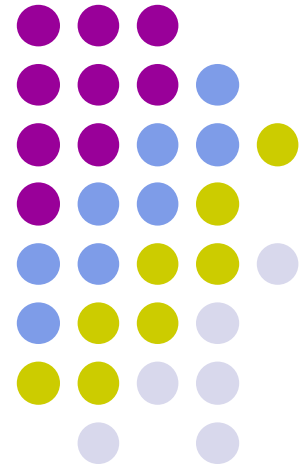
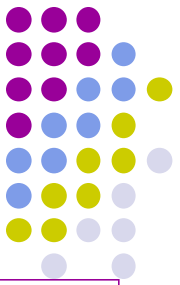


Tehnologije za Web

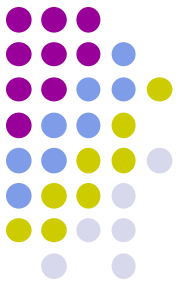


HTML, CSS i JavaScript



- ❑ HTML, CSS i JavaScript su jezici koji se koriste pri pravljenju interaktivnih web sajtova.
- ❑ Ovi jezici dopunjuju jedan drugog, i zbog toga se često koriste zajedno.
- ❑ **HTML**-om se definiše osnovni sadržaj i struktura web stranice.
- ❑ **CSS** služi za vizuelno sređivanje web stranice. Njim se ubacuje dizajn u web stranicu.
- ❑ **JavaScript** omogućava interaktivnost sa korisnicima, dinamičnost, kao i realizaciju drugih komplikovanijih klijentskih funkcionalnosti.

HTML, CSS i JavaScript (2)



HTML

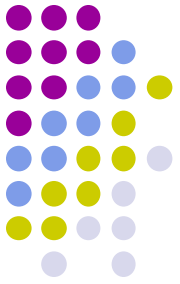


CSS



JavaScript

Liste



- Postoje tri vrste lista koje su definisane u HTML-u: unordered, ordered i description liste.

- Prva stavka
- Druga stavka
- Treća stavka

Unordered

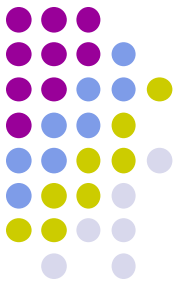
1. Prva stavka
2. Druga stavka
3. Treća stavka

Ordered

Prva stavka
Opis prve stavke
Druga stavka
Opis druge stavke
Treća stavka
Opis treće stavke

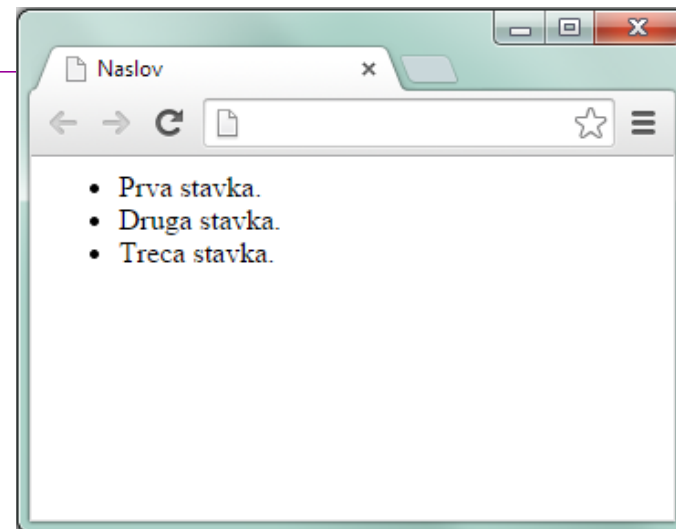
Description

Liste (2)



- ❑ Unordered lista se definiše elementom **ul** (unordered list).
- ❑ Element **ul** se sastoji od **li** (list item) elemenata, pri čemu svaki **li** element predstavlja jednu stavku liste.

```
<ul>
  <li>Prva stavka.</li>
  <li>Druga stavka.</li>
  <li>Treca stavka.</li>
</ul>
```

