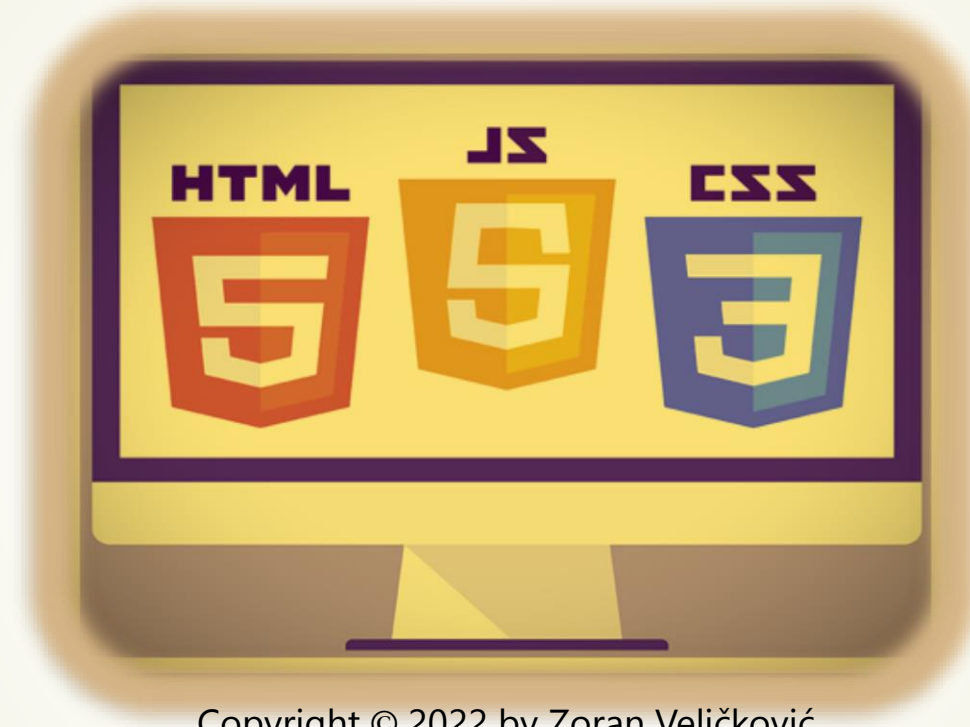




# Akademija tehničko-vaspitačkih strukovnih studija



Copyright © 2022 by Zoran Veličković

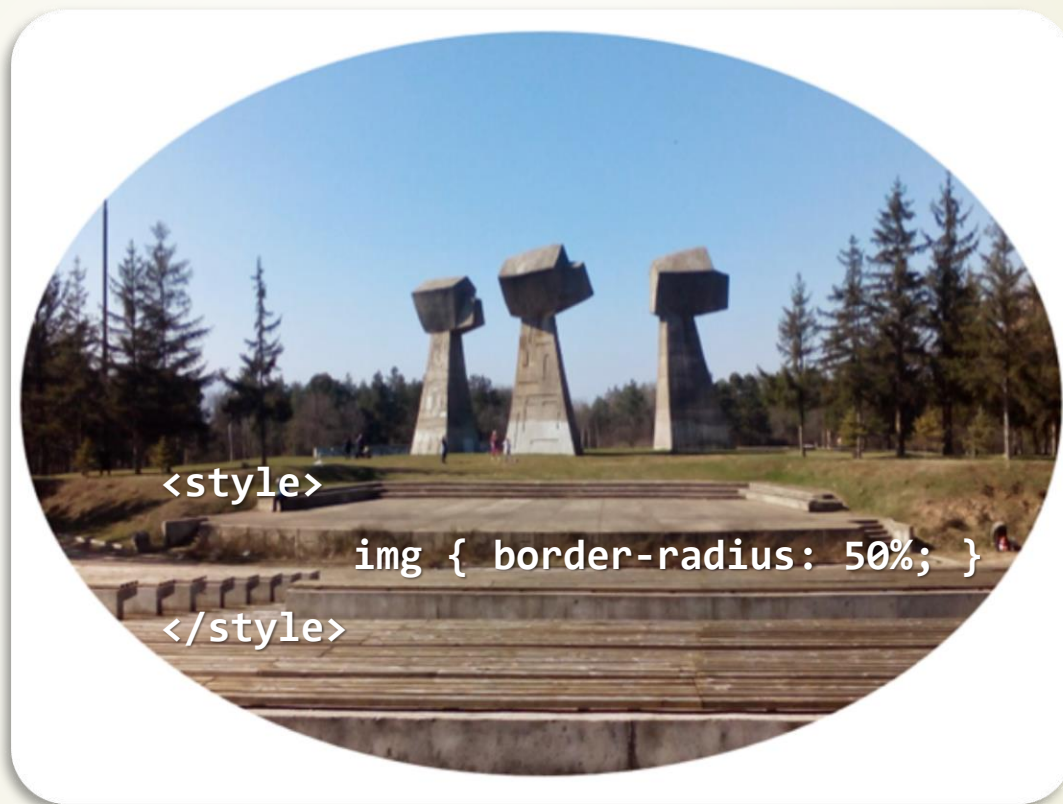
## INTERNET TEHNOLOGIJE

Prof. dr Zoran Veličković, dipl. inž. el.

2022/23.

Prof. dr Zoran Veličković, dipl. inž. el.

# Internet tehnologije



**Stilizacija Web dokumenata – CSS**

(6)



# Sadržaj



- ▶ Izgled Web stranice
  - ▶ Adaptacija izgleda Web stranice
- ▶ Kaskadni opisi stilova
  - ▶ CSS pravila
  - ▶ CSS i ugrađeni stilovi
  - ▶ W3.CSS spoljašnji opis stila
- ▶ CSS i atribut class
  - ▶ CSS i atribut id
  - ▶ CSS i fontovi
  - ▶ CSS i pozicioniranje objekta
- ▶ CSS 3 svojstva
- ▶ CSS i pseudoklase
- ▶ CSS ivice
  - ▶ CSS ivične slike
- ▶ CSS3 i Web čitači
- ▶ Novi selektori u CSS 3
- ▶ Uslovno stilizovanje
  - ▶ Terminalni uređaji i @media upiti
- ▶ Modeli okvira

# Izgled Web stranice

- Do sada je pokazano da jezik za označavanje Web dokumenta, kao što je **HTML**, koristi OZNAKE ugrađene **DIREKTNO** - U SAM HTML KOD da bi se **STILIZOVAO DOKUMENT**.
- Jezik za opis Web dokumenata **ISTOVREMENO** formira **STRUKTURU** dokumenta i vodi računa o njegovom **IZGLEDU**.
- **SAVREMENI KONCEPT** opisa Web dokumenata podrazumeva da se **NEZAVISNO** opiše **STRUKTURA DOKUMENTA**, a posebno **IZGLED DOKUMENTA** u Web pretraživaču.
- Tako, u **HTML 5** verziji jezika, za opis Web dokumenta se samo **VODI RAČUNA** o tome **KAKO** je formatiran sadržaj, a **NE** kako će on **IZGLEDATI** u Web čitaču.
- Poenta u HTML 5 je **TAČAN OPIS DOKUMENTA**, dok se izgled Web dokumenta **ADAPTIRA** preme **TIPU UREĐAJA** kojim klijent pristupa Web stranici.

