Postupak izrade i primjene predloška za web stranicu

Valjavec, Boris

Master's thesis / Diplomski rad

2016

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:971071

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-28**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

zir.nsk.hr

Image not found or type unknown

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
ELEKTROTEHNIČKI FAKULTET

Diplomski studij
Procesno računarstvo

POSTUPAK IZRADE I PRIMJENE PREDLOŠKA ZA WEB STRANICU

Diplomski rad

Boris Valjavec

Osijek, rujan 2016.

Sadržaj:

1.Uvod 1
2. HTML 4
2.1 Osnove HTML-a 4
2.2 Definiranje kodne stranice 5
2.3 Uređivanje teksta 5
2.3.1 Naslovi 5
2.3.2 Prijelom teksta 6
2.3.3 Novi red 6
2.3.4 Odlomak 6
2.3.5 Non–Breaking Space 8
2.3.6 Predefinirani tekst 8
2.4 Liste 9
2.4.1 Numerirane liste 9
2.4.2 Nenumerirane liste 11
2.4.3 Definicijske liste 12
2.5 Hiperveze 13
2.5.1 Lokalne hiperveze 14
2.5.2 Globalne hiperveze 14
2.5.3 Hiperveza na adresu e-pošte 14
2.6 Definiranje boja 15
2.7 Dodavanje slike 16
2.9 Meta oznake 16
2.9.1 Jezik 16
2.9.2 Autor 17
2.9.3 Autorska prava 17
2.10 Okviri 18
2.10.1 Definiranje okvira 18
2.10.2 INLINE okviri 19
2.11 Obrasci 21
2.11.1 Polja za unos podataka 21
2.11.2 Polje za unos teksta 22
2.11.3 Polje za upis lozinke 22
2.11.4 Potvrdni okviri 22
2.11.5 Opcijski gumbi 23
2.11.6 Izbornik 23
2.11.7 Gumbi 24
3.CSS 25
3.1 Kako uključiti CSS dokument u HTML kod 26
3.2 Sintaksa CSS–a 26
3.3 CSS selektori 28
3.3.1 Naziv elemenata 28
3.3.2 Klasa 28
3.3.3 Identifikator 29
3.4 Prioritet CSS pravila 29
3.5 Pseudo klase 31
3.5.1 Pseudo klase vezane uz korisničke akcije 31
3.5.2 Pseudo klase vezane uz unos podataka 32

3.6 Nasljeđivanje vrijednosti ... 33
3.7 Komentari ... 34
3.8 Oblikovanje teksta ... 35
3.8.1 Boja ... 35
3.9 Font ... 38
3.9.1 Vrsta fonta ... 38
3.9.2 Veličina fonta ... 39
3.9.3 Poravnavanje teksta ... 41
3.9.4 Velika i mala slova ... 41
3.9.5 Ukrašavanje teksta ... 42
3.10 Skraćeni način navođenja svojstva fonta ... 44
3.11 Oblikovanje elemenata ... 44
3.11.1 Definiranje boja u pozadini ... 45
3.11.2 Definiranje prikaza elemenata ... 45
3.12 Visina i širina ... 48
3.12.1 Visina ... 48
3.12.2 Širina ... 49
3.13 Padding ... 51
3.14 Margine ... 51
3.15 Dodavanje rubova ... 53
3.15.1 Stilovi ruba ... 53
3.15.2 Širina ruba ... 55
3.15.3 Zakrivljenost rubova ... 55
4. Predlošci ... 57
4.1 Izrada predloška ... 58
5.Zaključak ... 67

1.Uvod

LHC–Large Hadron Collider, odnosno akcelerator čestica , je stroj zbog kojega je CERN postao vrlo poznat. CERN je također mjesto u kojem je koncipiran World Wide Web, sustav međusobno povezanih hipertekst dokumenata kojima se pristupa putem Interneta.

Za nastanak World Wide Web–a odgovoran je Tim Berners–Lee koji je, radeći u odijelu za računalne usluge u okviru CERN–a osmislio koncept za koji u tome trenutku nije niti sanjao da će postati planetarno popularan i prihvaćen.

Naime, istraživanja u području nuklearne fizike obično podrazumijevaju suradnju institucija iz cijelog svijeta, a g. Berners – Lee smatrao je kako bi bilo dobro da istraživači iz različitih institucija imaju pristup svim informacijama u ovom području. I umjesto da se podaci nalaze na jednom mjestu, on je predložio da se dokumenti sa različitim sadržajem povežu tako da se među njima napravio određeni sustav referenci. Taj sustav bi funkcionirao tako da korisniku omogućio da na jednostavan način tokom čitanja jednog dokumenta prikaže sadržaj drugog, koji sadrži relevantne podatke u vidu dodatnog teksta, dijagrama ili formula. Tako bi se praktički stvorila mreža (engl. Web) podataka u elektroničkom obliku koji bi se nalazili na različitima računalima diljem svijeta.

Ovaj koncept je razrađen i formiran 1990. godine, kada je predložen i World Wide Web projekt, koji je zamišljen kao mreža hipertekst dokumenata koje je moguće prikazati upotrebom preglednika (engl. browser).

Hipertekst dokumenti su obogaćeni tekstualni dokumenti, u smislu da sadrže reference–linkove (engl. link, hyperlink) prema drugim dokumentima ili sadržajima (slikama, audio ili video zapisima).

Za pregledavanje HTML dokumenta je neophodan preglednik, odnosno posebna aplikacija zadužena za interpretaciju HTML koda i prikaz sadržaja korisniku u skladu sa tim kodom.

U današnje vrijeme postoji mnogo preglednika na tržištu, a neki od popularnijih su: Google Chrome, Mozilla Firefox, Opera, Safari i Microsoft Internet Explorer. Isto tako, svako računalo (i mobilni uređaj) posjeduje jedan od navedenih preglednika, kao i neki uređivač teksta (pod Windows operativnim sustavom je to Notepad ili Wordpad, Mac OS korisnici vjerojatno imaju dostupan TextEdit ili TextMate, dok kod Linux operativnih sustava podrazumijevani uređivač teksta zavisi od konkretne distribucije – često je to Vim, Emacs ili gEdit).