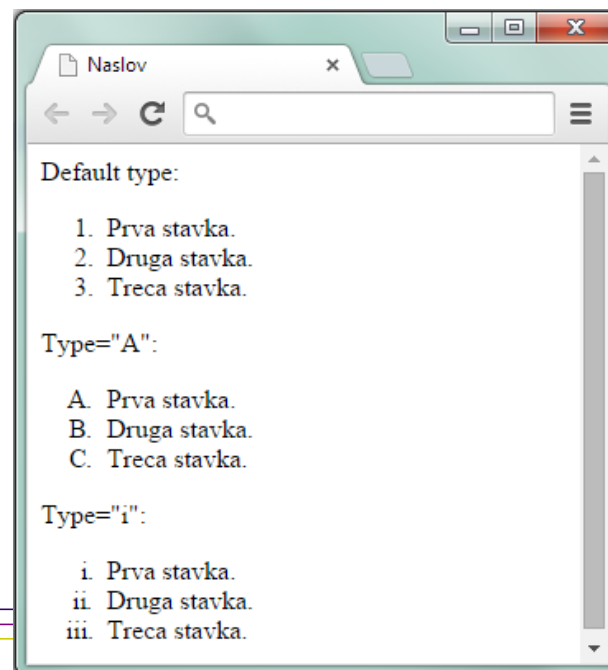


Liste (3)

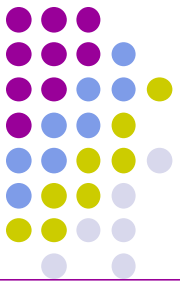


- ❑ Ordered lista se definiše elementom **ol** (ordered list).
- ❑ Stavke u listi se definišu na isti način kao i u unordered listama – koriste se **li** elementi.
- ❑ Moguće je definisati tip liste uz pomoć **type** atributa. Vrednosti koje ovaj atribut može imati su: 1, A, a, I, i.

```
<p>Default type:</p>
<ol>
  <li>Prva stavka.</li>
  . . .
<p>Type="A":</p>
<ol type="A">
  <li>Prva stavka.</li>
  . . .
<p>Type="i":</p>
<ol type="i">
  <li>Prva stavka.</li>
  . . .
```

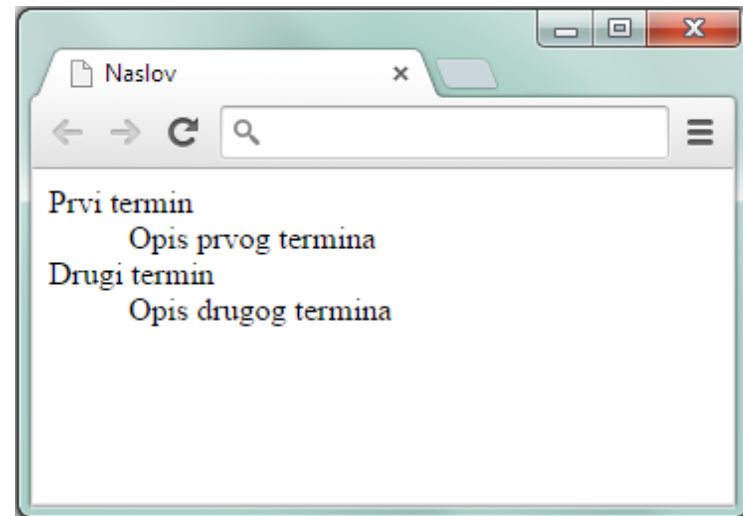


Liste (4)



- ❑ Description lista se definiše elementom **dl**.
- ❑ Unutar elementa **dl** nalaze se dva tipa podelemenata:
 - ❖ Element **dt** – unutar njega se navodi termin.
 - ❖ Element **dd** – unutar njega se navodi opis ili definicija termina.

```
<dl>  
  <dt>Prvi termin</dt>  
  <dd>Opis prvog termina</dd>  
  <dt>Drugi termin</dt>  
  <dd>Opis drugog termina</dd>  
</dl>
```

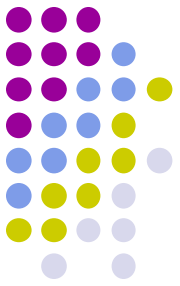


Forme



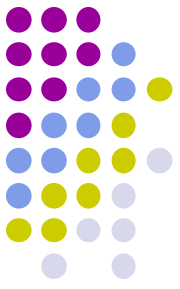
- ❑ Forme se koriste za prikupljanje podataka od korisnika.
- ❑ Forma se definiše elementom **form**.
- ❑ Unutar forme se između ostalog nalaze i **input** elementi, koji su ujedno i najvažniji deo formi.
- ❑ Input elementi imaju razne varijacije, koje su definisane vrednošću atributa **type**. Bitnije vrednosti koje ovaj atribut može imati su:
 - ❖ **text** – Definiše regularni tekstualni input.
 - ❖ **radio** – Definiše radio button-e.
 - ❖ **submit** – Definiše submit dugme kojim se podaci iz forme šalju na obradu.

