

Ajzenhamer Nikola  
Bukurov Anja

# UVOD U VEB I INTERNET TEHNOLOGIJE

13. oktobar 2018.

# Sadržaj

<b>Predgovor</b>	<b>3</b>
<b>1 Čas 3</b>	<b>5</b>
1.1 Uvod u HTML	5
1.1.1 Elementi i atributi, komentari	5
1.1.2 Struktura HTML dokumenta	5
1.1.3 HTML zaglavlje	6
1.1.4 HTML telo	7
1.1.5 HTML5 semantički elementi ( header , nav , ...)	7
1.1.6 Elementi za naslove ( h1 , ..., h6 )	8
1.1.7 Elementi p i span	8
1.1.8 Blokovski i linijski elementi	8
1.1.9 Grupisanje elemenata ( div ) i novi red ( br )	9
1.2 Uvod u CSS	11
1.2.1 CSS sintaksa	12
1.2.2 Dodeljivanje boje elementu	12
1.2.3 Formatiranje teksta	13
1.2.4 Dodavanje korisničkih fontova	14
1.3 Model kutije	17
1.3.1 Sadržaj elementa	18
1.3.2 Punjenje elementa	19
1.3.3 Ivica elementa	20
1.3.4 Pojas elementa	22
<b>2 Čas 4</b>	<b>27</b>
2.1 Pozicioniranje elemenata	27
2.1.1 Statičko pozicioniranje	27
2.1.2 Relativno pozicioniranje	27
2.1.3 Apsolutno pozicioniranje	27
2.1.4 Fiksno pozicioniranje	27
2.1.5 Z-pozicioniranje elemenata	27
2.1.6 Svojstvo display	27
<b>Dodatak</b>	<b>29</b>
Lista kodova	29



## Predgovor

Ovaj tekst predstavlja skriptu iz kursa "Uvod u Veb i Internet Tehnologije", na 2. godini smera Informatika na Matematičkom fakultetu Univerziteta u Beogradu. Skripta je koncipirana na osnovu materijala koleginice [Andelke Zečević](#) sa časova vežbi. Skripta je prateći materijal pre svega studentima koji ovaj kurs slušaju u okviru svojih studija, ali i svima Vama koji biste želeli da se upoznate sa ovom tematikom. Ovaj materijal ne može zameniti pohađanje vežbi niti drugu preporučenu literaturu.

Ovaj tekst je u ranoj fazi formiranja. Ukoliko ste pažljivi čitalac ove skripte, i ukoliko uočite bilo kakvu grešku ili propust, možete se javiti autorima na adresu **[nikola\\_ajzenhamer@math.rs](mailto:nikola_ajzenhamer@math.rs)**. Svi komentari, sugestije, kritike, ali i pohvale vezane za ovaj materijal su dobrodošli.

Autori



# 1 Čas 3

Cilj ovog časa je upoznavanje studenata sa osnovnim elementima jezika za obeležavanje teksta HTML i CSS, kao i sa modelom kutije u jeziku CSS.

## 1.1 Uvod u HTML

*HTML (HyperText Markup Language)* predstavlja najosnovniji blok za izgradnju Veba. Njime se opisuje logički sadržaj stranice<sup>1</sup>. Na primer, u HTML-u zadajemo koji deo stranice čini naslov ili paragraf, ili gde se nalazi slika.

### 1.1.1 Elementi i atributi, komentari

HTML stranica se izgrađuje od velikog broja HTML *elemenata*. Neki od njih definišu meta-informacije (odnosno, informacije o samoj stranici), a drugi definišu konkretan sadržaj. Struktura jednog elementa se sastoji od dve *etikete*, otvarajuće i zatvarajuće, između kojih se nalazi nekakav sadržaj (tekst ili ugnežđeni elementi):

```
<p>      <!-- Otvarajuca etiketa za element "p" -->
...      <!-- Sadrzaj elementa -->
</p>      <!-- Zatvarajuca etiketa za element "p" -->
```

Takođe, postoje elementi koji imaju samozatvarajuću etiketu. Njihova struktura je oblika:

```
<img>      <!-- Samozatvarajuca etiketa za element "img" -->
```

U nastavku teksta ćemo, kada uvodimo novi element, podrazumevati da je reč o elementu koji ima otvarajuću i zatvarajuću etiketu. Posebno ćemo naglasiti ukoliko je reč o samozatvarajućem elementu.

U prethodnim primerima smo videli i primer HTML komentara. Svaki HTML komentar počinje oznakom `<!--` i završava se oznakom `-->`. Sve što je unutar ove dve oznake neće biti prikazano na stranici, slično kao što komentari u programskom jeziku C ne utiču na kompiliranje i izvršavanje koda.

Elemente možemo dodatno okarakterisati *atributima* oblika `atribut="vrednost"`. Atributi se navode u okviru otvorene etikete. Na primer,

```
<body id="telo">
</body>
```

Postoje generički atributi koji se mogu primeniti na svim elementima, a postoje i specifični atributi za pojedine elemente. Na primer, svi elementi mogu imati atribut `id`, dok samo element `form` može imati atribut `onsubmit`.

### 1.1.2 Struktura HTML dokumenta

Osnovna struktura HTML dokumenta podrazumeva narednu strukturu:

```
<!DOCTYPE html>

<html>
  <head>
    <title>Naslov</title>
    <meta charset="UTF-8">
```

<sup>1</sup>HTML stranice se čuvaju kao HTML dokumenti, tj. sa ekstenzijom `html`.

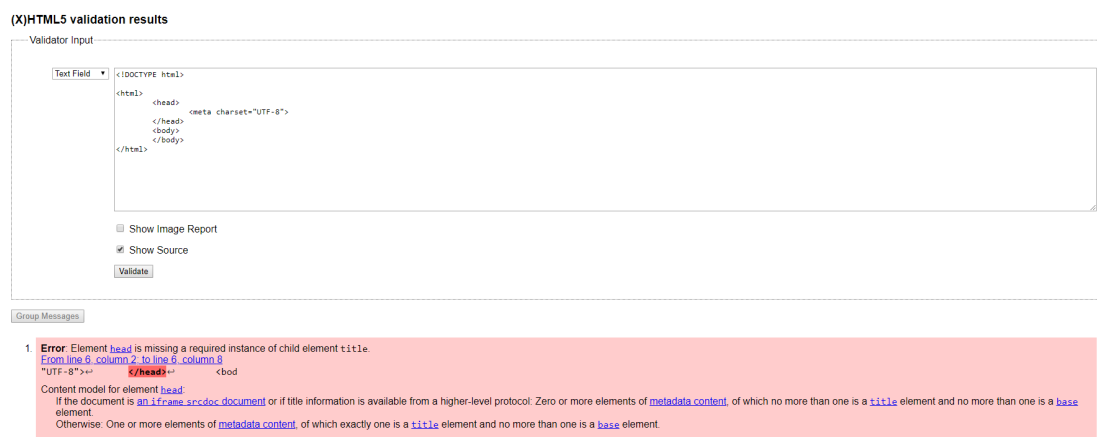
```
</head>
<body>
</body>
</html>
```

Deklaracija dokumenta `<!DOCTYPEhtml>` predstavlja posebnu vrstu elementa kojom se definiše verzija HTML dokumenta koja se koristi. Konkretno, ovom deklaracijom navodimo da ćemo koristiti HTML5 verziju. Elementom `html` definišemo sadržaj html dokumenta. HTML dokumenti imaju dva glavna dela:

- *zaglavlje* — sadrži informacije o samom dokumentu, kao i linkove ka definicijama stilova (CSS) i programskih elemenata (JavaScript). Ove informacije služe da opišu dokument i one nisu vidljive na stranici. Primeri informacija su: naslov, opis, jezik, ključne reči, itd. Zaglavlje je definisano elementom `head`.
- *telo* — sadrži vizuelne elemente koji će biti prikazani korisniku. Sve što se nalazi u telu dokumenta je vidljivo u okviru veb pregledača. Takođe, sve što želimo da bude vidljivo, moramo da stavimo u telo dokumenta. Telo je definisano elementom `body`.

Element `title` služi za davanje semantičkog naslova dokumenta. Ovaj naslov će se prikazivati u većini veb pregledača kao ime kartice u kojoj je dokument otvoren, ali i u rezultatima pretrage. Element `meta` nudi različite mogućnosti za dodavanje metainformacija dokumentu. Na primer, atributom `charset="UTF-8"` postavlja se UTF-8 kao kodna shema za dokument. U nastavku ćemo videti još neke primere metainformacija.

U prethodnom primeru takođe vidimo obaveznu strukturu svakog HTML dokumenta, definisanu HTML specifikacijom. Možemo testirati da li naš html dokument predstavlja validno napisan dokument pomoću HTML validatora. Jedan primer validatora je moguće pronaći na adresi <https://html5.validator.nu/>. Na primer, da smo izostavili element `title`, validator bi se pobunio i prikazao nam odgovarajuću poruku:



### 1.1.3 HTML zaglavlje

Kao što smo rekli, zaglavlje HTML dokumenta predstavljeno je elementom `head`. Takođe, upoznali smo se sa dva važna elementa zaglavlja: `title`, koji definiše naslov dokumenta i `meta` koji zadaje metainformacije o dokumentu. Najveći broj

metainformacija se zadaje u obliku para atributa `name="..."` i `content="..."`. U sledećoj tabeli dat je spisak nekih metainformacija i primer njihovog korišćenja:

Naziv meta-informacije	Primer upotrebe
language	<code>&lt;meta name="language" content="sr"&gt;</code>
author	<code>&lt;meta name="author" content="Ajzenhamer N, Bukurov A."&gt;</code>
keywords	<code>&lt;meta name="keywords" lang="sr" content="html, css, uvit"&gt;</code>
description	<code>&lt;meta name="description" content="Primer HTML i CSS datoteke"&gt;</code>
generator	<code>&lt;meta name="generator" content="Visual Studio Code"&gt;</code>

#### 1.1.4 HTML telo

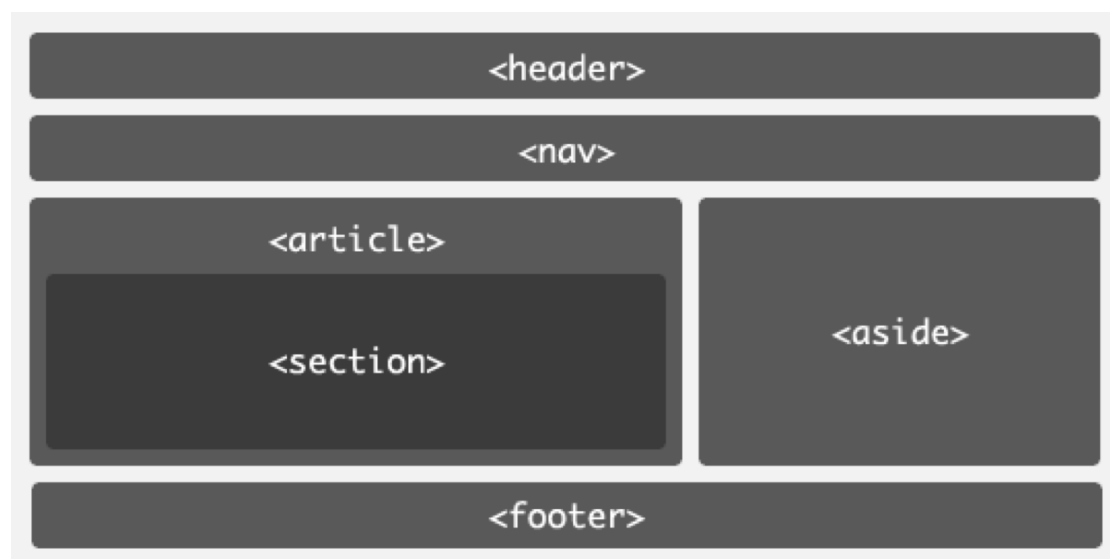
Telo HTML dokumenta je određeno elementom `body`. U okviru njega, potrebno je da vršimo ugnežđavanje elemenata koje želimo da prikazemo korisniku. Elementa ima raznih, i u nastavku ćemo se upoznati sa velikim brojem ovih elemenata.

#### 1.1.5 HTML5 semantički elementi (`header`, `nav`, ...)

HTML5 je uveo nove elemente koji označavaju glavne delove HTML dokumenata. Ti elementi su:

- `header` — element za naslovni sadržaj
- `nav` — element za navigaciju
- `main` — glavni sadržaj
- `footer` — element za završni sadržaj
- ...

Ovi elementi definišu isključivo semantičku organizaciju elemenata na stranici, odnosno, oni dodeljuju odgovarajuće značenje elementima koje sadrže i ne podrazumevaju nikakav poseban način prikazivanja tih elemenata. Naredna slika daje primer upotrebe ovih elemenata.





### 1.1.6 Elementi za naslove ( `h1` , ..., `h6` )

Tekst možemo istaknuti u vidu naslova koristeći elemente `<h1>` do `<h6>`. Oni takođe dodeljuju semantički značaj tekstu koji obeležavaju. Takođe, oni podrazumevaju i odgovarajući prikaz teksta koji je dat na sledećoj slici

# Naslov 1

## Naslov 2

### Naslov 3

#### Naslov 4

##### Naslov 5

###### Naslov 6

### 1.1.7 Elementi `p` i `span`

Pored naslova, tekst možemo obeležiti i elementima:

- `p` — definiše paragraf teksta. Podrazumevano, tekst obeležen ovim elementom se prostire celom širinom elementa u kojem se nalazi. Tako, na primer, ukoliko se nalazi u elementu `body`, tekst će se prostirati celom širinom prozora veb pregledača:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris ac ex in eros placerat elementum vel ac purus. Pellentesque iaculis, massa sed sollicitudin tincidunt, dolor lacus fermentum nibh, dictum ornare nulla ipsum a orci. Nulla imperdiet, est ac auctor tincidunt, tellus est condimentum risus, in tincidunt sem lectus eget enim. Nullam ut sapien ut libero tristique blandit. Pellentesque vitae lectus ultrices, placerat odio a, bibendum diam. Sed lobortis, felis in sagittis gravida, lectus ipsum laoreet lorem, sed imperdiet arcu augue id tellus. Aliquam finibus vulputate dui, a gravida felis egestas lacinia. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Duis eu sapien a mauris pellentesque ullamcorper et quis lacus.

