

INTERNET TEHNOLOGIJE I VEB DIZAJN

Korišćenje CSS-a na Veb stranici

predavač: dr Miloš Antonijević

Pojam i osnovni koncepti

Definicija:

- **CSS** je jezik koji se koristi za stilizovanje HTML elemenata, odnosno opisivanje načina prikaza HTML elemenata.
- Primenom CSS odvajamo prezentacioni stil stranice od informacija na Veb stranici.
- Prednosti CSS-a su:
 - Tipografija (veličina fonta, razmak između linija, razmak između slova) i raspored elemenata na stranici (margine, pozicije) se može bolje kontrolisati.
 - Stil je odvojen od strukture pa se može čuvati nezavisno od fajla sa HTML kodom.
 - Isti stil se može primeniti na više Veb stranica referenciranjem na CSS fajl.

Povezivanje CSS-a i HTML-a

- CSS se može koristiti u HTML kodu na tri načina:
 - Primenom inline stilizovanja – kada se CSS kod piše u telu Veb stranice u vidu **style** atributa u otvarajućem tagu određenog elementa.
 - Internim definisanjem stilova – kada se CSS stilovi definišu u samoj HTML stranici u okviru **head** sekcije u okviru **style** elementa.
 - Eksternim definisanjem stilova – kada se CSS stilovi nalaze u nezavisnom css fajlu koji se referencira u HTML kodu.
- Kada se koristi inline stilizovanje, CSS kod se odnosi samo na element u čijem se tagu nalazi.
- U slučaju internog i eksternog definisanja stilova, za označavanje elemenata na koje se odnosi CSS kod, koriste se **selektori**.
- Komentari u CSS se navode između /* i */.

Primeri povezivanja CSS-a i HTML-a

Inline:

```
<p style="color: red">Ovo je tekst crvene boje</p>
```

Internal:

```
<head>  
    <style>  
        p { color: red }  
    </style>  
</head>
```

External:

```
<head>  
    <link rel="stylesheet" href="styles.css">  
</head>  
  
styles.css  
<style>  
    p { color: red }  
</style>
```

Povezivanje CSS-a i HTML-a (2)

- Nedostatak **inline** definisanja stilova je neophodnost dupliranja koda u tagovima elemenata koji treba da sadrže iste stilove.
- Ovaj problem se rešava internim pisanjem CSS koda u okviru zaglavlja Veb stranice (**internal**). Kako bi se naznačilo na koji element se odnose navedeni stilovi koriste se **selektori** i **deklaracije**.
- Nedostatak internog pisanja CSS koda jeste neophodnost dupliranja stilova koji se žele koristiti na više HTML stranica.
- Ovaj problem se rešava smeštanjem CSS koda u eksterne fajlove koji se navode u svakoj Veb stranici koja koristi identične stilove (**external**).
- **Selektori** predstavljaju imena HTML elemenata (i opcionalno atributa), imena klasa ili id oznake elemenata na koji se odnose odgovarajući stilovi navedeni u okviru **deklaracija**.

Selektori

- Selektori u vidu naziva elemenata:

```
p,h1 { color: red }
```

- Selektori u vidu naziva klasa:

```
.imeklase #dete_element { background-color: red }
```

- Selektori u vidu id naziva:

```
#id_elementa { color: red }
```

- Selektori u vidu naziva atributa elemenata:

```
input[type=text] { border: 1px solid red }
```

Kaskadna pravila CSS-a

- Kaskadna pravila ili pravila prvenstva se odnose na redosled primene stilova u odnosu na mesto njihovog definisanja.
- Na ovaj način se omogućava definisanje opštijih stilova na eksterni način koja se po potrebi mogu „nadjačati“ njihovim redefinisanjem internom ili inline primenom.
- Eksterno definisani stilovi se prvo primenjuju.
- Ukoliko postoje interno definisani stilovi koji se odnose na iste selektore, oni će „nadjačati“ stilove definisane eksterno.
- Ukoliko postoje inline definisani stilovi koji se odnose na iste selektore, oni se primenjuju *preko* interno definisanih stilova.
- HTML tagovi ili atributi nadjačavaju sve stilove. Npr. tag će nadjačati inline formatiranje fonta.
- Ugnježdeni elementi (deca) nasleđuju stilove roditelja.