# Adaptacija izgleda Web stranice

- Godina 2016. predstavlja KONCEPTUALNU **PREKRETNICU** Web tehnologija.
- Prekretnica je izazvana činjenicom da je pristup Webu **MOBILNIM TELEFONOM** postao **DOMINANTAN**, što zahteva **NOVI PRISTUP** u razvoju Web stranica.
- Ovaj novi koncept je nazvan „**MOBILE FIRST**“ i podrazumeva razvoj Web aplikacije **PRIMARNO** za **MOBILNE TELEFONE**, a zatim njegovu adaptacija za ostale platforme!
- Cilj ovih tehnologija je da se razvije **EFIKASNA TEHNIKA** za **ADAPTACIJU IZGLEDA** Web sadržaja kako bi se **ADEKVATNO** PRIKAZAO na **SVIM** platformama.
- **ADAPTACIJA IZGLEDA** Web stranice se može obaviti i na:
  - **KLIJENTSKOJ STRANI** - kada se koristi tehnologija **KASKADNIH OPISA STILOVA - CSS** (engl. *Cascading Style Sheet*).
  - **SERVERSKOJ STRANI** - kada se koristi se specijalizovani programski jezik na serveru.



# Kaskadni opisi stilova - CSS

- **KASKADNI OPISI STILOVA** - **CSS** predstavljaju **SKUP PRAVILA** pomoću kojih se u potpunosti upravlja **RASPOREDOM** i **IZGLEDOM** objekata na Web stranici.
- Da se podsetimo, Web stranica **NEMA** unutrašnji koordinatni sistem koji bi pomogao **POZICIONIRANJU** Web objekata.
- **CSS** omogućava da se čitaču Weba direktno naloži **GDE** i **ŠTA** da stavi na Web stranicu.
- Pored pomenutog, **CSS PRAVILIMA** se takođe može definisati:
  - **Tip fonta,**
  - **Veličina i boja fonta,**
  - **Izgled teksta,**
  - **Pozicija elementa na stranici, ...**
- CSS je **SKUP ISKAZA** koji predstavljaju **CSS PRAVILA** koji se postavljaju između početne i završne HTML oznake **<style>** i **</style>**.

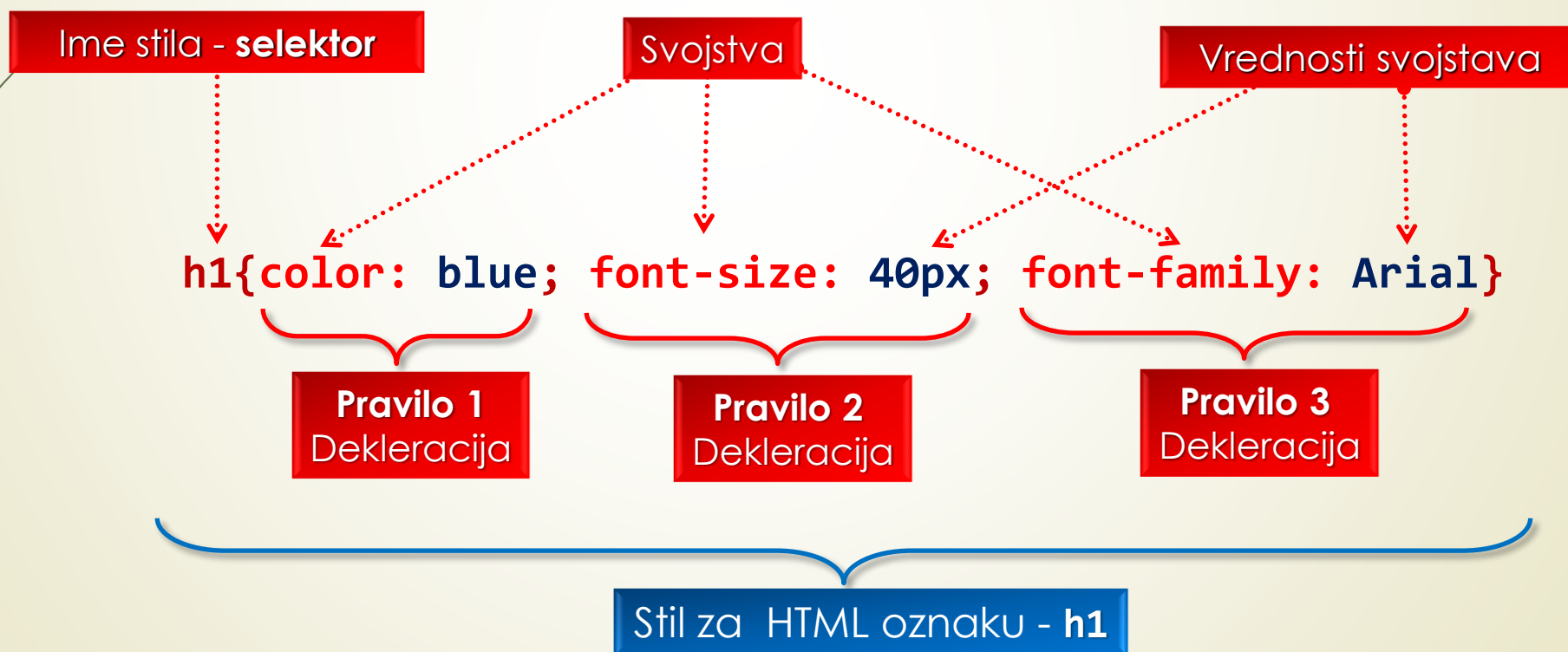
# CSS pravila

- Dakle, **CSS** precizno **UPRAVLJA IZGLEDOM**, **FONTOVIMA**, **BOJAMA**, **POZADINOM** i drugim **TIPOGRAFSKIM** efektima.
- **CSS** može da **MENJA IZGLED** i **FORMATIRANJE** neograničenog broja Web stranica promenom samo **JEDNOG DOKUMENTA** - **OPISA STILOVA**.
- **CSS** pravi efektivne i upečatljive Web stranice sa **MANJE KODA!**
- **KASKADNI OPISI STILOVA** omogućavaju pisanje naredbi za IZGLED i FORMATIRANJE u:
  - **ZAGLAVLJU** Web stranice ili
  - **SPOLJAŠNJOJ** datoteci.
- Opisi stilova se sastoje od **JEDNOG** ili **VIŠE PRAVILA**.
- **PRAVILA** koja su sastavni deo opisa stilova - CSS nalažu Web čitaču da **RASPOREDI** ili **FORMATIRA HTML ELEMENTE** definisane ovim pravilima.