Jezik koji se koristi za kreiranje hipertekst dokumenata naziva se HTML (engl.HyperText Markup Language) i zasniva se na oznakama (engl. tag) koje se koriste za strukturiranje i formatiranje sadržaja. Pored toga što je relativno jednostavan za savladavanje, HTML ne zahtjeva specijalno razvojno okruženje; u stvari, sve što je potrebno za kreiranje HTML dokumenta je običan uređivač teksta (engl. Notepad), koji je instaliran na svakom Windows sustavu kao dio osnovnog paketa. Također, postoje i mnogi alati koji omogućuju izradu web stranica i bez najosnovnijeg poznavanja HTML koda. Ti alati automatski generiraju HTML kod, prateći pritom upute koje korisnik zadaje preko korisničkog sučelja. Ovdje se govori o vizualnom uređivanju, gdje korisnik odabirom određenih alata stvara, odnosno uređuje web stranicu koja mu se u realnom vremenu prikazuje onako kako će se kasnije prikazivati u nekom od preglednika. Među najpopularnijim alatima za vizualnu izradu web stranica su Microsoftov FrontPage i Macromedia DreamWeavera.

Ručno pisanje koda u programu za obradu teksta, poprilično je komplicirano, već zbog same činjenice da korisnik ne vidi što zapravo radi, nego je prisiljen kodirati naslijepo.
Nakon dopisivanja nekoliko novih linija koda korisnik bi trebao prekinuti rad i svoj uradak pregledati u web pregledniku. Taj proces je zamoran i kod njega su greške neizbježne.

S druge strane, poznavanje HTML-a i CSS –a neophodno je za svako imalo ozbiljnije kreiranje i izrađivanje web stranica, a budući da će se ovdje upravo govoriti o tome, prije ikakve izrade predloška za web stranicu, potrebno je proći osnovne elemente HTML–a i CSS–a, nakon kojih će se izrada predložaka svesti samo na uređivanje CSS skripte i slaganje elemenata unutar Internet preglednika.

Opis rada:

Objasniti način izrade predloška za web stranicu. Izraditi predložak koji će biti primijenjen na nekoj web stranici. Objasniti način primjene predloška kao i način kako se predložak može primijeniti.

2. HTML

HTML odnosno *Hyper Text Markup Language* jeste ne programski jezik označavanja različitih sadržaja korištenjem tekstualnih struktura za opis dokumenta, kako bi se isti na strukturiran i uređen način prikazivali i dohvaćali putem Interneta.
Jezik je stvoren i standardiziran kao vlasništvo W3C (World Wide Web Consorcium).

2.1 Osnove HTML-a

HTML je strukturirani jezik u kojemu se za označavanje određenih sadržaja koriste oznake kojima se stvaraju, povezuju i strukturiraju elementi dokumenta koji se još nazivaju i elementi HTML-a, odnosno čvorovi dokumenta (engl. node).
Osnovna struktura HTML dokumenta uključuje oznake koje okružuju sadržaj i omogućavaju Internet pregledniku njihovu interpretaciju:

```
<html>
        <head>
                <title>Naziv dokumenta</title>
                <meta/>
        </head>
        <body>
                <p>Nekakav sadržaj</p>
        </body>
</html>
```

Sl.2.1 Struktura osnovnog HTML koda

Prva oznaka je **<*html*>**, a govori Internet pregledniku da slijedi početak HTML dokumenta. Tekst unutar oznaka **<head>** je informacija o zaglavlju dokumenta, koja neće biti prikazana u Internet pregledniku,**<title>** je naslov dokumenta, koji će biti prikazan u zaglavlju elementa, a sve unutar oznaka **<body>** predstavlja sadržaj HTML dokumenta.

2.2 Definiranje kodne stranice

Veliki broj korisnika Interneta i njegovih usluga je zasigurno, barem jednom, naišao na stranicu koja neispravno prikazuje znakove hrvatske abecede Do toga problema dolazi, kada kodna stranica nije ispravno definirana, pa se za prikaz teksta koristi pogrešni set znakova.

Prilikom izrade stranice, znakove poput *č,ć,đ,š,ž*, odnosno znakove koji nisu dio engleske abecede, preglednik će prikazati sa nekim simboličkim znakovima, koji će tekst učiniti teško čitljivim.

Kako bi se to izbjeglo, potrebno je definirati kodnu stranicu za srednje europske jezike (u koje spada i hrvatski), dodavanjem meta oznake unutar zaglavlja HTML dokumenta, odnosno:

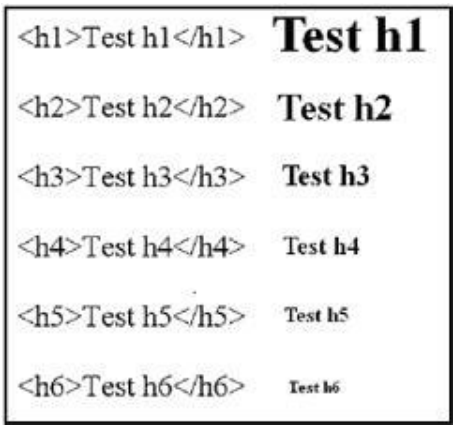
<meta http-equip="Content-Type" content="text/html; charset=iso-8859-2">

2.3 Uređivanje teksta

2.3.1 Naslovi

HTML dopušta definiranje naslova u šest razina korištenjem oznake ***<h>***.

Tako će, na primjer, naslov biti definiran sa oznakom ***<h1>***, a podnaslovi oznakama od ***<h2>*** do ***<h6>***.



Sl.2.2 HTML kod za naslove/podnaslove i njihov izgled u Internet pregledniku

Vidljivo je da se naslovi definiraju ,ovisno o njihovoj važnosti, sa brojem koji stoji uz oznaku "***h***". Naslovi sa najmanjim brojem su najveći, a oni sa najvećim brojem – najmanji.
Svi su prikazani kao masno otisnuti , te se automatski odvajaju od ostatka teksta.

