

Distribueret Programmering

Lektion 03: CSS

- ❖ CSS er en forkortelse for Cascading Style Sheets
- ❖ CSS definerer en websides layout
- ❖ Et layout er bl.a.:
 - ❖ farver, skrifttyper, baggrund, kanter og placering



CSS syntaks

- ♣ Syntaks for CSS:

selector, ... { property: value; ...}

- ♣ Eksempel:

```
p { color: blue; }
```

```
h1, h2, h3 { /* large heading */
```

```
color: white;
```

```
background-color: gray;
```

```
text-align: center;
```

```
}
```

```

<!-- eksempel.html -->
<!DOCTYPE html>
<html>
  <head>
    <title>Example</title>
    <link rel="stylesheet" href="example.css" />
  </head>
  <body>
    <h1>Lorem ipsum dolor</h1>
    <h2>Nullam gravida odio quis</h2>
    <h3>Aliquam at massa</h3>
    <p>Quisque venenatis nunc a risus
      varius condimentum. Nullam augue
      mauris, rhoncus eu, porta ac,
      pulvinar quis, quam. Nulla consequat,
      sapien sed sollicitudin aliquet,
      nisi velit malesuada tellus, quis
      viverra metus mi vel nunc.</p>
  </body>
</html>

```

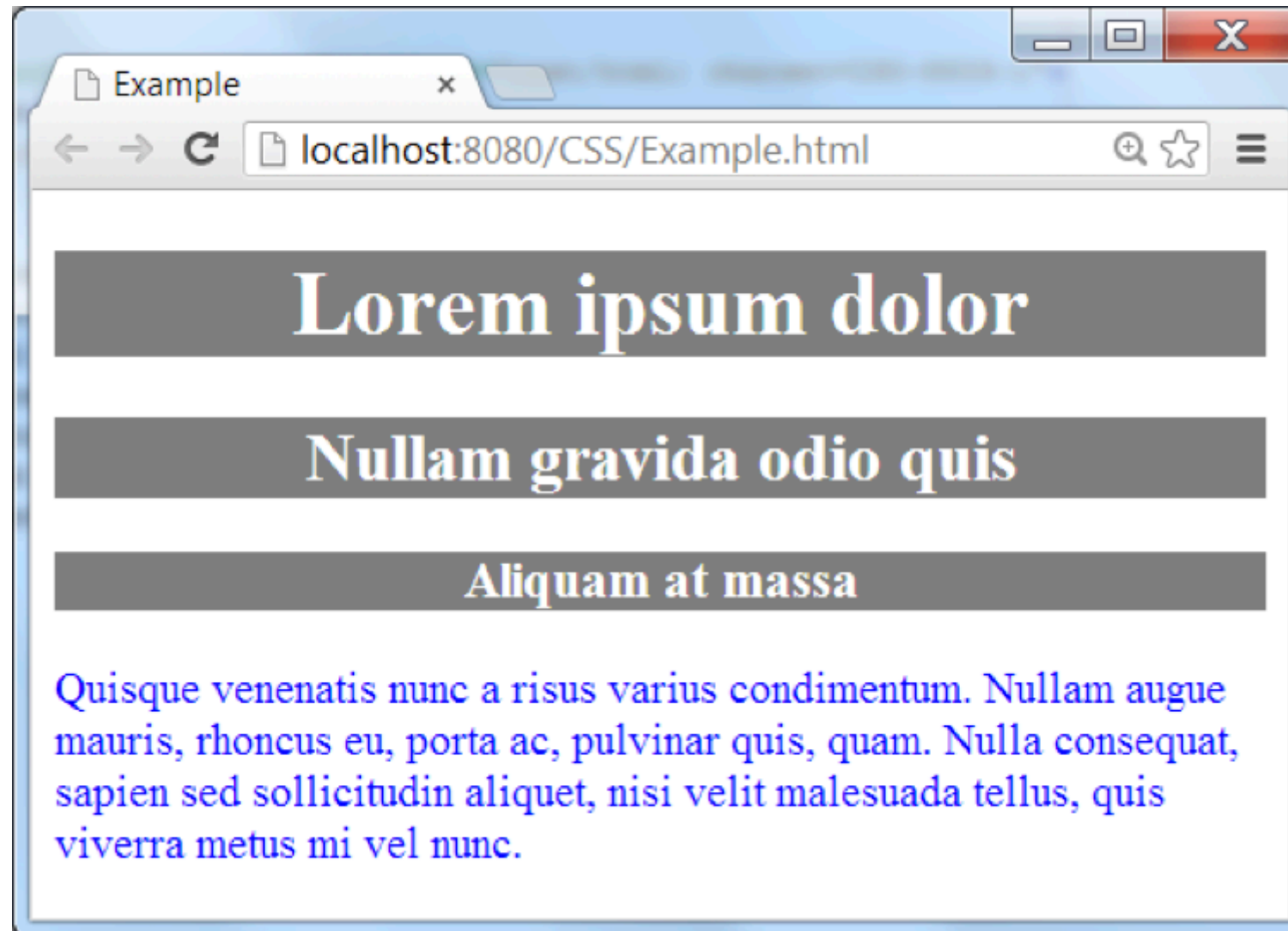
```

p {
  color: blue;
}

h1,h2,h3 { /* store overskrifter */
  color: white;
  background-color: gray;
  text-align: center;
}

body {
  background-image: url("html5.png");
}

```



CSS kan placeres 3 steder

External CSS i en separat fil specificeret i <head>

```
<link rel="stylesheet" href="filename.css"/
```

Internal CSS i et <style> element i <head>

CSS

```
<style >  
    p { color: blue; }  
</style>
```

Inline CSS som en attribut i et HTML element

```
<p style="color: blue;">
```

CSS i en separat fil

CSS *bør* placeres i en separat fil

Det har flg. fordele:

- ♣ det samme layout kan genbruges på tværs af websider
- ♣ og ændringer i layout skal kun foretages et sted

Selectors

❖ Vælg alle elementer:

`* {...}`

❖ Vælg alle elementer med et givet navn:

`p {...}`

❖ Vælg alle elementer med en af navne:

`h1, h2, h3 {...}`

❖ Vælg alle elementer med class="special":

`**.**special {...}`

❖ Vælg elementet med id="content":

`#content {...}`

❖ Vælg alle elementer med et givet navn og med class="special":

`p**.**special {...}`

Cascade

- ❖ Cascade i Cascading Style Sheets (CSS) er navnet på den algoritme, der definerer, hvordan property værdier fra forskellige kilder kombineres:
- ❖ Hvis et element får forskellige værdier for samme style (ie. CSS property), så vinder den med højest prioritet
- ❖ En style defineret tæt på et element har højere prioritet:
HTML > inline > internal > external > browser
- ❖ En mere specifik selector har højere prioritet:
id > class > combination > single
- ❖ En style inherited fra et omgivende element (fx color, font og text-align) har lavere prioritet
- ❖ Hvis flere har samme prioritet vinder den sidstnævnte

```

<!-- cascade.html -->
<!DOCTYPE html>
<html>
  <head>
    <title>Cascade</title>
    <link rel="stylesheet" href="cascade.css" />
  </head>
  <body>
    <h3>Lorem ipsum dolor</h3>
    <h3 class="special">Nullam gravida</h3>
    <h3 id="unique" class="special">
      Aliquam at massa</h3>
    <p>Quisque venenatis nunc a risus
      varius condimentum.</p>
    Nullam augue mauris, rhoncus eu,
    porta ac, pulvinar quis, quam.
  </body>
</html>

