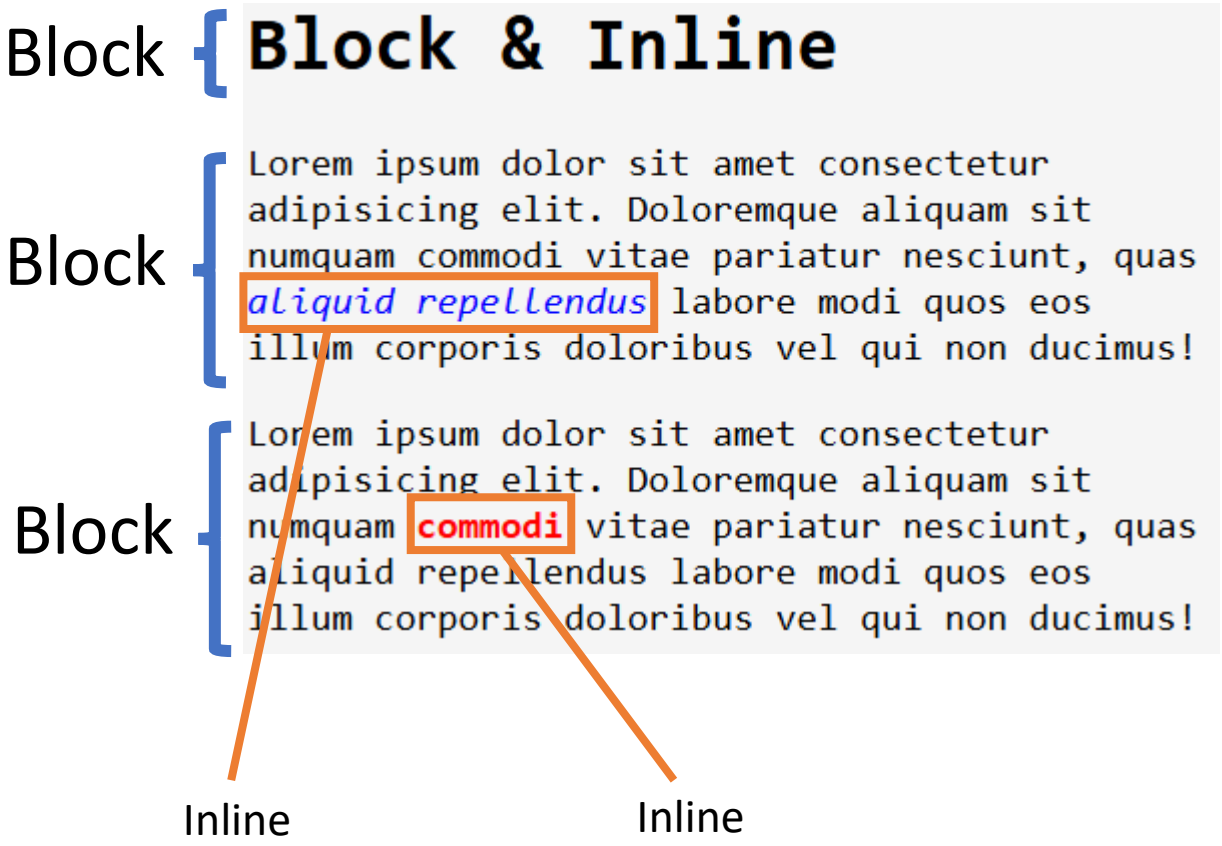
HTML 3

Block & Inline

Block und Inline

HTML-Elemente lassen sich grob in zwei Kategorien aufteilen:

- **Block:** Erzwingen einen Zeilenumbruch vor und nach ihnen, sie erzeugen also einen abgeschlossenen Block bzw. einen Absatz
- **Inline:** Im Unterschied dazu erzwingen Inline-Elemente *keinen* Zeilenumbruch, sie befinden sich im Textfluss und markieren Inhalte in einem Absatz / Block



Betonung / Hervorhebung

Um einzelne Begriffe oder Formulierungen im Text besonders zu betonen oder hervorzuheben, gibt es diverse Inline-Element. Die wichtigsten:

- `<em>` „Emphasis“, Betonung; wird standardmäßig *kursiviert*
- `<strong>` Starke Betonung; wird standardmäßig **gefettet**
- `<mark>` Textmarker; standardmäßig schwarz auf gelb

```
<p>
  Öffnen Sie <em>vorsichtig</em> die
  Abdeckhaube. <strong>Vorsicht!</strong>
  Es könnte heißer Dampf austreten.
  <mark>Tipp:</mark> Benutzen Sie ein
  sauberes Handtuch.
</p>
```

Öffnen Sie *vorsichtig* die Abdeckhaube.
Vorsicht! Es könnte heißer Dampf austreten.
Tipp: Benutzen Sie ein sauberes Handtuch.

Hoch- & tiefstellen / klein

Gelegentlich muss man einzelne Zeichen etwas höher oder tiefer als den übrigen Text stellen. Dazu gibt es zwei Tags:

- `<sup>` „superscript“, der markierte Bereich wird eine halbe Zeile höher gestellt
- `<sub>` „subscript“, der markierte Bereich wird eine halbe Zeile tiefer gestellt
- `<small>` Der Text wird auf 70% verkleinert

```
<p>
  Die chemische Formel für Wasser lautet
  H<sub>2</sub>O.
</p>
<p>
  Die Fläche eines Kreises berechnet
  sich wie folgt:
  A = &#960; &times; r<sup>2</sup><br>
  <small>
    <em>Anmerkung:</em> A = Fläche, r =
    Radius
  </small>
</p>
```

Die chemische Formel für Wasser lautet H_2O.