2.3.2 Prijelom teksta

Cijeli kod stranice može biti napisan u jednom jedinom retku. Sve razmake navedene u tekstu preglednik će prikazati kao jednostruke, bez obzira na to koliko puta su uneseni, a prelasci u novi red neće biti vidljivi.

2.3.3 Novi red

U slučaju prelaska u novi red, na mjestu prekida unosi se oznaka **<*br*>** (engl. break).Ta oznaka (najčešće) ne dolazi s nekim atributom, niti ima završni dio.

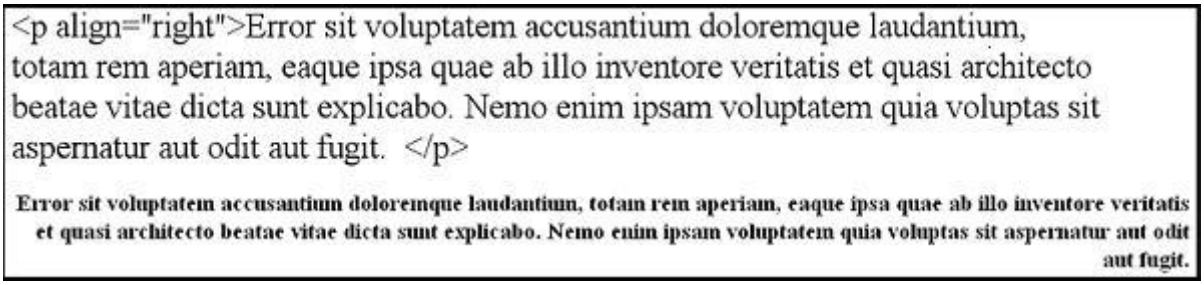
2.3.4 Odlomak

Tekst se može posložiti u odlomke ili paragrafe korištenjem oznake **<*p*>**. Oznaka kojom se definira odlomak, najčešće se koristi uz atribut <***align***>. Kao i kod naslova, vrijednosti atributa mogu biti *left, right, center i justify*, ovisno o tome kako se tekst želi poravnati.

Error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit.

Error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit.

Sl.2.3 Centrirani tekst

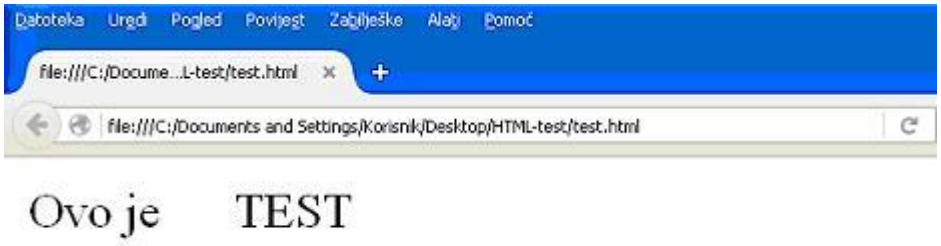


Sl.2.4 Tekst poravnat u desno

2.3.5 Non–Breaking Space

To je poseban znak koji se koristi za prikazivanje više uzastopnih razmaka. Korištenjem ove vrste razmaka izbjegava se neželjeni prijelom teksta između dviju ili više rijeci.

<p>Ovo je   TEST<p>

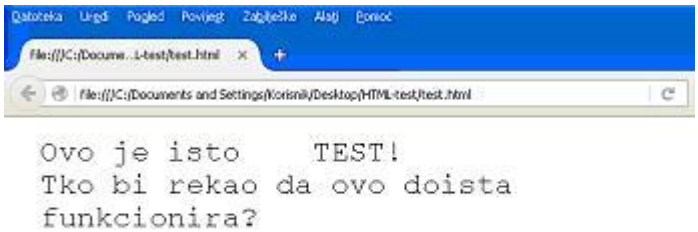


Sl.2.5 Primjer korištenje oznake

2.3.6 Predefinirani tekst

Tekst koji se nalazi unutar HTML koda može biti prikazan upravo onako kako je napisan u samom uređivaču teksta, sa svim razmacima i prijelazima u novi red. To se postiže korištenjem oznake ***<pre>***, kao što vidljivo u sljedećem primjeru:

<pre>
Ovo je isto TEST!
Tko bi rekao da ovo doista
funkcionira?
</pre>



Sl.2.6 Izgled predefiniranog teksta u Internet pregledniku

Sve navedeno unutar oznaka ***<pre>*** biti će ispisano fontom Courier.

2.4 Liste

Jednostavne su za definiranje i vrlo korisne. Razlikuju se:

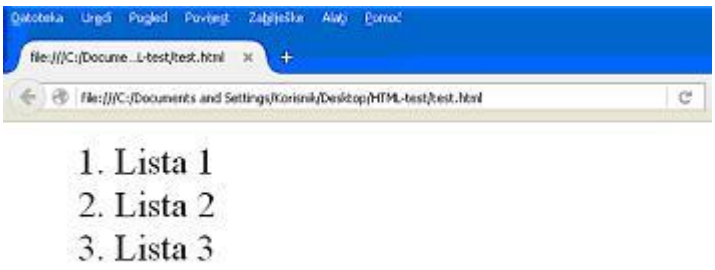
- numerirane
- nenumerirane
- definicijske liste

2.4.1 Numerirane liste

Koriste se za nabrajanje pojmova. Da bi se definirale, potrebne su dvije oznake. Prva služi za definiranje početka i kraja liste, a druga za definiranje elemenata liste.