Forme – “text” input



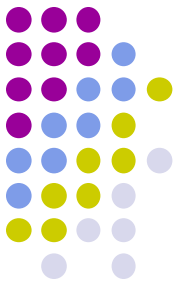
- ❑ Tekstualni input podrazumeva prazno polje za jednu liniju teksta.
- ❑ Definiše se na sledeći način:
`<input type="text" name="firstname">`
- ❑ Širina text input-a se definiše u broju karaktera, i to unutar atributa **size**.

Forme – “radio” input



- ❑ Kada korisnik treba da izabere jednu od više ponuđenih mogućnosti, koriste se radio button-i.
- ❑ Jedan radio button definiše se navođenjem **input** elementa čija je vrednost **type** atributa **radio**.
- ❑ Ukoliko više različitih radio button-a treba da isključuju jedan drugog, potrebno je svim tim button-ima staviti istu vrednost za **name** atribut.

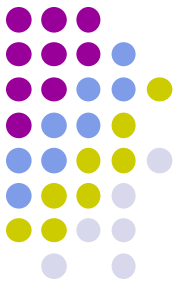
Forme – “radio” input (2)



- ❑ Unutar radio button-a potrebno je definisati i **value** atribut. Vrednost ovog atributa slaće se zajedno sa ostalim podacima forme ukoliko je baš taj radio button selektovan.
- ❑ Atribut **checked**, koji ne mora imati vrednost, se navodi u slučaju da određeni radio button pri inicijalizaciji stranice treba biti selektovan.
- ❑ Ovako bi izgledao jedan radio button:

```
<input type="radio" name="pol" value="muski" checked>Musko
```

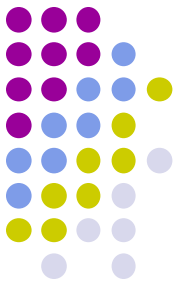
Forme – “submit” input



- ❑ “Submit” input elementom se definiše dugme čijim se klikom šalju svi podaci iz forme.
- ❑ Pri definiciji ove vrste input-a, atribut **type** treba da ima vrednost **submit**.
- ❑ Atributom **value** se definiše tekst koji će se ispisati na samom dugmetu.

```
<input type="submit" value="Pošalji">
```

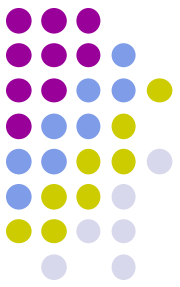
Forme - Primer



```
<form action="action_page.php">
Ime:<br>
<input type="text" name="ime">
<br>
Prezime:<br>
<input type="text" name="prezime">
<br><br>
<input type="radio" name="pol"
value="musko" checked>Musko
<br>
<input type="radio" name="pol"
value="zensko">Zensko
<input type="submit" value="Submit">
</form>
```

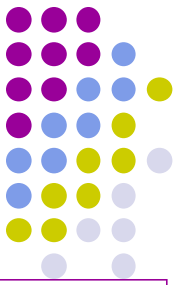
A screenshot of a web browser window. The browser has a single tab titled "Naslov". The address bar is empty. The page content displays a form with the following elements: a label "Ime:" followed by a text input field; a label "Prezime:" followed by a text input field; two radio buttons labeled "Musko" (which is selected) and "Zensko"; and a "Submit" button at the bottom.

Block i inline elementi



- ❑ Većina HTML elemenata se može svrstati u jedan od sledeća dva tipa: *block element* i *inline element*.
- ❑ Block elementi obično počinju i završavaju se novim redom. Neki od njih su: `<h1>`, `<p>`, ``, `<table>`.
- ❑ Inline elementi se prikazuju bez novih redova. Primeri inline elemenata su: ``, `<td>`, `<a>`, ``.