```

```

/* cascade.css */
body {color: purple;}

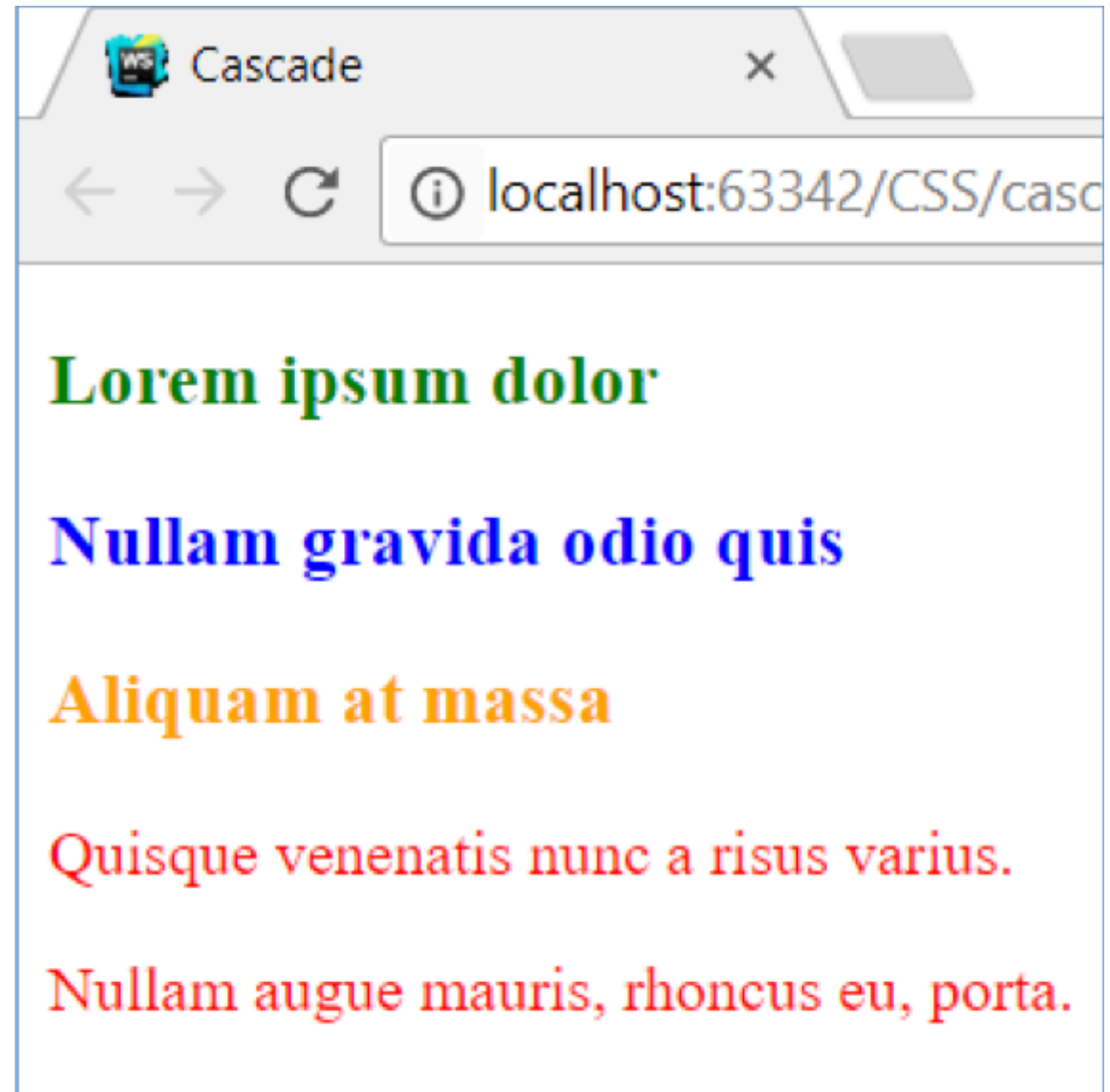
h3 {color: navy;}

.special {color: red;}

#unique {color: green;}

p {color: blue;}

```



Selectors - fortsat

❖ Vælg interne elementer:

`div p {...}`

❖ Vælg børne elementer:

`div > p {...}`

❖ Vælg søskende elementer:

`div + p {...}`

❖ Vælg ubesøgt links:

`a:visited {...}`

❖ Vælg elementet under markøren:

`p:hover {...}`

❖ Vælg hver andet barn:

`tr:nth-child(even) {...}`

```

<!DOCTYPE html>
<html>
  <head>
    <title>Selectors</title>
    <link rel="stylesheet"
          href="selectors.css" />
  </head>
  <body>
    <div>
      <header>
        <p>A</p>
      </header>
      <div>
        <p>B</p>
      </div>
    </div>
    <p>C</p>
    <a href="#">Link</a>
  </body>
</html>

```

```

div p {color: #000;}

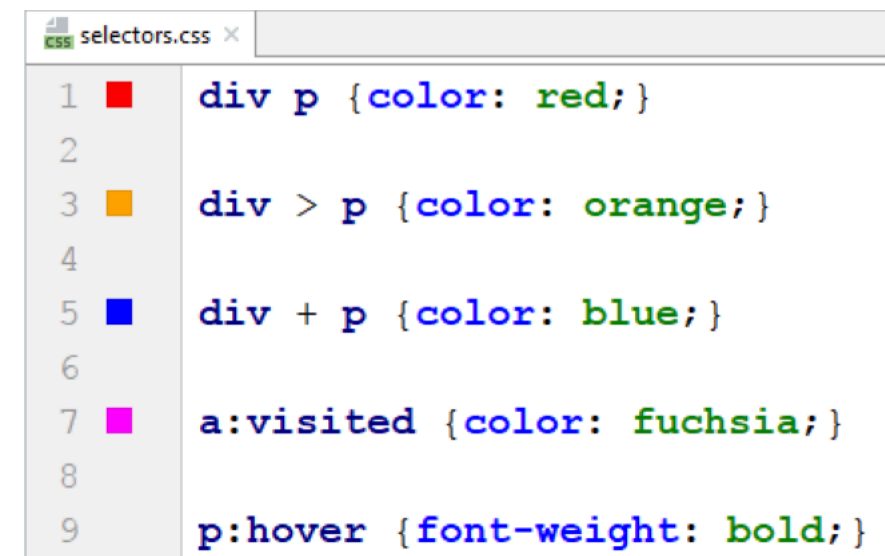
div > p {color: orange;}

div + p {color: blue;}

a:visited {color: fuchsia;}

p:hover {font-weight: bold;}

