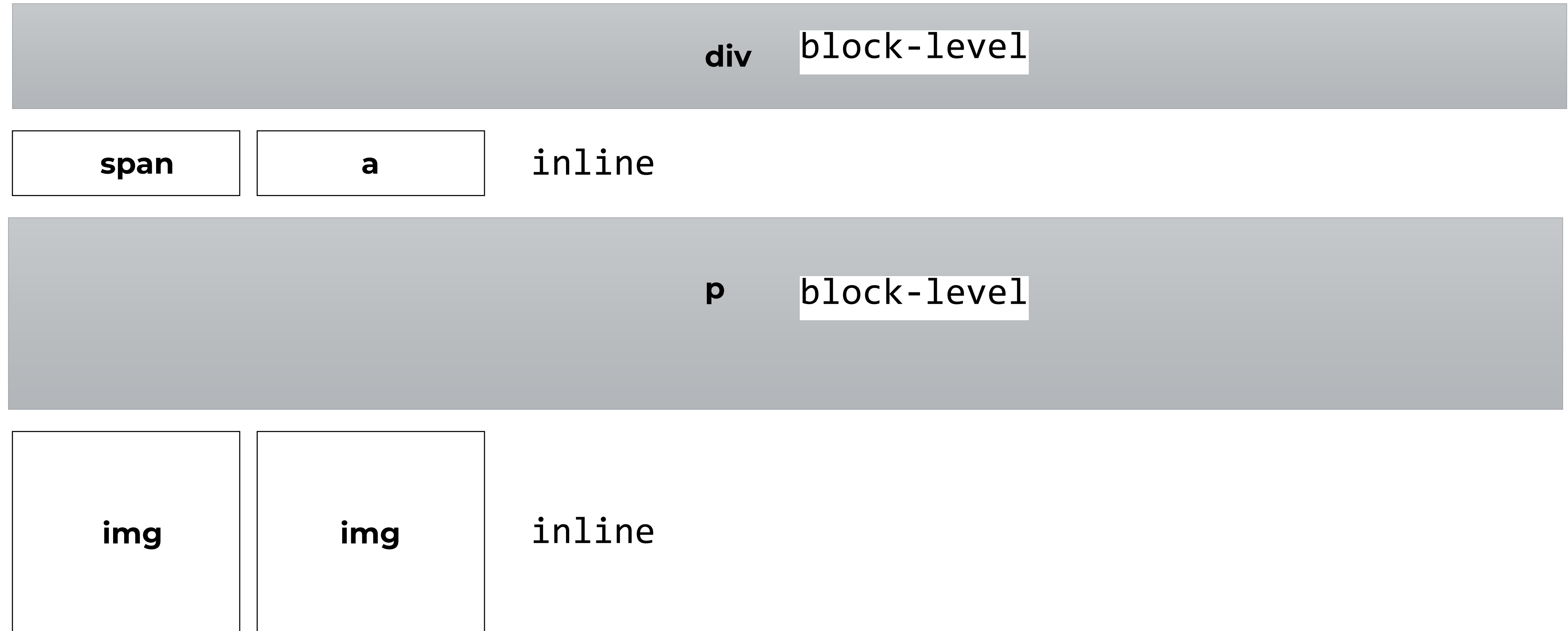
Fun with boxes

Benutze die Properties margin, padding und verschiedene Container zu platzieren.

Style sie mit border, background-color, box-shadow.



Layoutfluss



Block Level Elemente brauchen im **Layoutfluss die gesamte Breite**, auch wenn der Content nicht so viel Platz einnehmen würde.

Inline Elemente brauchen im Layoutfluss **bloss so viel Breite, wie ihr Inhalt benötigt**.

https://www.w3schools.com/html/html_blocks.asp

Block Level

Block-Level-Elemente

Ein Block-Level-Element beginnt immer auf einer neuen Zeile und nimmt die gesamte zur Verfügung stehende Breite ein (streckt sich nach links und rechts so weit wie möglich aus).