# Arhitektura CSS pravila

- CSS iskazi su uvek sledećeg oblika:

```
ime_stila { pravilo ili više_pravila }
```





# CSS pravila - grupni i kontekstualni selektori

- U vitičaste zagrade iza **IMENA STILA** se može staviti **JEDNO** ili **VIŠE PRAVILA** koja se **RAZDVAJAJU** tačkom i zarezom (;).
- Primer pravila koje poseduje **VIŠE DEKLARACIJA**:

**GRUPNI SELEKTORI**  
Odvojeni (,)

.....➤ h2, p, ul {color: green; font-family: arial}

.....➤ p ul {color: blue; font-family: ComicSans MT}

**KONTEKSTUALNI SELEKTOR**, primenjuje se samo na **ugneždjeni** HTML element (neuređenu listu u okviru pasusa)

- Osnovni atribut HTML oznake **<style>** je **"type"** koji u našim primerima uzima vrednost **"text/css"**.
- Između oznake **<style>** i **</style>** nalaze se **CSS PRAVILA** u različitim deklaracijama.

# Kaskadni opisi stilova u zaglavlju dokumenta

```
<html>
```

```
<head>
```

```
<style type="text/css">
```

... Ovde se pišu **CSS** pravila...

```
</style>
```

```
...
```

```
</head>
```

```
...
```

```
<body>
```

//... Sadržaj Web stranice...

```
</body>
```

```
</html>
```

Opis kaskadnih stilova se postavlja u **ZAGLAVLJU** HTML dokumenta  
`<head> </head>`

Kontejner za opis kaskadnih stilova `<style> </style>`

# CSS i ugrađeni stilovi

- Postoje nekoliko načina za **PRIDRUŽIVANJE** CSS PRAVILA u **HTML KOD** Web stranice:
  1. **DODAVANJE** PRAVILA za opise stilova **u HTML oznake** (engl. *inline*),
  2. **DIREKTNO** UGRAĐIVANJE opisa stilova **u HTML kod** (prethodni primer),
  3. **POVEZIVANJE** SA **SPOLJAŠNJI** opisima stilova (biće prikazano),
  4. **UVOŽENJE** spoljašnjeg opisa stilova (biće prikazano).
- Ako se koristi **DODAVANJE PRAVILA u HTML oznake** ili **DIREKTNO UGRAĐIVANJE** opisa stilova **u HTML kod**, čitač primenjuje **CSS PRAVILA** samo na elemente one **STRANICE U KOJOJ** se nalaze deklaracije.
- Ovaj način je pogodan samo u slučaju kada su **PRAVILA JEDINSTVENA** za tu WEB STRANICU.

# CSS stilovi sa više pravila

```
<head>
```

```
<title> Ugrađeni Cascading Style Sheet (CSS)</title>
```

```
<style type="text/css">
```

```
<!--
```

```
h1 {color:blue; font-size:40px; font-family:verdana}
```

```
p {color:white; background:green; font-family:helvetica; text-indent:2cm}
```

```
-->
```

```
</style>
```

```
</head>
```

Primer:  
Dva stila (**h1** i **p**) koji  
se sastoje od **VIŠE** pravila

# CSS i spoljašnji opisi stilova (1)

- **SPOLJAŠNJI OPISI STILOVA** su veoma pogodni kada **ISTI SKUP PRAVILA** treba primeniti na **VIŠE STRANICA**.
- Da bi se povezali spoljašnji opisi stilova sa tekućom Web stranicom, u zaglavlje Web stranice se umeće HTML oznaka „<link>“.
- Primer povezivanja **SPOLJAŠNJIH** opisa stilova u **HTML** dokument je realizovan sledećim kodom:

...

```
<head>
```

```
<title> Linked Cascading Style Sheet (CSS) </title>
```

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

```
</head>
```

...

Atributi oznake link: rel, href, type



# CSS i spoljašnji opisi stilova (2)

- Ako se pravila smeštaju u spoljašnjoj datoteci, **NE STAVLJAJU** se između početne i završne oznake **<style> </style>**!
- Primer kako se u spoljašnjoj datoteci definišu pravila:

```
h1 {color: blue; font-size: 40px; font-family: verdana}
```

```
p {color: white; background: green; font-family: helvetica; text-indent: 1cm}
```

```
body {background: #ADD8E6; color: maroon}
```

- **UVOŽENJE** CSS-a u HTML dokument se može realizovati i „**media query**“ pravilom **@import** koje dozvoljava **KOMBINOVANJE** sa drugim metodama.
- Zaglavlje Web stranice koje sadrži iskaz uvoženja (**@import**) i ugrađena CSS pravila kojima se zadaje izgled naslova drugog nivoa i menja izgled teksta tela stranice je dat kodom na sledećem slajdu:

# CSS i uvoženje stilova

```
<head>
```

```
  <title>Imported Cascading Style Sheet (CSS)</title>
```

```
  <style type="text/css">
```

```
    <!--
```

```
      @import url(styles.css);
```

```
      h2 {color:purple; font-size:30px; font-family:helvetica}
```

```
      body {color:black}
```

```
    -->
```

```
  </style>
```

```
</head>
```

Učitavaju se spoljašnji stilovi iz fajla `styles.css`

Menjaju se učitani stilovi **samo** za definisane oznake (h2, body) – ostali stilovi su **neizmenjeni**.

# Nepridružena pravila CSS-a

- CSS dozvoljava kreiranje i tzv. **NEPRIDRUŽENIH** – klasnih pravila.
- Ova nepridružena pravila se deklarišu **NEZAVISNO OD SELEKTORA** i mogu se pridružiti **VEĆEM BROJU** selektora (tamo gde pravilo ima smisla).
- Posledica ovog koncepta je da se jednom selektoru mogu pridružiti **RAZLIČITA** klasna CSS pravila u okviru jednog HTML dokumenta.
- Tako će se u sledećem primenu deklarirati dva nepridružena - klasna pravila **.shadow** i **.banner** koje će se primeniti na dve **<div>** oznake.
- Izbor nepridruženog pravila se obavlja „**class**“ atributom u okviru selektora na koji se primenjuje.

# CSS i nepridružena (class) pravila

```
<style type="text/css">
```

```
<!--
```

**.banner** – nepridruženo –  
klasno pravilo - klasa

```
.banner {
```

CSS svojstva

```
position: absolute;  
top: 7px;  
left: 1px;  
font-size: 100px;  
font-family: 'arial black' Impact Arial Helvetica Verdana; color: red }
```

```
.shadow {
```

CSS svojstva

```
position: absolute;  
top: 15px;  
left: 7px;  
font-size: 100px; font-family: 'arial black' Helvetica; color: gray }
```

**.shadow** – nepridruženo –  
klasno pravilo - klasa

```
-->
```



# CSS 3 lab. vežba-senka

```
</style>  
</head>  
<body bgcolor="white">  
  <div class = "shadow">Internet tehnologije CSS</div>  
  <div class = "banner">Internet tehnologije CSS</div>  
</body>  
</html>
```



Klasa "banner"

Klasa "shadow"



# CSS i više pridruženih pravila

```
<html>
  <head>
    <title>Primer: Korišćenje selektora klase</title>
    <style type="text/css">
      <!--
        p.critical {color:red; font-size:18pt; font-weight:bold}
        p.normal {color:black; font-size:12 pt}
      -->
    </style>
  </head>
  <body>
    <p class="critical"> Ovo je crveno, 18 point, bold face </p>
    <p class="normal"> Ovo je crno 12 point text </p>
  </body>
</html>
```

Dva pravila za  
ISTU HTML  
oznaku (p)