```



Måleenheder

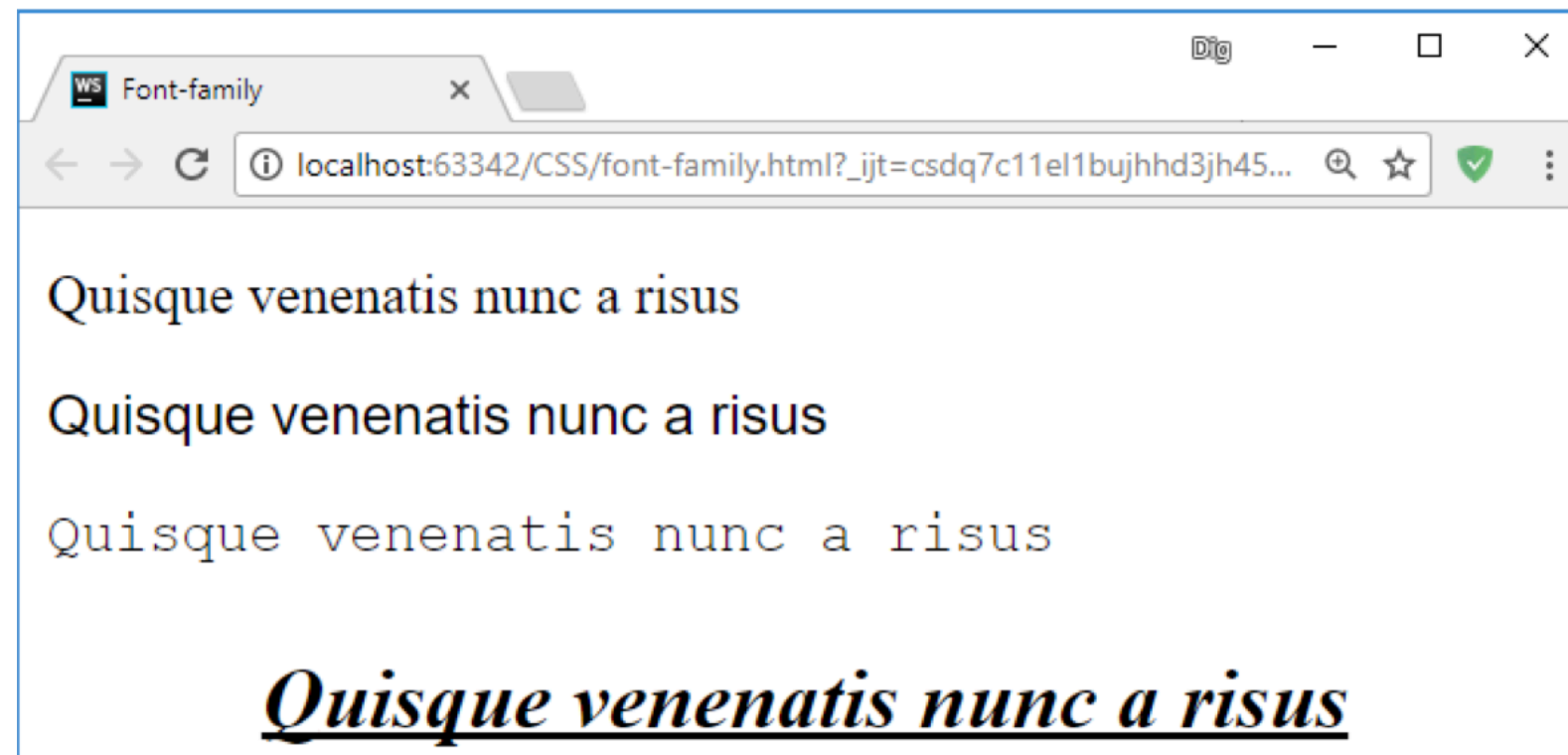
- ❖ Absolutte måleenheder:
cm mm
- ❖ Relative måleenheder:
px em (aktuel skriftstørrelse) rem (<html>'s skriftstørrelse)
- ❖ En procentvis værdi er relative til den tilsvarende værdi i det omgivende element: %
- ❖ Eksempler:
`table {width: 400px;}`
`img {width: 50%;}`

Farver

- ❖ Farver kan angives med et navn eller en RGB-værdi:
red green blue white ...
#FF0000 #Foo
rgb(255, 0, 0)
rgb(100%, 0, 0)
- ❖ RGBA farver udbygger RGB farver med en alfa-værdi - der angiver opacity (gennemsigtighed)
- ❖ En alfa-værdi er et tal mellem 0.0 (gennemsigtig) og 1.0 (uigennemsigtig)
- ❖ RGBA-værdier:
rgba(255, 0, 0, 0.5)
#FF000077 #F007 (CSS Color Module Level 4)

Skrifttype familier

- ❖ En `font-family` property angiver flere skrifttyper i prioriteret rækkefølge
- ❖ Den bruges til at vælge så ensartede skrifttyper som muligt – på tværs af browsere og operativsystemer




```

<!-- font-family.html -->
<!DOCTYPE html>
<html>
  <head>
    <title>Font-family</title>
    <link rel="stylesheet" href="font-family.css" />
  </head>
  <body>
    <div id="page">
      <p id="serif">Quisque venenatis
        nunc a risus</p>
      <p id="sans-serif">Quisque venenatis
        nunc a risus</p>
      <p id="courier">Quisque venenatis
        nunc a risus</p>
      <p id="default">Quisque venenatis
        nunc a risus</p>
    </div>
  </body>
</html>

