



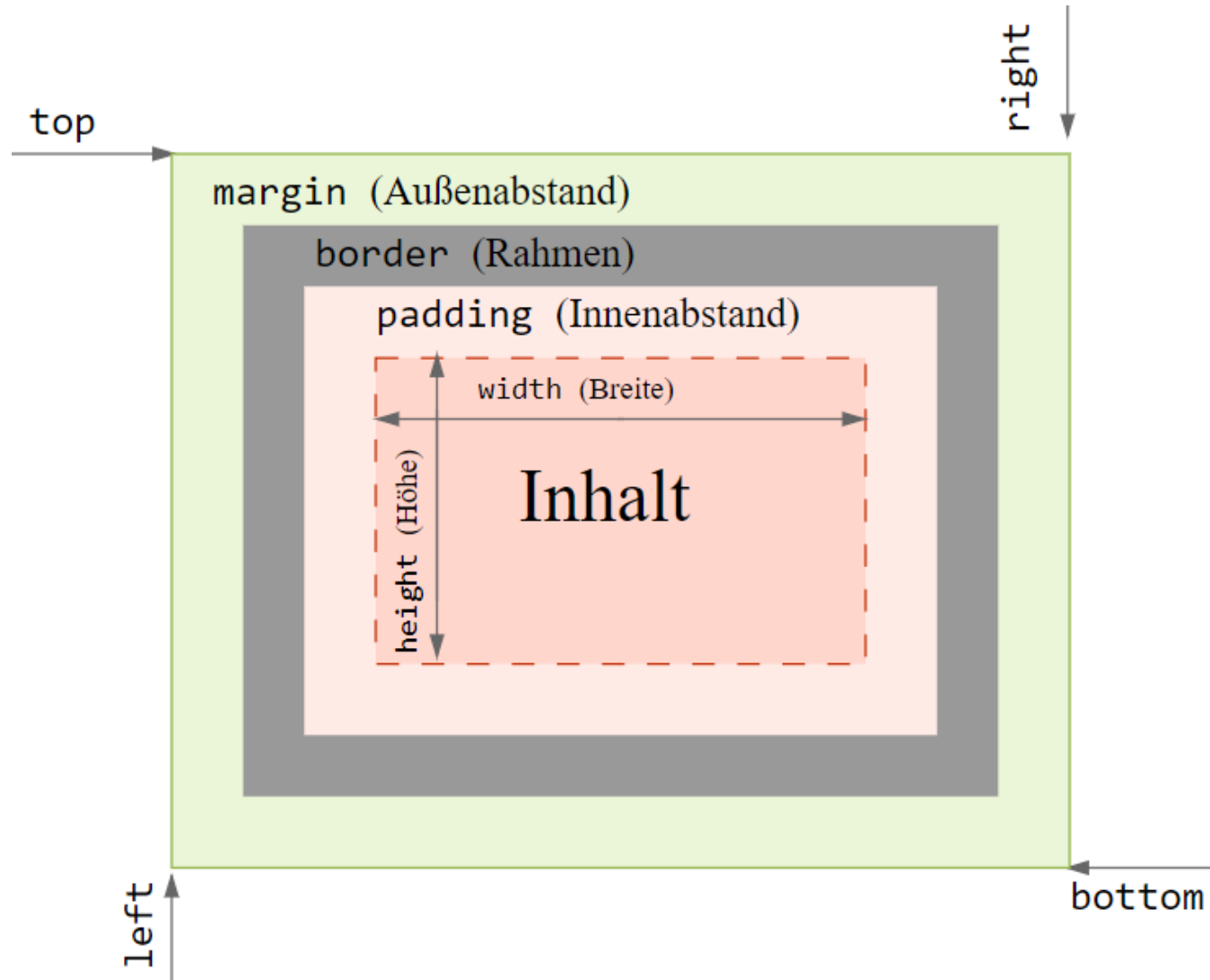
CSS Box-Modell



- Das Box-Modell beschreibt die rechteckigen Blöcke oder Boxen, die im Elementbaum erzeugt und auf dem Bildschirm dargestellt werden.
 - Es ist somit Grundlage jeden Layouts.
 - Die vom Browser erzeugte Box wird durch Höhe und Breite, aber auch durch Innen- und Außenabstände sowie Rahmen, festgelegt.
 - Abstände und Rahmen können für jede der vier Seiten einer Box einzeln festgelegt werden.
- Der Begriff „Box“ ist nicht mit dem Begriff „Element“ gleichzusetzen.
 - Nicht jedes Element erzeugt eine Box.
 - Boxen können auch erzeugt werden, wenn in einem Dokument kein Element als direktes Gegenstück existiert.



- Eine Box kann bestehen aus:
 - Dem Inhaltsbereich, also der Fläche, die durch Texte und Bilder oder Eigenschaften wie `width` und `height` vorgegeben wird,
 - einem Innenabstand (`padding`),
 - einem Rahmen (`border`) und
 - einem Außenabstand (`margin`).
- Wird eine Box positioniert, so beginnt
 - die linke Außenkante bei `left`,
 - die obere Außenkante bei `top`,
 - die rechte Außenkante bei `right` und
 - die untere Außenkante bei `bottom`.





- Ränder für die Box können per CSS festgelegt werden
 - padding – Innenabstand
 - border – Breite des Rahmens
 - margin – Außenabstand
- Die vier Seiten können einzeln adressiert werden
 - -top – oben
 - -right – rechts
 - -bottom – unten
 - -left – links



- Statt die vier Seiten der Box explizit anzusprechen, kann man auch eine Kurzform verwenden:
 - die Angaben werden beginnend bei top im Uhrzeigersinn notiert
 - drei Angaben: für top, left/right und bottom

```
/* Shorthand: */
```

```
padding: 5px 10px 15px 25px;
```

```
/* Ausführlich: */
```

```
padding-top: 5px;
```

```
padding-right: 10px;
```

```
padding-bottom: 15px;
```

```
padding-left: 25px;
```

```
/* Die Bedeutung ist aber gleich! */
```