Deklaracije pasusa

Primena klasnih pravila pasusa

# W3.CSS spoljašnji opis stila (1)

```
<!DOCTYPE html>
```

```
<html>
```

```
<title>W3.CSS</title>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<link rel="stylesheet" href="http://www.w3schools.com/lib/w3.css">
```

```
<body>
```

```
<div class="w3-container w3-teal">
```

```
// w3-container w3-red
```

```
<h1>My Header</h1>
```

```
</div>
```

```

```

Učitavanje spoljašnjeg stila

Kontekstualni selector

Primena spoljašnjeg stila

# W3.CSS spoljašnji opis stila (2)

```
<div class="w3-container">
```

```
<p>A car is a wheeled, self-powered motor vehicle used for transportation.</p>
```

```
</div>
```

Kontekstualni selector

```
<div class="w3-container w3-teal">
```

```
// w3-container w3-red
```

```
<h5>Footer</h5>
```

```
<p>Footer information goes here</p>
```

```
</div>
```

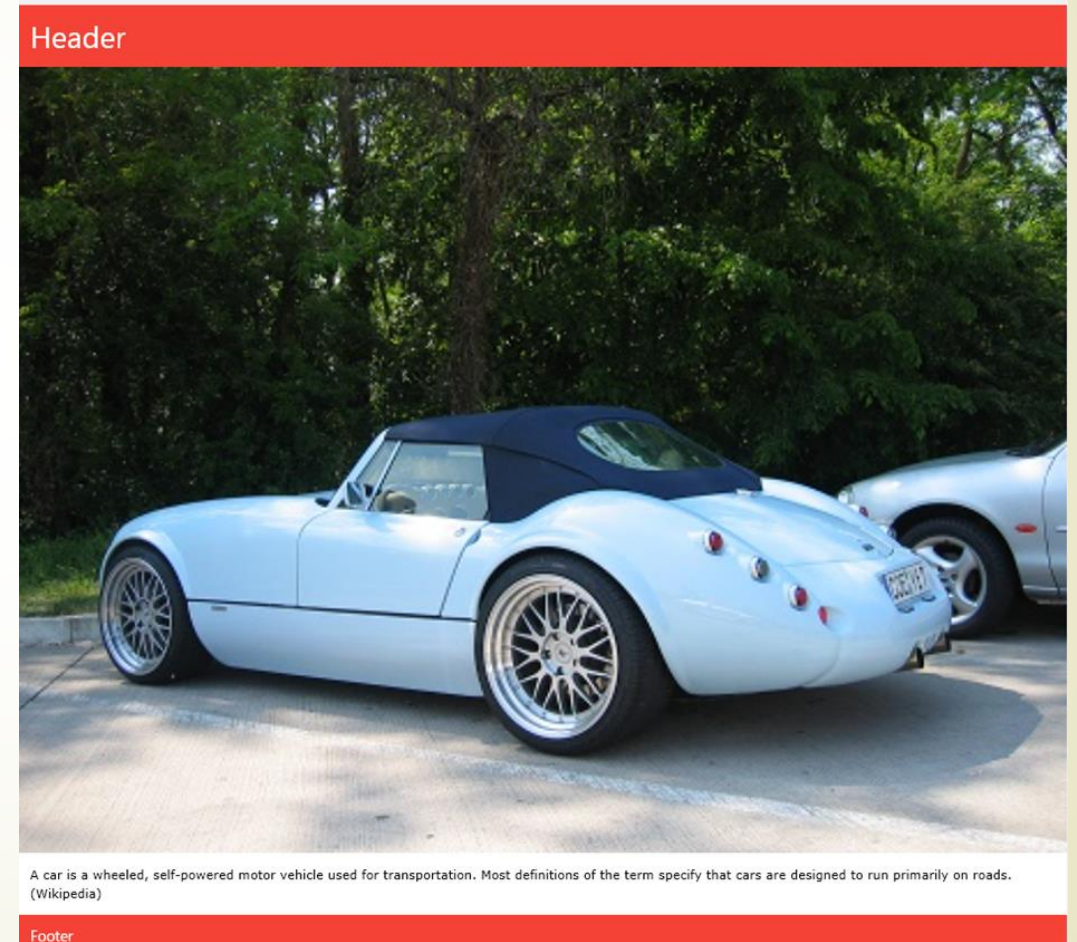
```
</body>
```

```
</html>
```

Primena stilova definisanih u spoljašnjoj datotrci



# W3.CSS spoljašnji opis stila (3)



# W3.CSS spoljašnji opis stila (4)

```
/* W3.CSS 4.10 February 2018 by Jan Egil and Borge Refsnes */
```

```
html{box-sizing:border-box}*,*:before,*:after{box-sizing:inherit}
```

```
/* Extr. from normalize.css by N.Gallagher and J. Neal git.io/normalize */
```

```
...
```

GRUPNI selektori

```
.w3-red,.w3-hover-red:hover{color:#fff!important;background-color:#f44336!important}
```

```
.w3-sand,.w3-hover-sand:hover{color:#000!important;background-color:#fdf5e6!important}
```

```
.w3-teal,.w3-hover-teal:hover{color:#fff!important;background-color:#009688!important}
```

```
...
```

```
@media (min-width:993px){
```

```
...
```

## The Container Class:

The w3-container class adds a 16px left and right padding to any HTML element.

The w3-container class is the perfect class to use for all HTML container elements like:

<div>, <article>, <section>, <header>, <footer>, <form>, and more.



# CSS i atribut id

- **REFERENCIRANJE** HTML objekata pomoću oznake "**class**" je veoma korisno, ali se i atribut "**id**" može upotrebiti kao **REFERENCA** na način prikazan u nastavku.
- Da bi se referencirao objekt pomoću **ATRIBUTA** "**id**", **PRAVILO SE MORA deklarirati** tako da se ispred vrednosti za identifikaciju stavi oznaka "**#**":

➔ **#text1 {font-size: 20px}**

- HTML kod koji koristi ovu referencu treba da sadrži sledeći iskaz:

```
<p id = "text1"> Neki tekst </p>
```

- Kada se CSS datoteci referencira element pomoću identifikatora **text1**, elementi sa tom vrednošću se **MENJAJU**, dok ostali elementi ostaju **NEPROMENJENI**.

# CSS i fontovi

- Vrednost svojstva "**font-family**" između vitičastih zagrada (`{ i }`) je deo je CSS DEKLARACIJE sa **REZERVNIM** FONTOVIMA.

```
p {font-family: helvetica, "times new roman", courier}
```

- **CSS SVOJSTVO** "**font-size**" pruža mogućnost da se zada **VELIČINA TEKSTA** Web stranice.
- Da bi se zadala veličina teksta u tačkama (engl. *pixels*), dodaju se slova **pt** brojčanoj vrednosti koja sledi iza svojstva "**font-size**" kao u sledećem primeru:

```
p {font-size: 12pt}
```

```
b {font-size: 18pt}
```

# CSS i pozicioniranje objekta (1)

- CSS SVOJSTVO "**position**" zadaje **GDE** će se na stranici smesti neki element – objekt Web stranice.
- **APSOLUTNO** i **RELATIVNO** pozicioniranje elemenata na Web stranici je omogućeno CSS-om:

```
<img src = "image1.jpg">
```

Inline CSS pravilo za pozicioniranje

```
<img style = "position: relative; left: 20px" src = "image2.jpg">
```

- Svojstvo "**text-align**" omogućava da se **PORAVNA TEKST** uz jednu od VERTIKALNIH IVICA Web stranice (levu ili desnu):

**KLASNI** –  
nepridruženi  
selektori

```
.left { text-align: left}  
.right { text-align: right}
```

# CSS i pozicioniranje objekta (2)

► Postoje **ČETIRI** svojstva **MARGINA**:

- `margin-left`,
- `margin-right`,
- `margin-top`,
- `margin-bottom`.