- `span` — definiše deo teksta, najčešće jednu reč ili deo rečenice (ali ne nužno). Veličina elementa zavisi od veličine teksta koji se nalazi kao njegov sadržaj.

### 1.1.8 Blokovski i linijski elementi

Prethodno opisani elementi su predstavnici dveju bitnih kategorija elemenata: blokovski i linijski<sup>2</sup>. Veb pregledači *blokovske elemente* tipično prikazuju sa praznim redom pre i nakon elementa. Možemo ih vizualizovati kao naslagane kutije. Blok elementi zauzimaju čitavu širinu elementa u kojem se nalaze. Za razliku od njih, *linijski elementi* mogu da počinju bilo gde u liniji (ne nužno u novom redu), i njihova veličina zavisi od njihovog sadržaja. Blokovski elementi mogu da sadrže linijske elemente i (obično) druge blok elemente, dok linijski elementi mogu da sadrže samo linijske elemente.

<sup>2</sup>Ove klase su definisane standardom pre HTML5, dok se u verziji HTML5 definišu nove kategorije.

Na primeru opisanih elemenata možemo demonstrirati ovo ponašanje. Na narednoj slici su obeleženi tekst koji se nalazi u okviru elementa `p` (levo) i tekst koji se nalazi u okviru elementa `span` (desno)

The following paragraph is a

**block-level element;**

its background has been colored to display both the beginning and end of the block-level element's influence.

The following span is an **inline element**; its background has been colored to display both the beginning and end of the inline element's influence.

### 1.1.9 Grupisanje elemenata (`div`) i novi red (`br`)

Grupisanje elemenata najčešće služi za izdvajanje logičkih celina, koje se potom veoma često i vizualno izdvajaju od ostalih celina. HTML5 semantički elementi omogućavaju jedan način grupisanja elemenata, na način na koji smo već videli. Dodatno, na raspolaganju nam je blokovski element `div`, koji takođe ne podrazumeva poseban način prikazivanja (osim što je u pitanju blokovski element, te ima njihova svojstva), već pomoću njega možemo da izdvojimo i, najčešće, imenujemo celine koje ćemo stilizovati na neki način.

Forsiranje prelaska u novi red se, pored korišćenja blokovskih elemenata, može izvesti i upotrebom samozatvarajućeg elementa `br`.

#### Zadatak 1.1:

Napisati HTML datoteku koja odgovara stranici na narednoj slici

## HTML i CSS

- Uvod -

### HTML

HTML (enlg. HyperText Markup Language, jezik za označavanje hiperteksta) je opisni jezik specijalno namenjen opisu web stranica. Pomoću njega se jednostavno mogu odvojiti elementi kao što su naslovi, paragrafi, citati i slično. Pored toga, u HTML standard su ugrađeni elementi koji detaljnije opisuju sam dokument kao što su kratak opis dokumenta, ključne reči, podaci o autoru i slično. Ovi podaci su opštepoznati kao meta podaci i jasno su odvojeni od sadržaja dokumenta. Aktuelna verzija standarda je HTML 5, a sam standard održava Konzorcijum za Веб (W3C, World Wide Web Consortium).

### CSS

CSS sintaksa se sastoji od opisa izgleda elemenata u dokumentu. Opis može da definiše izgled viših elemenata, a viših opisa može da definiše jedan element. Na taj način se opisi slažu jedan preko drugog da bi definisali konačni izgled određenog elementa (otuda naziv Cascading da bi se дочарало sлагање једног стила преко другог у дефинисању коначног изгледа елемента).

15. oktobar 2018.

### Rešenje:

Kod 1: `zadatak1/strana.html`

```
<!DOCTYPE html>

<html>
  <head>
    <title>HTML i CSS</title>
    <meta charset="UTF-8">
    <meta name="author" content="Ajzenhamer N, Bukurov A.">
    <meta name="generator" content="Visual Studio Code">
    <meta name="keywords" content="css, html, uvod, veb, uvit">
    <meta name="description" content="Uvod u HTML i CSS">
  </head>
```

```
<body>
  <!-- primer HTML komentara -->

  <header>
    <h1>HTML i CSS</h1>
    <h4>— Uvod —</h4>
  </header>

  <main>
    <!-- Deo o HTML-u -->
    <div>
      <h3>HTML</h3>
      <p>
        HTML (engl. HyperText Markup Language, jezik za označavanje
        hiperteksta) je opisni jezik specijalno namenjen opisu
        veb stranica. Pomocu njega se jednostavno mogu odvojiti
        elementi kao što su naslovi, paragrafi, citati i slično.
        Pored toga, u HTML standard su ugrađeni elementi koji
        detaljnije opisuju sam dokument kao što su kratak opis
        dokumenta, ključne reči, podaci o autoru i slično. Ovi
        podaci su opštepoznati kao meta podaci i jasno su
        odvojeni od sadržaja dokumenta.

        Aktuelna verzija standarda je HTML 5, a sam standard
        održava Konzorcijum za Veb (W3C, World Wide Web
        Consortium).

      </p>
    </div>

    <!-- Deo o CSS-u -->
    <div>
      <h3>CSS</h3>
      <p>
        CSS sintaksa se sastoji od opisa izgleda elemenata u
        dokumentu. Opis može da definiše izgled više elemenata,
        i više opisa može da definiše jedan element. Na taj
        način se opisi slažu jedan preko drugog da bi definisali
        konačni izgled određenog elementa (otuda naziv
        Cascading da bi se dočaralo slaganje jednog stila preko
        drugog u definisanju konačnog izgleda elementa).

      </p>
    </div>
  </main>

  <footer>
    <p>15. oktobar 2018.</p>
  </footer>

</body>
</html>
```

## 1.2 Uvod u CSS

CSS (engl. *Cascading Style Sheets*) je jezik koji služi za stilizovanje HTML elemenata<sup>3</sup>. CSS možemo koristiti na tri načina:

1. Navođenjem stilova u odvojenim CSS datotekama (uz njihovo linkovanje u HTML datoteci). Element oblika

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

navodi se u okviru `head` elementa HTML dokumenta. Ovim se omogućava da više HTML datoteka može da pristupi jednom CSS dokumentu (čime se postiže modularnost i smanjenje redundantnih stilova).