```

```

/* font-falily.css */
#serif {
  font-family: "Times New Roman", Times, serif;
}

#sans-serif {
  font-family: Arial, Helvetica, sans-serif;
}

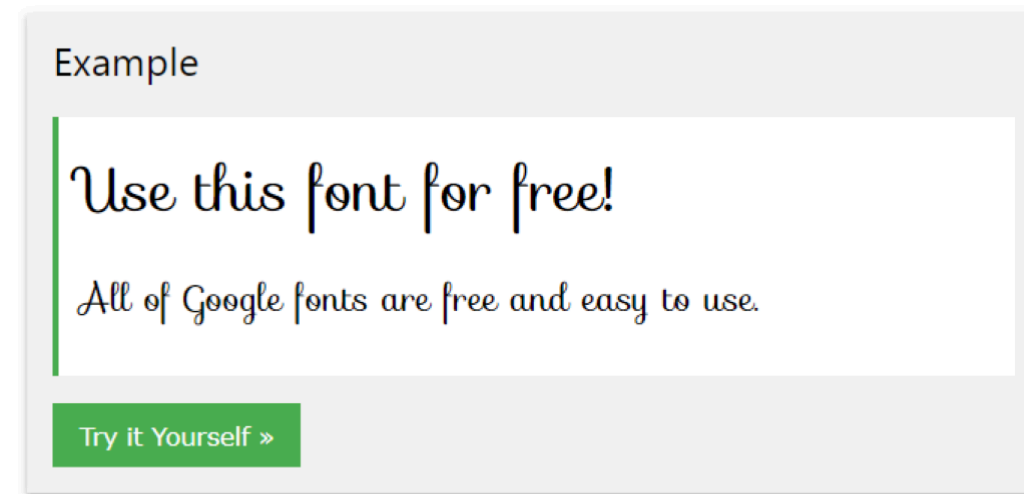
#courier {
  font-family: "Courier New", Courier, monospace;
}

#default {
  font-size: 1.5em;
  font-style: italic;
  text-decoration: underline;
  text-align: center;
}

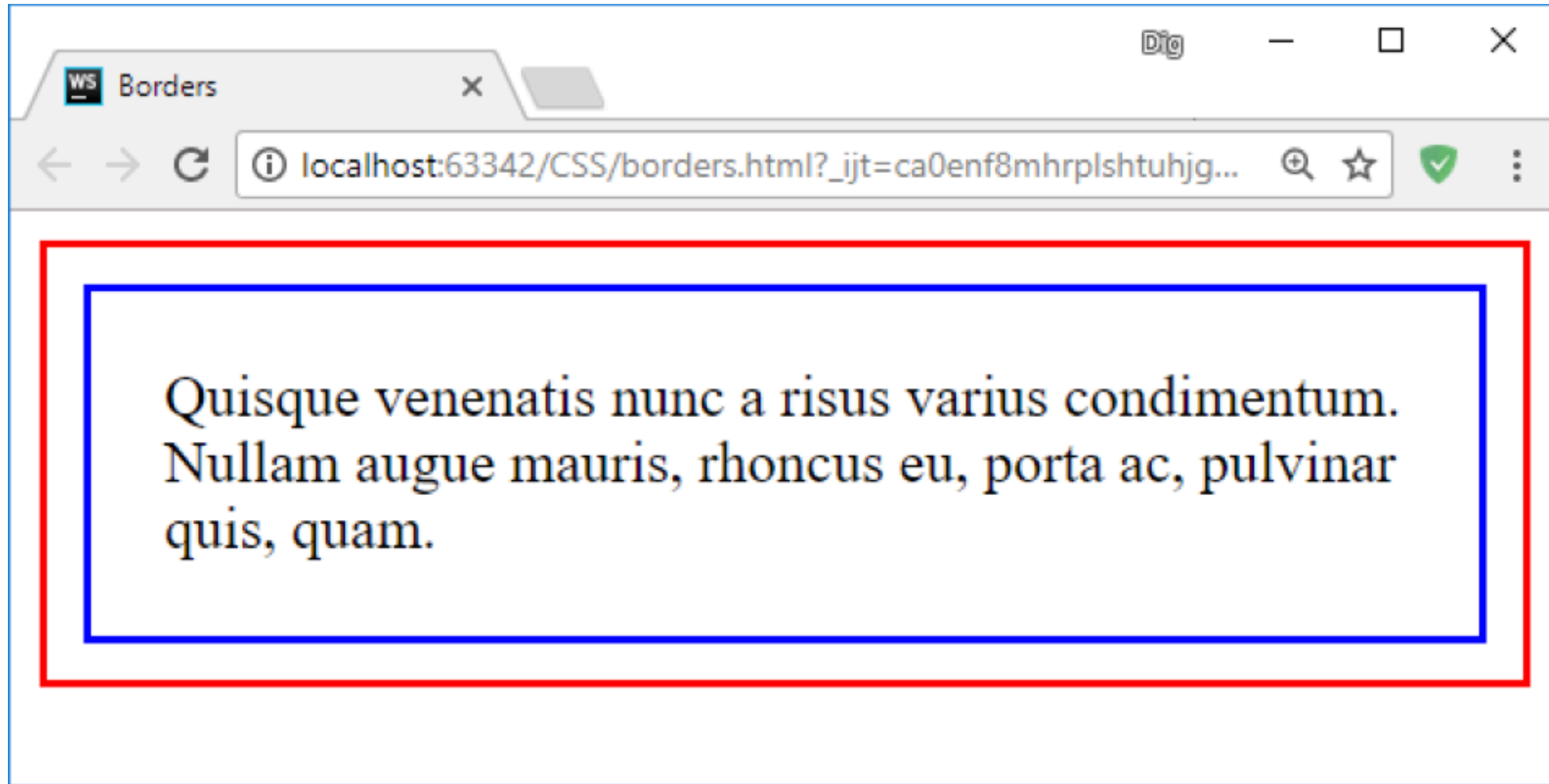
```

Online skrifttyper

- ❖ Et alternativ til skrifttype familier er at bruge en **online font service**
- ❖ Google Fonts tilbyder fx 884 forskellige skrifttyper



Kanter & luft



```
<!-- borders.html -->
<!DOCTYPE html>
<html>
  <head>
    <title>Borders</title>
    <link rel="stylesheet"
          type="text/css"
          href="borders.css" />
  </head>
  <body>
    <div>
      Quisque venenatis nunc
      a risus varius condimentum.
      Nullam augue mauris,
      rhoncus eu, porta ac,
      pulvinar quis, quam.
    </div>
  </body>
</html>
```

```
/* borders.css */
body {
  padding: 0;
  border: 2px solid red;
}

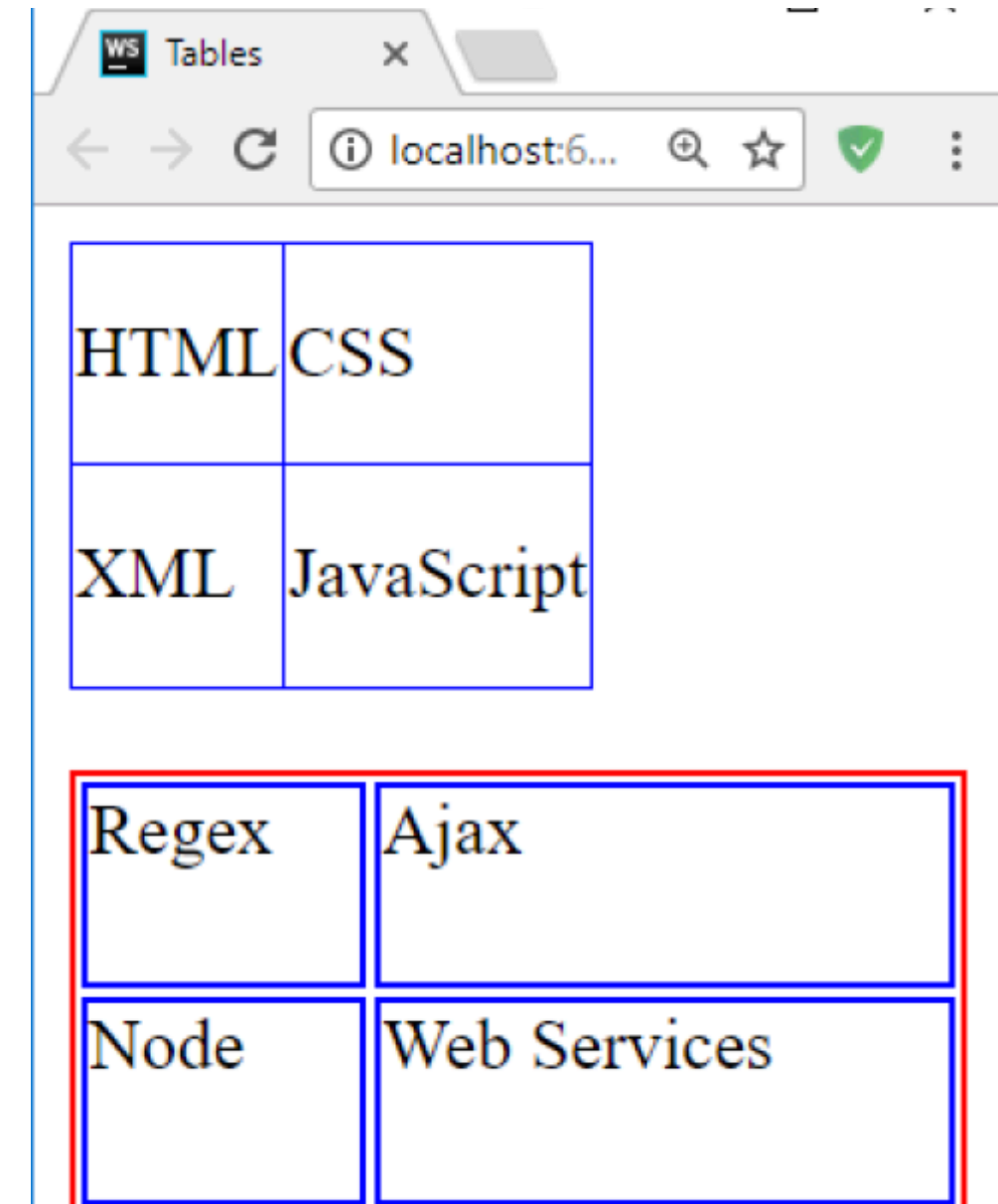
div {
  border: 2px solid blue;
  margin: 10px;
  padding: 20px;
}
```

