

CSS-Grid bietet die Möglichkeit ein Gestaltungsraster zu schaffen und die Positionierung der einzelnen Elemente vollständig durch CSS zu steuern. Damit wird eine klare Trennung des Gestaltungsrasters von der HTML-Textstruktur erreicht. Zusammen mit Media Queries erlaubt es den Aufbau flexibler Layouts.

Die Layout Gestaltung erfolgt in zwei Schritten:

1. Im ersten Schritt wird festgelegt, für welchen Bereich eines HTML-Dokumentes (container) CSS-Grids angewendet werden sollen und wie das Element in Rasterlinien eingeteilt wird.
2. Im zweiten Schritt werden die Elemente (items) innerhalb des Rasters positioniert (explizit für jedes Element).

```

① container {
②   display: grid ;
③   grid-template-columns: <track-size> ... ;
④   grid-template-rows: <track-size> ... ;
⑤   [ grid-gap: <line-size>; ]
⑥   [ grid-column-gap: <line-size>; ]
⑦   [ grid-row-gap: <line-size>; ]
⑧   [ justify-items: start | end | center | stretch; ]
⑨   [ align-items: start | end | center | stretch; ]
⑩
⑪ }
```

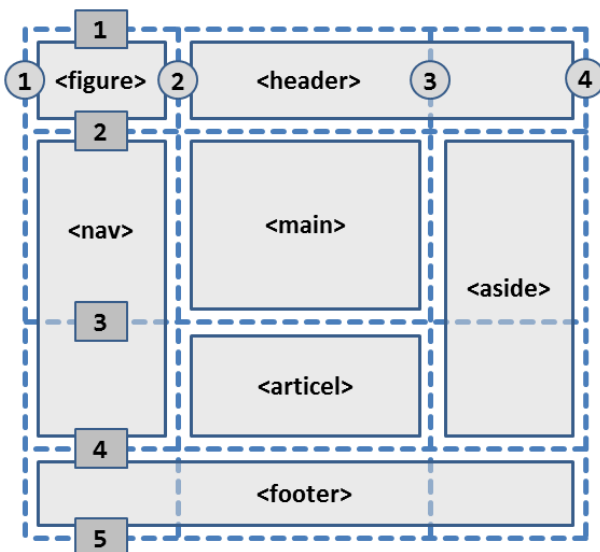
Eigenschaften des GRID-Containers

```

① item {
②   grid-column: <start-line> / <end-line> | <start-line> / span <value>;
③   grid-row: <start-line> / <end-line> | <start-line> / span <value>;
④   [ justify-self: start | end | center | stretch; ]
⑤   [ align-self: start | end | center | stretch; ]
⑥ }
```

Eigenschaften der GRID-Elemente

Im abgebildeten Beispiel wird ein Grid mit 3 Spalten und 4 Zeilen im Container `<body>` definiert.



```

body { display: grid;
      grid-template-columns: 100px 1fr 20%;/
      grid-template-rows:
      80px calc(100% - 310px) 150px 80px ;
      grid-gap:10px; }
figure { grid-column: 1; grid-row: 1; }
header { grid-column: 2/4; grid-row: 1; }
nav { grid-column: 1; grid-row: 2/4;}
main { grid-column: 2; grid-row: 2; }
aside { grid-column: 3; grid-row: 2/4;}
article{ grid-column: 2; grid-row: 3; }
footer { grid-column: 1/4; grid-row: 4; }
```

CSS-Formatierung

HTML-Container

```

<body> <figure> figure </figure>
      <header> header </header>
      <nav>   nav   </nav>
      <main>  main  </main>
      <aside> aside </aside>
      <article>article</article>
      <footer> footer </footer>    </body>
```

Die Spezifikation wurde mehrfach geändert. Im neuen Working-Draft definiert man damit nicht eigentlich Spalten, sondern vertikale **Rasterlinien**. Die **aktuelle Version** (Stand 3/2017) funktioniert zum Beispiel in Chrome ab 57, und in Firefox ab Version 52.