



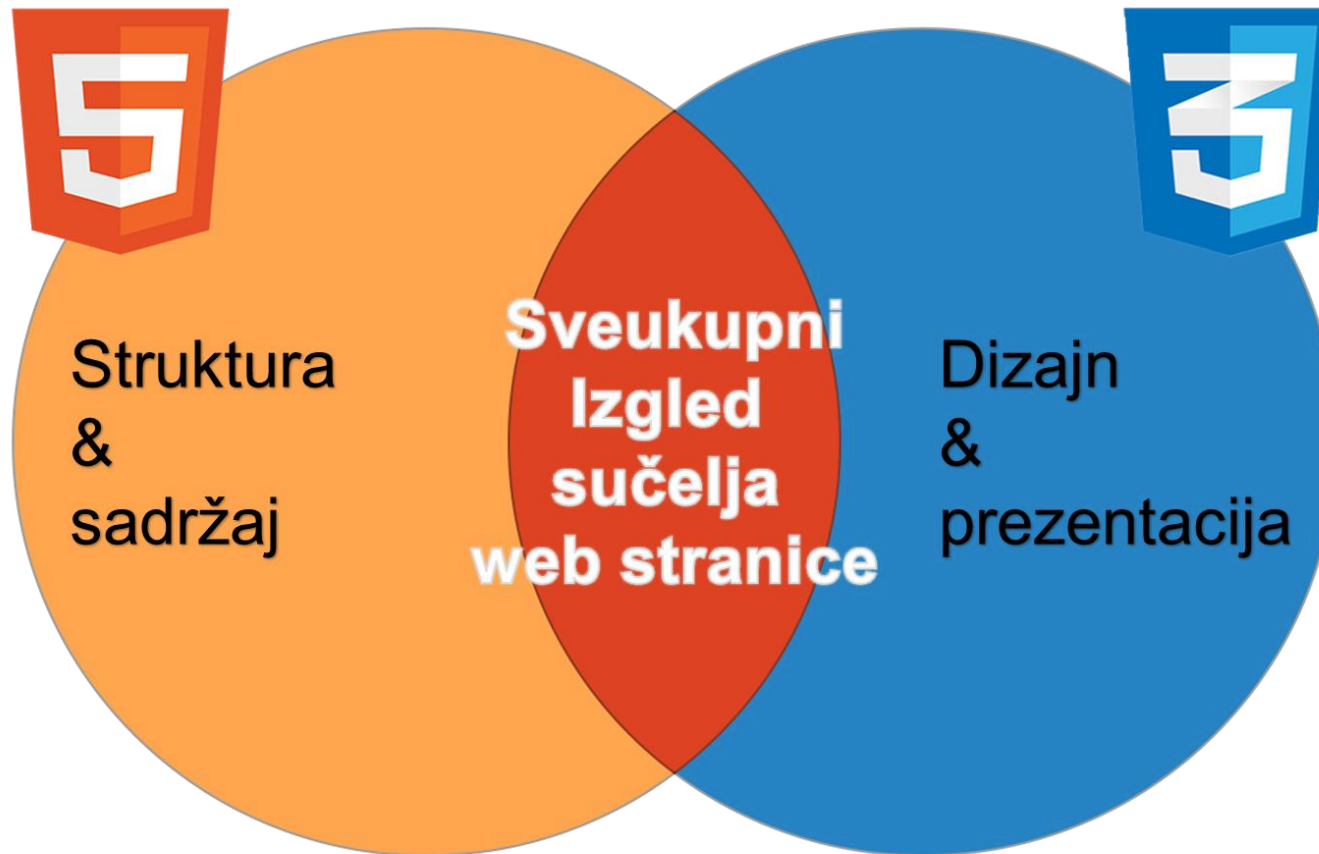
CSS

Uvod u CSS

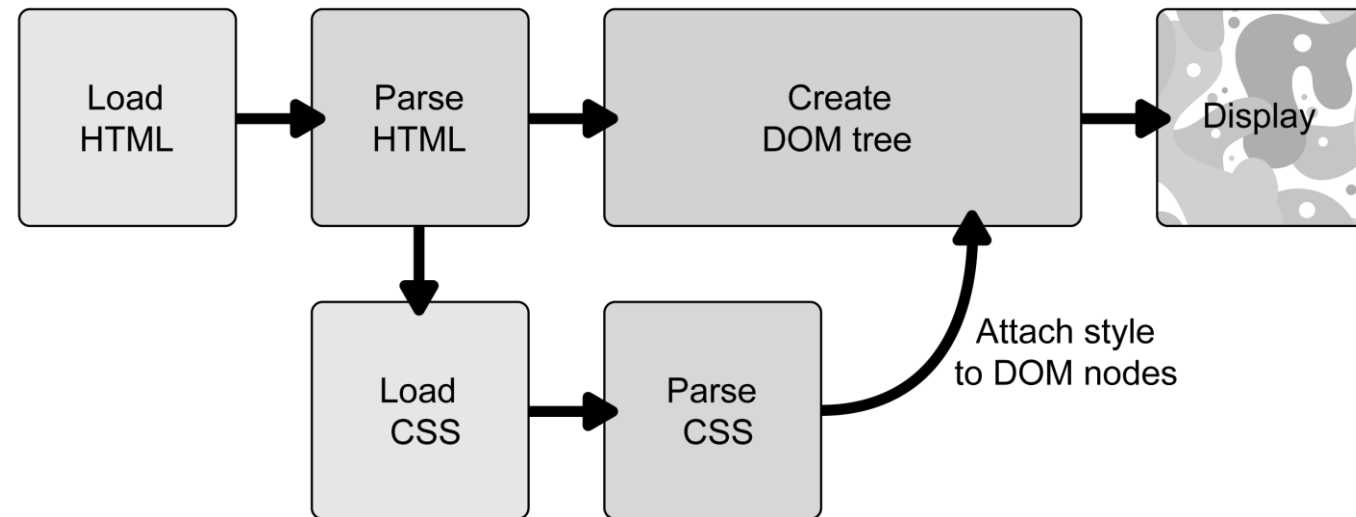
CSS

Cascading Style Sheets

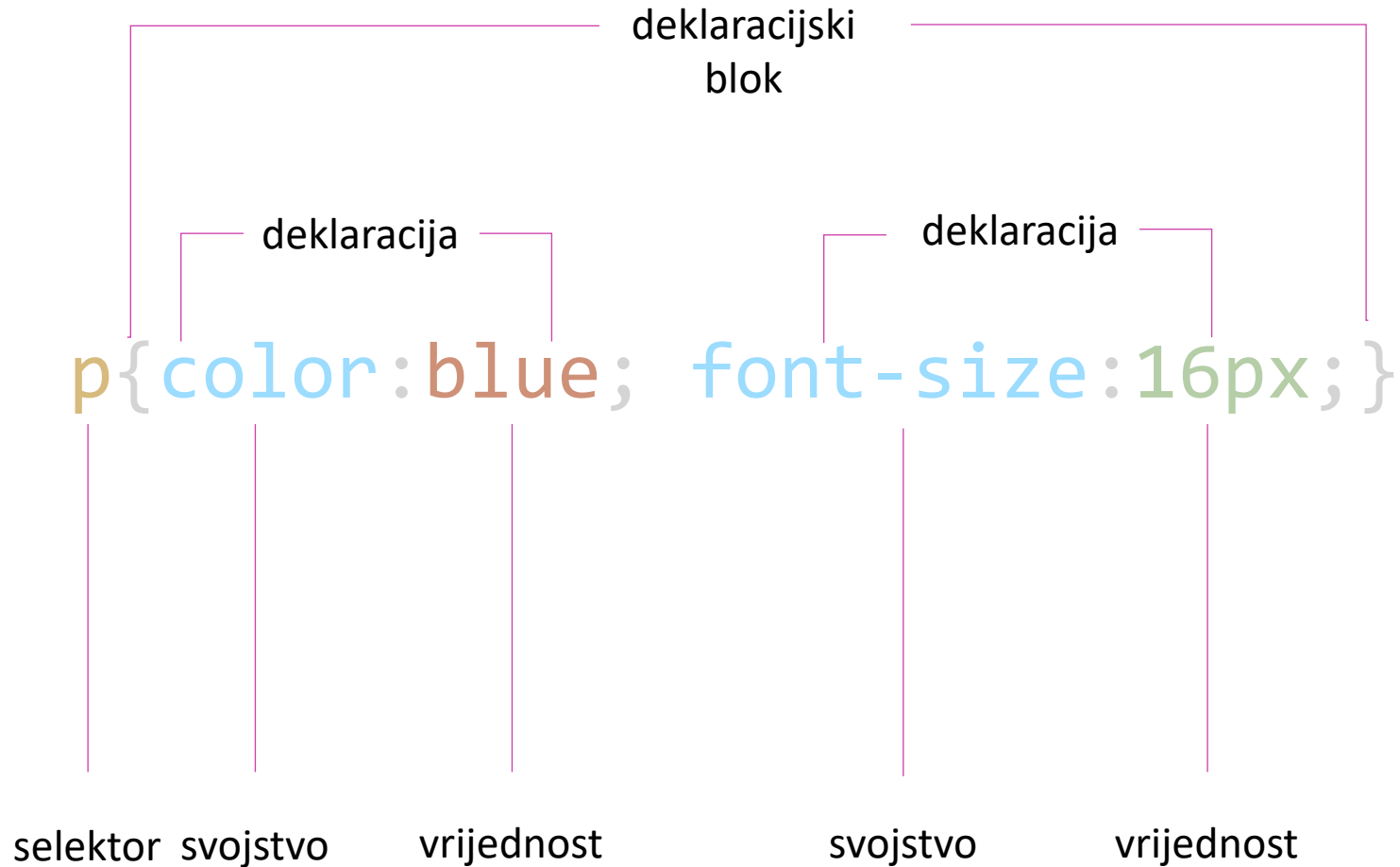




Kako funkcionira CSS?



Pravila CSS-a



```
h1 {  
  color: red;  
  font-size: 32px;  
}  
p {  
  color: blue;  
  font-size: 16px;  
}  
ul {  
  color: green;  
}
```

Hello World!

Lorem ipsum dolor sit, amet consectetur adipisicing elit. Quas, quos consequuntur ullam a consectetur, sint illo voluptas ducimus minus dolores consequatur. Exercitationem sequi amet culpa nulla harum ad officiis labore.

- Lorem ipsum
- Lorem ipsum
- Lorem ipsum

```
<h1>Hello World!</h1>
<p>Lorem ipsum dolor
sit, amet consectetur
adipisicing elit.</p>
<ul>
<li>Lorem ipsum</li>
<li>Lorem ipsum</li>
<li>lorem ipsum</li>
</ul>
```

```
h1 {
  color: red;
  font-size: 32px;
}
p {
  color: blue;
  font-size: 16px;
}
ul {
  color: green;
}
```

Hello World!

Lorem ipsum dolor sit, amet
consectetur adipisicing elit.

- Lorem ipsum
- Lorem ipsum
- Lorem ipsum

Kako primijeniti CSS na HTML

Linijski CSS

```
<p style="color:red;font-size:16px;">Odlomak teksta</p>
```

Kako primijeniti CSS na HTML

Unutarnji CSS

...

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <title>HTML dokument</title>
```

```
  <style>
```

```
    p {
```

```
      color: red;
```

```
      font-size: 16px;
```

```
    }
```

```
  </style>
```

```
</head>
```

```
<body>
```

...

Kako primijeniti CSS na HTML

Vanjski CSS

...

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <title>HTML dokument</title>
```

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

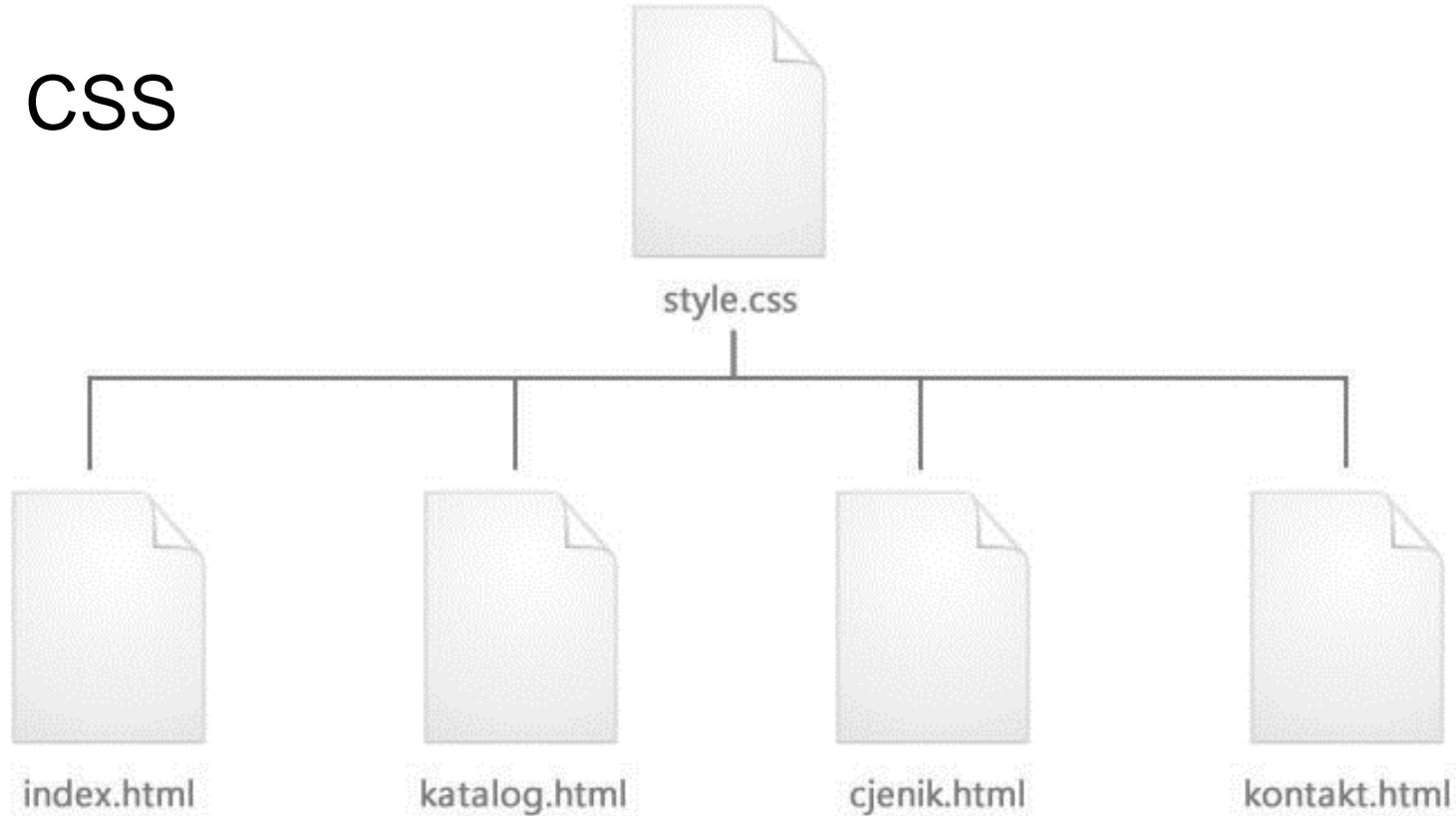
```
</head>
```

```
<body>
```

...