Početak i kraj numerirane liste definira se oznakom ***<ol>*** (engl. ordered list), a elemente koristeći oznaku ***<li>*** (engl. list item), koja se koristi posebno za svaki novi element liste:

```
<ol>
    <li>Lista 1</li>
    <li>Lista 2</li>
    <li>Lista 3</li>
</ol>
```



Sl.2.7 Numerirana lista

Ako je potrebno promijeniti način numeriranja elementa unutar liste, koristi se atribut <***type***>. Kod numerirane liste, vrijednost navedenog atributa može biti:

A–slova

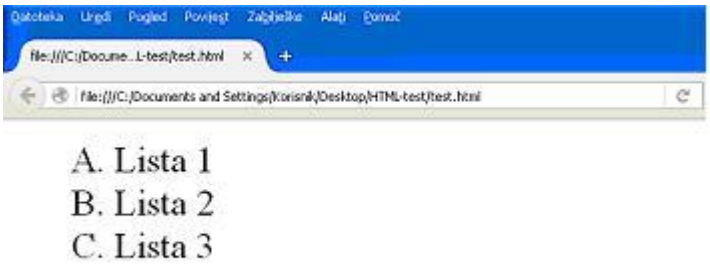
a–mala slova

I–rimski brojevi

i–mali rimski brojevi

1–arapski brojevi

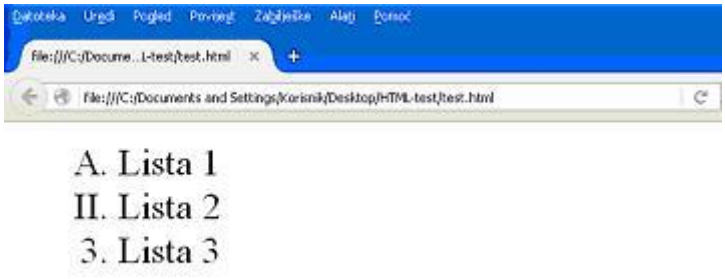
```
<ol type="A">
    <li>Lista 1</li>
    <li>Lista 2</li>
    <li>Lista 3</li>
</ol>
```



Sl.2.8 Numerirana lista s postavljenom vrijednosti atributa *type* (model 1)

Atribut može biti primijenjen na cijelu listu ili na svaki element liste zasebno, odnosno:

```
<ol>
<li type="A"> Lista 1</li>
<li type="I"> Lista 2</li>
<li type="1"> Lista 3</li>
</ol>
```

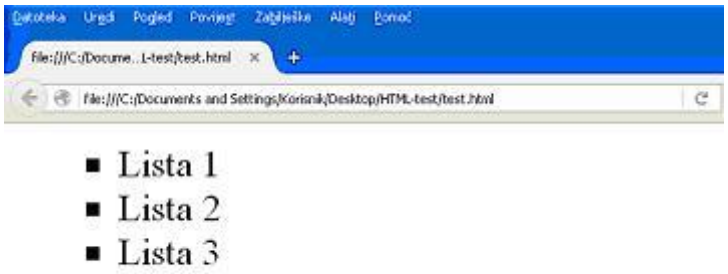


Sl.2.9 Numerirana lista s postavljenom vrijednosti atributa *type* (model 2)

2.4.2 Nenumerirane liste

Definiraju se na isti način kao i numerirane, samo se umjesto oznake **<*ol*>** koristi oznaka **<*ul*>** (eng. unordered list).

```
<ul type="square">
<li> Lista 1 </li>
<li> Lista 2 </li>
<li> Lista 3 </li>
</ul>
```

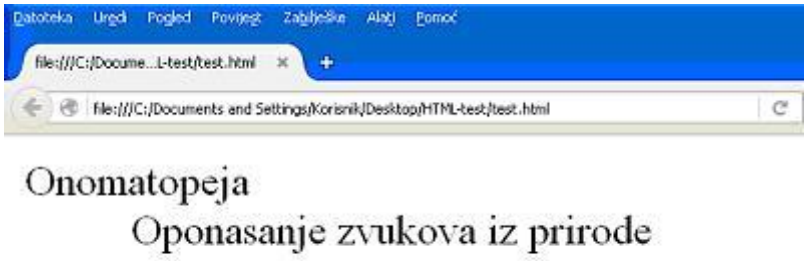


Sl.2.10 Nenumerirana lista

2.4.3 Definicijske liste

Za razliku od numerirane i nenumerirane, definicijsku listu sačinjavaju tri elementa. Početak i kraj definicijske liste, definira se oznakom **<*dl*>**, pojam koji se definira oznakom **<*dt*>**, a samu definiciju oznakom **<*dd*>**.

```
<dl>
<dt> Onomatopeja</dt>
  dd>Oponasanje zvukova iz prirode</dd>
</dl>
```



Sl.2.11 Primjer definicijske liste

2.5 Hiperveze

Hiperveze su elementi koji obične tekstualne dokumente čine hipertekstualnima. Koriste se za povezivanje tekstualnih dokumenata s raznim sadržajima unutar ili izvan web sjedišta.
Hiperveze mogu voditi na:
- adresu elektroničke pošte,
- mjesto unutar stranice
- neki drugi tekstualni dokument (doc, pdf, txt, html…)
- sliku (jpeg, gif, png…)
- audio/video zapis (wmv,rm,wav,mp3…) i slično.

Hiperveze se označavaju oznakom ***<a>***, koja dolazi od engleske riječi *anchor*, što bi u prijevodu značilo sidro. Ima svoj završni dio ****** te atribute kojima se definiraju njezina svojstva. Najčešće korišteni atributi su *href, name, target i title.*

href (engl. hyperlink reference) definira kamo hiperveza vodi (dokument, datoteku, adresu e-pošte ili na mjesto unutar dokumenta).

GOOGLE



Sl.2.12 Hiperveza na stranicu Google

name–se koristi za definiranje književne oznake, mjesta unutar dokumenta
target–određuje mjesto na kojem se hiperveza otvara (može biti jedan od okvira, novi prozor i slično)
title–opisuje hipervezu, a prikazuje se u obliku savjeta (engl. tip).

2.5.1 Lokalne hiperveze

Najčešće se definiraju relativnim vrijednostima atributa *href*, a vode na resurse unutar izvorne datoteke. Vrijednosti mogu biti relativne u odnosu na trenutačno otvorenu stranicu ili u odnosu na korijen izvorne datoteke.

2.5.2 Globalne hiperveze

Hiperveza koja vodi na resurs izvan izvorišne mape, naziva se globalnom hipervezom. Za njihovo definiranje koriste se apsolutne vrijednosti atributa *href*. Uz adresu, moraju sadržavati oznaku protokola koji se koristi, kako bi se moglo doći do željenog sadržaja.