Korišćenje boja u CSS-u

- Boje se u CSS mogu konfigurisati na četiri načina:
 1. Korišćenjem imena boje (https://www.w3schools.com/colors/colors_names.asp)
 2. Korišćenjem heksadecimalne vrednosti boje
 3. Navođenjem RGB decimalnih vrednosti (rgb, rgba(255, 99, 71, 0.5))
 4. Upotrebom HSL (Hue, Saturation, Lightness) notacije koja je nova u CSS3 verziji (https://www.w3schools.com/css/css_colors_hsl.asp).
- Ukoliko želimo da definišemo različite stilove u okviru istog elementa možemo koristiti element **span**, koji definiše logičku sekciju stranice koja nije „fizički“ odvojena od ostalih celina.

```
<p style="color: red">Ovo <span style="color: blue">je</span> <span style="color: green"> raznobojni</span> tekst.</p>
```


Definisanje pozadine u CSS-u

- **background-color** – definiše boju pozadine
`background-color: blue; opacity: 0.5;`
`background: rgba(0, 0, 255, 0.3)`
- **background-image** – definiše pozadinsku sliku
`background-image: url("pozadina.jpg");`
- **background-repeat** – određuje da li se pozadinska slika ponavlja po x ili y osi
`background-repeat: repeat-x;`
- **background-attachment** – određuje da li je pozadinska slika statična ili ne
`background-attachment: scroll;`
- **background-position** – određuje položaj pozadinske slike
`background-position: right top;`
- **background** – definiše karakteristike pozadine u jednoj deklaraciji
`background: #ffffff url("pozadina.png") no-repeat scroll left top;`

Definisanje ivica u CSS-u

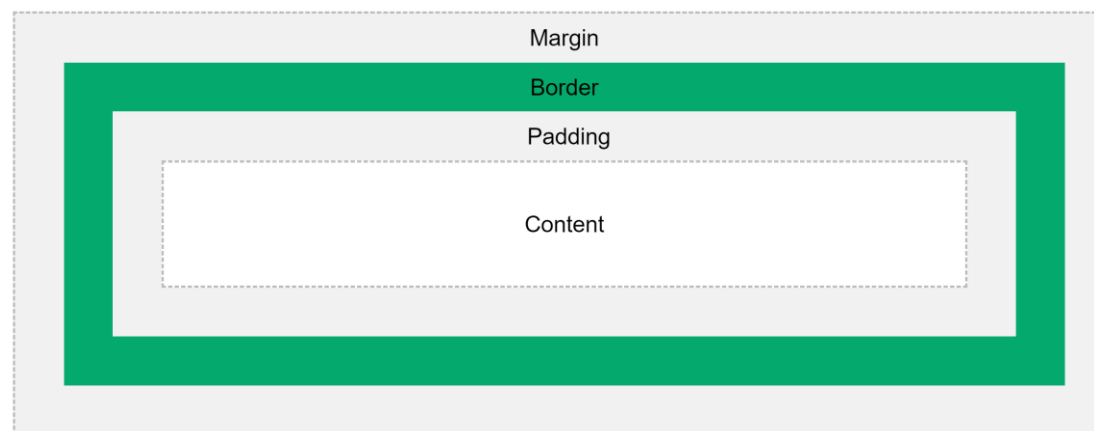
- **border-top(right,bottom,left)** – definiše gornju (desnu, donju, levu) ivicu
`border-top: 2px dotted red;`
- **border** – definiše karakteristike svih ivica u jednoj deklaraciji
`border: 1px solid black;`
- **border-top(right,bottom,left)-radius** – određuje zakrivljenost gornje (desne, donje, leve) ivice
`border-top-radius: 5px;`
- **border-radius** – određuje zakrivljenost svih ivica
`border: 2px solid black;`
`border-radius: 4px`

Definisanje margina i uvlačenja u CSS-u

- Margine predstavljaju prostor oko elemenata, izvan definisanih ivica.
- Uvlačenja predstavljaju prostor unutar sadržaja elementa, odnosno definisanih ivica.
- **margin-top(right,bottom,left)** – definiše gornju (desnu, donju, levu) marginu
`margin-top: 10px;`
- **margin** – definiše karakteristike svih margina u redosledu top, right, bottom, left
`margin: 5px 4px 10px 15px;`
- **padding-top(right,bottom,left)** – određuje uvlačenje sa gornje (desne, donje, leve) strane elementa
`padding-top: 5px;`
- **padding** – određuje uvlačenje sa svih strana elementa u redosledu top, right, bottom, left
`padding: 5px 10px;`