Kako primijeniti CSS na HTML

Vanjski CSS



Selektori

- jednostavni selektori
- selektori atributa
- pseudo-klase
- pseudo-elementi
- kontekstualni selektori
- grupni selektori

Selektori

Jednostavni selektori

- selektori HTML elmenata
- selektori atributa class
- selektori atributa id

Selektori

Jednostavni selektori

- selektori HTML elmenata

```
<h1>Hello World!</h1>
```

```
<p>Lorem ipsum dolor sit.</p>
```

```
<ul>
```

```
  <li>Lorem ipsum</li>
```

```
  <li>Lorem ipsum</li>
```

```
  <li>lorem ipsum</li>
```

```
</ul>
```

```
h1 {  
  color: red;  
  font-size: 32px;
```

```
}
```

```
p {  
  color: blue;  
  font-size: 16px;
```

```
}
```

```
ul {  
  color: green;  
}
```

Selektori

Jednostavni selektori

- selektori atributa class

```
<ul>  
  <li class="element1">Lorem ipsum</li>  
  <li class="element2">Lorem ipsum</li>  
  <li class="element3">lorem ipsum</li>  
</ul>
```

```
.element1 {  
  color: red;  
}  
.element2 {  
  color: green;  
}  
.element3 {  
  color: blue;  
}
```


Selektori

Jednostavni selektori

- selektori atributa id

```
<p id="odlomak1">Lorem ipsum</p>
```

```
<p id="odlomak2">Lorem ipsum</p>
```

```
<p id="odlomak3">Lorem ipsum</p>
```

```
#odlomak1 {  
    color: red;  
}
```

```
#odlomak2 {  
    color: green;  
}
```

```
#odlomak3 {  
    color: blue;  
}
```

Selektori

Selektori atributa

[attr]

[attr=val]

[attr~=val]

[attr^=val]

[attr|=val]

[attr\$=val]

[attr*=val]

Selektori

Pseudo-klase

:first-child

:last-child

:nth-child()

:first-of-type

:last-of-type

:nth-last-child()

Selektori

Pseudo-elementi

::after

::before

::first-letter

::first-line

::selection

Selektori

Kontekstualni selektori (*Combinators*)

descendant selector (razmak)

child selector (>)

adjacent sibling selector (+)

general sibling selector (~)

Selektori

Grupiranje selektora

```
p, a, table{...}
```

```
p, .element1, #odlomak1{...}
```

```
[title], a[target="_blank"]{...}
```

```
div a, header a{...}
```


```
...
```

Ključni koncepti

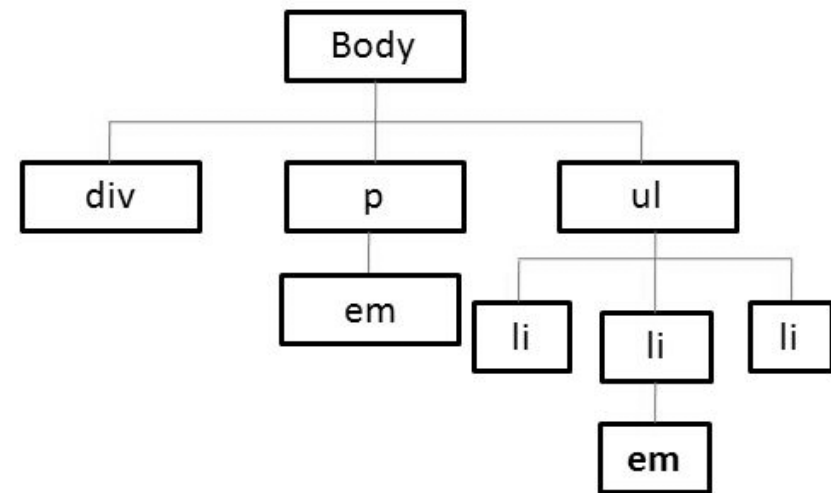
- nasljeđivanje
- grupiranje svojstava
- *box model*
- pozicioniranje
- responzivni i adaptivni dizajn
- *Media Queries*
- pristup *Mobile First*

Nasljeđivanje svojstava

Cascading order

	Lowest priority  Highest priority		
Stylesheet	Browser default	User	Author
Location	External	Embedded	Inline
Selectors	Contextual	Class	Id

Inheritance tree



Nasljeđivanje svojstava

Koncept kaskada (*cascading*)

- važnost i podrijetlo deklaracije
- specifičnost korištenog selektora
- redoslijed izvora

Nasljeđivanje svojstava

Koncept kaskada (*cascading*)

Važnost i podrijetlo deklaracije

- važnost deklaracije
 1. deklaracije normalne važnosti
 2. deklaracije s oznakom *!important*
 - `p{font-size:18px !important;}`
- podrijetlo deklaracije
 1. zadani stilovi preglednika koji se koriste kada nije postavljen nijedan drugi stil (*User-agent stylesheets*)
 2. prilagođeni stilovi koje je postavio korisnik (*User stylesheets*)
 3. stilovi koje je postavio autor/*developer* (*Author stylesheets*)

Nasljeđivanje svojstava

Koncept kaskada (*cascading*)

Važnost i podrijetlo deklaracije

'Cascading' order

1. zadani stilovi preglednika koji se koriste kada nije postavljen nijedan drugi stil (*User-agent stylesheets*)
2. prilagođeni stilovi koje je postavio korisnik (*User stylesheets*)
3. stilovi koje je postavio autor/*developer* (*Author stylesheets*)
4. deklaracije s oznakom *!important* u stilovima autora/*developer*a
5. deklaracije s oznakom *!important* koje je postavio korisnik (*User stylesheets*)

Nasljeđivanje svojstava

Koncept kaskada (*cascading*)

Specifičnost korištenog selektora

1. selektori HTML elemenata, pseudo-elementi
2. selektori atributa class, pseudo-klase, selektori atributa
3. selektori atributa id

Nasljeđivanje svojstava

Koncept kaskada (*cascading*)

Redoslijed izvora

Lokacija izvora

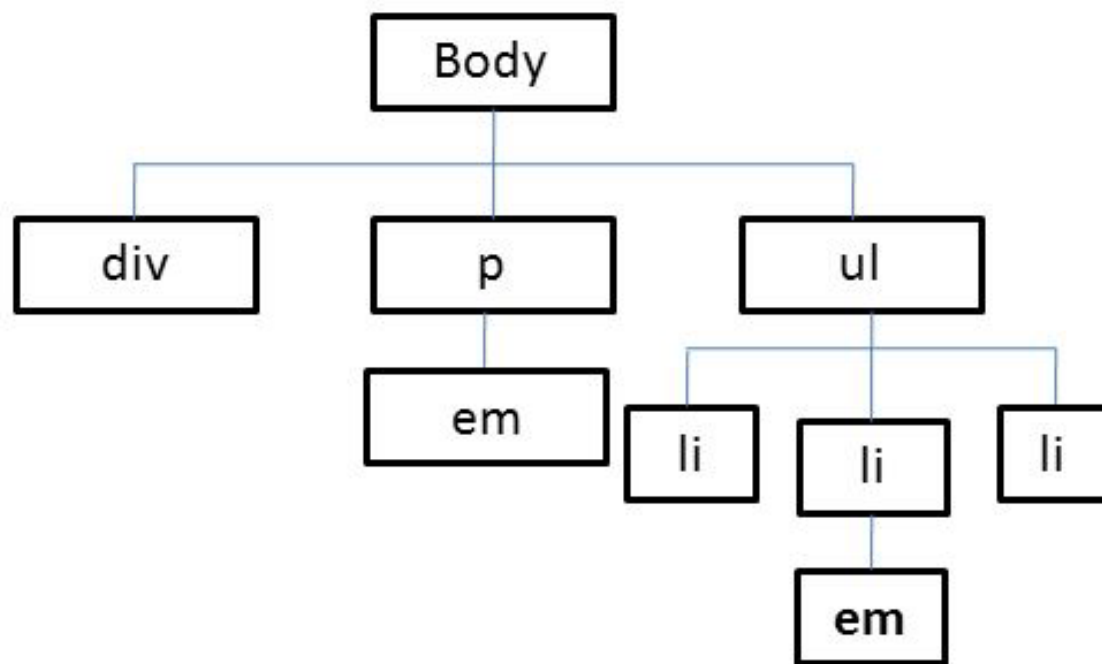
1. vanjski CSS
2. unutarnji CSS
3. linijski CSS

Redoslijed pisanja pravila

- Niže napisano pravilo „pobjeđuje”