2.5.3 Hiperveza na adresu e-pošte

Korisnicima se može omogućiti kontakt putem e-mail-a postavljanjem jednostavne hiperveze, odnosno.

Send e mail



Sl.2.13 Primjer hiperveze na adresu e-pošte

2.6 Definiranje boja

Boju u HTML–u može se definirati na dva načina; korištenjem njihovih naziva na engleskom jeziku (*aqua, black, blue,* itd.) ili korištenjem posebnog koda. Svaka boja se temelji na RGB modulu, a vrijednosti elemenata iskazuju se brojevima od 0 do 255, što grafički izgleda ovako:

#RRGGBB ,odnosno

000000	000033	000066	000099	0000CC	0000FF
003300	003333	003366	003399	0033CC	0033FF
006600	006633	006666	006699	0066CC	0066FF
009900	009933	009966	009999	0099CC	0099FF
00CC00	00CC33	00CC66	00CC99	00CCCC	00CCFF
00FF00	00FF33	00FF66	00FF99	00FFCC	00FFFF
330000	330033	330066	330099	3300CC	3300FF
333300	333333	333366	333399	3333CC	3333FF
336600	336633	336666	336699	3366CC	3366FF
339900	339933	339966	339999	3399CC	3399FF
33CC00	33CC33	33CC66	33CC99	33CCCC	33CCFF
33FF00	33FF33	33FF66	33FF99	33FFCC	33FFFF
660000	660033	660066	660099	6600CC	6600FF
663300	663333	663366	663399	6633CC	6633FF
666600	666633	666666	666699	6666CC	6666FF
669900	669933	669966	669999	6699CC	6699FF
66CC00	66CC33	66CC66	66CC99	66CCCC	66CCFF
66FF00	66FF33	66FF66	66FF99	66FFCC	66FFFF
990000	990033	990066	990099	9900CC	9900FF
993300	993333	993366	993399	9933CC	9933FF
996600	996633	996666	996699	9966CC	9966FF
999900	999933	999966	999999	9999CC	9999FF
99CC00	99CC33	99CC66	99CC99	99CCCC	99CCFF
99FF00	99FF33	99FF66	99FF99	99FFCC	99FFFF
CC0000	CC0033	CC0066	CC0099	CC00CC	CC00FF
CC3300	CC3333	CC3366	CC3399	CC33CC	CC33FF
CC6600	CC6633	CC6666	CC6699	CC66CC	CC66FF
CC9900	CC9933	CC9966	CC9999	CC99CC	CC99FF
CCCC00	CCCC33	CCCC66	CCCC99	CCCCCC	CCCCFF
CCFF00	CCFF33	CCFF66	CCFF99	CCFFCC	CCFFFF
FF0000	FF0033	FF0066	FF0099	FF00CC	FF00FF
FF3300	FF3333	FF3366	FF3399	FF33CC	FF33FF
FF6600	FF6633	FF6666	FF6699	FF66CC	FF66FF
FF9900	FF9933	FF9966	FF9999	FF99CC	FF99FF
FFCC00	FFCC33	FFCC66	FFCC99	FFCCCC	FFCCFF
FFFF00	FFFF33	FFFF66	FFFF99	FFFFCC	FFFFFF

Sl.2.14 RGB kodovi pojedinih boja

2.7 Dodavanje slike

Oznakom **<*img*>** se slike umeću na stranicu, a korištenjem atributa ***src*** (engl. source) definira se putanja do izvorišne datoteke.

<figure class="img-indent"><img src="images/Usluge/servis-slika1.jpg"alt=""></figure>

U navedenom primjeru, naziv datoteke servis-slika1.jpg ne nalazi se u istoj mapi kao i stranica.

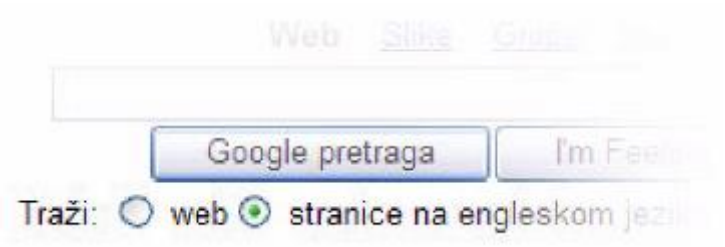
Oznaka **<*img*>** opisuje se sa nekoliko atributa, a neki od najčešći korištenih su:
***border*–**debljina ruba oko slike
***width*–**širina slike
***height*–**visina slike
***alt*–**alternativni tekst, prikazuje se u preglednicima koji ne podržavaju prikaz slika
***align*–**određuje položaj slike u odnosu na tekst
***hspace*–**horizontalni razmak slike od teksta ili nekog drugog elementa na stranici
***vspace*–**vertikalni razmak slike od teksta ili nekog drugog elementa na stranici

2.9 Meta oznake

Definiraju se unutar zaglavlja dokumenta a koriste se za opisivanje stranice ili davanje određene upute Internet preglednicima ili pretraživačima.
Meta oznake imaju najčešće atribute ***http-equip***, ***name*** *i* ***content***, a u nastavku će biti navedene neke od najčešće korištenih meta oznaka.

2.9.1 Jezik

Prilikom izrade stranice, moguće je dati informaciju koja ukazuje na jezik kojim je napisan sadržaj koji se nalazi na njoj. Tu informaciju koristi Internet pretraživač kada korisnik zatraži pretragu stranica na određenom jeziku



Sl.2.15 Pretraživanje stranica na hrvatskom jeziku pomoću Google pretraživača

Za definiranje jezika stranice, koriste se atributi "***http-equiv***" ***i*** "***content***", odnosno:

<meta http-equiv="Content-Language" content="hr">

Kratica *"hr"* ukazuje na to da se na stranici koristi hrvatski jezik

2.9.2 Autor

Je meta oznaka koja koristi dva atributa ***name*** *i* ***content.*** To je ustvari nevidljivi način potpisivanja stranice od strane samoga autora, odnosno:

<meta name="author" content="Boris Valjavec">

2.9.3 Autorska prava

Princip potpisivanja autorskih prava, jednak je principu potpisivanja stranice. Atribut ***name*** poprima vrijednost "***copyright***", a atribut ***content*** podatke o vlasniku autorskih prava.