2. Putem HTML elementa `style` (koji se takođe navodi u okviru `head` elementa koji ima atribut `type="text/css"`), u čijem sadržaju se nalaze stilovi. Navedeni stilovi važe za ceo dokument u čijem zaglavlju se nalaze, ali nisu vidljivi drugim HTML dokumentima.
3. Korišćenjem generičkog atributa `style` nad elementima<sup>4</sup>. Stilovi definisani na ovakav način su vidljivi i važe isključivo nad pojedinačnim elementom nad kojim su definisani.

Ono što je važno napomenuti jeste da često dolazi do preklapanja stilova. Na primer, moguće je nad celim dokumentom definisati da svi paragrafi budu obojeni crvenom bojom, a zatim posebno definisati za jedan paragraf da je obojen plavom bojom. U tu svrhu, definisana su pravila kojim se određuje koje svojstvo dolazi do izražaja u slučaju preklapanja.

Najveću težinu imaju stilovi definisani korišćenjem atributa `style`, zatim atributi definisani u zaglavlju HTML dokumenta (korišćenjem elementa `style` i ugnežđavanjem eksternih CSS dokumenata), i na kraju, najmanju težinu imaju podrazumevane vrednosti veb pregledača.

Takođe, potrebno je obratiti pažnju na redosled elemenata `style` i `link` (koji vrši ugnežđavanje eksternog CSS dokumenta) u zaglavlju HTML dokumenta. Na primer, pretpostavimo da eksterna CSS datoteka `plavi_naslov.css` definiše da naslov `h1` treba biti obojen u plavu boju. Takođe, neka je dat element `style` u zaglavlju koji definiše da naslov `h1` treba biti obojen u narandžastu boju. Na narednoj slici su dati rezultati kada se element `style` nalazi nakon elementa `link` (levo) i kada se element `link` nalazi nakon elementa `style` (desno). Vidimo da veću težinu ima ono pravilo koje je poslednje pronađeno (tj. ono je kaskadno poslednje složeno).

**This is a heading**

The style of this document is a combination of an external stylesheet, and internal style

**This is a heading**

The style of this document is a combination of an external stylesheet, and internal style

Ipak, postoji način da se forsira određeni stil, a to je navođenjem `!important` ključne reči u okviru CSS deklaracije. Korišćenjem ove ključne reči, navedena deklaracija će uvek imati prednost nad svim ostalima. Više o deklaracijama u nastavku teksta.

<sup>3</sup>Kaskadni stilske listovi se čuvaju kao CSS dokumenti, tj. sa ekstenzijom `css`.

<sup>4</sup>Da li razumete razliku između elementa `style` i atributa `style`?

### 1.2.1 CSS sintaksa

CSS jezik se sastoji od niza *pravila*. Svako pravilo je oblika:

```
selektor {
    svojstvo-1: vrednost_1;      /* deklaracija 1 */
    ...
    svojstvo-N: vrednost_N !important; /* deklaracija N */
}
```

Objasnimo sada svaku od predstavljenih stavki:

- *Selektor* predstavlja način za definisanje nad kojim HTML elementima će stil biti primenjen. Postoji veliki broj selektora, a mi ćemo predstaviti neke od njih:

Selektor	Primer	Objašnjenje
<i>Ime elementa</i>	<code>p</code>	Primeniće stil nad svim elementima <code>p</code>
<i>Identifikator</i>	<code>#moj-id</code>	Primeniće stil nad svim elementima koji imaju postavljen generički atribut <code>id="moj-id"</code> . Poželjno je da na nivou dokumenta postoji samo jedan element koji ima svoj identifikator, čime se on jedinstveno određuje.
<i>Klasa</i>	<code>.moja-kl</code>	Primeniće stil nad svim elementima koji imaju postavljen generički atribut <code>class="moja-kl"</code> . Stilovi se definišu pomoću klasa kada je potrebno na isti način predstaviti vizualno srodne elemente na stranici, na primer, kartice sa obaveštenjima.
<i>Ugnežđeni</i>	<code>sel1 sel2</code>	Primeniće stil na sve elemente odabrane sa <code>sel2</code> , a koji se nalaze u okviru elemenata odabranih sa <code>sel1</code> .
<i>Kompozicija</i>	<code>div p.m-kl</code>	Primeniće stil na sve pasuse (elemente <code>p</code> ) sa atributom <code>class="m-kl"</code> , a koji se nalaze u okviru elemenata <code>div</code> .
<i>Grupisanje</i>	<code>div, p</code>	Primeniće stil na sve elemente <code>div</code> i sve elemente <code>p</code> .

- *Deklaracija* opisuje način na koji se vrši stilizovanje selektovanih elemenata. Deklaracije se međusobno razdvajaju karakterom `;`.
- Svaka deklaracija sadrži par *svojstvo* i *vrednost*. Svaki element ima definisana svojstva koja je moguće primeniti nad njime. Takođe, svako svojstvo ima definisan opseg vrednosti koje može uzimati. U nastavku teksta ćemo se upoznati za raznim svojstvima i odgovarajućim vrednostima.

Komentari se u jeziku CSS navode između `/*` i `*/`.

Možemo testirati da li naš CSS dokument predstavlja validno napisan dokument pomoću CSS validatora. Jedan primer validatora je moguće pronaći na adresi <http://jigsaw.w3.org/css-validator/>.

### 1.2.2 Dodeljivanje boje elementu

U CSS jeziku, za dodeljivanje boje elementu možemo koristiti sledeća dva svojstva:

- `color` — zadaje boju teksta koji se nalazi u sadržaju elementa
- `background-color` — zadaje boju pozadine elementa

Vrednosti za boje se mogu specificovati na više načina, a mi navodimo sledeće:

- Pomoću imena boje, na primer, `Red`, `Orange`, `DodgerBlue`, `Violet`, `MediumSeaGreen`, itd. Spisak svih dostupnih imena boja je dostupan na [ovoj adresi \(w3schools.com\)](http://www.w3schools.com).
- Pomoću RGB i RGBA modela. Ovim modelima je potrebno zadati vrednosti iz opsega [0,255] za crvenu (R), zelenu (G) i plavu (B) boju čijim se "mešanjem" dobija oko 16,7M boja. Varijanta RGBA dodatno omogućava da se podesi prozirnost boje (alfa kanal, A) u opsegu [0.0, 1.0]. Na primer,

```
color: rgb(255, 0, 0); /* crvena boja */
background-color: rgba(0, 255, 0, 0.5); /* zelena boja sa prozirnoscu od 50% */
```

- Pomoću HEX modela. Ovim modelom se za svaku od crvene, zelene i plave, redom, navodi dvocifreni heksadekadni broj iz intervala [00 – FF]. Na primer,

```
color: #0000ff; /* plava boja */
```

Više o modelima boja se može pronaći na [ovoj adresi \(w3schools.com\)](http://www.w3schools.com).