Nasljeđivanje svojstava

Nasljeđivanje svojstava od roditelja/potomaka



Uređivanje teksta

Uređivanje teksta

- svojstva za uređivanje fontova
- svojstva za uređivanje teksta

Svojstva za uređivanje fontova

font-family

font-size

font-style

font-weight

font-variant

Svojstva za uređivanje teksta

color

text-align

text-decoration

text-transform

text-shadow

line-height

Uređivanje pozadina

Uređivanje pozadina

- pozadinska boja
- pozadinska grafika

Pozadinska boja

background-color

Pozadinska grafika

background-image

background-repeat

background-position

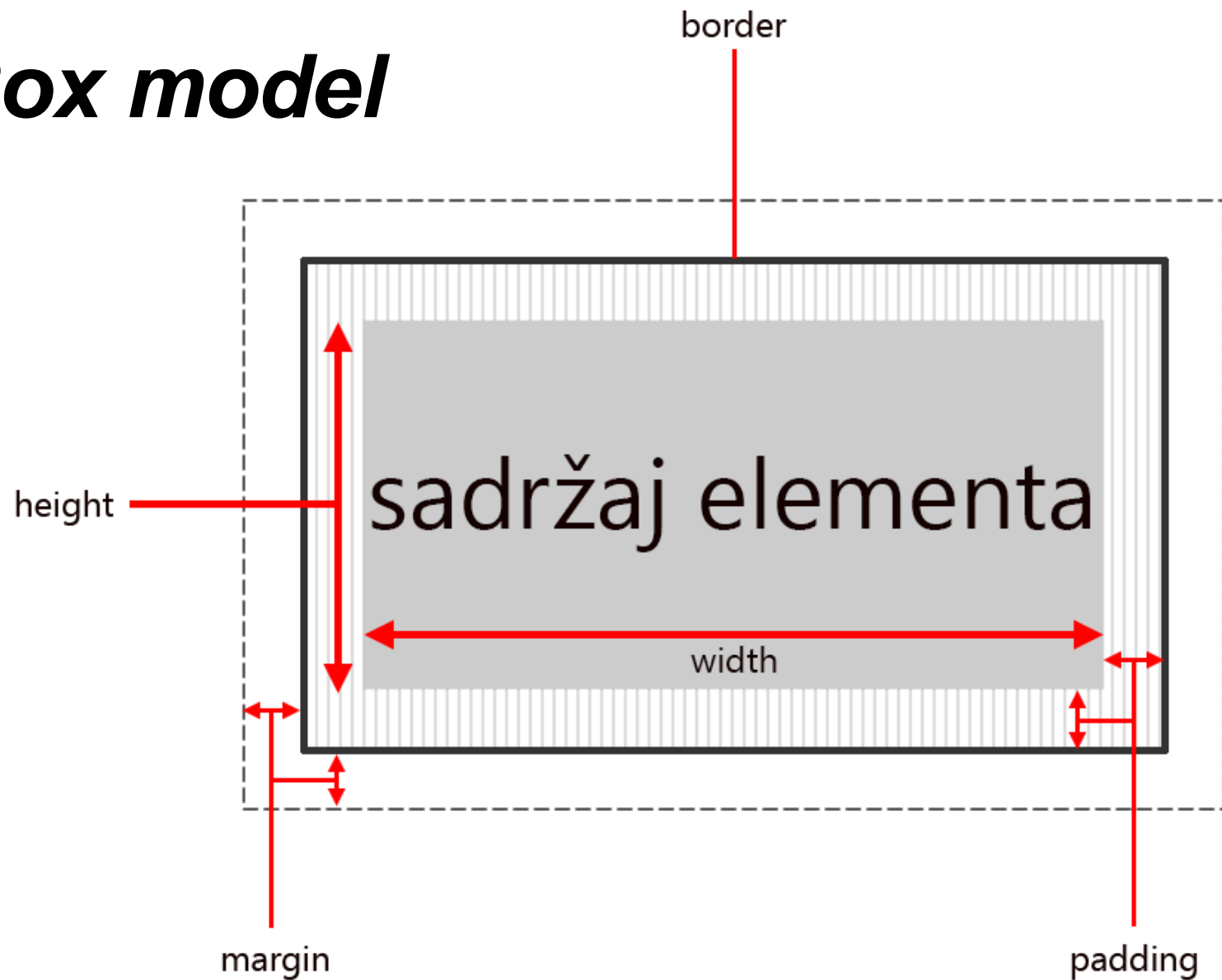
background-attachment

background-size

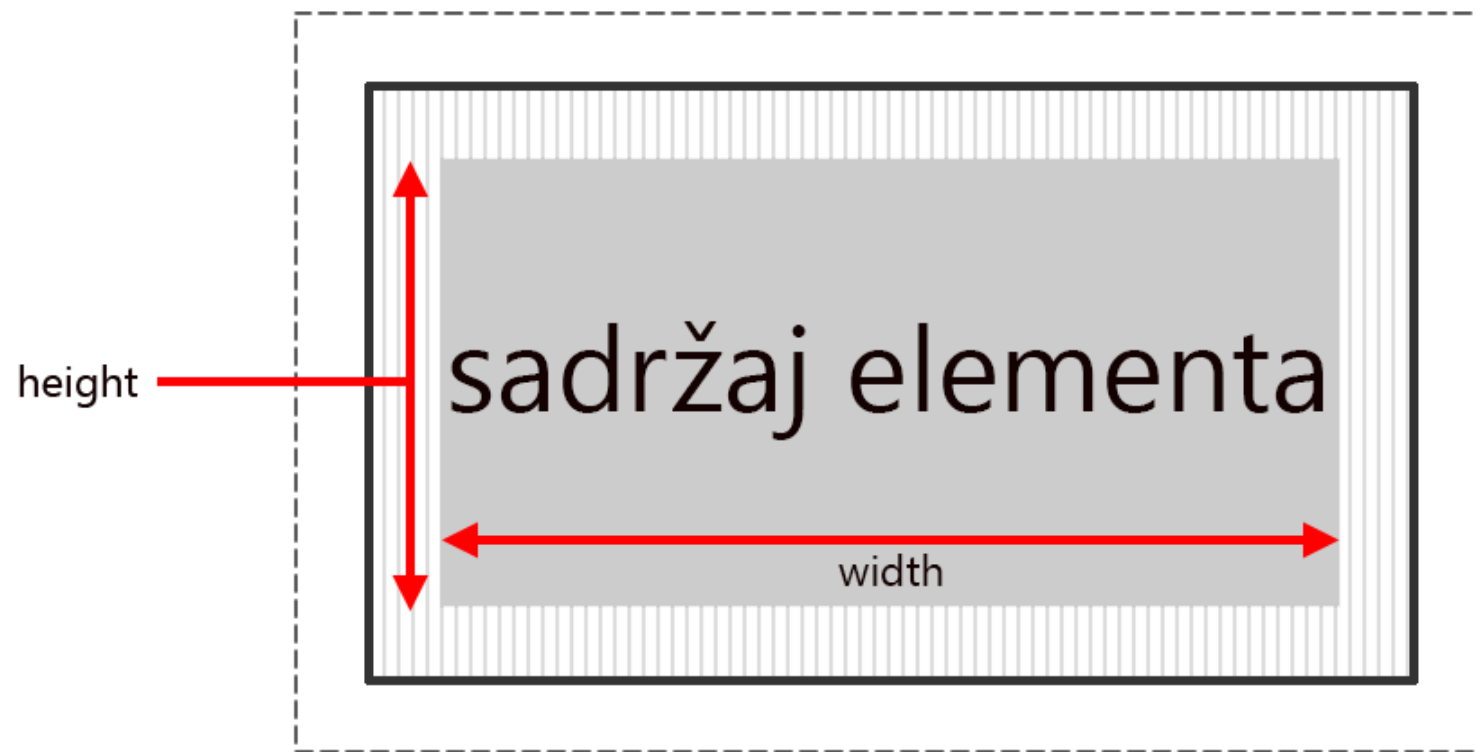
background

Uređivanje elemenata

Box model



Box model

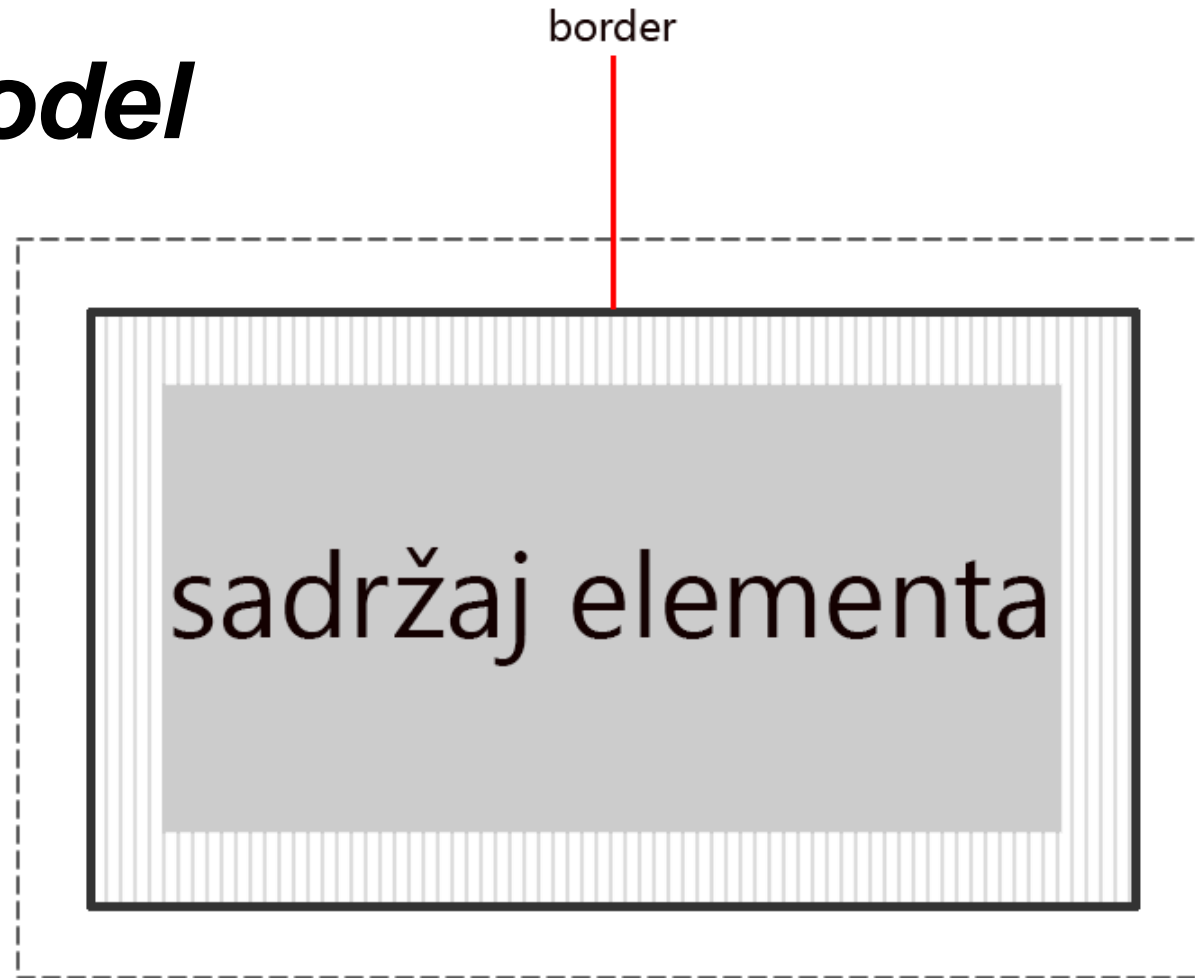


Box model

`width: auto|value|initial|inherit`

`height: auto|value|initial|inherit`

Box model



Box model

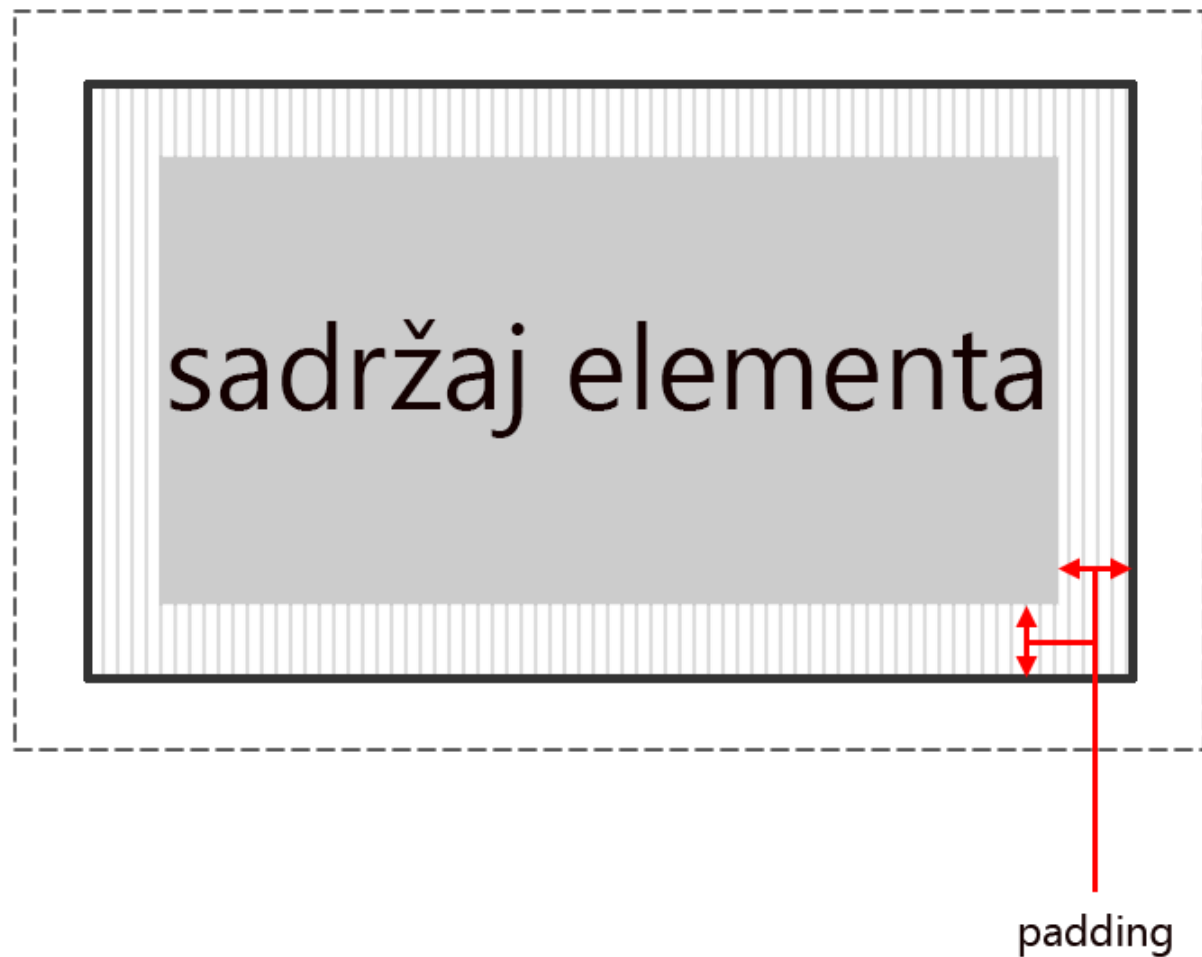
`border-width: medium|thin|thick|length|initial|inherit`

`border-style: none|hidden|dotted|dashed|solid|double|groove|ridge|inset|outset|initial|inherit`

`border-color: color|transparent|initial|inherit`

`border: border-width border-style border-color|initial|inherit`

Box model



Box model

`padding: length|initial|inherit`

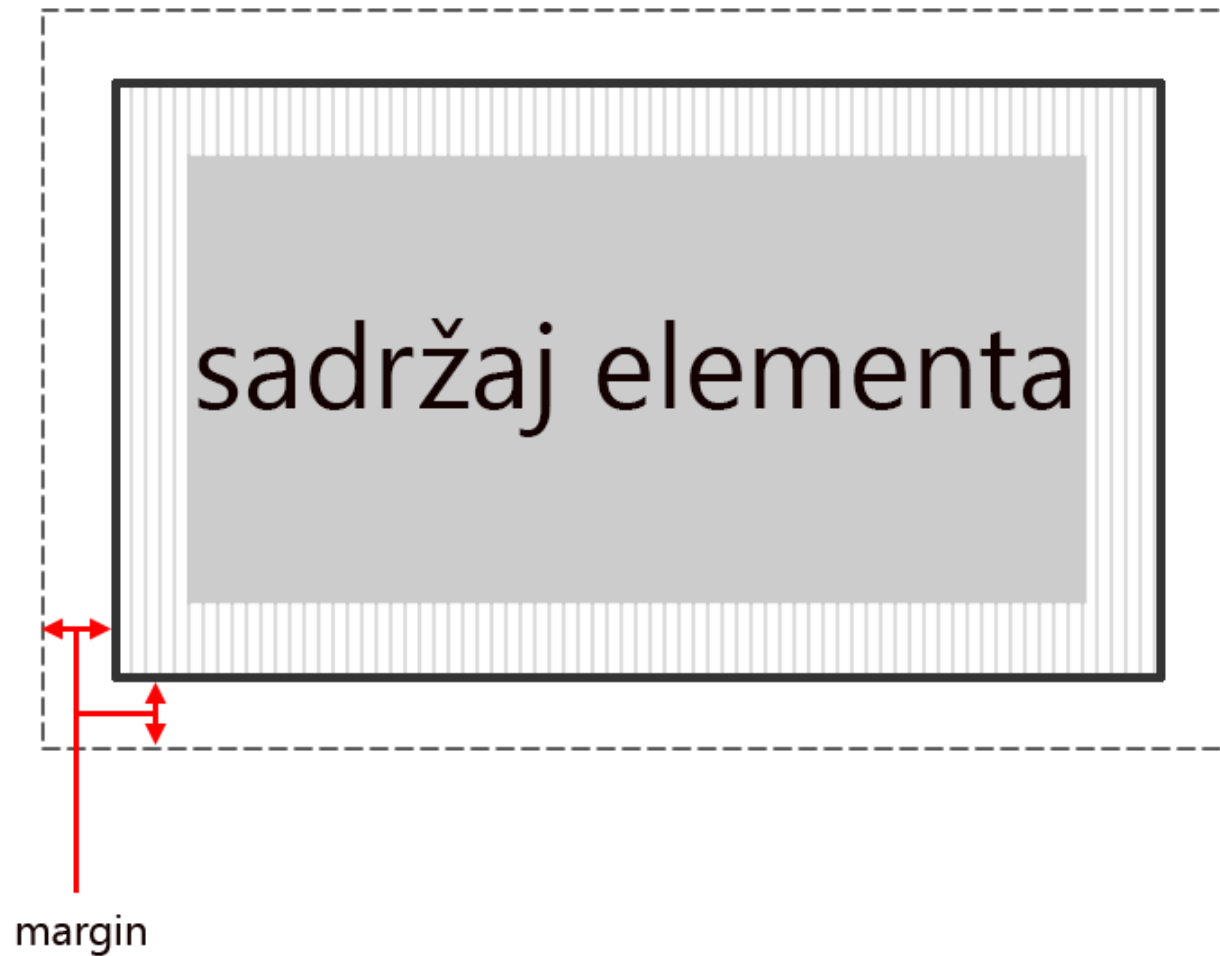
`padding-top: length|initial|inherit`

`padding-right: length|initial|inherit`

`padding-bottom: length|initial|inherit`

`padding-left: length|initial|inherit`

Box model



Box model

margin: length|initial|inherit

margin-top: length|initial|inherit

margin-right: length|initial|inherit

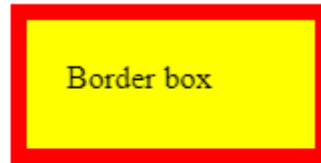
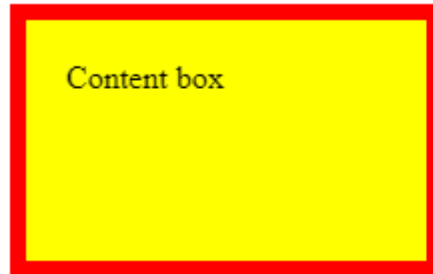
margin-bottom: length|initial|inherit

margin-left: length|initial|inherit

Box model

box-sizing

`box-sizing:` content-box | border-box



Box model

box-sizing

content-box

```
div {  
  width: 160px;  
  height: 80px;  
  padding: 20px;  
  border: 8px solid red;  
  background: yellow;  
}  
.content-box {  
  box-sizing: content-box;  
  /* Total width: 160px + (2 * 20px) + (2 * 8px) = 216px  
  Total height: 80px + (2 * 20px) + (2 * 8px) = 136px  
  Content box width: 160px  
  Content box height: 80px */  
}
```

border-box

```
div {  
  width: 160px;  
  height: 80px;  
  padding: 20px;  
  border: 8px solid red;  
  background: yellow;  
}  
.border-box {  
  box-sizing: border-box;  
  /* Total width: 160px  
  Total height: 80px  
  Content box width: 160px - (2 * 20px) - (2 * 8px) =  
  104px  
  Content box height: 80px - (2 * 20px) - (2 * 8px) =  
  24px */  
}
```

Kako preglednici prikazuju elemente

Elementi bloka
(*block-level elements*)

lorem ipsum

Elementi reda
(*inline-level elements*)

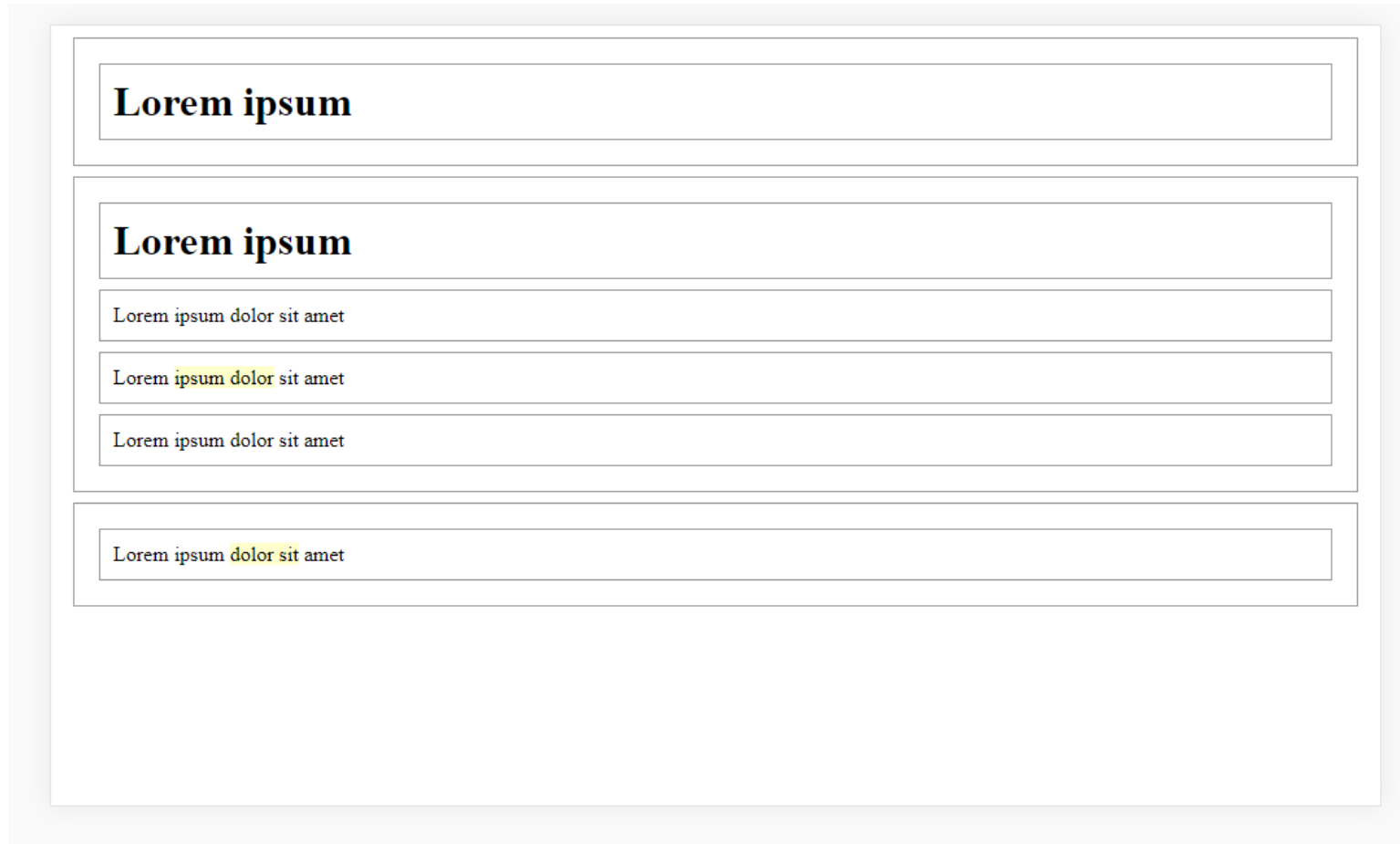
lorem ipsum

Kako preglednici prikazuju elemente

`display: block | inline | inline-block | none`

Kako su elementi razmješteni na stranici

Normal Flow



Pozicioniranje elemenata

Svojstvo *position*

`position: static | absolute | fixed | relative | sticky | initial | inherit`

Pozicioniranje elemenata

Svojstvo *position*

`position: static`

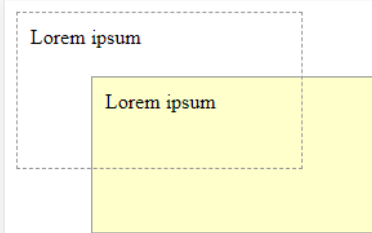
Lorem ipsum

```
#div1 {  
  position: static;  
}
```