<meta name="copyright" content="2016 Boris Valjavec. Sva prava zadržana.">

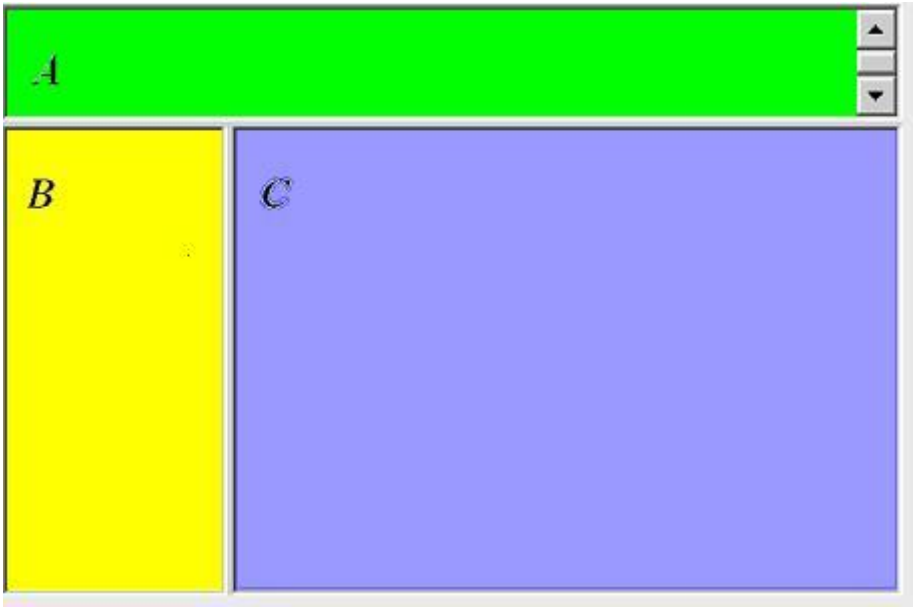
2.10 Okviri

Korištenjem okvira omogućeno je istovremeno prikazivanje više HTML dokumenata u prozoru preglednika. Svaki od dokumenata se nalazi u zasebnom okviru, a raspored i veličina okvira definiraju se u posebnom dokumentu koji ih sve sjedinjuje u jednu cjelinu.
Prednost korištenja ove metode je lakše održavanje i učitavanje web stranice. Nedostaci su kompleksnije definiranje hiperveza među stranicama, veći broj dokumenata od kojih se svaka stranica sastoji, problemi s ispisom cijele stranice, problemi s pretraživačima, problemi s adresama i slično. Većinu, ako ne i sve ove nedostatke, moguće je ispraviti, ali to zahtjeva više truda i vremena.

2.10.1 Definiranje okvira

Prilikom izrade koristi se oznaka ***<frameset>*** za postavljanje okvira i ***<frame>*** za definiranje dokumenta koji se prikazuje u okviru, to jest:

```
<html>
<frameset rows= "20%,80%">
    <frame src="a.html">
    <frameset cols="25%,75%">
    <frame src="ˇb.html">
    <frame src="ˇc.html">
</frameset>
</frameset>
</html>
```



Sl.2.16 Izgled okvira unutar Internet preglednika

Oznakom <***frameset***> i atributima ***rows*** i ***cols***, definiraju se broj i veličina retka (stupca) okvira. Atributom ***rows*** definira se broj okvira i njegova visina, a *atributom* ***cols*** definira se broj okvira i njegova širina. Vrijednosti atributa mogu biti zadane u pikselima ili postotcima. Unese li se za vrijednost atributa ***«*»*** visina/širina okvira biti će definirana kao preostala širina/visina stranice.

Oznaka nema završni dio, a neki od značajnijih atributa koji se koriste za opisivanje su:

frameborde-poprima vrijednost 0 ili 1, ovisno o tome ima li okvir rubove ili ne
name-koristi se za imenovanje okvira (ime svakom okviru dodjeljuje se samostalo, a imenovanje okvira posebno je važno za kreiranje hiperlinkova)
noresize-ne poprima vrijednosti, a koristi se za onemogućavanje promjene veličine okvira.
scrolling-može poprimiti vrijednosti "yes","no" ili "auto" a određuje prikaz klizača u okviru.

2.10.2 INLINE okviri

Ova vrsta okvira dodaje se na stranicu na isti način kao i element. Definira se oznakom <***frame***> i opisuje atributima:
name– ime okvira .
src–definira dokument prikazan u okviru
height–visina inline okvira

***width*–**širina inline okvira

***align*–** služi za određivanje položaja okvira na stranici

***frameborder*–**prikazuje ili skriva rubove okvira, ovisno o tome je li vrijednost atributa 1 ili 0.

2.11 Obrasci

Dizajneri internetskih stranica u današnje vrijeme se najčešće koriste obrascima kada ne žele javno objaviti adresu svoje elektroničke pošte. Uz količinu neželjene elektronske pošte koja svakodnevno pristiže, takva odluka je potpuno razumljiva.

Za definiranje se koristi oznaka ***<form>***, a unutar njenog početnog i završnog dijela dodaje se tekst obrasca i polje za unos podataka. Važno za znati je da svaki element obrasca mora biti imenovan, kako bi se unesena vrijednost mogla dodati varijabli istoga imena.

<input type="text" name="korisnik" size="15">



Sl.2.17 Primjer polja za unos

Podaci uneseni u obrazac šalju se metodom *GET* i/ili *POST*. Slanje, odnosno metoda slanja podataka, definira se pomoću atributa ***method***. Iz razloga što korištenje metode *GET* zahtjeva poznavanje jezika tipa ASP ili PHP, jednostavnije je koristiti metodu *POST* s kojom se izravno šalje na adresu elektroničke pošte. Atribut ***action*** imati će vrijednost istu kao i hiperveza na adresu
Kako podaci ne bi bili poslani kao privitak u poruci, nego kao tijelo poruke, dodaje se atribut ***enctype*** čija se vrijednost definira kao "text/plain".