### 1.2.3 Formatiranje teksta

Sada ćemo se upoznati sa nekim svojstvima kojima je moguće uticati na prikaz teksta:

- `font-family` — zadaje familiju fonta koji se koristi. Familija fonta se može zadati po imenu, na primer `Arial` ili `'DejaVu Serif'` (ukoliko ima znak razmaka, stavljaju se navodnici), ili se može nazvati generička familija fontova, pa se ostavlja veb pregledaču da odabere konkretan font iz te familije koji mu je dostupan. Na primer, `Sans-Serif`. Takođe, moguće je staviti više fontova odvojenih karakterom `,` da bi veb pregledač, u slučaju da mu nije dostupan neki font, prikazao odgovarajuću zamenu. Na primer

```
font-family: 'Neki kul font koji mozda nije dostupan', Arial, Sans-Serif;
```

- `font-size` — podešava veličinu slova. Vrednosti se mogu iskazati u pikselima (`px`), u `em` (važi  $1em = 16px$ ), ili u procentima, pri čemu se razmatra okolina teksta da bi se zaključio odnos. Ove jedinice se nazivaju *dužinama*, i više o njima se može pronaći na [ovoj adresi \(w3schools.com\)](http://www.w3schools.com).
- `font-style` — kontroliše iskošenosti slova. Vrednosti su: `normal` (uspravna slova), `italic` (iskošena slova) i `oblique` (nakošena slova, slično kao `italic`, ali manje podržano od veb pregledača).

- `font-weight` — kontroliše potamnijavanje slova. Vrednosti su: `normal` (podrazumevana debljina), `bold` (deblje), `bolder` (više podebljano), `lighter` (tanje) ili brojevi od 100 do 900 (sa korakom 100), pri čemu broj 400 odgovara vrednosti `normal`, a 700 odgovara vrednosti `bold`.
- `text-align` — kontroliše horizontalno poravnanje teksta. Vrednosti su: `left` (levo poravnato), `right` (desno poravnato), `center` (centrirano) i `justify` (poravnato sa obe strane).
- `letter-spacing` — podešava rastojanje između slova. Vrednosti su: `normal` (podrazumevano rastojanje) ili neka od dužina.
- `word-spacing` — podešava rastojanje između reči. Vrednosti su: `normal` (podrazumevano rastojanje) ili neka od dužina.
- `text-indent` — kontroliše nazubljanje prve linije teksta. Vrednosti su neka od dužina.

Više o fontovima se može pronaći na [ovoj adresi \(w3schools.com\)](http://w3schools.com), dok se više o tekstualnim promenama može pronaći na [ovoj adresi \(w3schools.com\)](http://w3schools.com).

#### 1.2.4 Dodavanje korisničkih fontova

Ukoliko želimo da koristimo svoj font, možemo ga definisati u CSS dokumentu pomoću `@font-face` naredbe. Njena struktura je oblika:

```
@font-face {  
  font-family: 'Naziv fonta';  
  src: url('./lokacija/do/mog/fonta/Naziv Fonta.ttf');  
}
```

Svojstvom `font-family` dodeljujemo naziv korisničkom fontu. Dalje se font može koristiti pomoću već opisanog postupka korišćenjem svojstva `font-family`. Svojstvom `src` zadajemo putanju ka datoteci koja sadrži informacije o fontu. Ona se može nalaziti u lokalnom računaru ili na nekoj veb lokaciji, ali se u oba slučaja navodi korišćenjem vrednosti `url('...')`.

#### Zadatak 1.2:

Napisati HTML datoteku i CSS datoteku koja odgovara stranici na narednoj slici

# HTML I CSS

## - Uvod -

### HTML

HTML (engl. **HyperText Markup Language**, jezik za označavanje hiperteksta) je opisni jezik specijalno namenjen opisu veb stranica. Pomocu njega se jednostavno mogu odvojiti elementi kao što su naslovi, paragrafi, citati i slično. Pored toga, u HTML standard su ugrađeni elementi koji detaljnije opisuju sam dokument kao što su kratak opis dokumenta, ključne reči, podaci o autoru i slično. Ovi podaci su opštepoznati kao meta podaci i jasno su odvojeni od sadržaja dokumenta. Aktuelna verzija standarda je HTML 5, a sam standard održava Konzorcijum za Veb (W3C, World Wide Web Consortium).

### CSS

CSS sintaksa se sastoji od opisa izgleda elemenata u dokumentu. Opis može da definiše izgled više elemenata, i više opisa može da definiše jedan element. Na taj način se opisi slažu jedan preko drugog da bi definisali konačni izgled određenog elementa (otuda naziv *Cascading* da bi se dočaralo slaganje jednog stila preko drugog u definisanju konačnog izgleda elementa).

15. oktobar 2018.

## Rešenje:

### Kod 2: zadatak2/strana.html

```
<!DOCTYPE html>

<html>
  <head>
    <title>HTML i CSS</title>
    <meta charset="UTF-8">
    <meta name="author" content="Ajzenhamer N, Bukurov A.">
    <meta name="generator" content="Visual Studio Code">
    <meta name="keywords" content="css, html, uvod, veb, uvit">
    <meta name="description" content="Uvod u HTML i CSS">

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

    <style type="text/css">
      /*
        Primer CSS komentara
      */
    </style>
  </head>

  <body>
    <!-- primer HTML komentara -->

    <header>
      <h1 id="glavni_naslov">HTML i CSS</h1>
      <h4 id="podnaslov">— Uvod —</h4>
    </header>

    <main>
      <!-- Deo o HTML-u -->
      <div>
        <h3>HTML</h3>
        <p class="intro html">
          HTML (engl. <span style="color: #DA5FA6;">HyperText Markup
            Language</span>, jezik za označavanje hiperteksta) je
```



opisni jezik specijalno namenjen opisu veb stranica. Pomocu njega se jednostavno mogu odvojiti elementi kao što su naslovi, paragrafi, citati i slicno. Pored toga, u HTML standard su ugrađeni elementi koji detaljnije opisuju sam dokument kao što su kratak opis dokumenta, ključne reči, podaci o autoru i slicno. Ovi podaci su opštepoznati kao meta podaci i jasno su odvojeni od sadržaja dokumenta.