Pozicioniranje elemenata

Svojstvo *position*

`position: relative`



```
#div1 {  
  position: relative;  
  left: 50px;  
  top: 50px;  
}
```


Pozicioniranje elemenata

Svojstvo *position*

`position: absolute`

parent div

Lorem ipsum

```
#parent {  
  position: relative;  
  left: 30px;  
  top: 30px;  
}
```

```
#div1 {  
  position: absolute;  
  left: 250px;  
  top: 50px;  
}
```

Pozicioniranje elemenata

Svojstvo *position*

`position: fixed`

```
#div1 {  
  position: fixed;  
  right: 0;  
  bottom: 0;  
}
```

Lorem ipsum

Pozicioniranje elemenata

Svojstva float i clear

`float:` none | left | right | initial | inherit

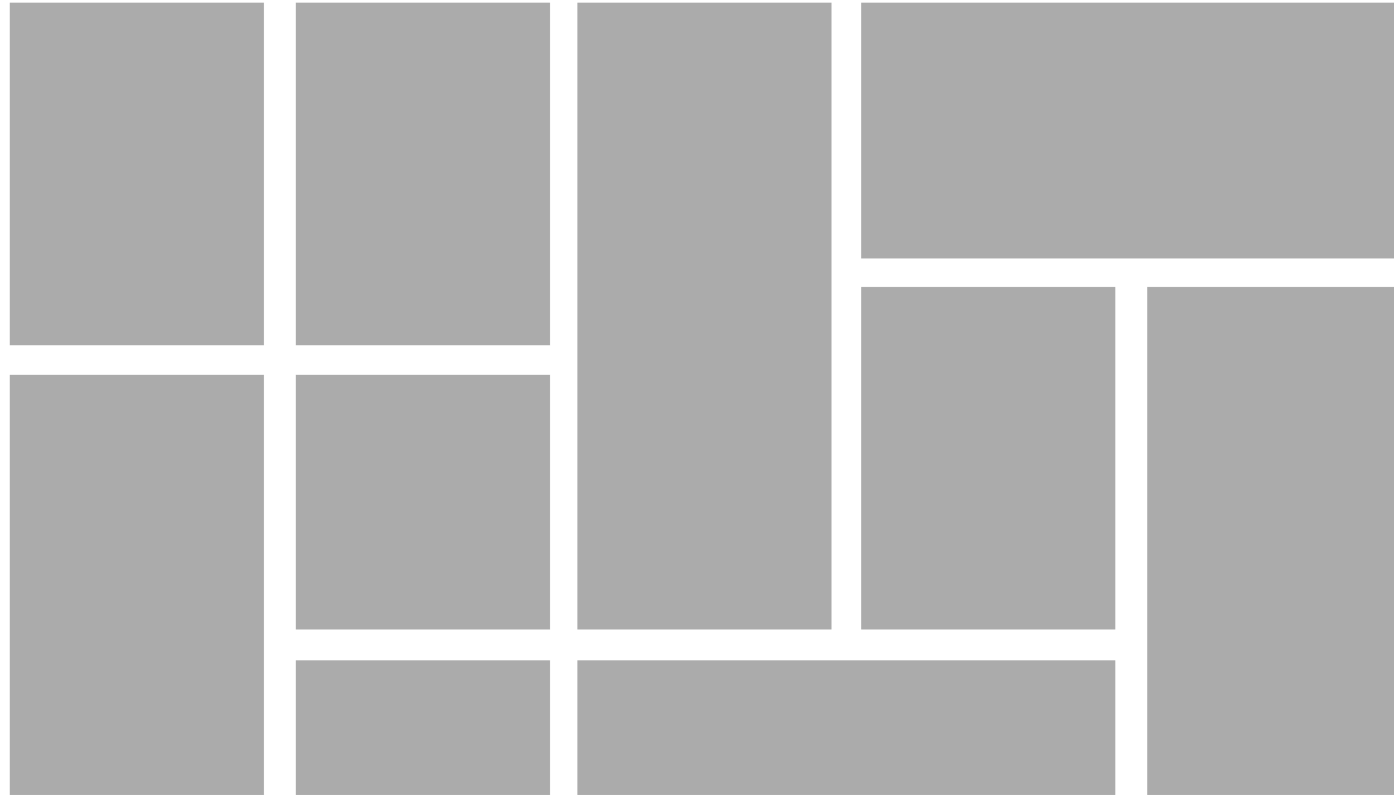
`clear:` none | left | right | both | initial | inherit

Izrada *layouta*

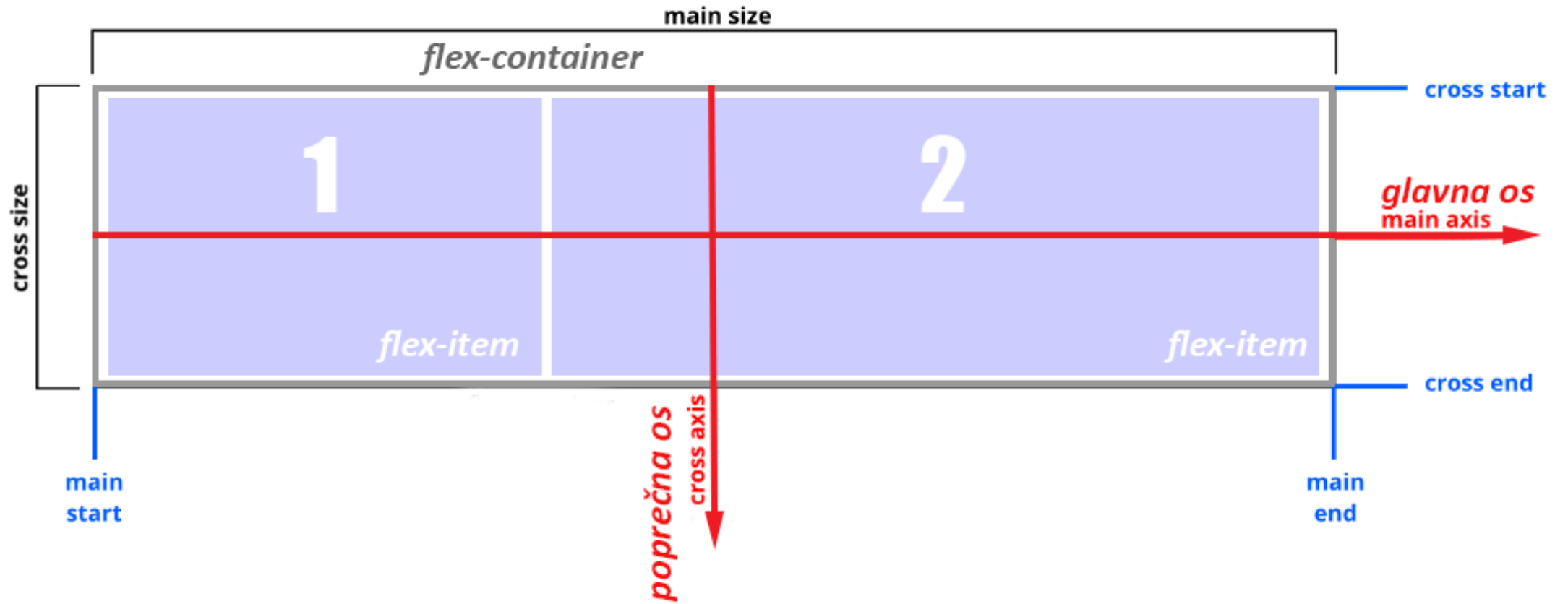
Tehnike izrade *layouta*

- tablice
- *Float*
- *Flexbox*
- *Grid*

Flexible Box Layout



Flexible Box Layout



Flexible Box Layout

Flex-container

`display:flex` | `inline-flex`

Flexible Box Layout

Flex-container

`flex-direction:` row | row-reverse | column | column-reverse

Flexible Box Layout

Flex-container

`flex-direction: row`



Flexible Box Layout

Flex-container

`flex-direction: row-reverse`



Flexible Box Layout

Flex-container

`flex-direction: column`



Flexible Box Layout

Flex-container

`flex-direction: column-reverse`



Flexible Box Layout

Flex-container

`flex-wrap:` nowrap | wrap | wrap-reverse

Flexible Box Layout

Flex-container

`flex-wrap: nowrap`



Flexible Box Layout

Flex-container

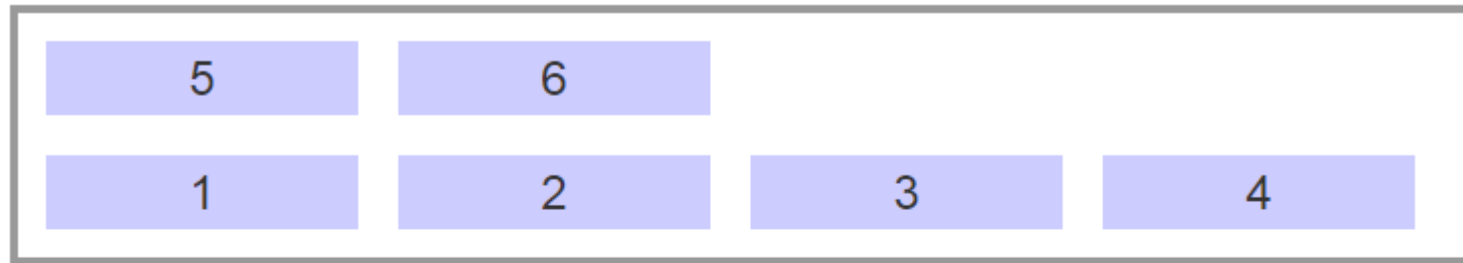
`flex-wrap: wrap`



Flexible Box Layout

Flex-container

`flex-wrap: wrap-reverse`



Flexible Box Layout

Flex-container

`flex-flow: <'flex-direction'> || <'flex-wrap'>`

Flexible Box Layout

Flex-container

`justify-content:` flex-start | flex-end | center |
space-between | space-around | space-evenly

