

# CSS

Novi Sad, 2017

# CSS - OSNOVNI POJMOVI

- CSS (Cascading Style Sheets) je jezik za opisivanje izgleda dokumenata napisanih u markup jezicima kao što su XML ili HTML.
- Pomoću CSS-a se mogu odrediti svi bitni elementi izgleda stranice
- U verziji 3 omogućava i različite efekte (transformacije) - ukidajući potrebu za komplikovanim animacijama, ili plug-inovima

# CSS - OSNOVNI POJMOVI

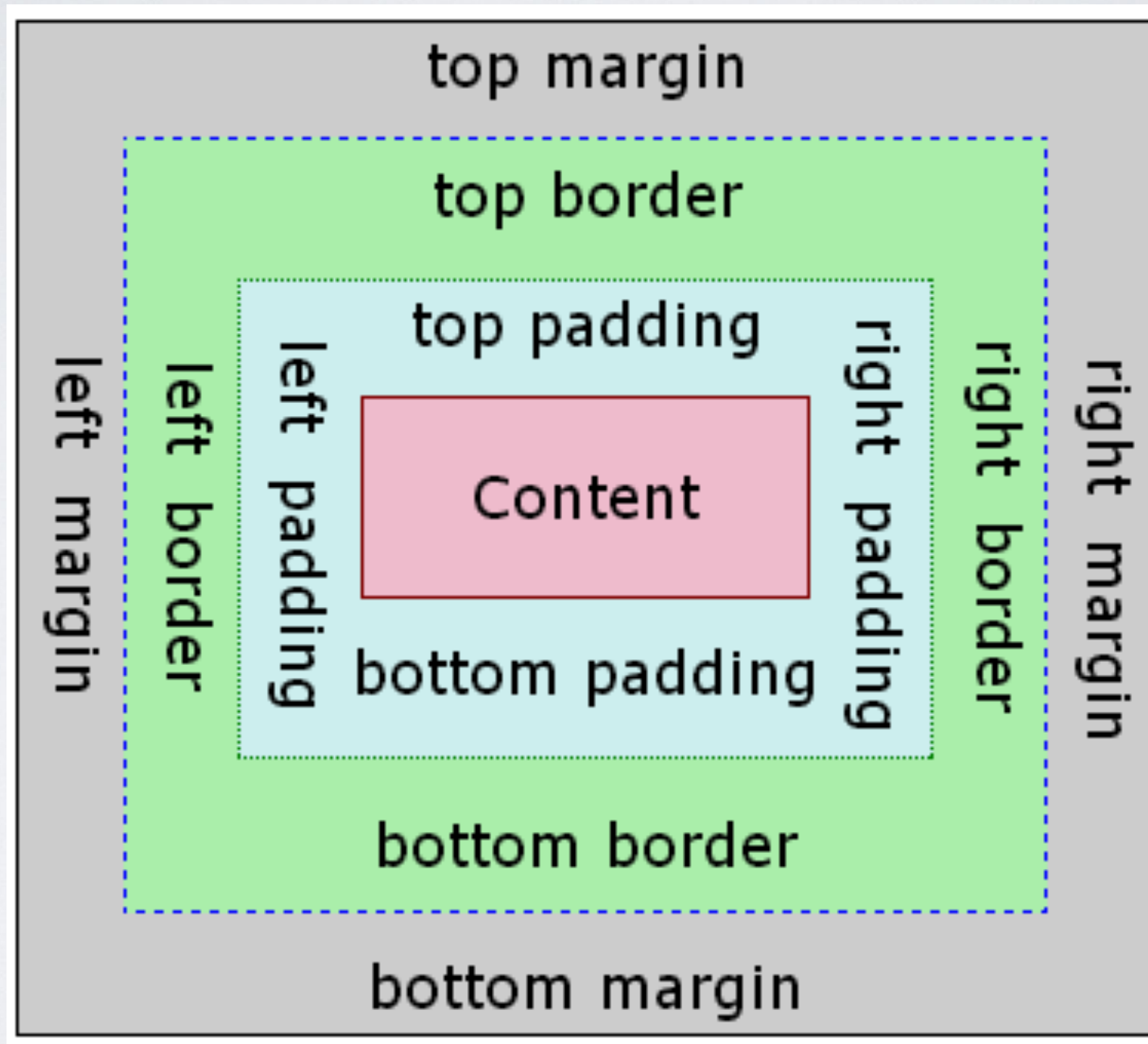
- Isti CSS može se koristiti za više od jedne stranice, što znači da stil celog veb sajta može biti prilagođen bez menjanja svake stranice posebno.
- Primena konzistentnih stilova na svim stranicama pomaže unificiranju vizuelnog identiteta web stranica.
- *Browseri* imaju predefinani skup stilova koje primenjuju ako korisnik ne definiše svoj stil za neke sadržaje.