...

```
body {margin-left: 20px; margin-right: 40px}
```

```
<div style="margin-left: 10%; margin-right: 10%">
```

...

```
</div>
```

...

CSS pravila za oznaku `<body>`

INLINE deklaracije za oznaku `<div>`

# CSS 3 svojstvo: border-radius

```
<!DOCTYPE html>
<html>
<head>
  <style>
    img { border-radius: 50%; }
  </style>
</head>
<body>
  <h2>Kružna slika</h2>
  <p>Korišćenje svojstva border-radius za kreiranje kružne slike:</p>
  
</body>
</html>
```

## Kružna slika

Korišćenje svojstva border-radius za kreiranje kružne slike:





# CSS 3 svojstvo: "text-shadow"

Pozicioniranje senke: Horizontalno, vertikalno, blur, boja senke

```
h1 {  
  text-shadow: 0 0 3px #FF0000;  
}
```

Internet tehnologije CSS 3

Dve senke

```
h1 {  
  text-shadow: 0 0 3px #FF0000, 0 0 5px #0000FF;  
}
```

Internet tehnologije CSS 3

Beli tekst sa više senki

```
h1 {  
  color: white;  
  text-shadow: 1px 1px 2px black, 0 0 25px blue, 0 0 5px darkblue;  
}
```

Internet tehnologije CSS 3

# CSS i pseudoklase

- Već je pokazano da se **NEPRIDRUŽENA** CSS pravila (zovu se i **CSS KLASE**) mogu pridružiti odgovarajućim HTML elementima.
- Operatorom tačka (.) se selektuje CSS NEPRIDRUŽENA **KLASA**, a ne HTML element.
- **PSEUDO-KLASE** (engl. *pseudo-class selector*) omogućavaju referenciranje HTML elemenata pomoću njegovih **OSOBINA**, kao što su recimo **POLOŽAJ U KODU** ili **AKTUELNO STANJE** HTML elementa.
- **SELEKTORI PSEUDO-KLASA** na početku imaju **DVOTAČKE** (: ili ::) i obično se navode **IZA** selektora tipa elementa, bez dodatnih razmaka.

```
p:nth-child(2) { background: #998877;}
```



Pseudo-klasom **nth-child** referencira **SVAKI DRUGI** **<p>** element (**pozicija** u HTML kodu)

# CSS pseudoklase aktuelnog stanja

- Primeri **PSEUDOKLASA** za HIPERTEKSTUALNE VEZE se baziraju na **AKTUELNOM STANJU** hipertekstualnih veza (posećene, neposećene, ...):

PSEUDOKLASA	ZNAČENJE
<code>a:link</code>	Neposećene hipertekstualne veze
<code>a:visited</code>	Posećene hipertekstualne veze: u listi posećenih lokacija Web čitača postoji ulaz koji pokazuje na cilj hiperveze.
<code>a:hover</code>	Pokazivač miša posetioca se nalazi nad hipervezom.
<code>a:active</code>	Pokazivač miša posetioca se nalazi nad hipervezom, a posetilac pritiska ili otpušta levi taster miša.

- Primeri:

```
a:link {color: darkgreen}
```

```
a:visited {color: magenta}
```

```
a:hover {font-size: 1.5em; font-style: italic; background-color: cyan}
```

# CSS i pseudo elementi položaja

- Pseudo elementi bazirani na POLOŽAJU U KODU:

PSEUDOKLASE	ZNAČENJE
<code>::first-line</code>	Ovaj selektor primenjuje stil prve linije specificiranog elementa (pre :).
<code>::first-letter</code>	Ovaj selektor primenjuje stil prvog slova specificiranog elementa.
<code>::before</code>	Koristi se za insertovanje sadržaja pre specificiranog elementa, bez dodavanja karaktera izvornom dokumentu.
<code>::after</code>	Koristi se za insertovanje sadržaja posle specificiranog elementa, bez dodavanja karaktera izvornom dokumentu.

- Primeri:

```
p::before {content:"Once upon a time: "; font-weight:bold; color:purple;}
```

```
p::first-line {letter-spacing: 8px;}
```

```
p::first-letter {font-size: 300%; color: orange;}
```

# CSS ivice

- ▶ CSS svojstvo "**border**" omogućava nezavisno kontrolisanje debljine, stila i boje **SVAKE IVICE** oko objekta.
- ▶ CSS omogućava iscrtavanje **IVICA** oko sledećih objekata:
  - ▶ slika, pasusa,
  - ▶ teksta unutar oznaka **<span> </span>**,
  - ▶ elemenata i objekata unutar oznaka **<div> ... </div>**:

```
h1 {border: solid 3px black}
```

- ▶ Ako se želi kontrolisanje svih **ČETIRI IVICA POSEBNO**, zadaju se vrednosti za **border-style**, **border-width** i **border-color** za sve stranice:

```
h1 { border-top-style: solid; border-left-style: solid;
      border-right-style: solid; border-bottom-style: solid;
      border-top-width: 3px; border-left-width: 3px; ...
    }
```



# CSS ivične slike i vodeni žig

- ▶ CSS svojstvo "**background-image**" je odgovorno za postavljanje **IVIČNIH SLIKA**:

```
.body {  
    background-image: URL(images/TileFilename.gif);  
    background-repeat: repeat-y  
}
```

- ▶ **VODENI ŽIG** ostaje na **ISTOM MESTU** dok posetilac pomera sadržaj stranice.

- ▶ CSS pruža svojstva za pozadinu koja omogućavaju da se smesti **JEDNA KOPIJA VODENOG ŽIGA** na pozadinu stranice i **FIKSIRA** je na tom mestu:

```
.body {  
    background-image: url(images/LightBulb.gif);  
    background-repeat: no-repeat;  
    background-position: center;  
    background-attachment: fixed  
}
```

# CSS i svojstvo float

- CSS svojstvo "**float**" omogućava raspoređivanje objekata **UZ IVICE SLIKA** ili uz neke **DRUGE OBJEKTE**.
- Za pomeranje slike levo od pasusa, dodaje se svojstvo **float** oznaci `<img>`.