`<address>`
`<article>`
`<aside>`
`<blockquote>`
`<canvas>`
`<dd>`
`<div>`
`<dl>`
`<dt>`
`<fieldset>`
`<figcaption>`

`<figure>`
`<footer>`
`<form>`
`<h1>-<h6>`
`<header>`
`<hr>`
``
`<main>`
`<nav>`
`<noscript>`
``

`<p>`
`<pre>`
`<section>`
`<table>`
`<tfoot>`
``
`<video>`

Inline

Ein Inline-Element beginnt nicht in einer neuen Zeile und nimmt nur so viel Breite wie nötig ein.

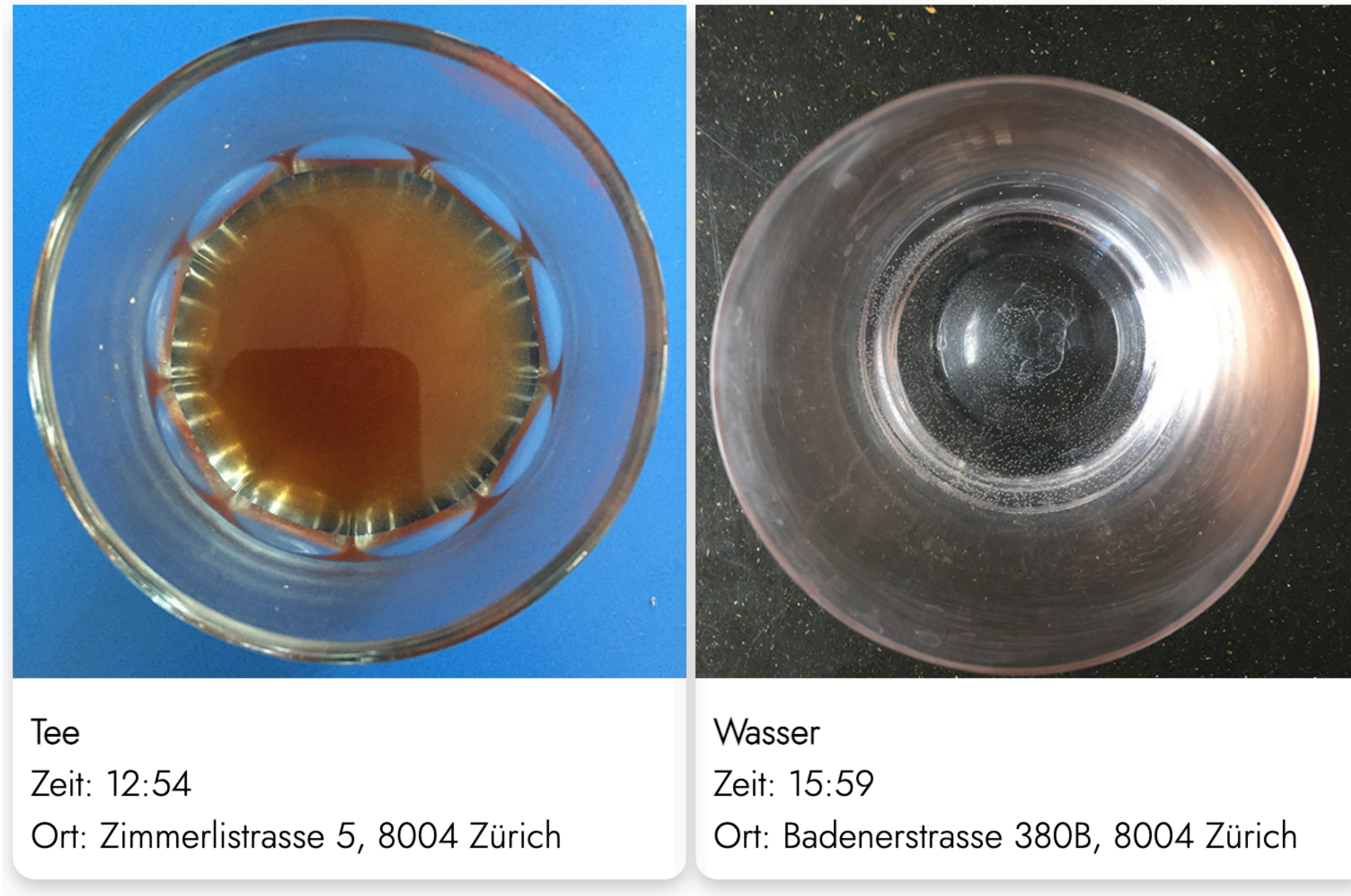
<code><a></code>	<code><map></code>
<code><abbr></code>	<code><object></code>
<code><acronym></code>	<code><output></code>
<code></code>	<code><q></code>
<code><bdo></code>	<code><samp></code>
<code><big></code>	<code><script></code>
<code>
</code>	<code><select></code>
<code><button></code>	<code><small></code>
<code><cite></code>	<code></code>
<code><code></code>	<code></code>
<code><dfn></code>	<code><sub></code>
<code></code>	<code><sup></code>
<code><i></code>	<code><textarea></code>
<code></code>	<code><time></code>
<code><input></code>	<code><tt></code>
<code><kbd></code>	<code><var></code>
<code><label></code>	