Baggrund og billeder

- ✦ Baggrundsfarve:
background-color: #111
- ✦ Baggrundsbillede:
background-image: url("duke.gif")
- ✦ Billed gentagelse
background-repeat: no-repeat
default værdi: repeat
andre værdier: repeat-x repeat-y

Tabeller

- ❖ Layout:
table-layout: fixed
default værdi: auto
- ❖ Border:
border-collapse: collapse
default værdi: separate
- ❖ Andre attributter:
padding
border



HTML	CSS
XML	JavaScript

Regex	Ajax
Node	Web Services

```

<!DOCTYPE html>
<html>
  <head>
    <title>Tables</title>
    <link rel="stylesheet" href="tables.css" />
  </head>
  <body>

    <table class="coll">
      <tr>
        <td>HTML</td>
        <td>CSS</td>
      </tr>
      <tr>
        <td>XML</td>
        <td>JavaScript</td>
      </tr>
    </table>

    <br />

    <table class="sep">
      <tr>
        <td>Regex</td>
        <td>Ajax</td>
      </tr>
      <tr>
        <td>Node</td>
        <td>Web Services</td>
      </tr>
    </table>
    <table class="coll">
      <tr>
        <td>X</td>
        <td>0</td>
      </tr>
      <tr>
        <td>0</td>
        <td> </td>
      </tr>
    </table>

  </body>
</html>

```

```

table.coll {
  border-collapse: collapse;
  height: 100px;
}

table.sep {
  width: 200px;
  height: 100px;
}

table.sep td{
  vertical-align: top;
}

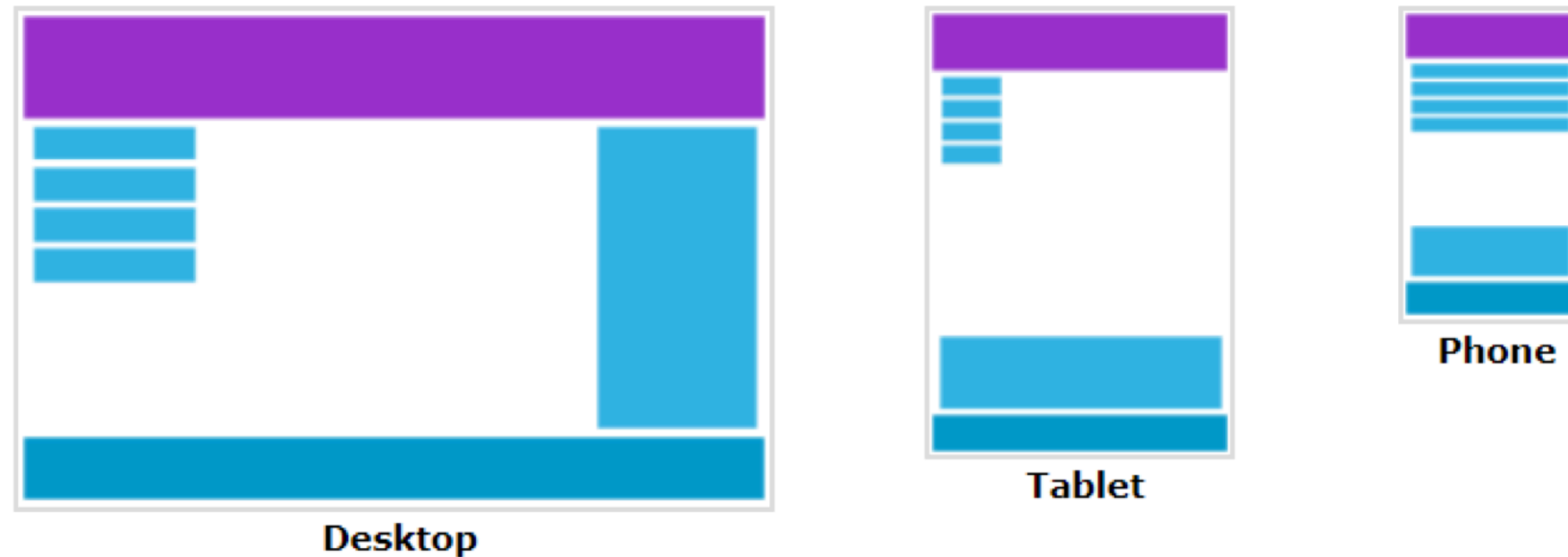
table {
  border: 1px solid red;
}

td {
  border: 1px solid blue;
}

```

Responsive Web Design

- ♣ Responsive webdesign får en webside til at se godt ud på alle størrelser af enheder
- ♣ Tilpasser automatisk websiden til størrelsen på browserens vindue
- ♣ Responsive web design anvender kun HTML og CSS – ikke JavaScript