Aktuelna verzija standarda je HTML 5, a sam standard održava Konzorcijum za Veb (W3C, World Wide Web Consortium).

```

    </p>
  </div>

  <!-- Deo o CSS-u -->
  <div>
    <h3>CSS</h3>
    <p id="css" class="intro">
      CSS sintaksa se sastoji od opisa izgleda elemenata u
      dokumentu. Opis može da definiše izgled više elemenata,
      i više opisa može da definiše jedan element. Na taj
      način se opisi slažu jedan preko drugog da bi definisali
      konačni izgled određenog elementa (otuda naziv
      Cascading da bi se dočaralo slaganje jednog stila preko
      drugog u definisanju konačnog izgleda elementa).

    </p>
  </div>
</main>

<footer>
  <p>15. oktobar 2018.</p>
</footer>

</body>
</html>

```

Kod 3: zadatak2/stil.css

```

@font-face {
  font-family: 'Nas font';
  src: url('cool_font.ttf');
}

#glavni_naslov {
  text-align: center;
  font-family: 'Nas font', Arial, Sans-Serif;
  font-size: 50px;
}

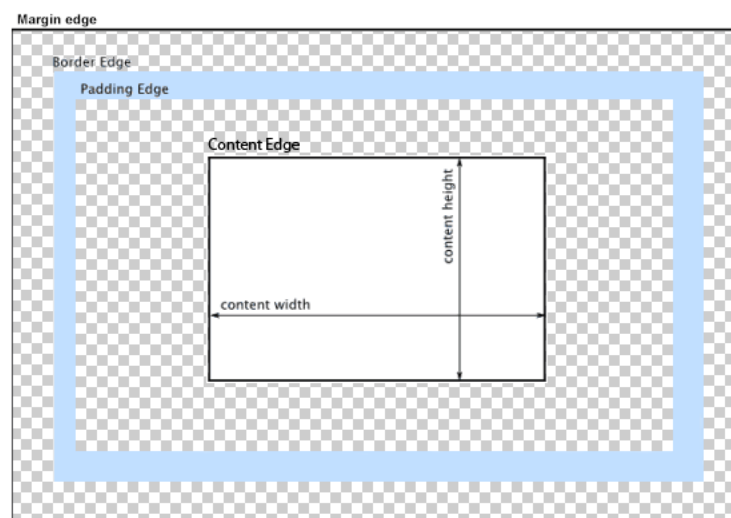
#podnaslov {
  text-align: center;
  font-size: 25px;
  font-family: Sans-Serif;
}

```

```
}  
  
.intro {  
  color: SlateBlue;  
}  
  
.html {  
  font-size: 14px;  
  font-family: 'Tahoma', 'Arial', sans-serif;  
  letter-spacing: 2px;  
  text-indent: 20px;  
  text-align: justify;  
}  
  
#css {  
  color: rgb(60, 154, 109);  
  font-style: italic;  
}  
  
main div h3 {  
  font-weight: bold;  
}  
  
header h1, header h4 {  
  background-color: violet;  
  color: white;  
}
```

### 1.3 Model kutije

Sve HTML elemente možemo posmatrati kao pravougaonike na ekranu, te se stoga zovu ponekad i *kutije*. U jeziku CSS, termin *model kutije* opisuje kutiju koja okružuje sadržaj svakog elementa. Naredna slika demonstrira prikaz modela kutije



Dakle, model kutije se sastoji od sledećih elemenata (elementi su nabrajani tako da svaki sledeći uokviruje sve prethodne):

- *Sadržaj* (engl. *Content*) — kutija u kojoj se prikazuje tekst, slika ili drugi sadržaj HTML elementa.
- *Punjenje* (engl. *Padding*) — prostor oko sadržaja, ali koji se još uvek smatra "unutrašnošću" elementa, s obzirom da se nalazi unutar ivice. Punjenje je uvek prozirno.
- *Ivica* (engl. *Border*) — kutija koja deli unutrašnjost elementa od spoljašnjosti (odn. od drugih elemenata).
- *Pojas* (engl. *Margin*) — prostor oko ivice elementa, koji pre svega služi da stvori bezbedni pojas oko elementa i odvoji ga od okolnih elemenata (otuda i njegov prevod). Pojas je uvek proziran.

### 1.3.1 Sadržaj elementa

Veličina kutije sadržaja zavisi od dve stvari. Prvo, koja je vrsta elementa u pitanju (linijski i blokovski), a zatim i od postavljenih svojstava `width` (širina) i `height` (visina).

Veličina kutije sadržaja linijskih elemenata zavisi isključivo od veličine njihovog sadržaja. Na linijske elemente nije moguće uticati svojstvima `width` i `height`. Naredni primer ilustruje ovo ponašanje

```
<span id="s1">TEST SADRZAJ</span>
<span id="s2">TEST SADRZAJ</span>

#s1 {
  /* height i width svojstvo nemaju nikakvog efekta */
  height: 100px;
  width: 1000px;
  background-color: rgb(193, 22, 193);
}
```

Kao što vidimo na narednoj slici, iako smo podesili visinu i širinu, linijski element `span` je zauzeo samo onoliko prostora koliko mu je neophodno za njegov sadržaj.

TEST SADRZAJ TEST SADRZAJ

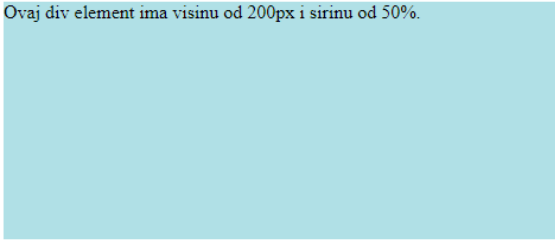
Za razliku od njih, na veličinu sadržaja blokovskih elemenata se može uticati ovim dvama svojstvima. Oni mogu uzimati naredne vrednosti: `auto` (prepušta se veb pregledaču da izračuna širinu/visinu elementa, što je podrazumevano ponašanje), dužina ili procenat. Naredni primer ilustruje ovo ponašanje

```
<body>
<div>
  Ovaj div element ima visinu od 200px i sirinu od 50%.
</div>
</body>
```

```
div {  
  height: 200px;  
  width: 50%;  
  background-color: powderblue;  
}
```

Kao što vidimo na narednoj slici, blokovski element `div` ima ispravno podešenu visinu i širinu.

Ovaj div element ima visinu od 200px i širinu od 50%.



### 1.3.2 Punjenje elementa

Punjenje elementa možemo zadati u jeziku CSS pomoću svojstva `padding` koje istovremeno podešava gornje, desno, donje i levo punjenje. Međutim, ukoliko želimo da specifikujemo posebnu veličinu punjenja na nekoj strani, možemo iskoristiti neko od svojstava `padding-top`, `padding-right`, `padding-bottom` ili `padding-left`. Vrednosti za sva opisana svojstva su: dužina ili procenat. Podrazumevana vrednost je 0.