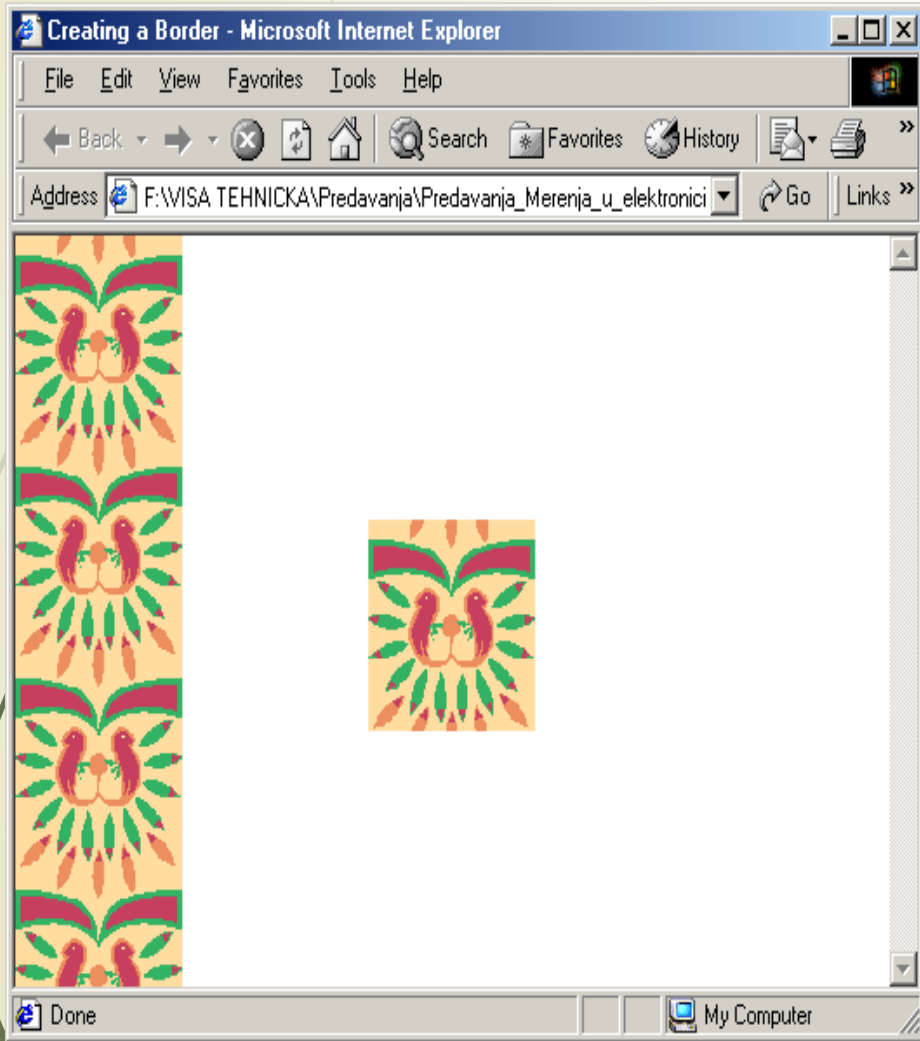
```
<img style = "float: left" border = "0" src =  
        "Programmer.gif" width = "146" height = "148">
```

```
h6 {float: left}
```

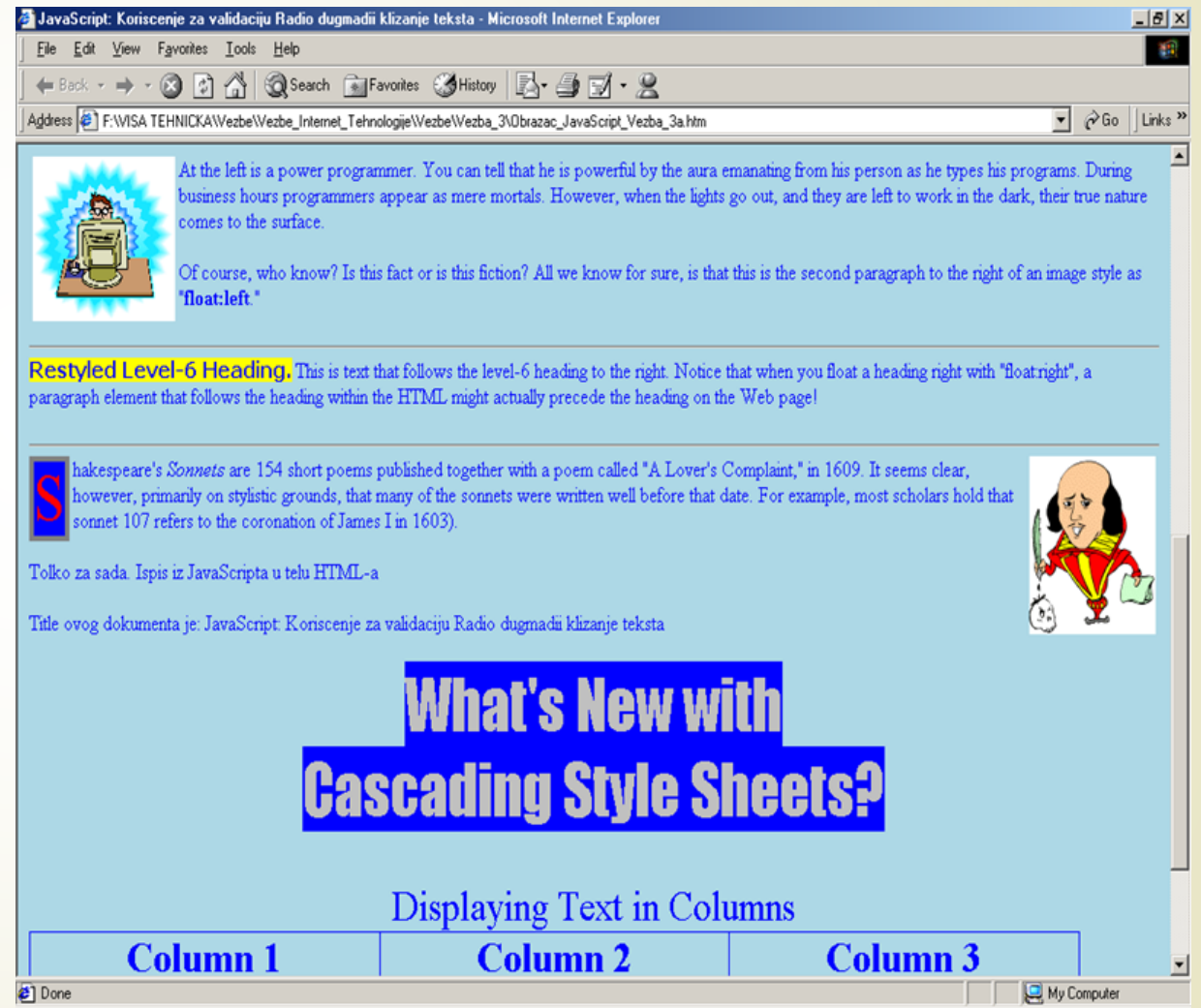
```
<img style = "float: right" border = "0" src =  
        "images/Shakespeare.gif" width = "110" height = "132">
```

- Dodatak praktikuma sadrži sva CSS svojstva sa vrednostima koje im se mogu dodeliti.