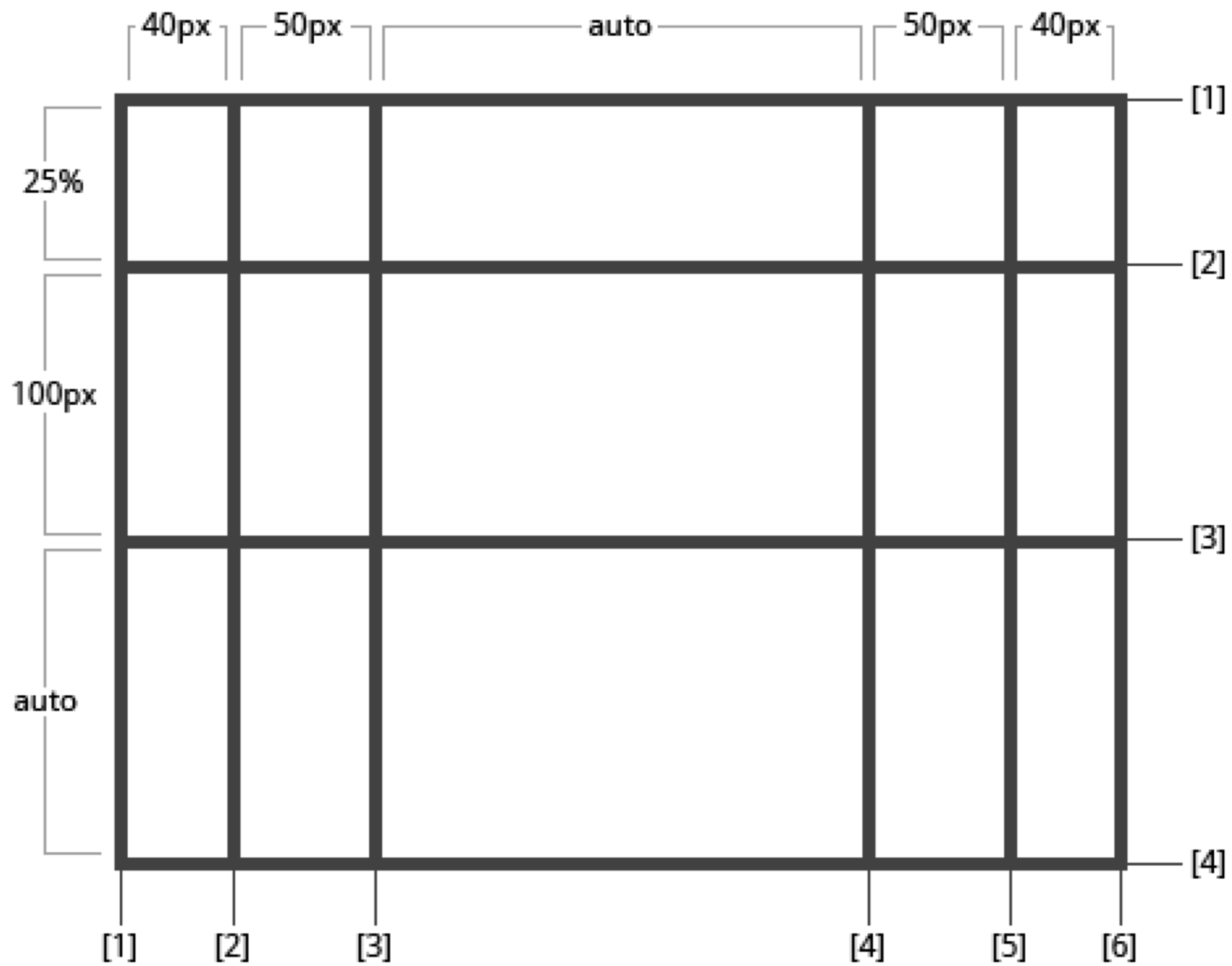
Webside design med grid

Angiv typen af grid

```
.container {  
  display: grid | inline-grid | subgrid;  
}
```

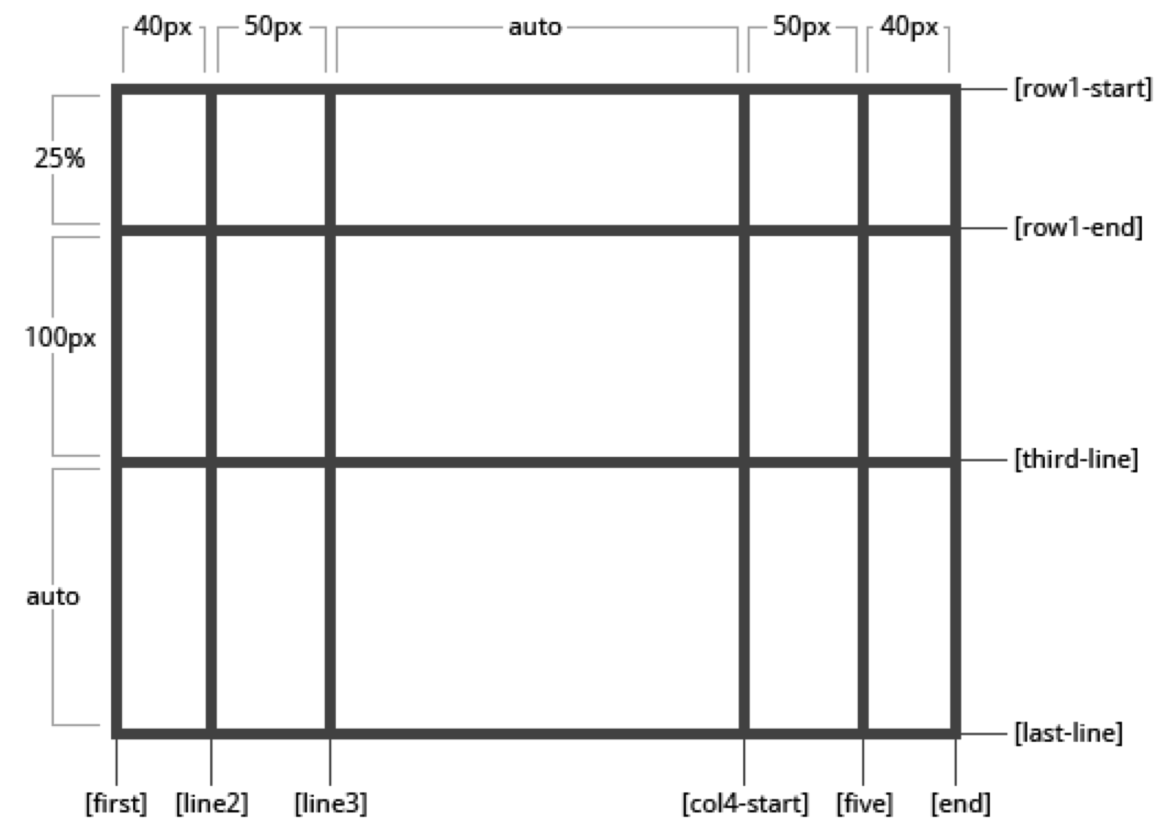
Design grid

```
.container {  
  grid-template-columns: 40px 50px auto 50px 40px;  
  grid-template-rows: 25% 100px auto;  
}
```



Det er også muligt at navngive linjerne så det er nemmere at designe

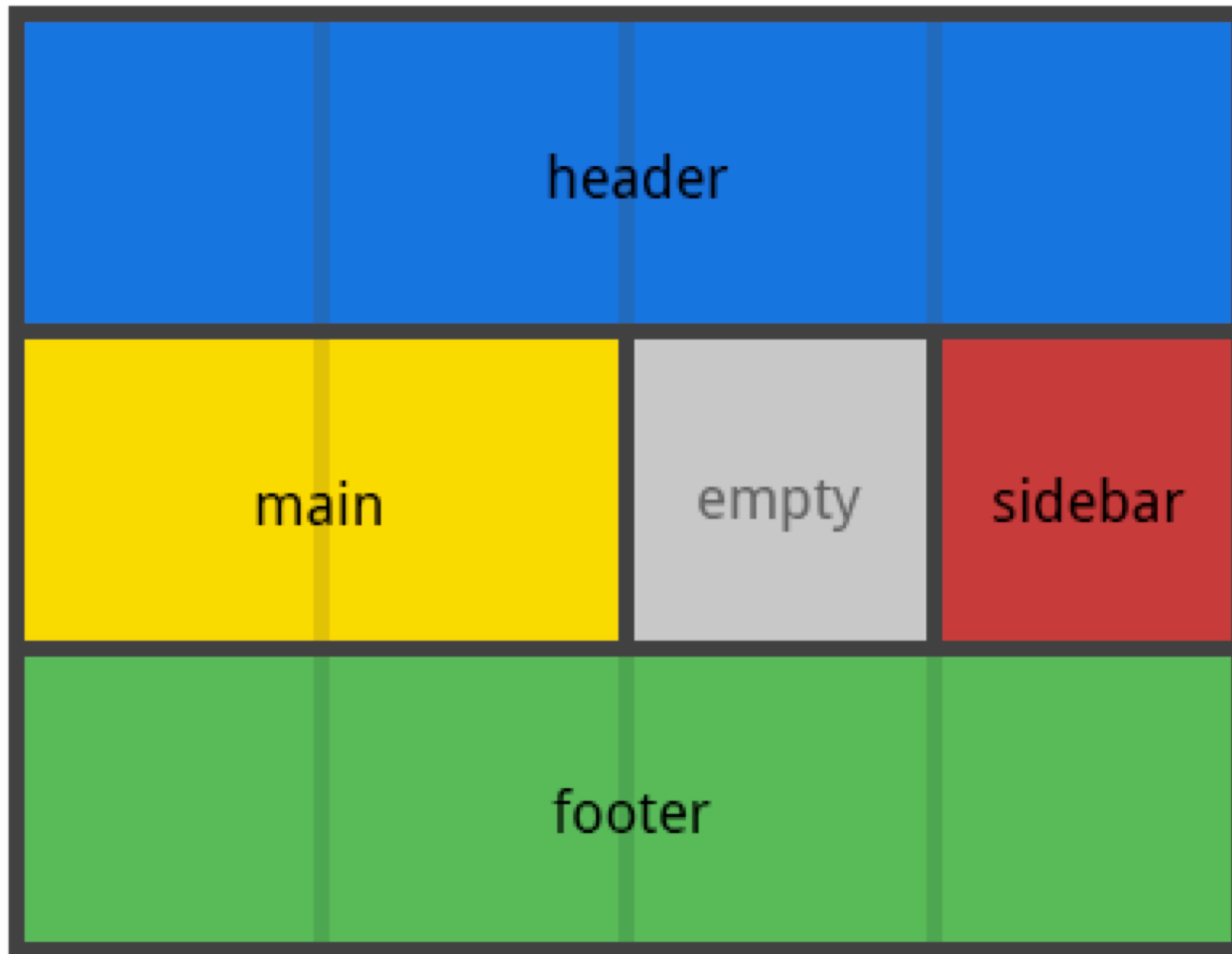
```
.container {  
  grid-template-columns: [first] 40px [line2] 50px [line3] auto [col4-start] 50px [five] 40px [end];  
  grid-template-rows: [row1-start] 25% [row1-end] 100px [third-line] auto [last-line];  
}
```



```
.container {  
  grid-template-columns: repeat(3, 20px) 5%;  
}
```

```
.container {  
  grid-template-columns: 1fr 1fr 1fr;  
}
```