# CSS - DODAVANJE U DOKUMENT

- putem eksternog dokumenta sa css stilovima  
`<link rel="stylesheet"  
href="putanja_do_css_fajla" />`
- internom deklaracijom stilova u style tagu u head tagu
- *inline* specifikacijom u *style* atributu elementa

# BOX MODEL

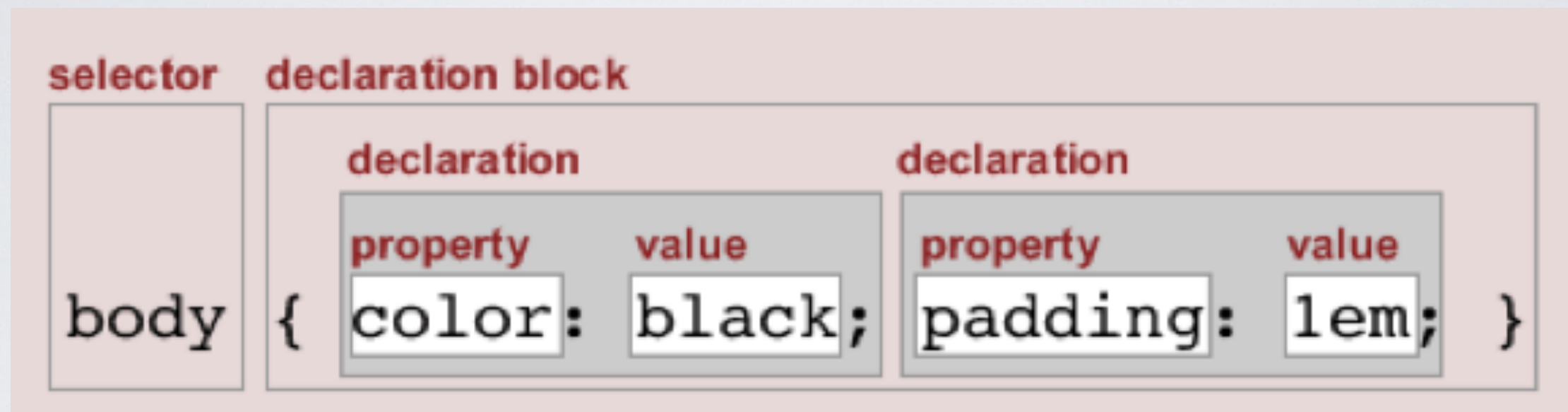


# CSS - SINTAKSA

- Skup CSS pravila se sastoji od *selektora* i *bloka deklaracija*
- **Selektor** definiše na koji element (ili elemente) se primenjuju pravila navedena u bloku deklaracija
- Svaka deklaracija sastoji se od naziva atributa (property-ja) i vrednosti na koju se postavlja u formatu  
*property: value*



# CSS - SINTAKSA



# CSS SELEKTORI

- Određuje na koje XML ili HTML elemente se primenjuje CSS pravilo. Selektor govori *browseru* šta treba da stilizuje (koje elemente treba da pronađe u stablu dokumenta).
- postoje tri osnovne vrste selektora
  - element selektori
  - **id** selektori
  - **class** selector



# CSS

## ELEMENT SELEKTORI

- Element selektori pronalaze sve elemente čiji naziv taga odgovara onom koji je naveden u selektoru
- primer:

```
p{  
    background: #363636;  
    padding: 10px;  
}
```

# CSS

## ID SELEKTORI

- ID selektori pronalaze jedan element koji sadrži *id* atribut čiji vrednost odgovara onoj koja je navedena u selektoru.
- ID selektor sadrži znak # na početku selektora
- primer: `<div id="container">...</div>`  
`#container{`  
    `background: #363636;`  
    `padding: 10px;`  
`}`

# CSS

## CLASS SELEKTORI

- class selektori pronalaze elemente koji sadrži *class* atribut koji sadrži naziv klase koji odgovara onom u selektoru.
- CLASS selektor sadrži znak . na početku selektora

- primer:

```
<p class="center no-margin">...</p>
```

```
<div class="no-margin">...</div>
```

```
.center{  
    text-align="center";  
}
```

```
.no-margin{  
    margin: 0;  
}
```