<form method="POST" action=mailto:valjavec.boris@gmail.com enctype="text/plain">

2.11.1 Polja za unos podataka

Polje za unos teksta definira se oznakom ***<input>*** i atributom ***type***, koji ima vrijednost "text". Atributom **<size>**određuje se širina polja (odnosno određuje ju broj znakova koji mogu biti uneseni u polje). Vrijednosti polja zadaju se atributom ***<value>***.

2.11.2 Polje za unos teksta

Pri definiranju polja za unos teksta, koristi se oznaka <***textarea***>. Atribut ***rows***, određuje broj redaka, a atribut ***cols*** širinu polja. Moguće je unaprijed zadati vrijednost navedenog elementa, a to može biti učinjeno na sljedeći način:

<input type="text" name="korisnik" size="15">



Sl.2.18 Polje za unos teksta

2.11.3 Polje za upis lozinke

Vrijede ista pravila kao i za definiranje običnih tekstualnih polja. Razlika je jedino u vrijednosti atributa ***type***, koji je u ovom slučaju "***password***".

<input type="password" name="korisnik" size="15">



Sl.2.19 Polje za unos lozinke

2.11.4 Potvrdni okviri

Potvrdni okviri se definiraju oznakom <***input***>. Vrijednost atributa ***type*** je "checkbox", a atributom ***value*** zadaje se vrijednost potvrdnog okvira.

<input type="checkbox" name="accept" value="yes">

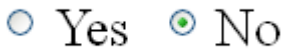


Sl.2.20 Potvrdni okvir

2.11.5 Opcijski gumbi

Opcijski gumbi definiraju se oznakom ***<input>***. Atribut ***type*** poprima vrijednost "radio", a atributom ***value***, definira se vrijednost odabranog opcijskog gumba.
Svi opcijski gumbi iste skupine moraju imati isto ime definirano atributom ***name***. To omogućuje odabir samo jedne vrijednosti iz niza ponuđenih opcija te grupe.

<input type="radio" name="answer" value="yes" checked>Yes
<input type="radio" name="answer" value="no" checked>No



Sl.2.21 Primjer opcijskih gumba

Također, oznakom ***<input>,*** ovisno o tome je li vrijednost atributa ***type*** *"submit"* ili *"reset"*, mogu se definirati gumbi za slanje , odnosno brisanje unesenih podataka.

2.11.6 Izbornik

Prilikom izrade izbornika, koriste se dvije oznake:
<select>–koristi se za definiranje početka/kraja izbornika
<option>–koristi se za definiranje svake ponuđene opcije u izborniku (isto kao što oznaka ***<li>*** definira elemente liste).

Koliko će opcija izbornik imati, određuje se oznakom ***<select>*** i atributom ***size***.

<select size="1" name="tip">
<option selected>Mercedes</option>
<option>...<option>

<option>...<option>

</select>



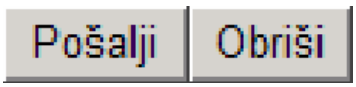
Sl.2.22 Padajući izbornik

2.11.7 Gumbi

Još jedan element obrasca koji se definira oznakom <***input***>. Ovisno o tome je li vrijednost atributa ***type*** *submit ili reset*, definira se gumb za slanje ili za brisanje unesenih podataka. Atribut ***value*** određuje što će pisati na gumbu.

<input type="submit" value="Send" name="button_1>

<input type="reset" value="Clear" name="button_2>



Sl.2.23 Gumbi za slanje i brisanje teksta

3.CSS

CSS je jezik koji uz HTML služi za oblikovanje internetskih stranica. Ujedno je i osnovna tehnologija na kojoj se temelji današnji web.

Kratica CSS znači "*Cascading Style Sheets*", a sami pojam "*style sheet*" koristi se za datoteku koja sadrži CSS kod. To jest, "*style sheet*" *je* datoteka koja definira elemente, stil, odnosno sami izgled web stranice.

Pojam "*Cascading*" s druge strane, označava kaskadnu primjenu CSS–pravila, a može se napisati tako da pravilo bude primijenjeno na sve, neke ili za točno određene elemente.

Ranije se oblikovanje web stranica vršilo uz pomoć HTML–a, pri čemu se stvorio problem miješanja sadržaja i strukture s kodom, čija je svrha bila isključivo prezentacija.

Pojavom CSS–a nastojao se riješiti taj problem. Ideja CSS-a bila je odvajanje prezentacijskog koda u zasebne datoteke i njegovo definiranje pomoću jednostavnih pravila koja se mogu odnositi na više elemenata odjednom.

Prva verzija CSS–a, definirana je krajem 1996, no do njenog usvajanja prošlo je dosta vremena. Dugo Internet preglednici nisu dosljedno implementirali CSS specifikaciju, pa se autori nisu mogli pouzdati da će stranice izgledati približno isto u svim preglednicima.

3.1 Kako uključiti CSS dokument u HTML kod

HTML dokument se povezuje sa CSS datotekom uz pomoć linka (element link uvijek mora biti unutar HTML elementa ***head***):

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

Prilikom korištenja ovog elementa za uključivanje CSS–a, atribut ***rel*** mora imati vrijednost "stylesheet" a atribut ***type*** "text/css".
Atribut ***href*** se postavlja na putanju CSS datoteke koja se uključuje.

Najčešći načini korištenja relativne putanje prikazani su u tablici :

Putanja	**Opis**
style.css	Datoteka *style.css* i HTML datoteka nalaze se u istoj mapi
css/style.css	Datoteka *styke.css* se nalazi u mapi *css*. Mapa *css* i HTML datoteka se nalaze u istoj mapi
../css/style.css	Datoteka *style.css* nalazi se u mapi *css*. Mapa u kojoj se nalazi HTML–datoteka i mapa *css* nalaze se u istoj mapi.