# CSS i svojstvo float/ivična slika



Ivična slika



Relativni položaj elemenata na Web starnici

# W3C: svojstvo background-\*

CSS Tutorial

CSS HOME

CSS Introduction

CSS Syntax

CSS Selectors

CSS How To

CSS Comments

CSS Colors

CSS Backgrounds

Background Color

Background Image

Background Repeat

Background Attachment

Background Shorthand

CSS Borders

CSS Margins

CSS Padding

CSS Height/Width

CSS Box Model

CSS Outline

CSS Text

CSS Fonts

CSS Icons

CSS Links

CSS Lists

CSS Tables

CSS Display

CSS Max-width

CSS Position

CSS Z-index

## CSS Backgrounds

< Previous

Next >

The CSS background properties are used to add background effects for elements.

In these chapters, you will learn about the following CSS background properties:

- `background-color`
- `background-image`
- `background-repeat`
- `background-attachment`
- `background-position`
- `background` (shorthand property)

## CSS background-color

The `background-color` property specifies the background color of an element.

### Example

# Nova svojstva u CSS3 i Web čitači

- ▶ Dobar deo novih CSS 3 svojstava je u **EKSPIRIMENTALNOJ** UPOTREBI.
- ▶ Da bi različiti Web čitači ispravno renderovali nova CSS 3 svojstva, u ovoj fazi razvoja specifikacije, je neophodno postaviti odgovarajuće **PREFIKSE** kojima će se identifikovati primenjeni Web čitač.
- ▶ U upotrebi su sledeći prefiksi:
  - ▶ **-moz** za Mozilla Firefox
  - ▶ **-webkit** za Safari i Google Chrome
  - ▶ **-ms** za Internet Explorer
- ▶ Primer upotrebe **box-sizing** svojstva:
  - ▶ **-moz-box-sizing: border-box;**
  - ▶ **-webkit-box-sizing: border-box;**



# Novi selektori u CSS 3 (1)

- CSS 3 obezbeđuje referenciranje HTML objekata pomoću **BILO KOG ATRIBUTA**.
- Može se napisati pravilo koje menja samo **<p>** elemente koji imaju atribut sa imenom **"name"** sa vrednošću **mytext**:

```
p[name = "mytext"] {font-size: 20px}
```

Selektuje pasuse sa atributom name postavljenim na "mytext".

- Sledeći **SELEKTORI** zahtevaju da **IME ATRIBUTA**:
  - počinje datim karakterima (^);
  - da se završava datim karakterima (\$);
  - da sadrži dati podniz (\*):

```
p[name^ = "my"] {font-size: 20px}
```

```
p[name$ = "my"] {font-size: 20px}
```

```
p[name* = "my"] {font-size: 20px}
```

# Novi selektori u CSS 3 (2)

- U **CSS 3** je dodato još nekoliko **NOVIH SELEKTORA** označenih simbolima: **>**, **\***, **~**.

- Selektor "**>**" menja **DRUGI ELEMENT** kada mu je roditelj prvi element:

```
div > p.mytext {color: #9000;} 
```

- Selektor "**+**" referencira **DRUGI ELEMENT** kada mu neposredno **PRETHODI** prvi element:

```
p.mytext + p {color: #9000;} 
```

- Selektor "**~**" je sličan prethodnom i ne zahteva da se menjani element ne mora nalaziti neposredno **ISPRED** prvog elementa:

```
p.mytext ~ p {color: #9000;} 
```

- Više o selektorima kada budemo radili temu vezanu za **JavaScript**, koji koristi CSS selektore za **REFERENCIRANJE HTML OZNAKA**.

# Nova CSS 3 svojstva

- ▶ Pored bavljenja izgledom i formatiranjem, specifikacija **CSS 3** se sve više usmerava na **OBLIK** i **POKRET** na Web stranici.
- ▶ Zapravo, nova CSS 3 svojstva pružaju one mogućnosti koje je nekada nudio samo **JavaScript**.
- ▶ Ovo je obimna specifikacija, tako će biti naveden samo deo novih svojstava:
  - ▶ **border-radius**
  - ▶ **box-shadow**
  - ▶ **text-shadow**
  - ▶ **border-image**
  - ▶ **outline**
  - ▶ **column-count, column-gap, column-rule**
  - ▶ **transform:scale, rotate, translate,**
  - ▶ **3D transformacije,**
  - ▶ **prelazi,**
  - ▶ **animacije, ...**

## Polaroid Images / Cards

Senka formirana box-shadow svojstvom :



Niš, Srbija

# Uslovno stilizovanje i @media upiti

- Web tehnologija poznata pod nazivom **MEDIA UPITI** (engl. *media queries*) je od strane **W3C**-a preporučena za **ADAPACIJU PRIKAZA** Web stranice (najnovija specifikacija nosi oznaku **CSS 3 (4)** i doživela je potpunu afirmaciju).
- **MEDIA UPITI** se deklarišu pravilom **@media** i omogućavaju **USLOVNO STILIZOVANJE OBJEKATA** na Web stranici.
- USLOVI STILIZOVANJA zavise od **TIPA MEDIJA**, odnosno, **SPECIFIČNOSTI** pristupnog terminalnog uređaja.
- ANALIZIOM PODATAKA dobijenih **@media** upitima, može se obaviti **ADAPTACIJA STILOVA**, odnosno, **IZGLEDA WEB STRANICE**.
- Pojedini **STILOVI** se mogu primeniti na ČITAVU **KLASU UREĐAJA** ili pak samo na određenim **TIPOVIMA**.
- U prvom redu, adaptivni Web dizajn podrazumeva **PRILAGOĐAVANJE** – adaptaciju Web sadržaja **ŠIRINI** i **VISINI** displeja terminalnog uređaja.

# Terminalni uređaji i media upiti

- **CSS 3** specifikacija preporučuje adaptaciju izgleda Web stranice na osnovu **MOGUĆNOSTI** uređaja za prikazivanje, a **NE** na osnovu **TIPA UREĐAJA!**
- Tehnika **MEDIA UPITA** se veoma često koristi za **ISPORUČIVANJE ADAPTIRANIH STILOVA** za tablete ili mobilne telefone kao što su iPhone i Android.
- Media upiti se mogu koristiti za ustanovljavanje **SLEDEĆIH KARAKTERISTIKA** terminalnih uređaja:
  - širina i visina **DISPLEJA**,
  - širina i visina **PRIKAZA**,
  - **ORIJENTACIJA** (da li je tablet/telefon u pejzažnom ili portretnom režimu,
  - **REZOLUCIJA** ekrana.
- Na osnovu **USTANOVljenih KARAKTERISTIKA** terminalnog uređaja koristi se **CSS** (engl. *Cascading Style Sheets*) **TEHNOLOGIJA** za adaptiranje sadržaja **PRE** prikaza.