Definisanje širine i visine u CSS-u

- Širina i visina se mogu definisati u formi:
 - Dimenzija u pikselima
 - Dimenzija u procentima od veličine roditelja
- **width** – definiše širinu elementa
`width: 500px;`
`width: 60%;`
- **height** – definiše visinu elementa
`height: 300px;`
`height: 100%;`
- **max-width(height)** – određuje najveću dozvoljenu širinu (visinu) elementa
`width: 100px; max-width: 500px;`



Definisanje osobina fonta u CSS-u

- U CSS-u postoji 5 generičkih tipova fonta: Serif, Sans-serif, Monospace, Cursive i Fantasy.
- Kada se definiše font teksta, poželjno je navesti nekoliko imena kao rezervne (backup) varijante. Obično se započinje imenom željenog fonta, a završava se imenom generičkog fonta
- Ukoliko ime fonta sadrži više od jedne reči, piše se pod navodnicima.
- Fontovi koji su podrazumevano instalirani na najvećem broju uređaja i browser-a spadaju u grupu Web safe (bezbednih za Veb) fontova. Npr. Arial (sans-serif), Verdana (sans-serif), Helvetica (sans-serif), Tahoma (sans-serif), Times New Roman (serif), Georgia (serif), Courier New (monospace).
- **font-family** – određuje tip fonta
font-family: "Times New Roman", serif;

Definisanje osobina fonta u CSS-u (2)

- **font-style** – definiše stil fonta (normal, italic)
font-style: italic;
- **font-weight** – definiše debljinu fonta (normal, bold, bolder, lighter, 100-900)
font-weight: bold;
font-weight: 400;
- Pored generičkih fontova, moguće je koristiti fontove koji se nalaze na lokalnom računaru i google fontove.

```
<head>
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Play">
  <style>
    body {
      font-family: "Play", sans-serif;
    }
  </style>
</head>
```

Definisanje linkova u CSS-u

- Pored standardnih osobina teksta (boje, fonta, pozadine itd.), u slučaju linkova moguće je definisati stilove u zavisnosti od stanja u kome se link nalazi.
- Moguća stanja linka su:
 - **a:link** – „normal“, neposećen link
 - **a:visited** – link koga je korisnik prethodno posetio
 - **a:hover** – link kada korisnik pređe kursorom miša preko njega
 - **a:active** – link u trenutku kada je korisnik kliknuo na njega

```
a:link { color: red; }
```

```
a:visited { color: green; }
```

```
a:hover { color: cyan; }
```

```
a:active { color: blue; }
```

Pozicioniranje elemenata u CSS-u

- Pozicioniranje elemanta u okviru Veb stranice i određivanje njihovog međusobnog položaja obavlja se korišćenjem pet metoda:
 - static – podrazumevano, elementi se postavljaju redom kako su navedeni na stranici
 - relative – isto kao static, sa razlikom korišćenja svojstava top, right, bottom i left za određivanje pozicije u odnosu na očekivanu static poziciju
 - fixed – relativno u odnosu na ivice browser-a
 - absolute – relativno u odnosu na roditelja koji ima pozicioniranje različito od static ili html elementa
 - stiky – porazumevano relativno, a fiksirano u slučaju kad korisnik skroluje stranicu
- Nakon navođenja odgovarajuće metode, za tačno definisanje pozicije elemenata mogu se koristiti svojstva **top**, **right**, **bottom** i **left**.

Primer pozicioniranja u CSS-u

```
<head>
  <style>
    .container { position: relative; }

    .topleft { position: absolute; top: 8px; left: 16px; font-size: 18px; }

    img { width: 100%; height: auto; opacity: 0.3; }
  </style>
</head>
<body>
  <div class="container">
    
    <div class="topleft">Top Left</div>
  </div>
</body>
</html>
```

CSS kombinatori

- Kombinatori se koriste za opisivanje odnosa između selektora
- Postoji četiri vrste kombinatora:
 - **space** – selektor potpomaka
 - **>** - selektor dece
 - **+** - selektor narednog (susednog) elementa koji ima istog roditelja
 - **~** - selektor svih narednih elemenata sa istim roditeljem

```
div p { color: red };
```

```
div > p { color: red };
```

```
div + p { color: red };
```

```
div ~ p { color: red };
```

PITANJA?