Elementi **div** i **span**



- ❑ Elementi **div** i **span** nemaju nikakvu posebnu semantiku. Njihova svrha je da budu kontejneri za druge elemente i tekst.
- ❑ Jedina razlika između elemenata **div** i **span** jeste ta što je **div** block element, a **span** inline element.
- ❑ Ovi elementi najviše se koriste u kombinaciji sa CSS-om – u njih se smeštaju elementi i tekst koji treba da se stilizuju.

Formatiranje grupe HTML oznaka



- ❑ Element **div** ograđuje jedan blok
- ❑ Koristan je za poravnavanje teksta i pridruživanje određenih karakteristika određenom delu dokumenta, da se ne bi morale koristiti za svaki element bloka posebno
- ❑ Pomoću oznake `<div>` može se dodeliti jedan ili više atributa formatiranja grupi HTML oznaka

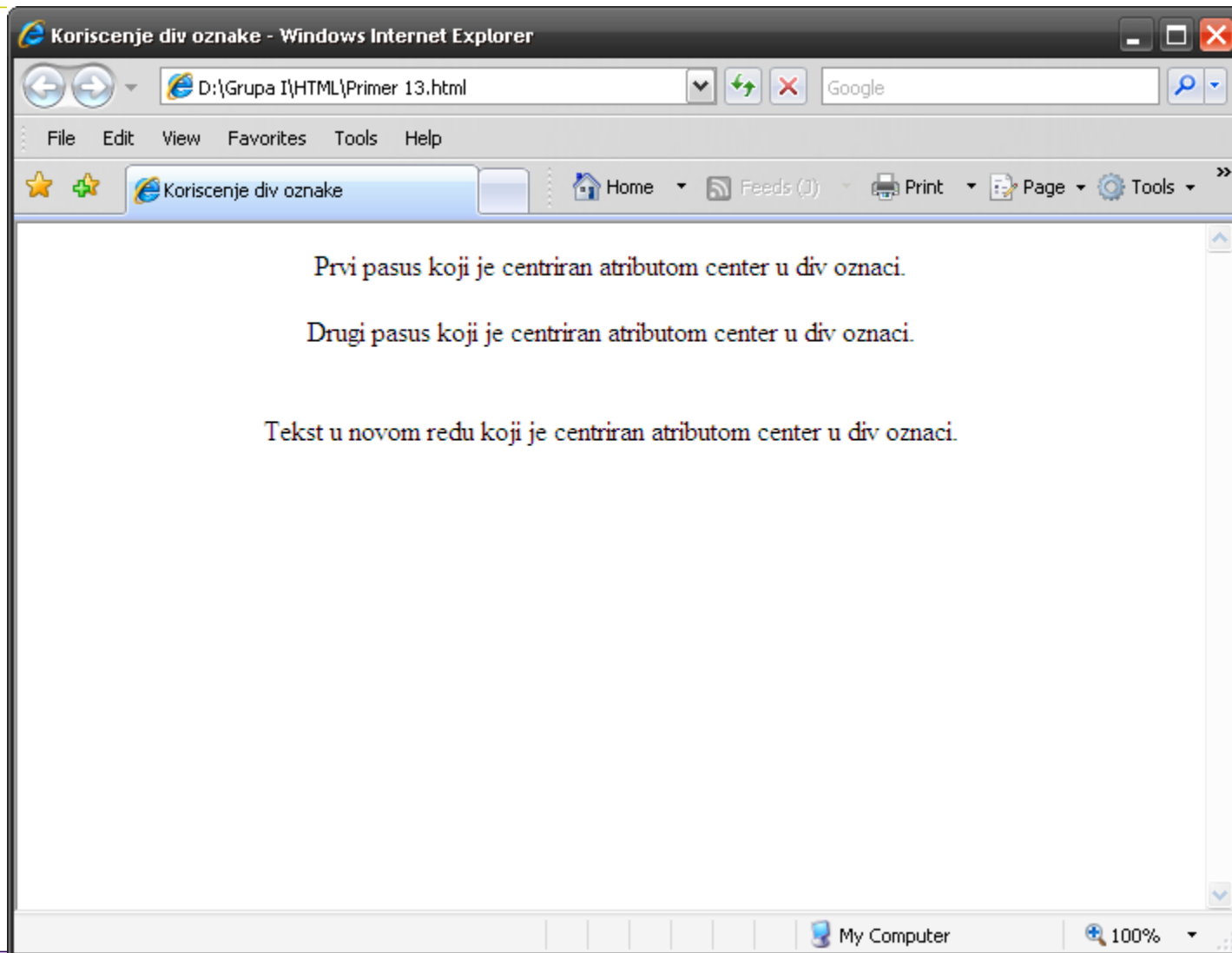
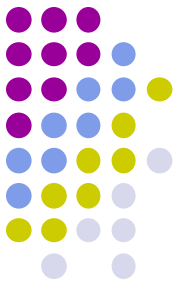
Primer 13.



```
<html>
  <head>
    <title>Veličina slova u naslovu</title>
  </head>
  <body>
    <div align="center">
      <p> Prvi pasus koji je centriran atributom center u div
      oznaci. </p>
      <p> Drugi pasus koji je centriran atributom center u div
      oznaci. </p>
      <br> Tekst u novom redu koji je centriran atributom
      center u div oznaci.
    </div>
  </body>
</html>
```