# Kreiranje media upita

- **MEDIA UPITI** se sastoje od ključne reči **@media** iza koje se navodi lista **MEDIJA TIPOVA** (all, print, screen i speech) praćena **GRUPNIM PRAVILOM** koje se može sastojati od JEDNOG ili VIŠE **LOGIČKIH IZRAZA**.
- **LOGIČKI IZRAZI** (engl. *expressions*) mogu uzeti samo vrednosti BINARNE VREDNOSTI (istina ili laž) i mogu se kombinovati rečima **and**, **not** i **only** koje predstavljaju LOGIČKE FUNKCIJE.
- Opšti oblik media upita se može predstaviti na sledeći način:

```
@media not|only mediatype and (expressions) {  
    // specifična stilizacija -CSS-kod;  
}
```

- Ideja adaptacije izgleda Web sadržaja se svodi na formiranje **RAZLIČITIH STILOVA** za **RAZLIČITE KARAKTERISTIKE** media.
- Uslovno UVOŽENJE eksternog stila **Pravila\_A** se može obaviti na sledeći način:

```
<link rel="stylesheet" media="mediatype and|not|only (expressions)" href="Pravila_A.css">
```

# Specifičnosti media upita

- Rezultat media upita je **ISTINIT** samo ako se navedeni tip medija **PODUDARA** sa tipom uređaja na kojem se dokument prikazuje i **SVI** izrazi u medijskom upitu su istiniti.
- **LOGIČKI IZRAZ** se može formirati od niza **SPECIFIČNOSTI** terminalnog uređaja koje su od interesa za prikaz.
- Neke od najčešće korišćenih specifičnosti su:
  - **grid** – Terminalni uređaj podržava mrežu (engl. *grid*) ili bitmapu;
  - **height** – Visina vidljivog dela Web stranice (engl. *viewport*);
  - **max-height** – Maksimalna visina područja prikaza (prozor Web čitača);
  - **max-resolution** – Maksimalna rezolucija uređaja;
  - **min-resolution** – Minimalna rezolucija uređaja;
  - **max-width** – Maksimalna širina područja prikaza (prozor Web čitača);
  - **min-aspect-ratio** – Minimalni odnos između širine i visine područja prikaza;
  - **min-device-width** – Minimalna širina uređaja (ekran računara);
  - **min-device-height** – Minimalna visina uređaja (ekran računara);
  - **orientation** – Orijentacija viewport-a (landscape ili portrait mod);
  - **resolution** – Rezolucija izlaznog uređaja.

# Primer media upita

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body { background-color: lightblue;}
      @media screen and (min-width: 400px)
        body {background-color: lightgreen;}
    </style>
  </head>
  <body>
    <p>Media upit će biti primenjen jedino ako je media tip screen i viewport 400px
    ili veći.</p>
  </body>
</html>
```

# Redosled elemenata na Web stranici

- Redosled elemenata na Web stranici se podrazumevano definiše prema njihovom tipu koji može biti:
  - Tip **BLOKA** (engl. *block*)
  - Tip **REDA** (engl. *inline*)
- Elementi tipa **BLOKA** se postavljaju **JEDAN ISPOD DRUGOG**, dok se elementi tipa **REDA** postavljaju **JEDAN PORED DRUGOG**.
- Web stranica predstavlja niz – grupu okvira postavljenih u skladu sa odgovarajućim CSS pravilima.
- CSS pravila se najčešće primenjuju na niz - grupu okvira kada se nazivaju **MODEL OKVIRA**.
- U primeni su dva modela okvira:
  - **tradicionalni** model okvira
  - **fleksibilni** model okvira (CSS 3)

# CSS3 Fleksibilni model okvira

- ▶ **NOVA CSS3** svojstva za **POZICIONIRANJE** objekta:
  - ▶ **Display:**
    - ▶ **Inline, Inline Block, Block**
  - ▶ **Table:**
    - ▶ **Table Cell, Table Row, Flex, Inline Flex**
  - ▶ **Float:**
    - ▶ **Left, Right, None**
  - ▶ **Flexbox:**
    - ▶ **Flex Container, Flex Direction, Justity Content, Align Items, Align Content, Wrap, Align Self, Order, Margin Auto.**
  - ▶ **Visibility:** **visible, invisible.**



# Tradicionalni model okvira

- ▶ Da bi se **MODEL OKVIRA** uspešno primenio mora se **RAZDVOJITI STRUKTURA** HTML dokumenta od njegove **PREZENTACIJE**.
- ▶ Pomoću **<div>** elemenata prave se okviri na ekranu kojima se dodeljuje:
  - ▶ pozicija,
  - ▶ veličina,
  - ▶ boja,
  - ▶ ivice, ...
- ▶ **Web aplikacije** i **Ajax** tehnologija su imale značajan uticaj na koncept modela okvira.
- ▶ Osnovna ideja koju su oni doneli je da se okviri „**omotavaju**“ oko drugih okvira kako bi se rasporedili horizontalno.
- ▶ Čitavo telo Web stranice se formira kao **GRUPA OKVIRA** koje treba centrirati i čija veličina mora da se definiše kao određena vrednost.
- ▶ HTML oznaka **<div>** se koristi kao **OMOTAČ** grupe okvira.

# Deklarisanje novih HTML 5 elemenata

- ▶ Da bi se osiguralo ispravno tretiranje novih HTML 5 elemenata od strane Web čitača, treba ih deklarirati kao elemente bloka korišćenjem oznake **<div>**.

- ▶ Primer deklaracije novih HTML 5 oznaka:

```
header, section, footer, aside, nav, article, figure, figcaption, hgroup
{
    display: block;
}
```

- ▶ Svojstvo **display** može uzeti vrednosti: **block**, **inline** i **inline-block**.
- ▶ Prvi element modela okvira je uvek **<body>** i uvek se postavlja horizontalno.
- ▶ Primer oznake body koja zauzima 100% širine Web stranice.

```
body
{
    text-align: center;
}
```

# Fleksibilni model okvira (1)

- ▶ Većinu probleme koji nastaju primenom tradicionalnih okvira treba da reši **FLEKSIBILNI MODEL**.
- ▶ Kod ovog modela, okviri predstavljaju **VIRTUELNE REDOVE I KOLONE**.
- ▶ Ovo će omogućiti potpuno **UPRAVLJANJE** rasporedom, položajem i veličinom okvira, rasporedom okvira unutar drugih okvira kao i načinom na koji oni **DELE I KORISTE** dostupan prostor na stranici.
- ▶ Dimenzije elemenata se se **SMANJUJU** ili **POVEĆAVAJU** u zavisnosti od DOSTUPNOG PROSTORA.
- ▶ Fleksibilni kontejner je element koji omogućava **PRILAGODLJIVOST** njegovog sadržaja.
- ▶ Kod fleksibilnog kontejnera se uvode pojmovi **HORIZONTALNA** i **VERTIKALNA ORIJENTACIJA**.
- ▶ Fleksibilni okvir se definiše svojstvom **display** koje uzima vrednost **flex** za element bloka, za element reda uzima se vrednost **inline-flex**.
- ▶ U fleksibilnom kontejneru orijentacija sadržaja se opisuje glavnom (engl. *main axis*) i poprečnom (engl. *cross*) osom.
- ▶ Po glavnoj osi se raspoređuje sadržaj, a poprečna je vertikalna na nju.

## Fleksibilni model okvira (2)

- ▶ Definišu se počeci i krajevi glavne i poprečne ose.
- ▶ Da bi se neki element u fleksibilnom kontejneru učinio fleksibilnim, mora se svojstvu **flex** dodeliti vrednost definisanu sa tri parametra **flex-grow**, **flex-shrink** i **flex-basis**.
- ▶ Fleksibilni model okvira spada u **NAPREDNE TEHNIKE** adaptacije Web stranice.