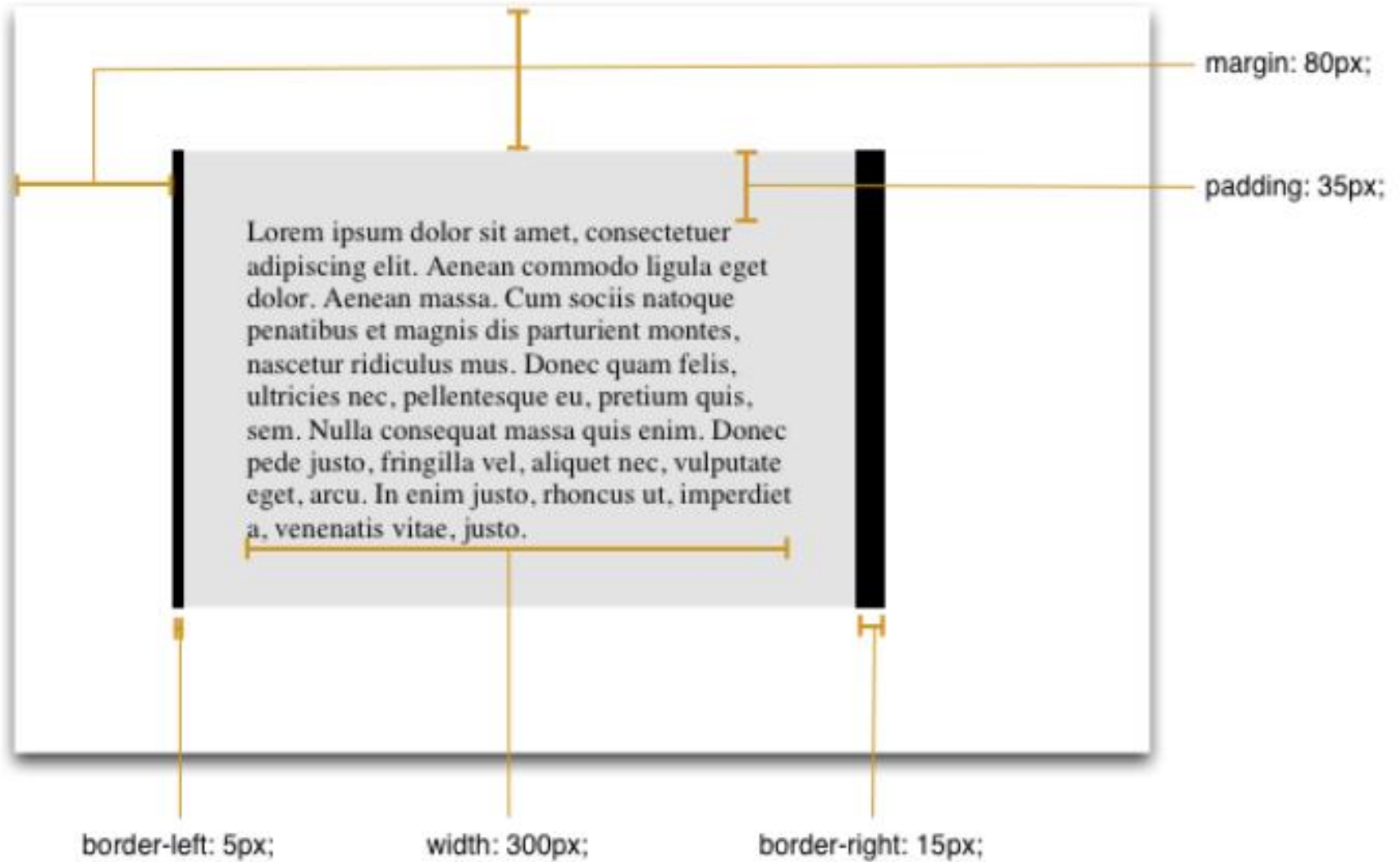
- Für Anfänger irritierend ist, dass mehrere Boxen nicht nebeneinander dargestellt werden, obwohl deren Gesamtbreite genau der Breite des Eltern-Elements entspricht.
- Dies liegt daran, dass alle Bestandteile einer Box addiert werden und nicht nur die Breiten $\text{SUM}(\text{width})$ und Höhen $\text{SUM}(\text{height})$:
 - Die Gesamtbreite errechnet sich statt dessen aus:
margin-left + border-left + padding-left +
width +
padding-right + border-right + margin-right
 - Die Gesamthöhe einer Box wird ebenso errechnet aus:
margin-top + border-top + padding-top +
height +
padding-bottom + border-bottom + margin-bottom



```
<head>
  <style>
    .boxed{
      width: 300px;
      padding: 35px;
      margin: 80px;
      border: #353d2d;
      border-right: 15px solid;
      border-left: 5px solid;
      background-color: #e3e3e3;
    }
  </style>
</head>
<body>
<p class="boxed">Lorem ipsum...</p>
</body>
```




Ein Box-Beispiel als CSS-Klasse: box.html





- Angabe der gewünschten Größe
 - width – Breite
 - height – Höhe
- Angabe der maximalen und minimalen Größe
 - min-width, max-width – minimale und maximale Breite
 - min-height, max-height – minimale und maximale Höhe
- overflow – Was passiert, wenn der Inhalt nicht zur Größe passt?
 - visible – Inhalt ragt über den Rand hinaus!
 - hidden – Inhalt wird am Rand abgeschnitten!
 - scroll – Inhalt wird verborgen, es gibt aber Scrollbalken!
 - auto – Der Browser entscheidet, wo Scrollbalken benötigt werden!