Flexible Box Layout

Flex-container

`justify-content: flex-start`



Flexible Box Layout

Flex-container

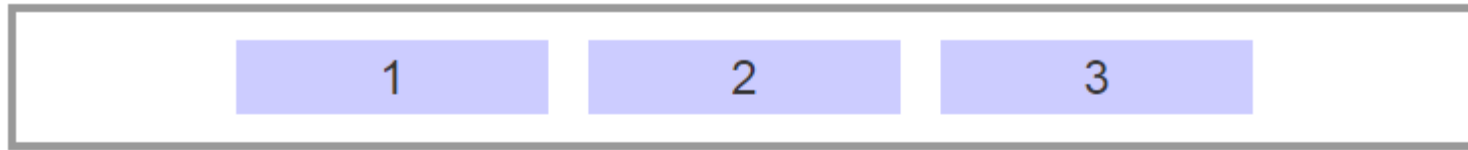
`justify-content: flex-end`



Flexible Box Layout

Flex-container

`justify-content: center`



Flexible Box Layout

Flex-container

`justify-content: space-between`



Flexible Box Layout

Flex-container

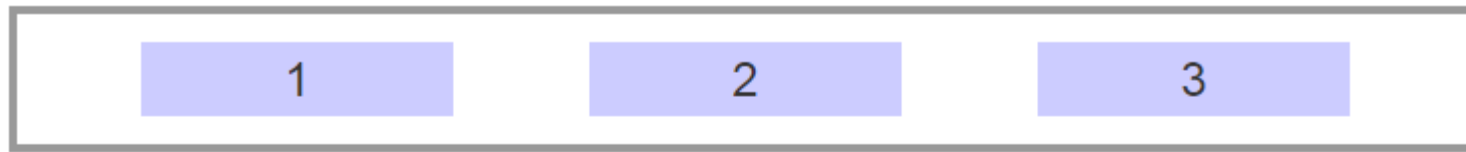
`justify-content: space-around`



Flexible Box Layout

Flex-container

`justify-content: space-evenly`



Flexible Box Layout

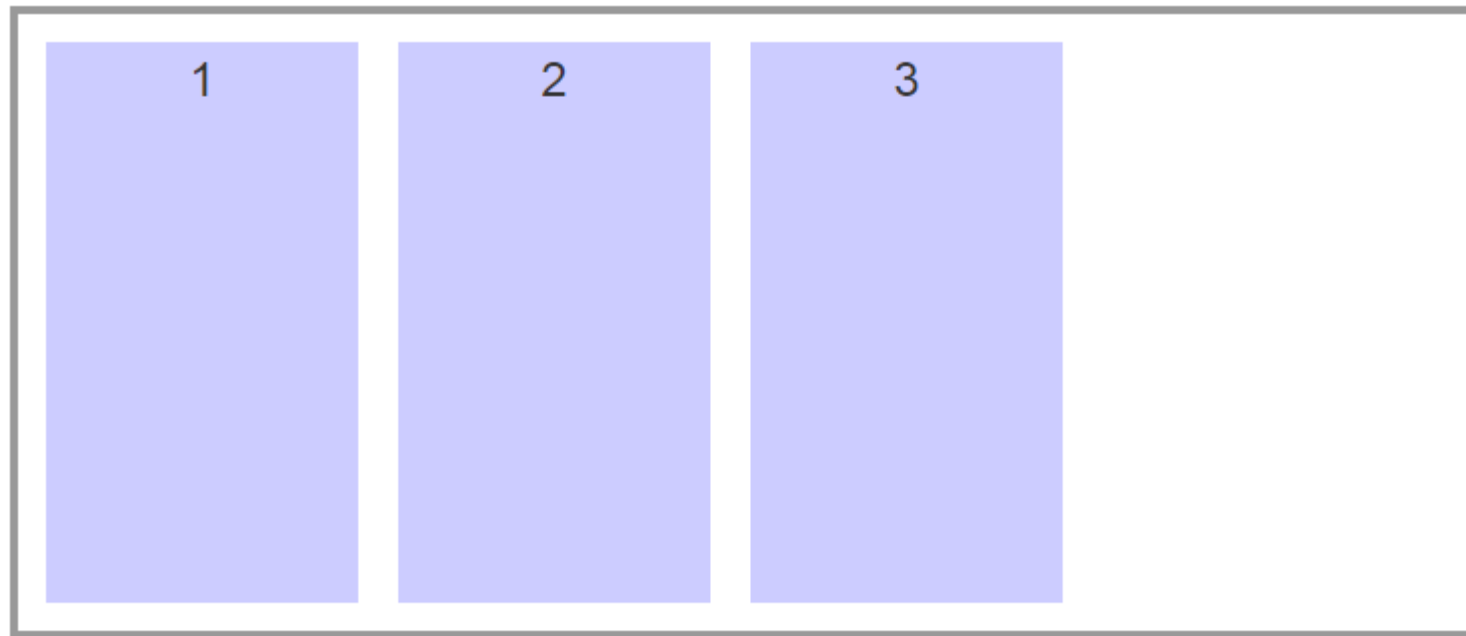
Flex-container

`align-items:` stretch | flex-start | flex-end | center | baseline

Flexible Box Layout

Flex-container

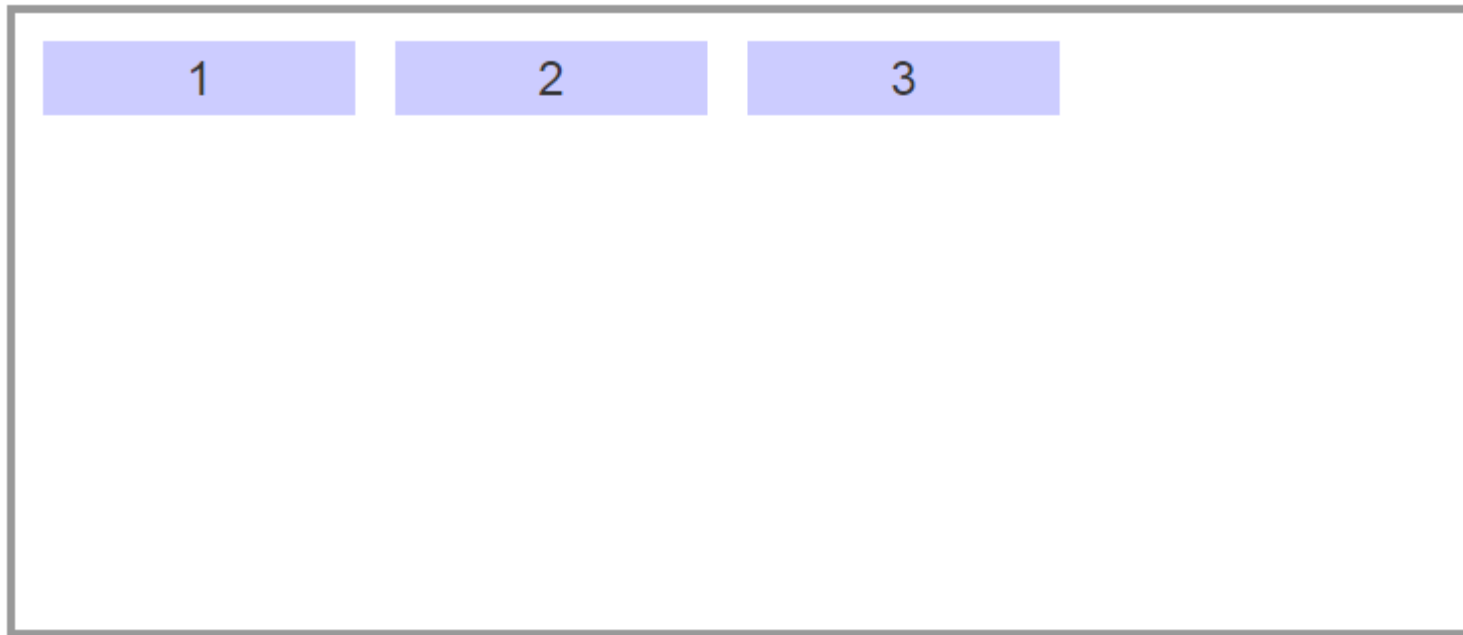
`align-items: stretch`



Flexible Box Layout

Flex-container

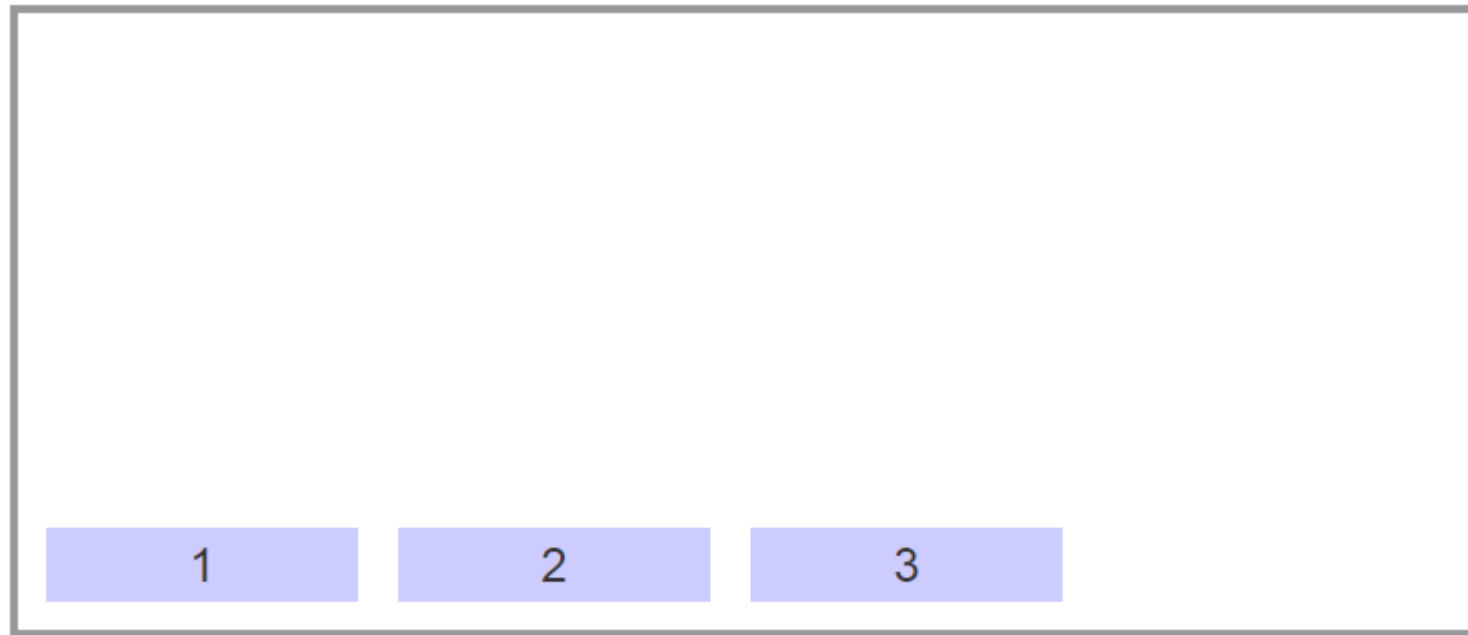
`align-items: flex-start`



Flexible Box Layout

Flex-container

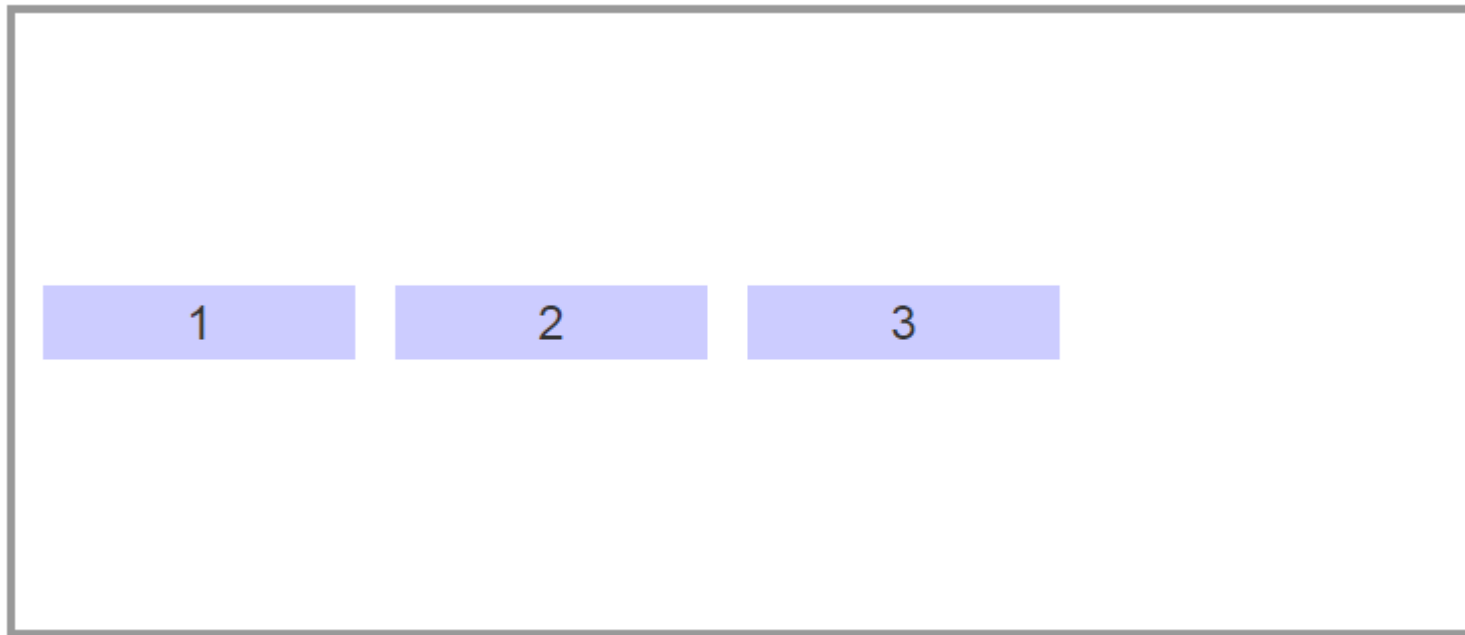
`align-items: flex-end`



Flexible Box Layout

Flex-container

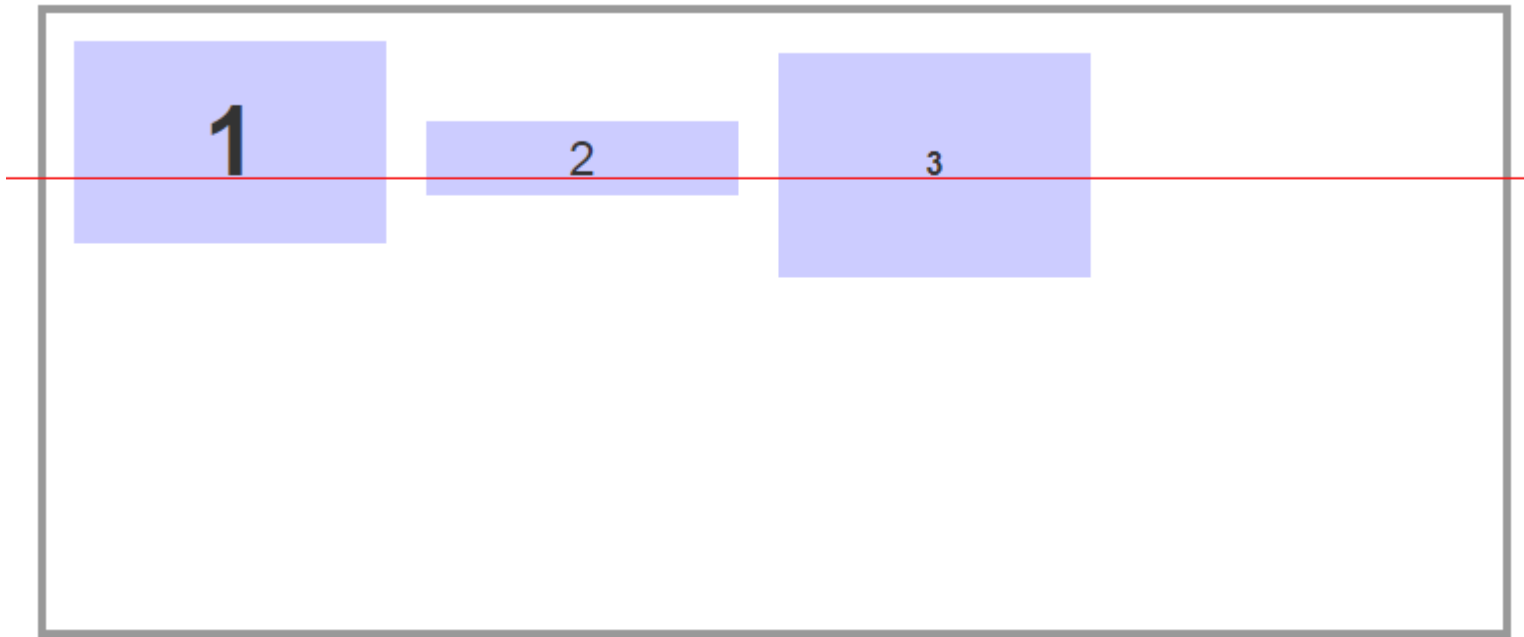
`align-items: center`



Flexible Box Layout

Flex-container

`align-items: baseline`



Flexible Box Layout

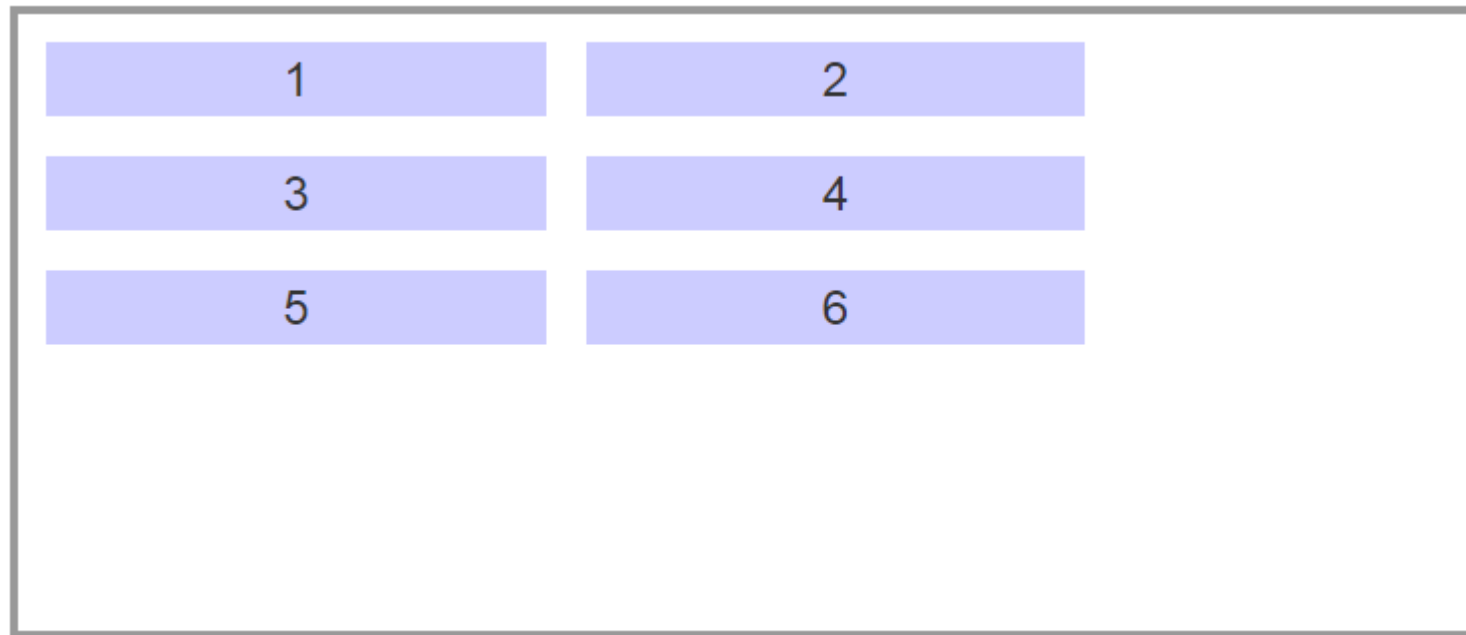
Flex-container

`align-content`: `flex-start` | `flex-end` | `center` | `space-between` | `space-around` | `stretch`