display property

```
li {  
  display: inline;  
}  
span {  
  display: block;  
}  
h1.hidden {  
  display: none;  
}  
.card {  
  display: inline-block;  
}
```

https://www.w3schools.com/css/css_display_visibility.asp

simples card layout



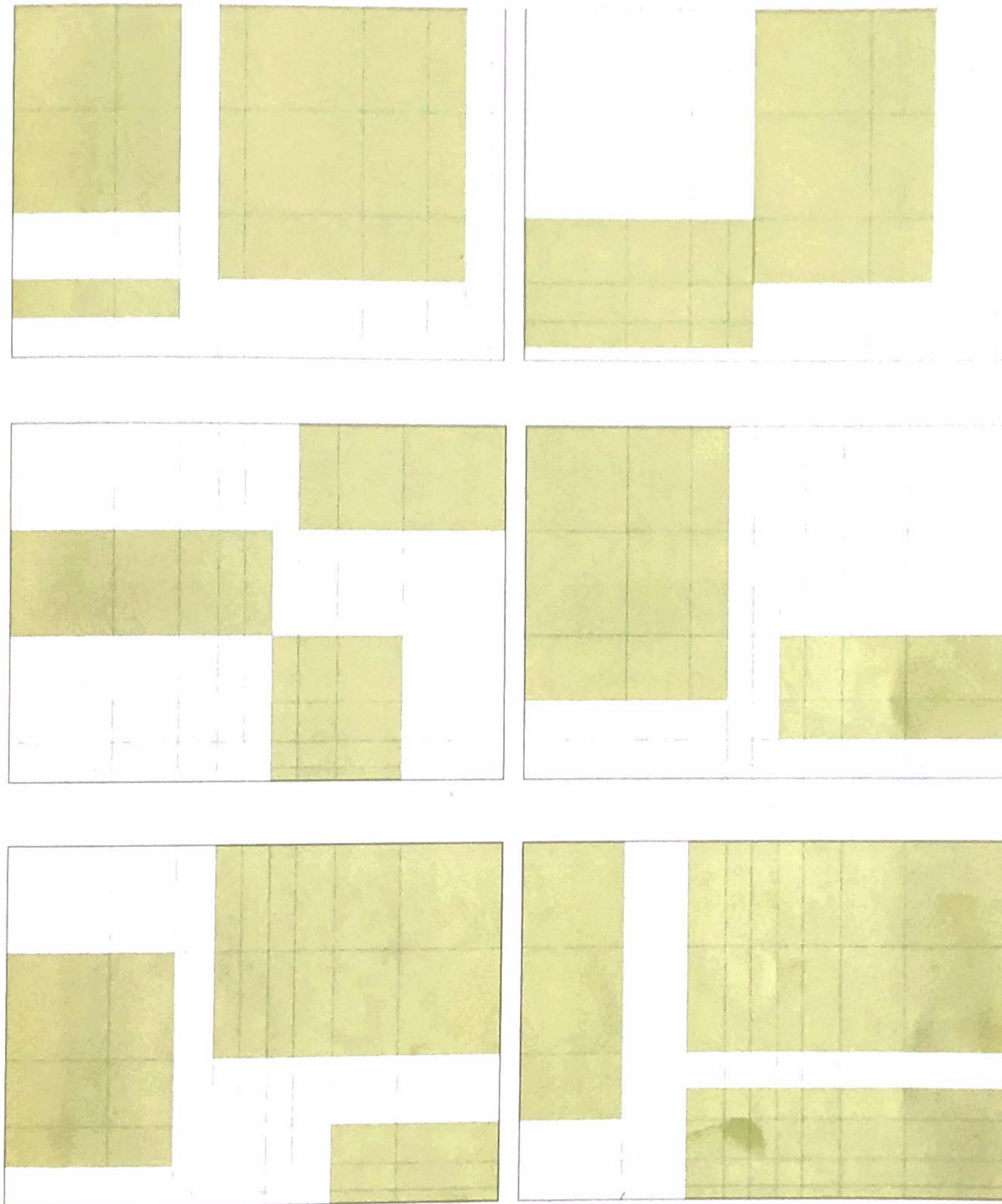
```

<!-- HTML -->
<div class="card">
  
  <div class="info">
    <h4>Tee</h4>
    <p>Zeit: 12:54</p>
    <p>Ort: Zimmerlistrasse 5, 8004 Zürich</p>
  </div>
</div>

/*--CSS--*/

.card {
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2);
  border-radius: 10px;
  width: 300px;
  display: inline-block; /*positioniert die cards nebeneinander*/
}
img {
  width: 100%;
}
.info {
  font-family: 'Jost', sans-serif;
  padding: 0.5rem;
}
h4,
p {
  margin: 0px;
  padding: 0px;
}

```

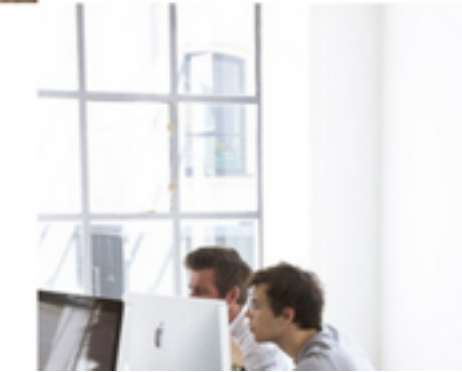



Die Grafiken oben illustrieren, dass der Raster von Le Corbusier bei der Gestaltung von Layouts sehr unterschiedlich und flexibel eingesetzt werden kann. Die Formen ergeben sich durch den Raster und Le Corbusiers Zahlenansatz und bilden so eine harmonische Einheit.