Ein Overflow-Beispiel: overflow.html



```
<head>
  <style>
    p {
      font-family: Helvetica, Arial, Sans-Serif;
      font-size: 20px;

      /* --- Wichtig --- */
      overflow: visible;
      /* --- Wichtig --- */

      border: 5px solid;
      border-color: olive;
      width: 200px;
      max-height: 140px;
    }
  </style>
</head>
<body>
<p>Lorem ipsum...</p>
</body>
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean commodo ligula eget dolor. Aenean massa. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec quam felis, ultricies nec, pellentesque eu, pretium quis, sem. Nulla consequat massa quis enim.



Ein Overflow-Beispiel: overflow.html



Lorem ipsum dolor sit
amet, consectetur
adipiscing elit. Aenean
commodo ligula eget
dolor. Aenean massa.
Cum sociis natoque

```
<head>
  <style>
    p {
      font-family: Helvetica, Arial, Sans-Serif;
      font-size: 20px;

      /* --- Wichtig --- */
      overflow: hidden;
      /* --- Wichtig --- */

      border: 5px solid;
      border-color: olive;
      width: 200px;
      max-height: 140px;
    }
  </style>
</head>
<body>
<p>Lorem ipsum...</p>
</body>
```



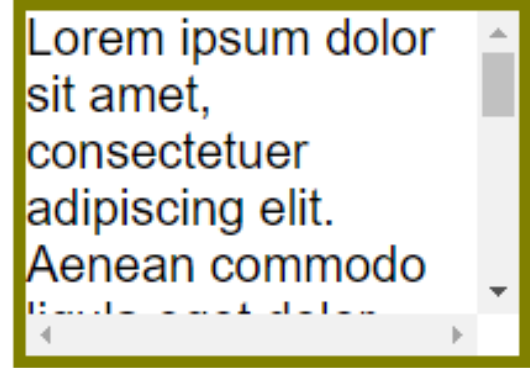
Ein Overflow-Beispiel: overflow.html



```
<head>
  <style>
    p {
      font-family: Helvetica, Arial, Sans-Serif;
      font-size: 20px;

      /* --- Wichtig --- */
      overflow: scroll;
      /* --- Wichtig --- */

      border: 5px solid;
      border-color: olive;
      width: 200px;
      max-height: 140px;
    }
  </style>
</head>
<body>
<p>Lorem ipsum...</p>
</body>
```





- display – legt fest, wie ein Element angezeigt wird
 - none – nicht anzeigen und auch keinen Platz dafür freihalten
 - inline – im laufenden Textfluss anzeigen
 - block – Block erzwingen, d.h. in neuer Zeile anzeigen
 - list-item – wie block, aber mit Aufzählungszeichen davor
 - run-in – kontextabhängig als block oder inline darstellen

- visibility – legt fest, ob ein Element angezeigt wird
 - hidden – nicht anzeigen. Platz wird aber freigehalten
 - visible – anzeigen, Default



```
<head>
  <style>
    .box {
      display: block;
      height: 100px;
      width: 100px;
      background-color: red;
    }
  </style>
</head>
<body>
  <span>Ich bin ein Text</span>
  <div class="box">Test</div>
  <span>Ich bin noch ein Text</span>
</body>
```

Ich bin ein Text
Test

Ich bin noch ein Text



```
<head>
  <style>
    .box {
      display: inline;
      height: 100px;
      width: 100px;
      background-color: red;
    }
  </style>
</head>
<body>
  <span>Ich bin ein Text</span>
  <div class="box">Test</div>
  <span>Ich bin noch ein Text</span>
</body>
```

Ich bin ein Text **Test** Ich bin noch ein Text



- border-color – Rahmenfarbe
 - Einschränkung auf eine Seite über border-top-color, border-left-color ...
- border-style – Rahmenstil
 - none – kein Rahmen
 - dotted – gepunktet
 - dashed – gestrichelt
 - solid – durchgezogen
 - double – doppelt durchgezogen
 - inset, outset – 3D-Effekt eingedrückt bzw. hervorstehend
 - Einschränkung auf eine Seite über border-top-style, border-left-style ...