- ❑ Snimite dokument pod nazivom Primer13.html na D disk (D:\Grupa I\HTML\Primer13.html)

Primer 13.



Uvod u CSS



Uvod

- **CSS = Cascading Style Sheets**
- **Kaskadni opisi stilova (CSS)** predstavljaju skup odrednica (pravila) pomoću kojih se u potpunosti upravlja rasporedom i izgledom stranice
- Unutar HTML-a pre verzije 3.2 nije bilo moguće vršiti formatiranje teksta.
- Verzija 3.2 je to omogućila, ali je sintaksa za formatiranje bila integrisana u sam HTML. To je činilo HTML nepreglednim i obimnim.



Uvod (2)

- Radi postizanja bolje preglednosti, nakon verzije HTML-a 4.0 uveden je CSS.
- Osnovne prednosti CSS-a:
 - Razdvajanje glavnog sadržaja stranice od koda namenjenog za dizajn.
 - CSS kod se može iskoristiti nad više HTML stranica, čime se izbegava ponovno kucanje istog koda.



Gde pisati CSS kod?

- Postoje tri načina na koja se CSS ubacuje u HTML stranicu. CSS kod se može definisati:
 1. Unutar **style** atributa određenog HTML elementa.
 2. Unutar **style** elementa.
 3. U eksternom fajlu koji se onda referencira iz HTML dokumenta.
- Ponekad se može dogoditi da se na navedena tri mesta iste stvari definišu na različit način. Zbog toga je definisan prioritet nad ova tri načina, koji odgovara redosledu u kojem su načini iznad navedeni.
- Primer: ukoliko je boja slova u **style** atributu setovana na plavo, a u **style** elementu na crveno, browser će navedeni tekst obojiti plavom bojom zbog gore navedenog prioriteta.



Inline CSS

- CSS kod koji se nalazi unutar **style** atributa.
- Ovo je najlošiji pristup pisanju CSS koda, s obzirom da se na ovaj način sve navedene pogodnosti CSS-a gube.
- Ovaj pristup koristiti samo kada ne postoji drugo rešenje.

```
<p style="color:blue;margin-left:30px;">Paragraf.</p>
```



Internal CSS

- CSS koji se nalazi u okviru **style** elementa.
- Element **style** se navodi unutar **head** elementa HTML dokumenta.
- CSS kod koji je tu definisan se primenjuje na ceo HTML dokument.
- Ovaj pristup je nešto bolji od prethodnog (inline CSS-a), budući da je CSS kod grupisan na jednom mestu – u zaglavlju dokumenta.
- Ovim pristupom preglednost koda je bolja, no, višestruko korišćenje istog koda i dalje nije moguće.