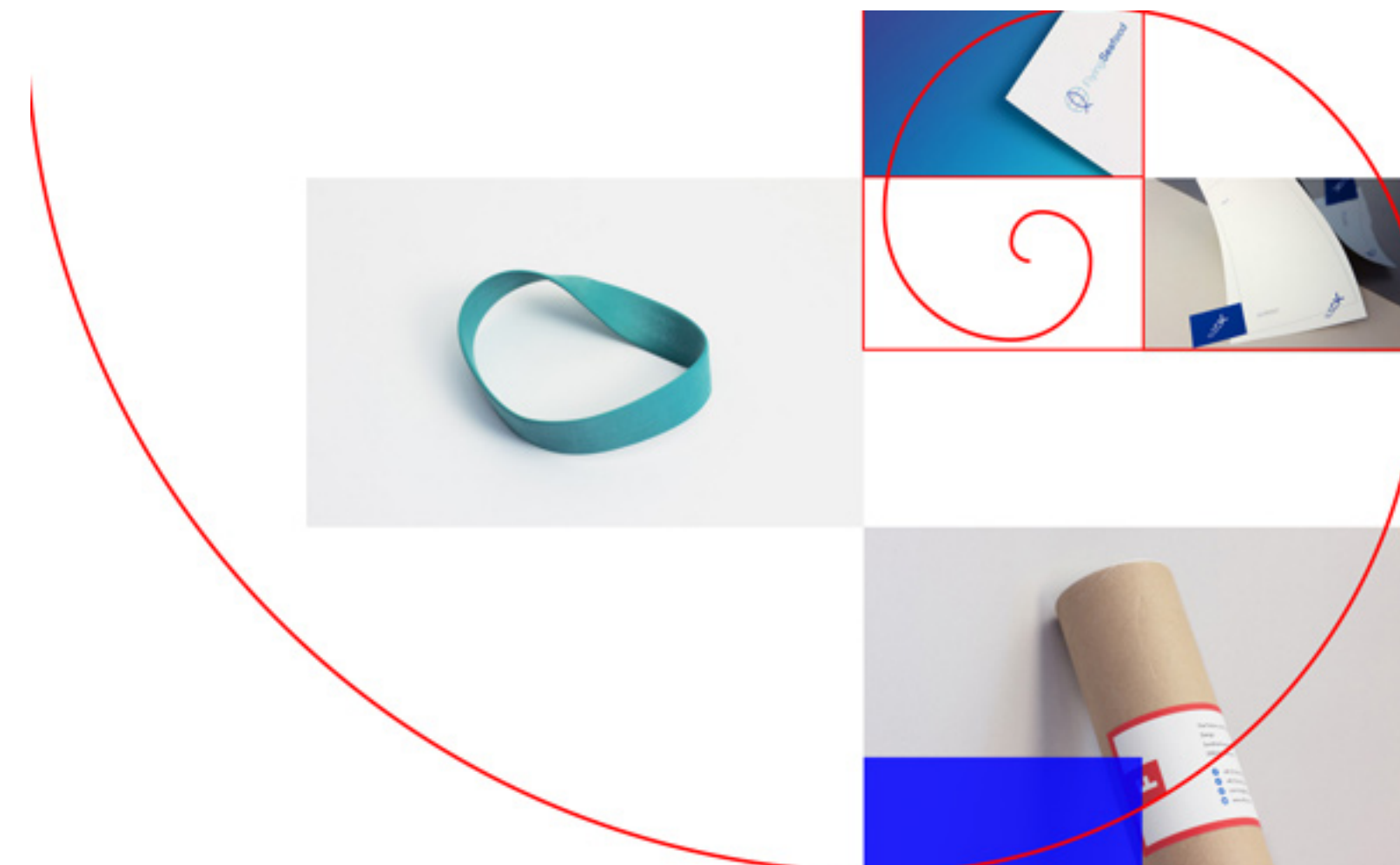
Y.

Feed Studio Work

Our team has a wide range of expertise offering hands-on development and design of objects, spaces, software, multimedia projects and printed matters.



We work with li
from cultural in
organizations, I
think of our clie
embrace their c



Code along

Versuche mit den Elementen, die du jetzt kennengelernt hast, deine Getränke darzustellen.

Es kann ganz einfach sein, von oben nach unten aufgelistet.

Du kannst auch das Layout auf der Seite vorher nachbauen und anpassen.

Fortgeschrittenere können natürlich auch flexbox oder grid ausprobieren, oder Filter einbauen.