Naredni primer i prateća slika ilustruju postavljanje punjenja za svaku stranu ponaosob. Dodatno, element `div` iz primera sadrži crnu ivicu kako bi se lakše razumeo efekat punjenja. Primetimo da će svojstvo `background-color` takođe obojiti i boju punjenja.

```
<body>  
<div>  
  Ovaj div element ima gornje punjenje od 50px, desno punjenje od 30px,  
  donje punjenje od 100px i levo punjenje od 280px.  
</div>  
</body>
```

```
div {  
  border: 1px solid black;  
  background-color: lightblue;  
  padding-top: 50px;  
  padding-right: 30px;  
  padding-bottom: 100px;  
  padding-left: 280px;  
}
```

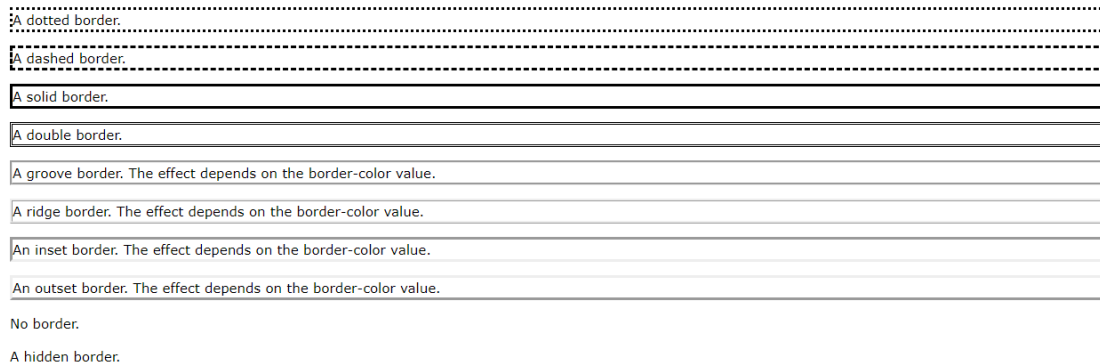
Ovaj div element ima gornje punjenje od 50px, desno punjenje od 30px, donje punjenje od 100px i levo punjenje od 280px.

Punjenje proizvodi isti efekat i za linijske i za blokove elemente.

### 1.3.3 Ivica elementa

U CSS jeziku, svojstvom `border` je moguće postaviti stil, širinu i boju kutije koja odvaja unutrašnjost elementa od njegovog pojasa (odnosno, od okolice ukoliko pojas ne postoji). Ipak, prvo ćemo se upoznati sa svojstvima koja postavljaju ove tri stavke odvojeno, pre nego što pokažemo kako ih je moguće ukombinovati pomoću svojstva `border`.

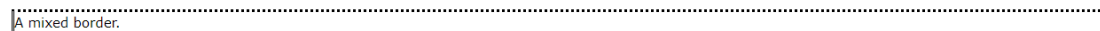
**Stil ivice** Stil ivice definišemo pomoću svojstva `border-style`. Moguće vrednosti su: `dotted`, `dashed`, `solid`, `double`, `groove`, `ridge`, `inset`, `outset`, `none` ili `hidden`. Naredna slika ilustruje svaki od ovih stilova.



Takođe, ukoliko želimo da definišemo različite stilove za svaku ivicu ponaosob, umesto navođenja jedne vrednosti (čime podešavamo sve četiri ivice), možemo navesti četiri vrednosti koje će postaviti stilove za gornju, desnu, donju i levu ivicu, redom. Na primer, rezultat narednog pravila

```
p.mix {
  border-style: dotted dashed solid double;
}
```

dat je na narednoj slici



Napomenimo da nijedna od preostala dva svojstva, širina i boja, neće biti vidljiva ukoliko nije podešen stil ivice.

**Širina ivice** Širinu ivice definišemo pomoću svojstva `border-width`. Moguće vrednosti su: dužine, `thin`, `medium` ili `thick`. Poput svojstva za stil ivice, i ovo svojstvo može imati jednu ili četiri vrednosti. Semantika ovih dveju deklaracija je identična kao za prethodno svojstvo. Podrazumevana vrednost za širinu je `1px`.

**Boja ivice** Boju ivice definišemo pomoću svojstva `border-color`. Moguće vrednosti su: ime boje, boja zadata u HEX modelu, boja zadata u RGB ili RGBA modelima ili `transparent`. Kao i prethodna dva svojstva, i ovo svojstvo može imati jednu ili četiri vrednosti, uz identičnu semantiku. Podrazumevana vrednost za boju je `black`.

**Navođenje konkretne ivice** Svako od prethodna tri svojstva ima i varijantu kojom se navodi na koju stranu ivice svojstvo treba da se primeni. Tako, na primer, pored svojstva `border-style`, postoje i svojstva `border-top-style`, `border-right-style`, `border-down-style` i `border-left-style`. Analogno važi i za svojstva `border-width` i `border-color`.

**Svojstvo `border`** Dakle, ukoliko želimo za ivicu da detaljno specifikujemo izgled, moramo da navedemo tri deklaracije, po jednu za stil, širinu i boju. Navedene tri deklaracije se mogu skratiti u jednu tako što se koristi svojstvo `border` (ili neko od svojstava `border-top`, ..., `border-left` za svaku ivicu ponaosob). Vrednost ovog svojstva čine: (1) vrednost za stil (obavezno), (2) vrednost za širinu i (3) vrednost za boju, odvojenu znakom razmaka. Ukoliko se ne navede neka od (2) ili (3), korišće se podrazumevane vrednosti. Na primer, CSS pravilom

```
p {  
  border: 5px solid red;  
}
```


proizvodi se



dok se CSS pravilom

```
p {  
  border-left: solid 10px;  
  background-color: lightgrey;  
}
```

proizvodi



**Svojstvo `border-radius`** Svojstvo `border-radius` služi za kontrolu zakrivljenosti ćoškovia elementa. Moguće vrednosti su: dužina ili procenat. Ukoliko navedemo jednu vrednost, svi ćoškovi će se zakriviti. Alternativno, možemo navesti četiri vrednosti koje će zakriviti gornji-levi, gornji-desni, donji-desni i donji-levi ćošak, redom. Na primer, CSS naredbama

```
#example1 {  
  border: 2px solid red;  
  padding: 10px;  
  border-radius: 25px;  
}  
  
#example2 {  
  border: 2px solid red;  
  padding: 10px;  
  border-radius: 50px 20px 50px 50px;  
}
```

se naredni HTML kod

```
<h2>border-radius: 25px:</h2>  
  
<div id="example1">  
  <p>Svojstvo border-radius definise zakrivljenje coskova elementa.</p>  
</div>  
  
<h2>border-radius: 50px 20px 50px 50px:</h2>  
  
<div id="example2">  
  <p>Mozemo staviti jednu vrednost za sve coskove, ili cetiri vrednosti,  
    za svaki cosak od gornjeg-levog do donjeg-levog po jednu.</p>  
</div>
```

formatira u prikaz na narednoj slici

#### **border-radius: 25px:**

Svojstvo border-radius definise zakrivljenje coskova elementa.