Die Fläche eines Kreises berechnet sich wie folgt: $A = \pi \times r^2$

Anmerkung: A = Fläche, r = Radius

Weitere Inline-Elemente

Es gibt etliche Inline-Elemente für die verschiedensten Fälle. Einige Beispiele:

`<cite>` Markierung von Zitaten, standardmäßig *kursiv*

`<code>` Markierung von Programmcode, standardmäßig `Courier`

`<del>` Für die Markierung gelöschter Passagen, standardmäßig ~~durchgestrichen~~

`<ins>` Hinzufügungen zu einem Text, standardmäßig <u>unterstrichen</u>

```
<p>
  Descartes' zentrale Erkenntnis: <cite>Cogito,
  ergo sum</cite>
</p>
<p>
  Mit <code>let</code> werden in JavaScript
  Variablen deklariert
</p>
<p>
  Änderungen in den AGB haben wir
  <del>durchgestrichen</del>, Hinzufügungen
  werden <ins>unterstrichen</ins>
</p>
```

Descartes' zentrale Erkenntnis:
Cogito, ergo sum

Mit `let` werden in JavaScript Variablen deklariert

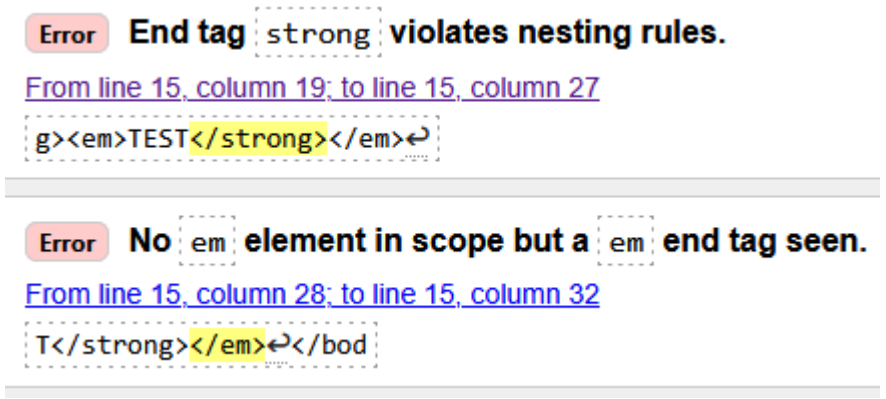
Änderungen in den AGB haben wir ~~durchgestrichen~~, Hinzufügungen werden <u>unterstrichen</u>

Verschachteln („Nesting“)

HTML-Elemente können verschachtelt werden („Nesting“). Dabei gilt es einige Regeln zu beachten:

- **Block-Elemente** können Inline-Elemente enthalten. Block-Elemente innerhalb von Block-Elementen sind erlaubt (z.B. `<li>` innerhalb von `<ul>`)
- **Inline-Elemente** stehen *innerhalb* von Block-Elementen und können Inline-Elemente enthalten, in der Regel aber *keine* Block-Elemente

Beim Verschachteln muss man die Reihenfolge der öffnenden / schließenden Tags beachten: `<strong><em>Test</em></strong>` ist korrekt, `<strong><em>Test</strong></em>` dagegen falsch!



1. Fehlermeldung: Die Nesting-Regeln wurden verletzt
Daher kommt es zur
2. Fehlermeldung: wg. Verletzung der Nesting-Regeln wird das öffnende `<em>` nicht erkannt, dem späteren schließenden `</em>` kann daher kein öffnendes Tag zugeordnet werden

Neutrale Elemente (Container)

`<div>` Neutrales **Blockelement**, kann beliebige Inhalte enthalten, auch weitere `<div>`. Erzeugt einen Absatz, hat ansonsten keine Eigenschaften wie z.B. Abstände. Wird zur Strukturierung benötigt, wird aber erst mit CSS und JavaScript interessant.

`<span>` Neutrales **Inline-Element**, kann weitere Inline-Elemente und insbesondere `<span>` enthalten. Wird benötigt, um mit CSS und JavaScript gezielt auf bestimmte Inhalte zugreifen zu können. Hat ansonsten keine sichtbaren Auswirkungen.

Weitere Strukturelemente

Mit HTML5 wurden zahlreiche weitere neutrale Blockelemente eingeführt, um Inhalte auf Webseiten besser zu strukturieren. Eine kleine Auswahl:

- `<header>` Kopfbereich, z.B. Logo
- `<nav>` Für Navigationselemente, meist Teil von `<header>`
- `<main>` Umfasst den Hauptbereich der Seite
- `<article>` Abgeschlossene Beiträge (etwa Blogpostings)
- `<section>` Fasst mehrere Abschnitte / Beiträge zusammen
- `<footer>` Fußzeilen
- `<aside>` Randleisten (etwa Werbung)

Die Seite wird in header, main und footer strukturiert

Im header steht eine h1, gefolgt von einem nav, der Navigation (= Menü)

Im main steht der eigentliche Inhalt der Seite, aufgeteilt in

- sections (hier aus Platzgründen nur eine), die wiederum
- articles enthalten

Sowohl section als auch artikel haben Überschriften (h2 bzw. h3)

Nach main folgt abschließend die Fußzeile im footer. Hier stehen üblicherweise Copyright o.ä.

```
<body>
  <header>
    <h1>BlaBlaBla</h1>
    <nav>
      <!-- Das Menü -->
    </nav>
  </header>
  <main>
    <section>
      <h2>Überschrift</h2>
      <article>
        <h3>Überschrift</h3>
        <p>Irgendwas</p>
      </article>
      <article>
        <h3>Überschrift</h3>
        <p>Irgendwas</p>
      </article>
    </section>
  </main>
  <footer>
    <p>Alle Inhalte ©2026</p>
  </footer>
</body>
```