# CSS

## UNIVERZALNI SELEKTOR

- poseban selektor koji označava da se stil odnosi na sve elemente

- primer:

```
* {  
    font-family: Roboto, Verdana, sans-serif;  
}
```

```
div.poem *{  
    font-family: serif;  
}
```

# GRUPISANJE SELEKTORA

- Kada postoje CSS selektori sa istim definicijama stilova, radi konciznosti (i verovatne potrebe da se promena propagira) pogodno je selektore grupisati
- primer:  

```
h1, h2, p.center{  
    text-align="center";  
}
```



# SELEKTORI ZA POTOMKE (PODREĐENE ELEMENTE)

- navode se kada je potrebno da se stil primeni na željene potomke određenog elementa.
- Postoje dva oblika:
  - ul li{...} - selektuje sve *li* elemente koji su potomci ul elementa, bez obzira da li su direktni potomci ili dublje ugneždeni
  - ul >li{...} - selektuje samo one *li* elemente koji su direktni potomci ul elementa



# SELEKTORI - PSEUDO KLAZE

- omogućavaju da se selektuju elementi koji zadovoljavaju neko posebno *stanje*
- `selector:pseudo-class{...}`  
Primer:  
`a:hover{...}`
- neke od često korišćenih u poslednje vreme su:  
`:first-child, :last-child, :nth-child, :empty...`

# KOMBINOVANJE SELEKTORA

- moguće je kombinovati selektore kako bi se selektovali proizvoljni specifični elementi u dokumentu
- primer:  
`div.container div.row > p:nth-child(2n)`

# JEDINICE MERA U CSS-U

- razlikujemo apsolutne i relativne
- apsolutne izražavaju veličinu u zadatim mernim jedinicama, najčešće u pikselima (px), ali je moguće koristiti i cm, mm, in, pc, pt
- relativne jedinice mera izražavaju veličine u odnosu na neke referentne vrednosti



# RELATIVNE JEDINICE MERA U CSS-U

- omogućavaju preslaganje sadržaja stranice u skladu sa veličinom prikaza
- najčešće se koristi % - 100% se najčešće odnosi na veličinu roditeljskog elementa.
  - Ako se radi o tekstu 100% je podrazumevana veličina fonta za dati prikaz (najčešće ako zoom strane nije menjan 16px).
- em - jedinica mere kod koje 1em odgovara visini “em” razmaka trenutno postavljene veličine fonta
- ex - jedinica mere kod koje se gleda visina malog slova x tekućeg fonta
- vh - 1% trenutne visine *viewport*-a
- vw - 1% trenutne širine *viewport*-a
- vmin - 1vh ili 1vw šta god je manje

# FORMATIRANJE TEKSTA

- najčešće stvari za formatiranje teksta su oblik slova, veličina slova, poravnanje, “težina”, boja, boja pozadine...
- U CSS stilovima se podešavaju vrednosti sledećih atributa
  - color
  - direction
  - letter-spacing
  - line-height
  - text-align
  - text-decoration
  - text-indent
  - text-shadow
  - text-transform
  - unicode-bidi
  - vertical-align
  - white-space
  - word-spacing
  - font-family
  - font-style
  - font-weight



# FORMATIRANJE BLOKOVA

- opet se vraćamo na box model
- podešavaju se sledeći atributi
  - margin - moguće je podešavati sve ili pojedinačno
  - padding - moguće je podešavati sve ili pojedinačno
  - border - debljina ivice box-a
  - width, height - dato u apsolutnim ili relativnim jedinicama
  - min-width, max-width, min-height, max-height



# FORMATIRANJE BLOKOVA DISPLAY

- *display*: atribut koji govori kako se blok prikazuje u odnosu na ostatak sadržaja
- vrednosti:
  - block
  - inline-block
  - none - blok više ne zauzima prostor u stranici kao da ni ne postoji (za razliku od visibility:hidden, kada se ne vidi ali “rezerviše” prostor)

# FORMATIRANJE BLOKOVA

## POSITION

- *position*: atribut koji govori kako se blok pozicionira
- vrednosti:
  - static - podrazumevano
  - relative
  - absolute
  - fixed

# FLOAT | CLEAR

- *float*: omogućava da se element “izdvoji” iz normalnog rasporeda

```
img{  
  float: left;  
  margin: 5px 20px 5px 0;  
}
```