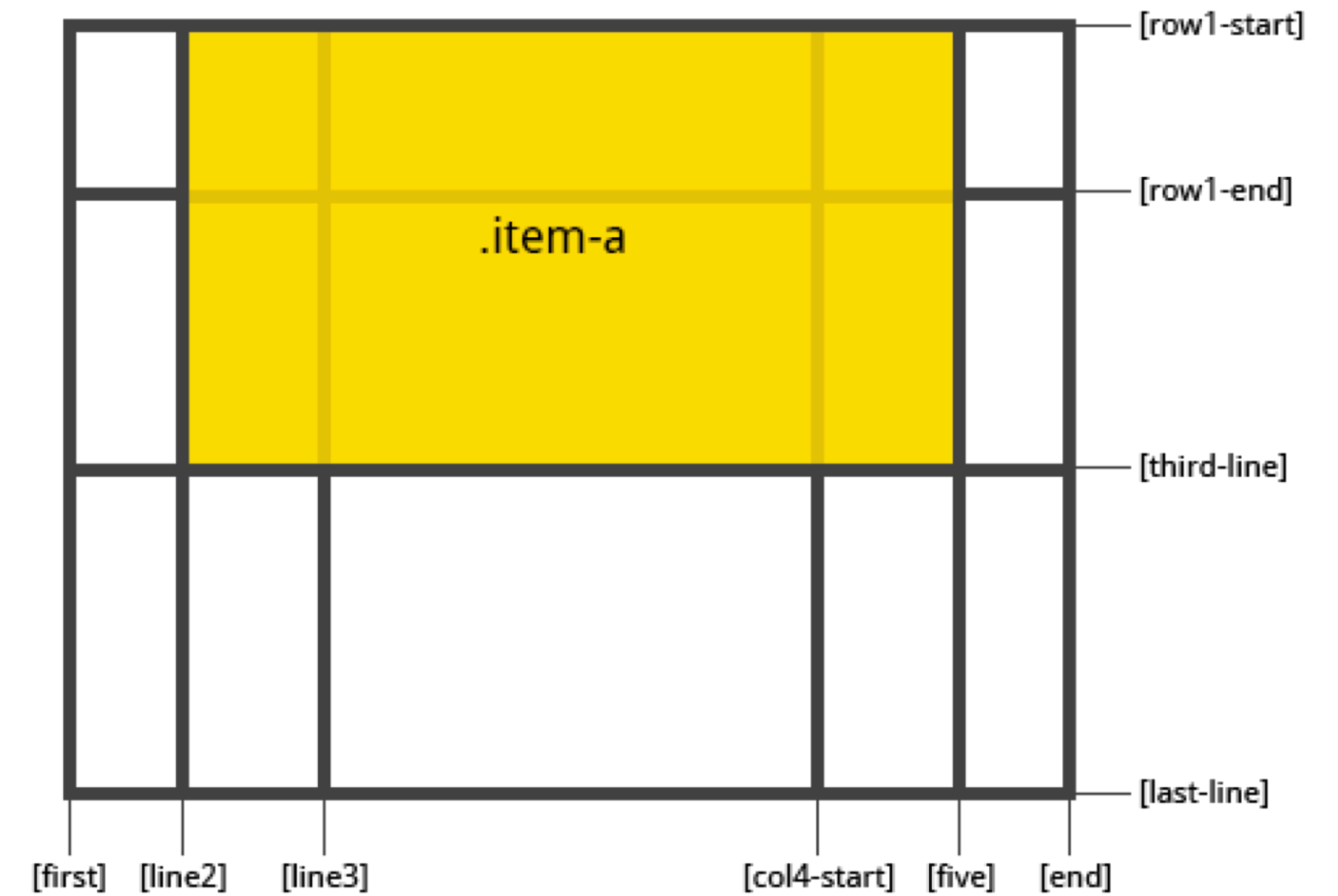
```
.container {  
  grid-template-columns: 1fr 50px 1fr;  
}
```