Internal CSS (2)

```
<html>
<head>
  <style type="text/css">
    body {background-color:yellow;}
    h1 {color: red;
        font-family: sans-serif;}
    p {background-color: #00ff00;font-family: Verdana;}
  </style>
</head>
<body>
<h1>Prvi naslov</h1>
<h2>Drugi naslov</h2>
<p>Tekst paragrafa</p>
</body>
</html>
```

Prvi naslov

Drugi naslov

Tekst paragrafa



External CSS

- CSS koji se definiše u eksternom fajlu, a potom se taj fajl referencira iz HTML-a.
- Ovaj pristup je najbolji i treba ga koristiti kad god je to moguće.
- Na ovaj način se CSS u potpunosti izdvaja iz HTML-a, čime se dobija pregledan kod.
- Tako izdvojen kod se može referencirati od strane više HTML dokumenata, te se time izbegava dupliranje istog CSS koda.



External CSS (2)

- Referenciranje eksternog CSS fajla se vrši korišćenjem **link** HTML elementa koji u generalnom slučaju služi za linkovanje eksternih resursa.
- **link** se navodi unutar **head** elementa.
- Unutar **link** elementa potrebno je navesti nekoliko atributa: **rel**, **type** i **href**. (Postoji još atributa koji se mogu navesti unutar **link** elementa, a ovde su navedeni najbitniji.)



External CSS (3)

- Atributom **rel** se definiše relacija između HTML dokumenta i linkovanog resursa, tj. definiše se šta linkovani resurs HTML dokumentu predstavlja. U slučaju CSS-a vrednost ovog atributa je **stylesheet**.
- Atributom **type** se definiše tip linkovanog resursa. U slučaju CSS-a navodi se vrednost **text/css**.
- Atributom **href** se definiše url do linkovanog resursa. (Uglavnom je to relativna putanja u odnosu na HTML dokument.)

```
<head>  
<link rel="stylesheet" type="text/css" href="style.css">  
</head>
```



Primer - External CSS

```
<html>
<head>
  <TITLE> primer poziva CSS dokumenta </TITLE>
  <link rel="stylesheet" type="text/css" href="test.css" />
</head>
<body>
  <h1>Prvi naslov </h1>
  <p> Tekst izmedju prva dva naslova </p>
  <h1>Drugi naslov </h1>
  <p> Tekst izmedju drugog i treceg naslova </p>
  <h1>Treci naslov </h1>
  <p> Tekst posle treceg naslova </p>
</body>
</html>
```



Primer - External CSS

Primer test.css

test.css je šablon, koji ima sledeći kod:

```
H1 {  
font-family: sans-serif;  
color: #ff0000;  
}  
P {  
font-family: Verdana;  
color: #000055;  
}
```

Prvi naslov

Tekst izmedju prva dva naslova

Drugi naslov

Tekst izmedju drugog i treceg naslova

Treci naslov

Tekst izmedju posle treceg naslova



Sintaksa CSS-a

- CSS kod se sastoji od CSS pravila.
- Svako pravilo sačinjeno je od selektora i deklaracionog bloka.
- Selektorom se definiše **koje** HTML elemente treba stilizovati.
- Deklaracionim blokom se definiše **kako** HTML elemente treba stilizovati.



Sintaksa CSS-a (2)

body

```
{  
  background-color: #d0e4fe;  
}
```

.orangetext

```
{  
  color: orange;  
  text-align: center;  
}
```

#title

```
{  
  font-family: "Times New Roman";  
  font-size: 20px;  
}
```

SELEKTORI

DEKLARACIONI
BLOKOVI



Selektori

- Svrha im je da selektuju HTML elemente koji treba da se stilizuju.
- Postoji više načina selektovanja elemenata:
 - Korišćenjem CSS klasa
 - Korišćenjem ID-a elemenata
 - Navođenjem imena elementa (npr. **p**, **body**, ...)
 - Korišćenjem pseudo klasa
 - Korišćenjem selektora * koji selektuje sve elemente
- Pored navedenih, postoji još načina selektovanja, ali su ovi najbitniji.



Selekcija pomoću CSS klasa

- Class selektor selektuje sve elemente koji za vrednost **class** atributa imaju baš onu klasu koja je navedena unutar selektora.
- Selektor se navodi tako što se ispred imena klase stavlja tačka.

```
.bluetext {  
    text-align: center;  
    color: blue;  
}
```

```
<p class="bluetext">Tekst na  
koji će stil iz klase "bluetext" biti  
primenjen.</p>
```



Selekcija pomoću ID-a elementa

- ID selektor selektuje element čiji je ID jednak vrednosti koja je navedena u selektoru.
- ID elementa navodi se u istoimenom atributu.
- ID vrednost treba da bude jedinstvena unutar datog HTML dokumenta. Drugim rečima, ne treba da postoje dva HTML dokumenta sa istom vrednošću **id** atributa.
- Zbog navedenog, ovaj selektor se može koristiti samo za stilizovanje jednog HTML elementa po dokumentu.