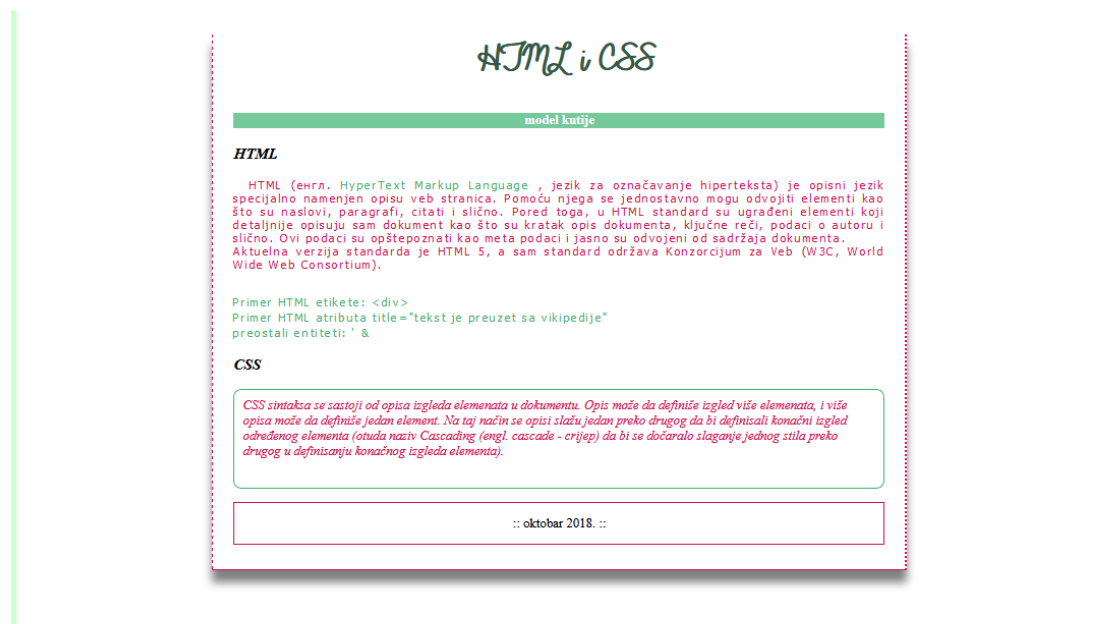
#### **border-radius: 50px 20px 50px 50px:**

Mozemo staviti jednu vrednost za sve coskove, ili cetiri vrednosti, za svaki cosak od gornjeg-levog do donjeg-levog po jednu.

### **1.3.4 Pojas elementa**

#### **Zadatak 1.3:**

Napisati HTML datoteku i CSS datoteku koja odgovara stranici na narednoj slici

**Rešenje:****Kod 4: zadatak3/strana.html**

```
<!DOCTYPE html>

<html>
  <head>
    <title> Model kutije </title>
    <meta charset='UTF-8'>
    <link rel='stylesheet' href='stil.css'>
  </head>

  <body>

    <header>
      <h1> HTML i CSS </h1>
      <h4> model kutije </h4>
    </header>

    <main>
      <div>
        <h3> HTML </h3>
        <p class='intro html' title='tekst je preuzet sa Vikipedije'>
          HTML еһгп(. <span style='color:mediumseagreen;'> HyperText
            Markup Language </span>, jezik za označavanje
            hiperteksta) je opisni jezik specijalno namenjen opisu
            veb stranica. Pomoću njega se jednostavno mogu odvojiti
            elementi kao što su naslovi, paragrafi, citati i slično.
            Pored toga, u HTML standard su ugrađeni elementi koji
            detaljnije opisuju sam dokument kao što su kratak opis
            dokumenta, ključne reči, podaci o autoru i slično. Ovi
            podaci su opštepoznati kao meta podaci i jasno su
            odvojeni od sadržaja dokumenta.
```



```

        <br>
        Aktuelna verzija standarda je HTML 5, a sam standard
        održava Konzorcijum za Veb (W3C, World Wide Web
        Consortium).
        <br>
        <br>
    </p>
    <span class='example html' title='tekst je preuzet sa
    Vikipedije'>
        Primer HTML etikete: &lt;div&gt;
        <br>
        Primer HTML atributa title=&quot;tekst je preuzet sa
        vikipedije&quot;
        <br>
        preostali entiteti: &apos; &amp;
    </span>
</div>

<div>
    <h3> CSS </h3>
    <p id='css' class='intro'>
        CSS sintaksa se sastoji od opisa izgleda elemenata u
        dokumentu. Opis može da definiše izgled više elemenata,
        i više opisa može da definiše jedan element. Na taj
        način se opisi slažu jedan preko drugog da bi definisali
        konačni izgled određenog elementa (otuda naziv
        Cascading (engl. cascade – crijep) da bi se dočaralo
        slaganje jednog stila preko drugog u definisanju
        konačnog izgleda elementa).

    </p>
</div>
</main>

<footer>
    <p> :: Oktobar 2018. :: </p>
</footer>

</body>
</html>

```

Kod 5: zadatak3/stil.css

```

@font-face{
    /* Vanessa Bays @ http://bythebutterfly.com */
    font-family: 'TestFont';
    src: url('test_font.ttf');
}

body{
    /* leva i desna margina postavljene na auto omogućavaju centriranje –
    preduslov je promeniti sirinu tako da bude manja od 100% */
    width: 800px;
    margin-left:auto;
}

```

```
margin-right:auto;

border-left: 1px dashed rgb(211, 17, 79);
border-right: 3px dotted rgb(211, 17, 79);
border-bottom: 1px solid rgb(211, 17, 79);

padding-left: 25px;
padding-right:25px;

box-shadow: 0px 15px 10px 0px gray;
}

header{
    text-align: center;
}

header h1{
    text-align: center;
    font-family: 'TestFont', cursive;
    font-size: 50px;
    color: rgb(53, 88, 71);
}

header h4{
    background-color: mediumseagreen;
    color: white;
    /* TODO: eksperimentisati sa razlicitim vrednostima providnosti – od 0
        do 1 */
    opacity: 0.7;
}

.intro{
    color: rgb(211, 17, 79);
}

.example {
    color: mediumseagreen;
}

.html{
    font-size: 14px;
    font-family: 'Tahoma', 'Arial', sans-serif;
    letter-spacing: 2px;
    text-indent: 20px;
    text-align: justify;
}

#css{
    border: 2px solid mediumseagreen;
    border-radius: 10px;
    padding: 10px;
    color: rgb(211, 17, 79);
```

```
    font-style: italic;
    /* TODO: eksperimentisati sa razlicitim visinama bloka */
    height: 100px;
    overflow: auto;
}

main div h3{
    font-weight: bold;
    font-style: oblique;
}

footer{
    text-align: center;
    text-transform: lowercase;
    border: 2px solid rgb(211, 17, 79);
    margin-bottom: 30px;
}
```

## 2 Čas 4

Cilj ovog časa je upoznavanje studenata sa pojmovima vezanim za *pozicioniranje* elemenata.

### 2.1 Pozicioniranje elemenata

#### 2.1.1 Statičko pozicioniranje

#### 2.1.2 Relativno pozicioniranje

#### 2.1.3 Apsolutno pozicioniranje

#### 2.1.4 Fiksno pozicioniranje

#### 2.1.5 Z-pozicioniranje elemenata

#### 2.1.6 Svojstvo `display`



## Dodatak

### Lista kodova

1	zadatak1/strana.html . . . . .	9
2	zadatak2/strana.html . . . . .	15
3	zadatak2/stil.css . . . . .	16
4	zadatak3/strana.html . . . . .	23
5	zadatak3/stil.css . . . . .	24