- *clear* kontroliše ponašanje elemenata oko (posle) float elemenata



# COLUMN COUNT

- *column-count* omogućava da se sadržaj slaže u željeni broj kolona

```
.clanci {
```

```
  -webkit-column-count: 2; /* Chrome, Safari, Opera */
```

```
  -moz-column-count: 2; /* Firefox */
```

```
  column-count: 2;
```

```
}
```

# SELEKTORI ZA ATRIBUTE

- *element[attr]* - omogućava da se selektuju svi elementi koji imaju atribut *attr* bez obzira koja je njegova vrednost
- *element[attr="value"]* - omogućava da se selektuju svi elementi koji imaju atribut *attr*, ali da je pri tome njegova vrednost baš *value*
- često se koristi kod stilizovanja formi



# CSS 3 NOVINE

- *boje* - sada ih je moguće zadati i kao rgba, hsl, i hsla vrednosti
- *text-overflow* - kontroliše kako se prikazuje da tekst ne staje u zadati blok. Moguće vrednsoti su *clip* i *ellipsis*
- mogućnost linkovanja na eksterne fontove

```
@font-face {
```

```
    font-family: mojnaziv;
```

```
    src: url(superfont_light.woff);
```

```
}
```



# CSS 3 NOVINE

- *border-radius* - omogućava da se direktno CSS-om zadaju zaobljene ivice elementima - ovo omogućava i kreiranje krugova čisto css-om
  - kao što je moguće postaviti background-image property, sada je moguće postaviti i border-image (obično neki pattern kojim se iscrtava stilizovana linija)
- **box-shadow, text-shadow** - omogućava da se bilo na blok ili samo na text postavi senka
  - text-shadow: 2px 2px 5px red;
  - prvi broj je x offset, drugi y offset. opcioni treći broj je radius rasipanja senke, a četvrti je boja

# CSS 3 TRANSFORMACIJE

- 2D

- *translate()*
- *rotate()*
- *scale()*
- *skewX()*
- *skewY()*
- *matrix()*

- 3D

- *matrix3d*
- *(n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n,n)*
- *translate3d(x,y,z)*
- *translateX(x)*
- *translateY(y)*
- *translateZ(z)*
- *scale3d(x,y,z)*
- *scaleX(x)*
- *scaleY(y)*
- *scaleZ(z)*
- *rotate3d(x,y,z,angle)*
- *rotateX(angle)*
- *rotateY(angle)*
- *rotateZ(angle)*
- *perspective(n)*



# CSS 3 TRANZICIJE

- *transition-delay* - vremenski interval u sekundama pre nego tranzicija počne
- *transition-duration* - trajanje tranzicije u sekundama
- *transition-property* - govori na koju osobinu objekta se tranzicija primenjuje
- *transition-timing-function*:
  - *ease, linear, ease-in, ease-out, ease-in-out, cubic-bezier(n,n,n,n)*
- *transition* - omogućava da se sve podesi u jednom propertyju



# CSS 3 MEDIA QUERIES

- omogućavaju da se za iste elemente definišu potpuno različiti stilovi u zavisnosti od konteksta u kome se koriste
- sastoje se od @media

```
@media not | only mediatype and (expressions) {  
    CSS-Code;  
}
```

# CSS 3 MEDIA QUERIES

- Zasniva se na konceptima da se za različite tipove medija može postići bolje korisničko iskustvo ako se prikaz prilagodi
- mediatype
  - najčešći: all, print, screen, speech



# CSS 3 MEDIA QUERIES

- izrazima se utvrđuje na osnovu koje osobine medija se vrši izbor
- najčešće se koriste height, width, min-width, max-width, min-height, max-height, aspect-ratio, resolution, monochrome, color-index...
- lista svih  
[http://www.w3schools.com/cssref/css3\\_pr\\_mediaquery.asp](http://www.w3schools.com/cssref/css3_pr_mediaquery.asp)



# CSS 3 MEDIA QUERIES

- Responsive design se postiže najčešće na osnovu breakpointa - tipičnih veličina ciljanih uređaja

- Mobile

@media only screen and (min-device-width : 320px) and (max-device-width : 480px) {...}

- Tablets

@media only screen and (min-device-width : 768px) and (max-device-width : 1024px) {...}

- Mobile Portrait

@media only screen and (max-device-width : 320px) {...}

- Mobile Landscape

@media only screen and (min-device-width : 321px) { ...}

- Tablet Portrait

@media only screen and (min-device-width : 768px) and (max-device-width : 1024px) and (orientation : portrait) {...}

- Tablet Landscape

@media only screen and (min-device-width : 768px) and (max-device-width : 1024px) and (orientation : landscape) {...}