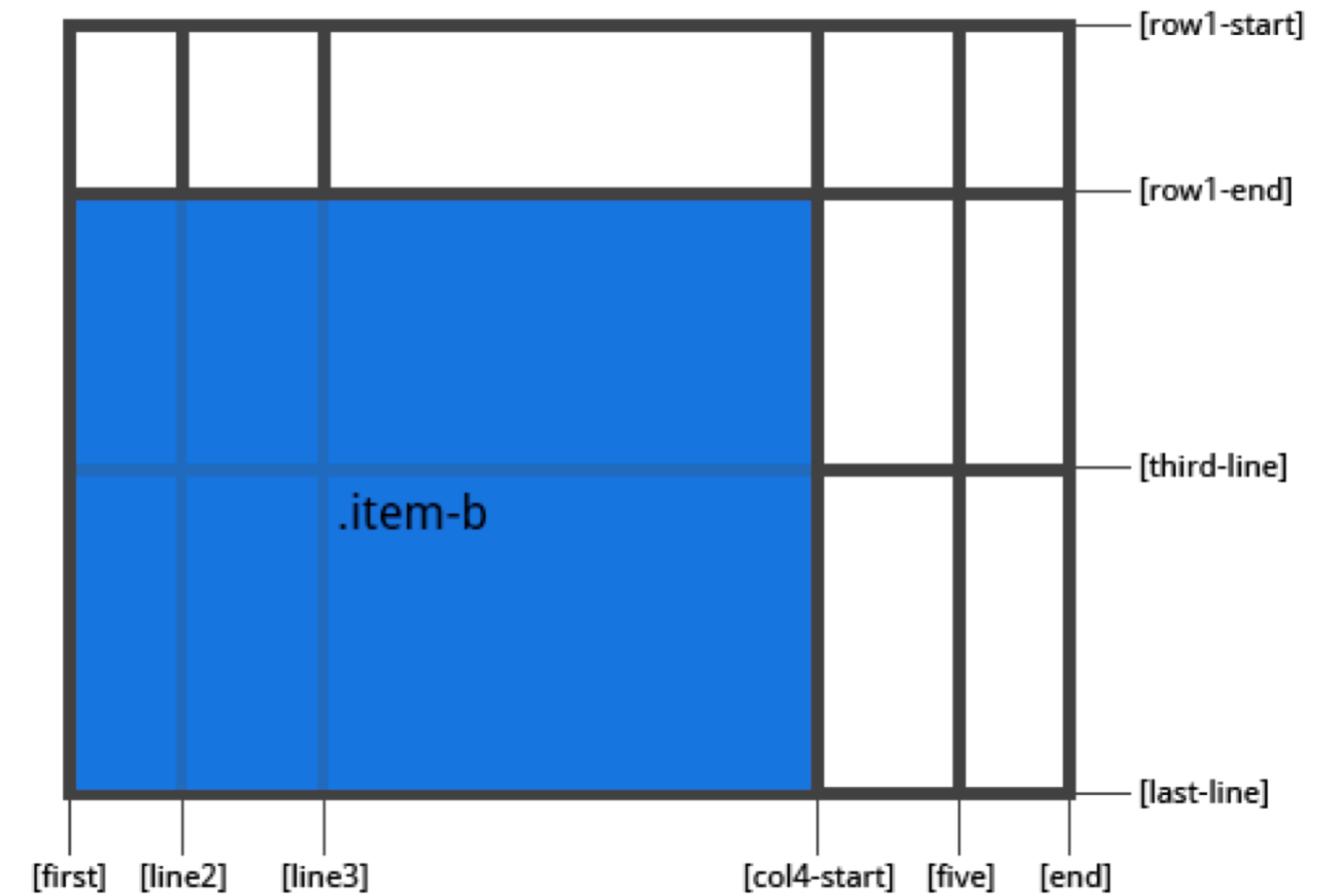
```
.item-a {  
  grid-area: header;  
}  
.item-b {  
  grid-area: main;  
}  
.item-c {  
  grid-area: sidebar;  
}  
.item-d {  
  grid-area: footer;  
}
```

```
.container {  
  grid-template-columns: 50px 50px 50px 50px;  
  grid-template-rows: auto;  
  grid-template-areas:  
    "header header header header"  
    "main main . sidebar"  
    "footer footer footer footer";  
}
```

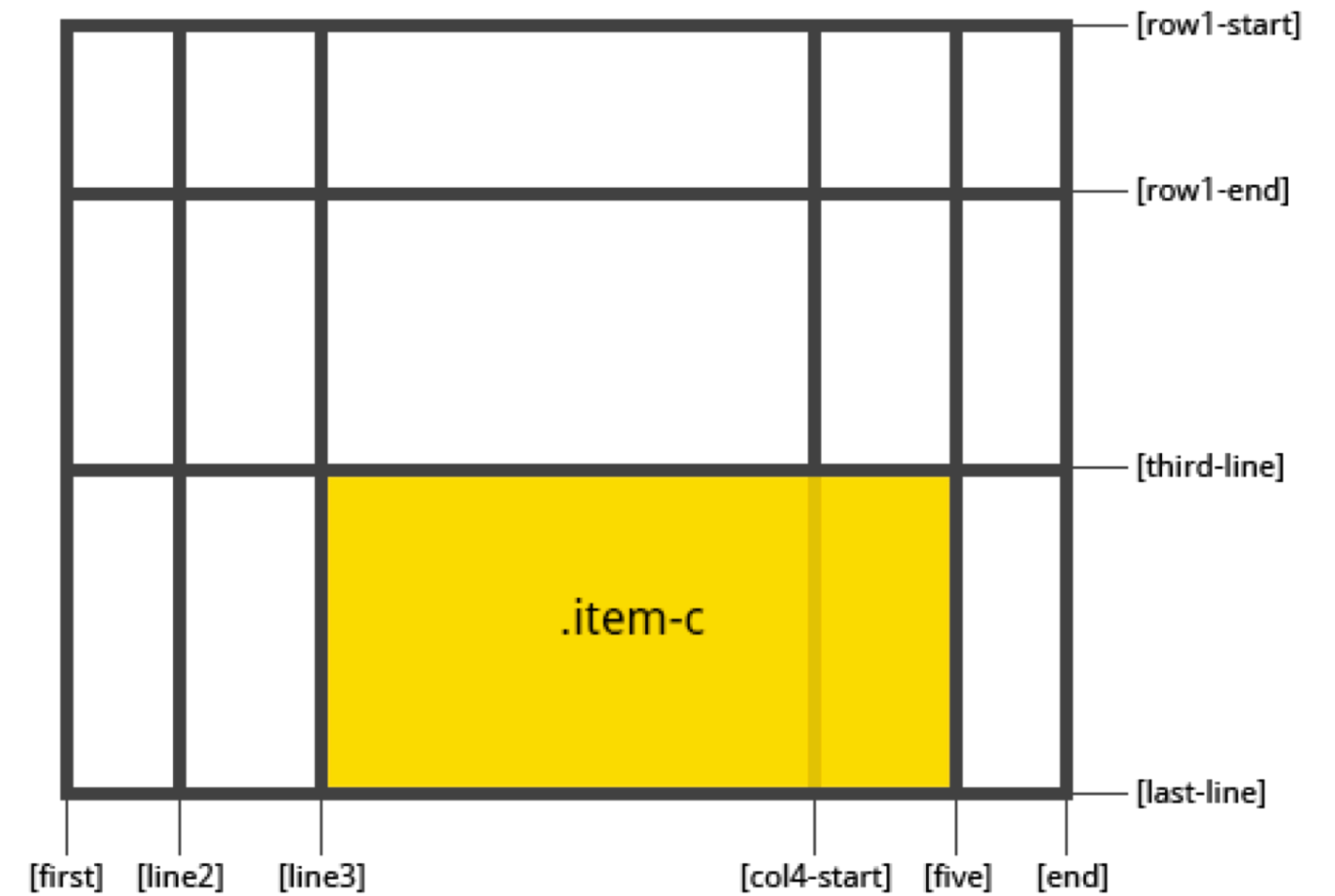
```
.item-a {  
  grid-column-start: 2;  
  grid-column-end: five;  
  grid-row-start: row1-start  
  grid-row-end: 3  
}
```



```
.item-b {  
  grid-column-start: 1;  
  grid-column-end: span col4-start;  
  grid-row-start: 2  
  grid-row-end: span 2  
}
```




```
.item-c {  
  grid-column: 3 / span 2;  
  grid-row: third-line / 4;  
}
```



```

.item-d {
  grid-area: header
}

/* short-shorthand for
grid-row-start +
grid-column-start +
grid-row-end +
grid-column-end:
*/

.item-d {
  grid-area: 1 / col4-start / last-line / 6
}

```