Flexible Box Layout

Flex-container

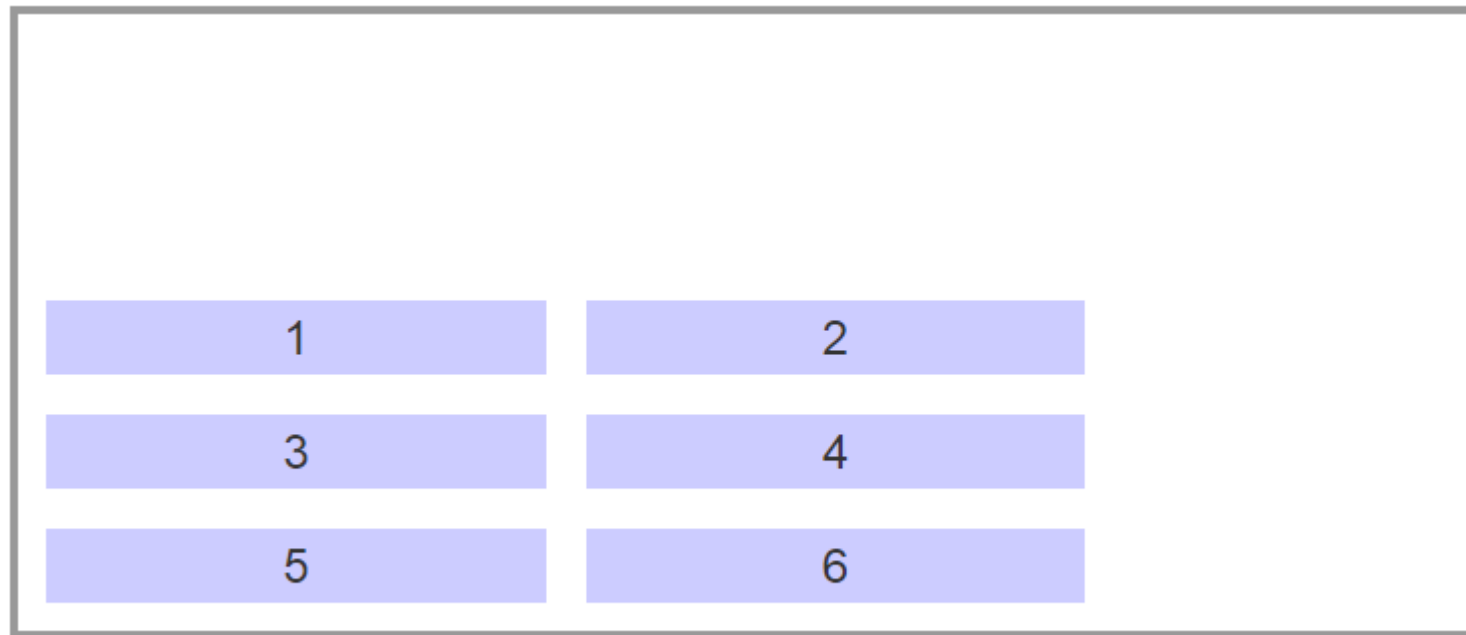
`align-content: flex-start`



Flexible Box Layout

Flex-container

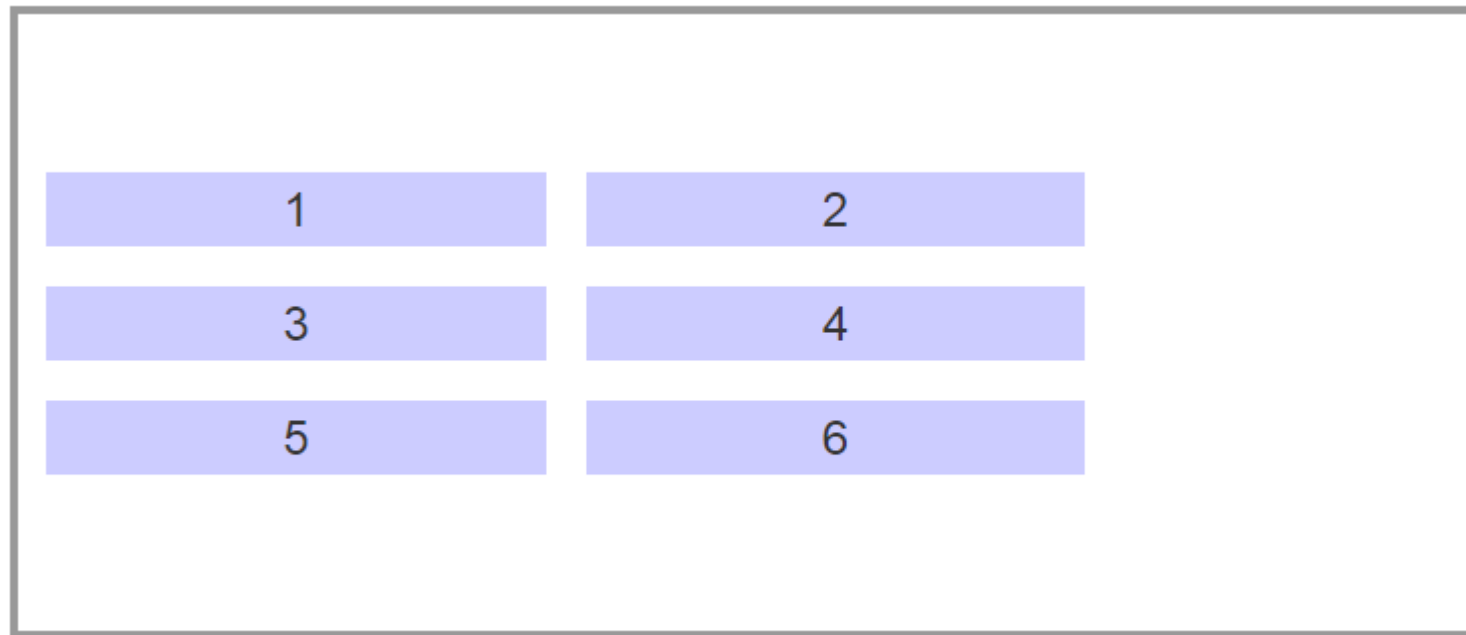
`align-content: flex-end`



Flexible Box Layout

Flex-container

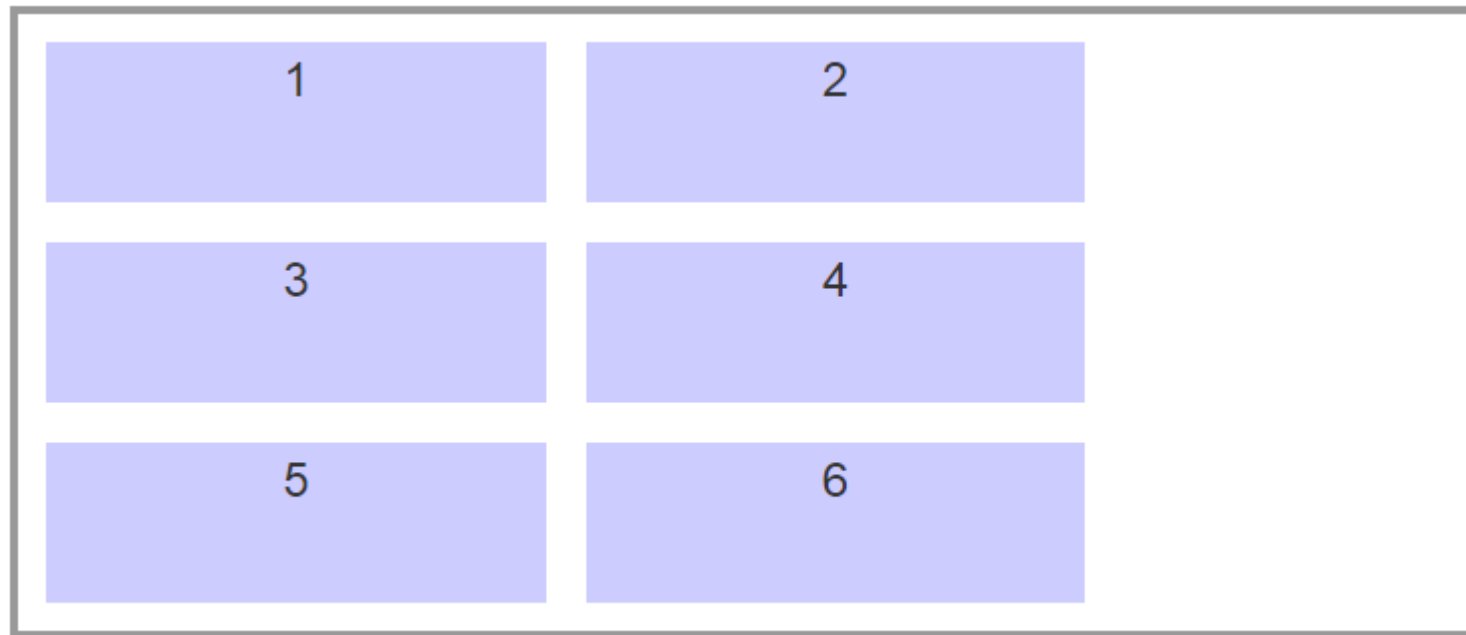
`align-content: center`



Flexible Box Layout

Flex-container

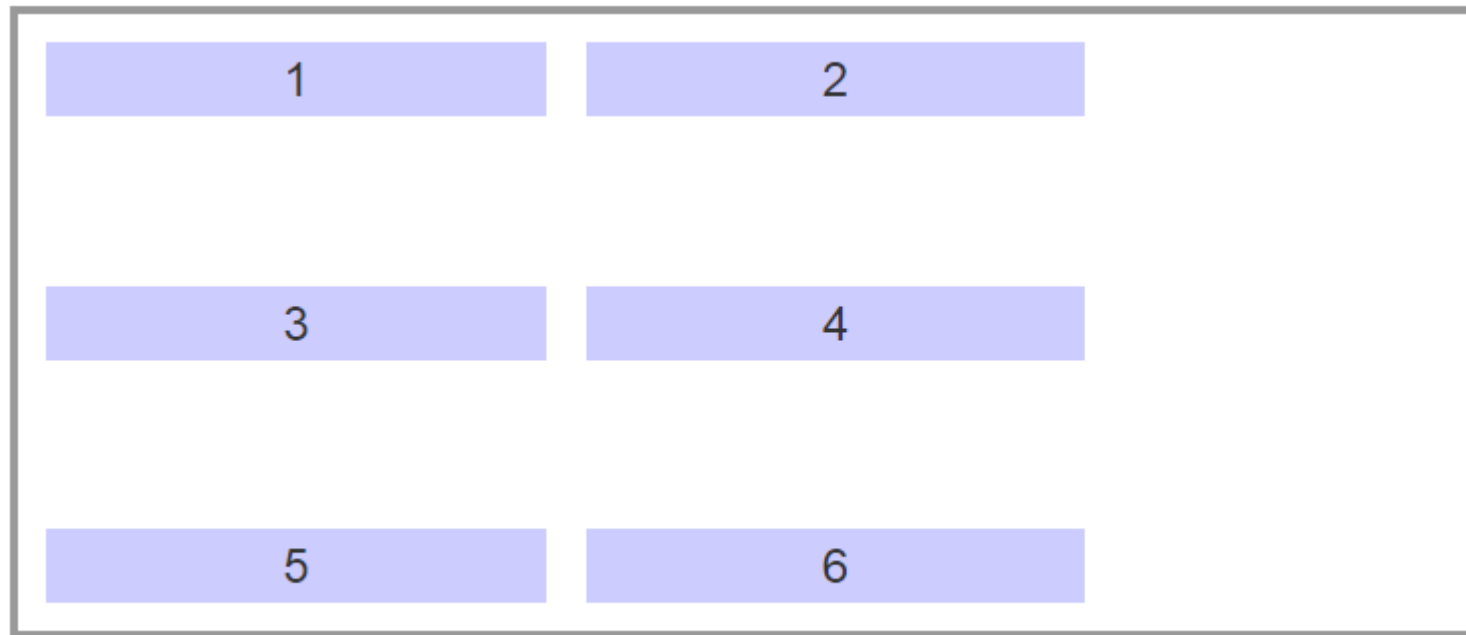
`align-content: stretch`



Flexible Box Layout

Flex-container

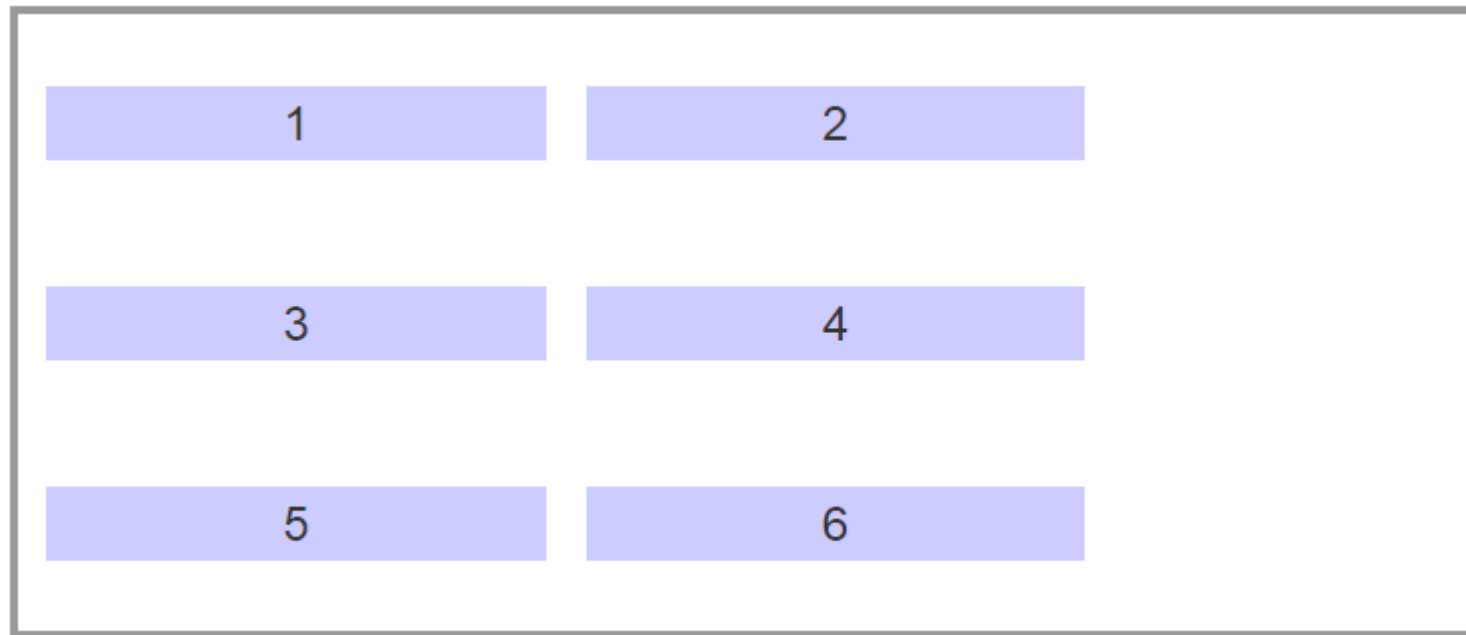
`align-content: space-between`



Flexible Box Layout

Flex-container

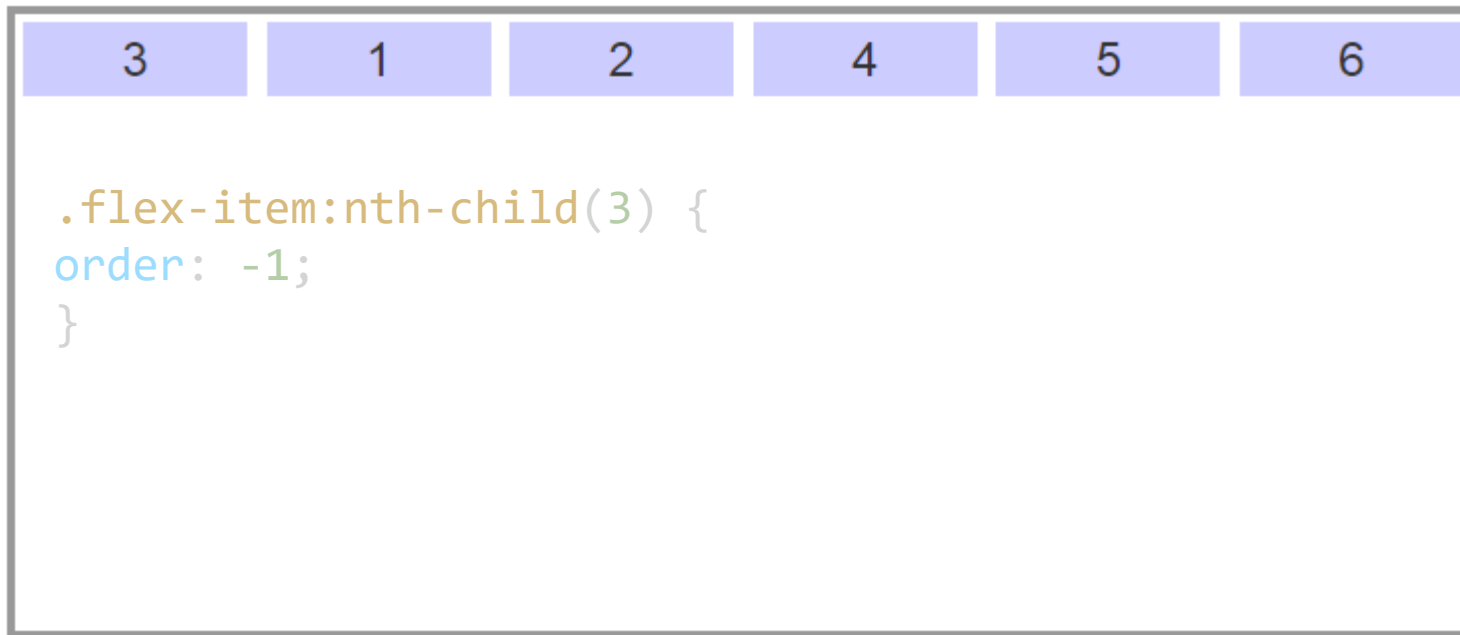
`align-content: space-around`



Flexible Box Layout

Flex-items

`order: <integer>`



Flexible Box Layout

Flex-items

`flex-grow: <number>`

1	2	3	4	5	6
---	---	---	---	---	---

```
.flex-item {  
  flex-grow: 1;  
}  
.flex-item:nth-child(3) {  
  background: red;  
}
```


Flexible Box Layout

Flex-items

`flex-grow: <number>`

1	2	3	4	5	6
---	---	---	---	---	---

```
.flex-item {  
  flex-grow: 1;  
}  
.flex-item:nth-child(3) {  
  background: red;  
  flex-grow: 2;  
}
```

Flexible Box Layout

Flex-items

`flex-shrink: <number>`

1	2	3	4	5	6
---	---	---	---	---	---

```
.flex-item:nth-child(3) {  
  background: red;  
  flex-shrink: 2;  
}
```

Flexible Box Layout

Flex-items

`flex-basis: number|auto`

1	2	3	4	5	6
---	---	---	---	---	---

```
.flex-item:nth-child(3) {  
  background: red;  
  flex-basis: 300px  
}
```

Flexible Box Layout

Flex-items

`flex:` `<'flex-grow'> || <'flex-shrink'> || <'flex-basis'>`

Flexible Box Layout

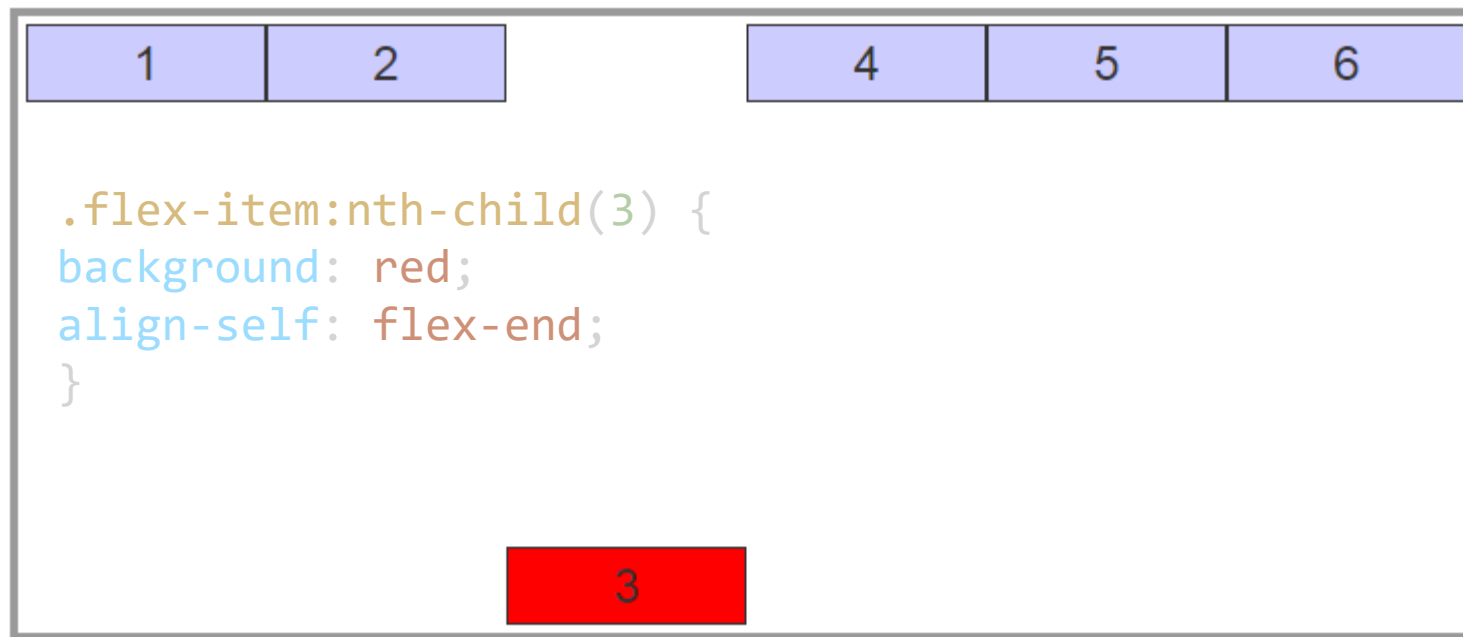
Flex-items

`align-self:` auto | flex-start | flex-end | center | baseline | stretch

Flexible Box Layout

Flex-items

`align-self: flex-end`



Transformacije, tranzicije, animacije

Transformacije

transform

Transformacije

CSS metode transformacije

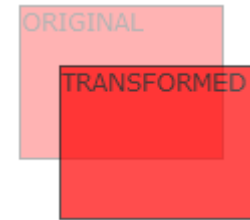
- `translate()`
- `rotate()`
- `scaleX()`
- `scaleY()`
- `scale()`
- `skewX()`
- `skewY()`
- `skew()`

Transformacije

CSS metode transformacije

translate()

```
div {  
  transform: translate(50px, 100px);  
}
```



Transformacije

CSS metode transformacije

rotate()

```
div {  
  transform: rotate(20deg);  
}
```

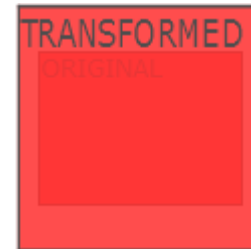


Transformacije

CSS metode transformacije

scale()

```
div {  
  transform: scale(2,3);  
}
```



Transformacije

CSS metode transformacije

scaleX()