Selekcija pomoću ID-a elementa

- ID selektor se unutar CSS-a navodi tako što se stavi taraba ispred željenog ID-a.

```
#text {  
    text-align: center;  
    color: blue;  
}
```

```
<p id="text">Tekst na koji će stil  
biti primenjen.</p>
```



Primer - External CSS

```
<html>
<head>
  <TITLE> primer poziva CSS dokumenta </TITLE>
  <link rel="stylesheet" type="text/css" href="test.css" />
</head>
<body>
  <h1>Prvi naslov </h1>
  <p> Tekst izmedju prva dva naslova </p>
  <h1>Drugi naslov </h1>
  <p id="text">Tekst na koji ce biti primenjen ID selektor.</p>
  <h1>Treci naslov </h1>
  <p class="greentext">Tekst na koji ce stil iz klase greentext biti
  primenjen.</p>
</body>
</html>
```



Primer - External CSS

Primer test.css

test.css je šablon, koji ima sledeći kod:

```
H1 {
font-family: sans-serif;
color: #ff0000;
}
P {
font-family: Verdana;
color: #000055;
}
.greentext {
    color: green;
}
#text {
    background-color: yellow;
    color: magenta;
}
```

Prvi naslov

Tekst izmedju prva dva naslova

Drugi naslov

Tekst na koji ce biti primenjen ID selektor.

Treci naslov

Tekst na koji ce stil iz klase greentext biti primenjen.



Selekcija pomoću imena elementa

- Elementi se mogu selektovati i navodeći ime elementa.
- U tom slučaju se selektuju svi elementi tog imena koji se nalaze na datoj HTML stranici.

```
p {  
    text-align: center;  
    color: blue;  
}
```

```
<p>Tekst na koji će stil biti  
primenjen.</p>
```



Selekcija pomoću pseudo klasa

- Pseudo klase su predefinisane CSS klase koje se u okviru selektora navode tako što njihovom imenu prethodi dvotačka (:imeklase).
- One definišu posebno stanje određenog elementa. Primer: klasa **hover** selektuje dati element samo kada se strelica miša nalazi iznad njega.
- Primeri pseudo klasa: **link** (neposećen link), **visited** (posećen link), **hover** (miš je iznad elementa), **active** (selektovan link), **first-child** (element je prvo dete nekog roditeljskog elementa), itd.



Selekcija svih elemenata

- Navođenjem zvezde (*) na mesto selektora stilizuju se svi HTML elementi.

```
* {  
    text-align: center;  
    color: blue;  
}
```



Kombinovanje različitih tipova selektora

- Ukoliko se različiti tipovi selektora navedu sukcesivno bez razmaka unutar istog selektora, tada je rezultat selekcije jednak preseku rezultata svakog navedenog tipa selektora pojedinačno.
- Primer: `p.imeklase#idelementa:hover` - u ovom slučaju će se selektovati `p` element čiji `class` atribut ima vrednost “imeklase”, koji ima id “idelementa” i preko kojeg trenutno stoji strelica miša.



Kombinovanje različitih tipova selektora

- Ukoliko se različiti tipovi selektora navedu sukcesivno sa razmakom unutar istog selektora, tada se selekcija vrši postupno. Prvo se izvrši selekcija sa prvim navedenim selektorom na nivou celog dokumenta. Potom se vrši selekcija sa drugim selektorom, ali ne na nivou celog dokumenta već samo nad child elementima elemenata koji su selektovani u prethodnoj fazi. Ovaj postupak se ponavlja dok se svi navedeni selektori ne iskoriste.
- Primer: **p .imeklase #idelementa** - u ovom slučaju će se selektovati element koji ima id “idelementa”, a koji se nalazi unutar elementa klase “imeklase”, koji se opet nalazi unutar nekog **p** elementa.



Kombinovanje različitih tipova selektora

- Ukoliko se različiti selektori navedu tako da su razdvojeni zarezom, tada je rezultat selekcije jednak uniji rezultata svakog navedenog selektora pojedinačno.
- Primer: **p**, **.imeklase**, **#idelementa** – u ovom slučaju biće selektovani svi **p** elementi, svi elementi koji imaju klasu “imeklase”, kao i element koji ima id “idelementa”.



Deklaracioni blokovi

- Sastoje se iz deklaracija koje su jedna od druge odvojene tačka-zarezom, a koje se zajedno nalaze unutar uglastih zagrada.
- Deklaracija se sastoji od property-ja i vrednosti koja je dodeljena tom property-ju.
- Property i njegova vrednost su razdvojeni dvotačkom.

```
{ color: red; text-align: center; }
```



CSS komentari

- CSS komentari se navode između simbola `/* i */`.
- Komentari se ne interpretiraju od strane browser-a.

```
p {  
    color: red;  
    /* Prvi komentar */  
    text-align: center;  
}  
  
/* Drugi  
komentar*/
```



Boje u CSS-u

- Boje se u CSS-u se uglavnom definišu na jedan od sledećih načina:
 1. Kao heksadecimalni broj - `#0000ff`
 2. Kao RGB ili RGBA vrednost - `rgb(0, 0, 255)`,
`rgba*(0, 0, 255, 0.5)`
 3. Pomoću predefinisanih imena boja - `blue`
- Sva tri načina navođenja boja su ravnopravna, i daju isti krajnji rezultat.

* Vrednosti RGBA boja su nastavak RGB vrednosti boja sa alfa kanalom - koji određuje neprozirnost boje.



CSS merne jedinice

- Pri formatiranju web stranice često je potrebno definisati veličinu elementa, udaljenost između elemenata, veličinu margine, itd.
- CSS-om je definisano nekoliko mernih jedinica koje se koriste pri definisanju gore navedenih stvari:

em	Mera relativna u odnosu na veličinu fonta datog elementa. (2 em = 2 puta veće od veličine fonta)
ex	Mera relativna u odnosu na visinu malog slova "x" datog elementa.
ch	Mera relativna u odnosu na širinu karaktera "0" (nula).
rem	Mera relativna u odnosu na veličinu fonta root elementa HTML stranice.
vw	Mera relativna u odnosu na 1% širine viewport-a.
vh	Mera relativna u odnosu na 1% visine viewport-a.
vmin	Mera relativna u odnosu na 1% manje stranice viewport-a.
vmax	Mera relativna u odnosu na 1% veće stranice viewport-a.
%	Značenje procenta zavisi od konteksta u kom je naveden.
px	Pikseli.

CSS merne jedinice (2)

- Postoje i apsolutne merne jedinice koje ne zavise od veličine ekrana – na svakom ekranu će njihova fizička veličina biti ista.
- Ove merne jedinice treba koristiti samo u slučaju kada se unapred znaju dimenzije medijuma na kom će biti prikazan dokument.
- Primer: apsolutne merne jedinice se mogu koristiti ukoliko je HTML dokument namenjen za štampu. Tada se dimenzije papira unapred znaju, te je pogodno koristiti apsolutne jedinice.



APSOLUTNE MERNE JEDINICE



RELATIVNE MERNE JEDINICE



CSS merne jedinice (3)

- Spisak apsolutnih mernih jedinica koje su podržane u CSS-u je dat ispod:

cm	Centimetar
mm	Milimetar
in	Inč (1 in = 2,54 cm)
pt	Point (1 pt = 1/72 in)
pc	Picas (1 pc = 12 pt)



Box model

- Pri raspoređivanju HTML elemenata i određivanju njihovih veličina, neophodno je biti upoznat sa CSS box modelom.
- Box model definiše “omotač” oko svakog HTML elementa koji se sastoji iz sledećih stvari: paddinga, bordera i margine.



Box model (2)

- **Content** predstavlja sadržaj samog HTML elementa.
- **Paddingom** se definiše razmak između contenta i bordera. Padding nije vidljiv – predstavlja običan prazan prostor.
- **Borderom** se definiše linija koja ide oko contenta i paddinga.
- **Margina** služi za definisanje razmaka između elementa i njegove okoline. Kao i padding, ni margina nije vidljiva.