```
div {  
  transform: scaleX(2);  
}
```

Transformacije

CSS metode transformacije

scaleY()

```
div {  
  transform: scaleY(3);  
}
```

Transformacije

CSS metode transformacije

skew()

```
div {  
  transform: skew(20deg, 10deg);  
}
```



Transformacije

CSS metode transformacije

skewX()

```
div {  
  transform: skewX(20deg);  
}
```


Transformacije

CSS metode transformacije

skewY()

```
div {  
  transform: skewY(20deg);  
}
```

Transformacije

CSS metode transformacije

```
matrix()
```

```
matrix(a, b, c, d, e, f)
```

Tranzicije

- transition
- transition-delay
- transition-duration
- transition-property
- transition-timing-function

Tranzicije

transition: property duration transition-timing-function delay

Tranzicije

```
div {  
  transition-property: property;  
  transition-duration: duration;  
  transition-timing-function: timing-function;  
  transition-delay: delay  
}
```

```
div{transition: property duration transition-timing-function delay;}
```

Animacije

- `@keyframes`
- `animation`
- `animation-name`
- `animation-duration`
- `animation-delay`
- `animation-iteration-count`
- `animation-direction`
- `animation-timing-function`
- `animation-fill-mode`

Animacije

- `@keyframes`
 - Definira samu animaciju, njezine ključne kadrove i svojstva koja se animiraju
- `animation`
 - Dodjeljuje animaciju određenom elementu i definira kako je animiran

Animacije

```
@keyframes animationname {keyframes-selector {css-styles;}}
```


Animacije

```
@keyframes animacija1 {  
  0% {  
    top: 0px;  
    background: red;  
    width: 100px;  
  }  
  100% {  
    top: 200px;  
    background: yellow;  
    width: 300px;  
  }  
}
```

Animacije

```
@keyframes animacija1 {  
  0% {  
    top: 0px;  
    left: 0px;  
    background: red;  
  }  
  50% {  
    top: 100px;  
    left: 100px;  
    background: yellow;  
  }  
  100% {  
    top: 0px;  
    left: 0px;  
    background: red;  
  }  
}
```

Animacije

`animation:` name duration timing-function delay iteration-count direction fill-mode play-state

Animacije

```
div {  
  animation: animacija1 5s linear 2s infinite alternate;  
}
```

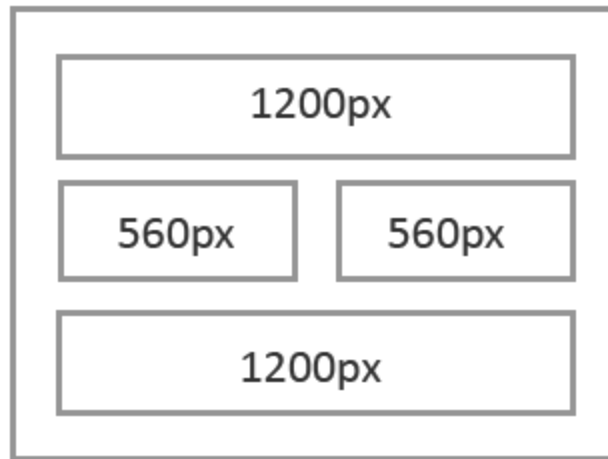
Animacije

```
div {  
  animation-name: animacija1;  
  animation-duration: 5s;  
  animation-timing-function: linear;  
  animation-delay: 2s;  
  animation-iteration-count: infinite;  
  animation-direction: alternate;  
}
```

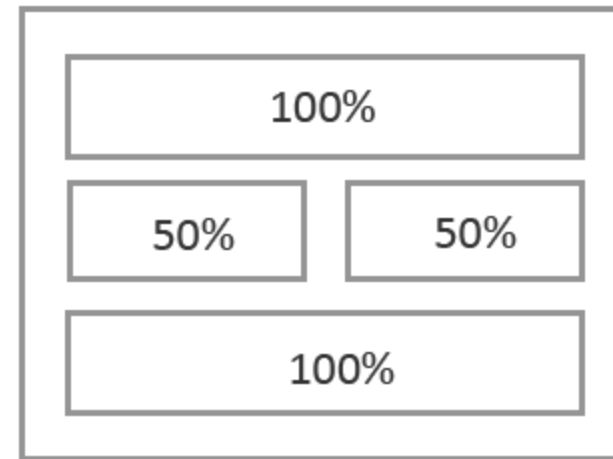
Responzivni dizajn

Fiksni i fluidni *layout*

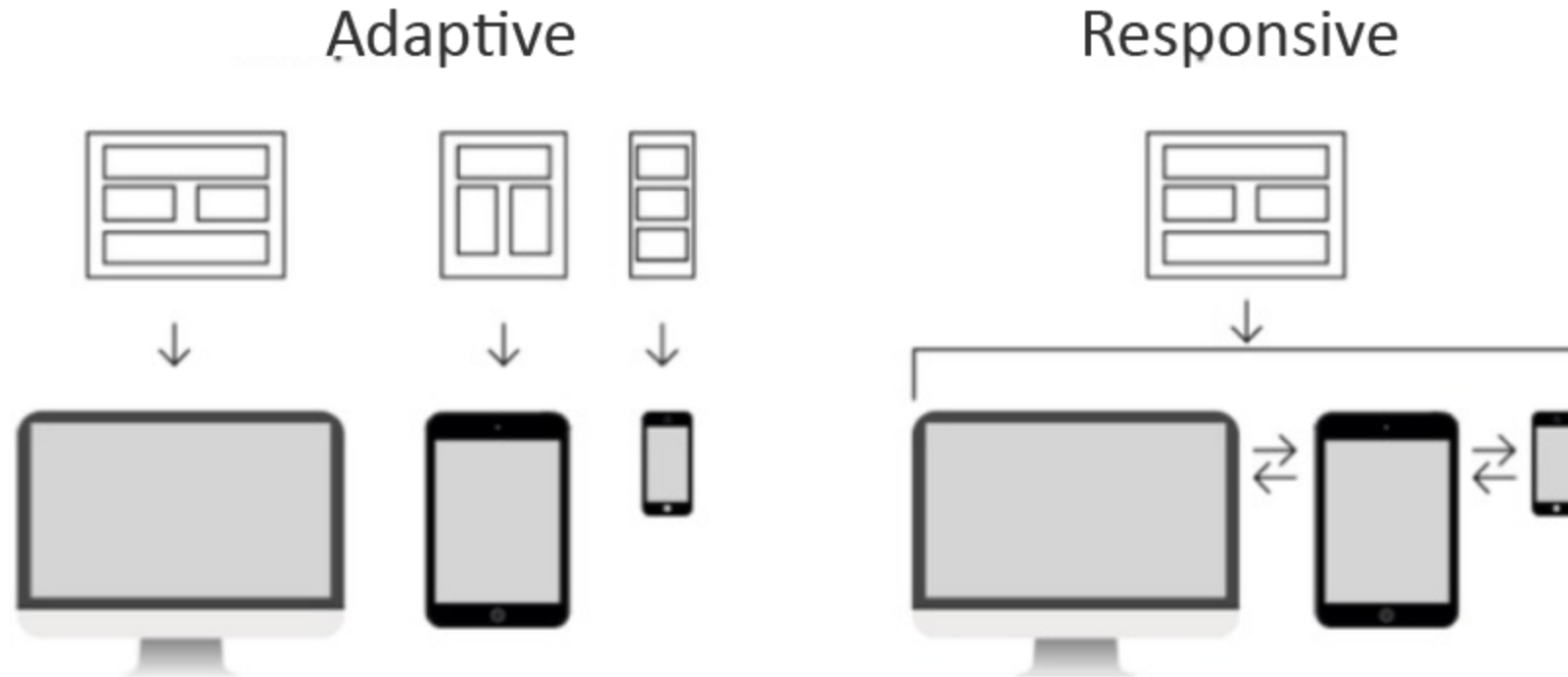
Fixed



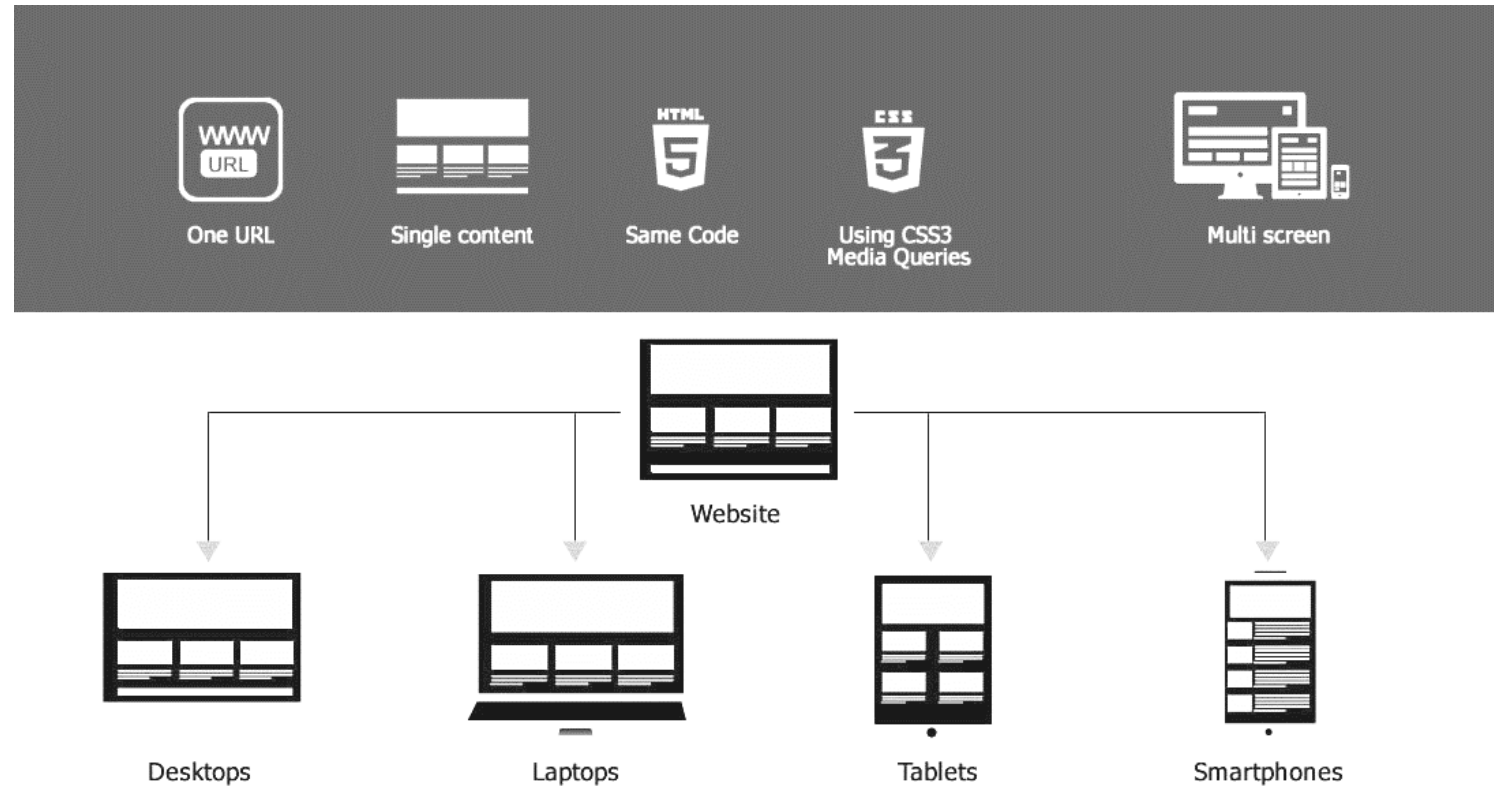
Fluid



Adaptivni i responzivni *layout*



Responzivni dizajn



Media Queries

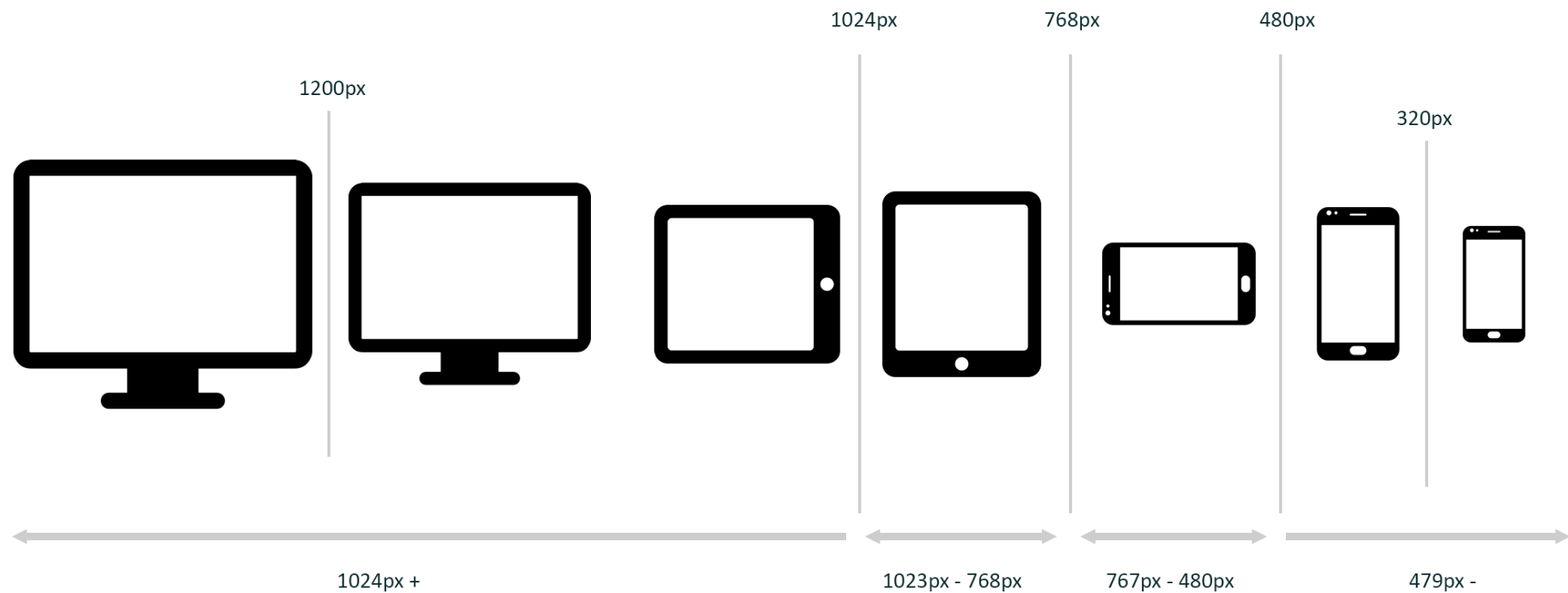


Media Queries

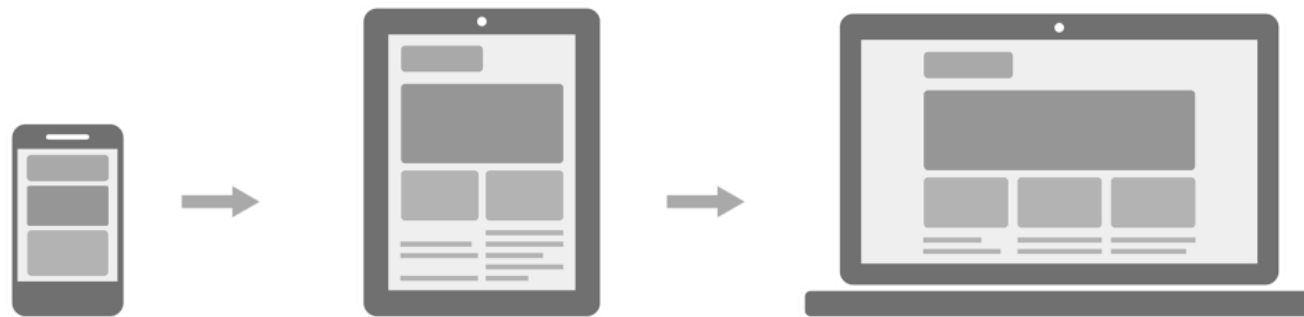
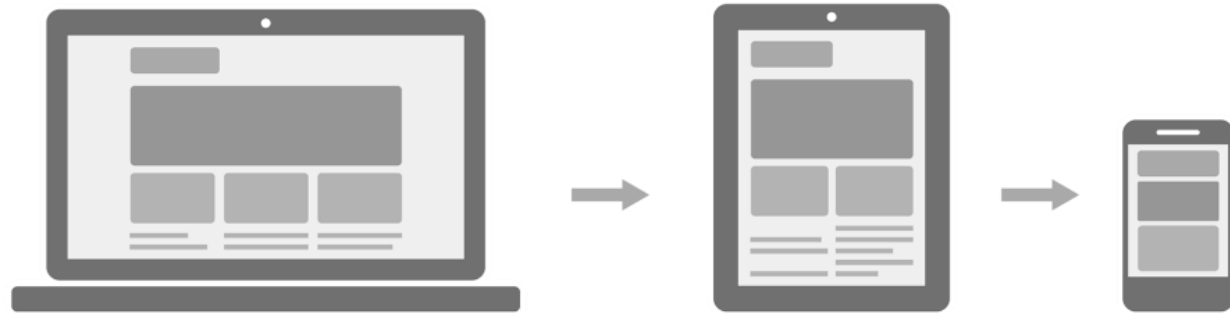
```
@media not|only mediatype and|not|only (media feature) {  
    .my-code { ... }  
}
```

```
@media (media feature) {  
    .my-code { ... }  
}
```

Breakpoints

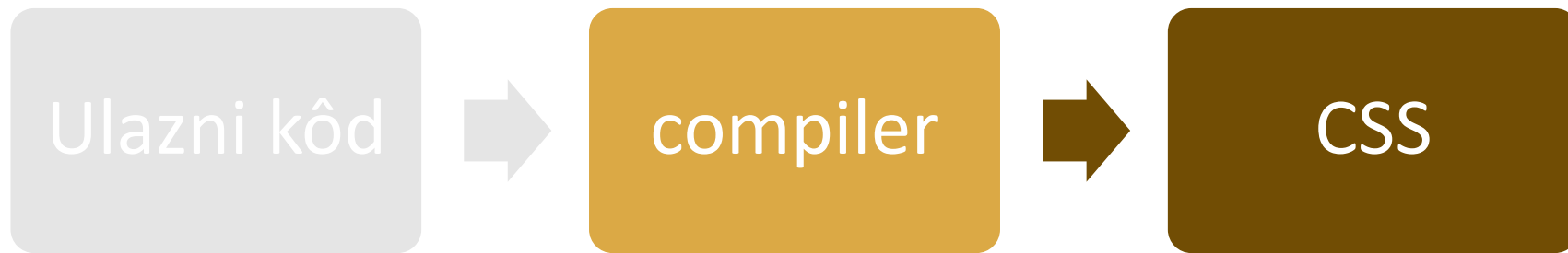


Desktop first i Mobile first



CSS Preprocessori

Što su preprocesori i kako rade?



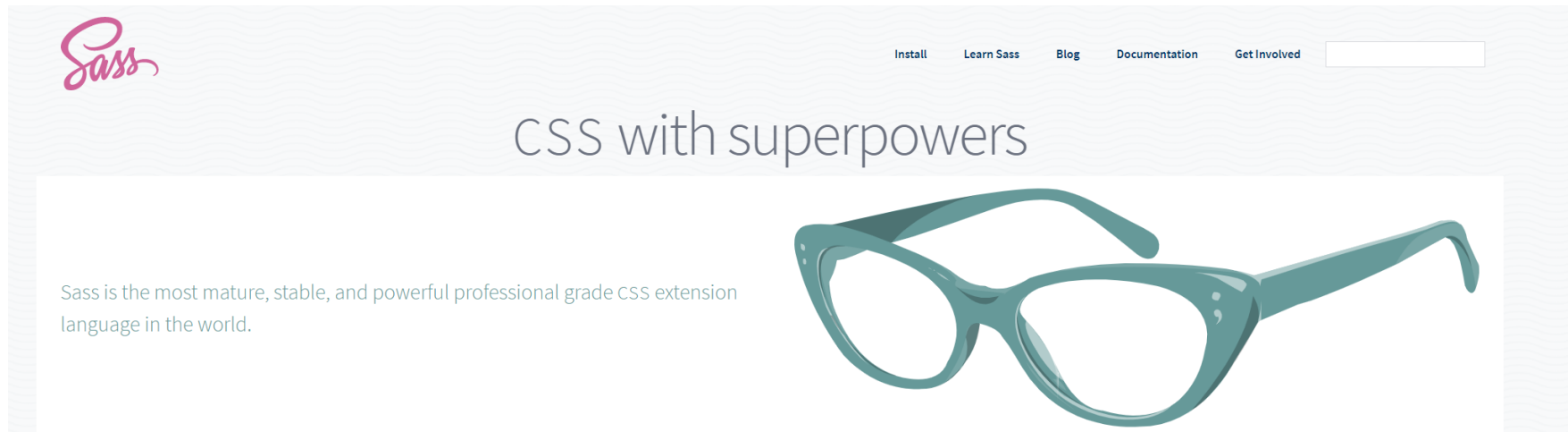
Prednosti korištenja

- više mogućnosti
- čišći i jednostavniji kôd
- veća fleksibilnost pri radu
- *Snippets & Libraries*
- štede vrijeme i trud
- ...

Popularni preprocesori



Sass



<https://sass-lang.com/>

Sass vs. SCSS

`/* CSS */`

```
.content-navigation {  
border-color: #3bbfce;  
color: #2b9eab;  
}
```

```
.border {  
padding: 8px;  
margin: 8px;  
border-color: #3bbfce;  
}
```

`/* SASS */`

```
$blue: #3bbfce  
$margin: 16px
```

```
.content-navigation  
border-color: $blue  
color: darken($blue, 9%)
```

```
.border  
padding: $margin / 2  
margin: $margin / 2  
border-color: $blue
```

`/* SCSS */`

```
$blue: #3bbfce;  
$margin: 16px;
```

```
.content-navigation {  
border-color: $blue;  
color: darken($blue, 9%);  
}
```

```
.border {  
padding: $margin / 2;  
margin: $margin / 2;  
border-color: $blue;  
}
```

Mogućnosti

- *Variables*
- *Nesting*
- *Mixins*
- *Imports*
- *Loops*
- ...

Variable

```
$red: #833;
```

```
body {  
  color: $red;  
}
```

Nesting

```
.markdown-body {  
  p {  
    color: blue;  
  }  
}
```

```
&:hover {  
  color: red;  
}  
}
```

Extend

```
.button {  
  ...  
}  
.push-button {  
  @extend .button;  
}
```

Composing

```
@import './other_sass_file';
```


Mixins

```
@mixin heading-font {  
  font-family: sans-serif;  
  font-weight: bold;  
}  
h1 {  
  @include heading-font;  
}
```

Mixins

```
@mixin font-size($n) {  
  font-size: $n * 1.2em;  
}  
body {  
  @include font-size(2);  
}
```

Mixins

```
@mixin pad($n: 10px) {  
  padding: $n;  
}  
body {  
  @include pad(15px);  
}
```

For loops

```
@for $i from 1 through 4 {  
  .item-#{$i} { left: 20px * $i; }  
}  
mixin pad($n: 10px) {  
  padding: $n;  
}  
body {  
  @include pad(15px);  
}
```

Each loops

```
$menu-items: home about services contact;
```

```
@each $item in $menu-items {  
  .photo-#{ $item } {  
    background: url('images/#{ $item }.jpg');  
  }  
}
```

Conditionals

```
@if $position == 'left' {  
  position: absolute;  
  left: 0;  
}  
@else {  
  position: static;  
}
```

Maps

```
$map: (key1: value1, key2: value2, key3: value3);
```

```
map-get($map, key1)
```

Hvala na
pažnji!