Tab3.1 Načini korištenja relativne putanje

3.2 Sintaksa CSS–a

CSS kod ima svoja pravila. Na primjer:

```
p
{
  color: blue;
  font size: 12px;
  float: center
}
```

Ovako napisan kod, postavlja svim ***"p" elementima*** boju teksta plavu, veličinu fonta 12 piksela te poravnava kompletan tekst u sredinu.

Dio pravila koji određuje na koje se elemente pravilo odnosi naziva se *selektor*. Svako pravilo mora se započeti selektorom, a najjednostavniji je upravo naziv elementa.

CSS nije osjetljiv na prazan prostor, pa vitičaste zagrade nije potrebno pisati u zasebne redove, ali se to uzima kao jedno od nepisanih pravila koje se koristi radi lakše čitljivosti samoga koda.

Unutar vitičastih zagrada najprije se navodi svojstvo koje se postavlja (***color, font size, float*** i slično). Nakon što se navede svojstvo koje se želi postaviti, piše se dvotočka, a iza nje vrijednost na koju se postavlja to svojstvo. Svojstvo i vrijednost zajedno čine deklaraciju.

3.3 CSS selektori

Svako CSS pravilo započinje selektorom, a upravo selektor određuje za koji HTML element vrijedi to pravilo. Kao selektor mogu koristiti nazivi, klase ili identifikatori.

3.3.1 Naziv elemenata

To je ujedno i najjednostavniji primjer selektora. Ako element ima naziv "P", zapis će izgledati ovako:

```
p
{
  color: blue
}
```

3.3.2 Klasa

Kako bi se preciziralo na koje elemente se odnosi pojedino pravilo, te elemente potrebno je označiti klasom određenog imena, korištenjem atributa ***class***:

```
<p class="test">
  Ovo je primjer klase.
</p>
```

Kada se unutar selektora navodi klasa, potrebno je prije imena klase staviti točku:

```
.test
{
  color: blue;
}
```

Definirano pravilo će vrijediti samo za elemente koji imaju postavljenu klasu "test".

3.3.3 Identifikator

Osim klase, HTML elementi se mogu označiti identifikatorom, korištenjem atributa ***id***.

```
<p id="test2">
        Ovo je drugi test.
</p>
```

Princip definiranja je isti kao i za ranije navedene primjere, jedina razlika je u tome da ispred identifikatora mora stajati oznaka "#", odnosno:

```
#test2
{
  color: red;
}
```

Pravilo će se primijeniti samo na elemente s identifikatorom *test2*. Identifikator mora biti jedinstven, to jest: na HTML stranici identifikator ne bi trebao biti dva puta upotrebljen.

3.4 Prioritet CSS pravila

U CSS-u prioritet se određuje samim selektorom. Selektor sa većom specifičnošću imati će veći prioritet.

Navođenje klase je specifičnije nego navođenje naziva elementa, pa će tako drugo pravilo imati veći prioritet i na primjer, ispisani tekst će poprimiti svojstva definirana unutar klase.

Specifičnije od klase je navođenje identifikatora, pa pravilo koje u selektoru navodi identifikator, imati će veći prioritet od oba navedena pravila.

Pojednostavljeno, specifičnost selektora se računa na sljedeći način:

- specifičnost naziva elementa jednaka je 1
- specifičnost klase jednaka je 10
- specifičnost identifikatora jednaka je 100
- ako selektor ima više naziva elemenata, klasa ili identifikatora, njihove vrijednosti se zbrajaju i na taj način se dobiva ukupna vrijednost selektora.

Primjer:

Selektor (1)***P span strong*** ima specifičnost 3, jer sadrži tri naziva elementa.

Selektor (2)***test strong***, specifičnost iznosi 11 iz razloga što sadrži jednu klasu i jedan naziv elementa.

Dakle selektor (2) ima veću specifičnost od selektora (1).

U slučaju kada dva selektora imaju istu specifičnost, veći prioritet će imati pravilo koje je navedeno poslije.

3.5 Pseudo klase

To su izrazi u CSS–u, koji omogućuju selektiranje elemenata na sličan način kao što to čine klase. Koriste se poput klasa, ali se za razliku od klasa ne postavljaju na elemente u HTML–u.

3.5.1 Pseudo klase vezane uz korisničke akcije

Ovisno o korisničkim akcijama, moguće je stilizirati elemente i na taj način dodatno naglasiti informaciju o tome koji je element korisnik odabrao, te mu na tako olakšati navigaciju i interakciju s web stranicom.

Najčešće korištene pseudo klase:

-link: odabire link *(element a)*, koji korisnik nije posjetio

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

-visited, selektira link koji je korisnik već posjetio

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

-active, selektira element na koji je korisnik upravo kliknuo

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

-hover, selektira element preko kojeg korisnik trenutno prelazi mišem. To može biti link ili neki dugi element tipa ***div,span,*** i slično.

```
div:hover
{
        color: purple;
}
```

-focus, selektira element koji je trenutačno u fokusu

```
input: focus
{
        color: yellow;
}
```

3.5.2 Pseudo klase vezane uz unos podataka

Ako iz nekog razloga korisniku nije dopušten unos teksta ili klik na element, to je potrebno predočiti na vizualan način. Preglednici imaju svoj zadani stil, s kojim prikazuju onemogućene elemente, no taj stil se može promijeniti uz pomoć dvije pseudo klase:

-enabled, selektira elemente koji su omogućeni

```
input:enabled
{
 color:black;
}
```

-disabled, selektira elemente koji su onemogućeni

```
input:disabled
{
 color:red;
}
```

Moguće je koristiti i specijalne elementi za unos (kvačica ili dugme za odabir), na koje se mogu primijeniti dvije pseudo klase koje selektiraju element ovisno o njegovu stanju (označen ili ne označen).

-checked, selektira elemente koji su označeni (elementi input tipa *checkbox* ili *radio*).

```
input:checked
{
  color:blue;
}
```

-indeterminate, selektira elemente koji nisu označeni

```
input:interminate()
```

3.6 Nasljeđivanje vrijednosti

Elementi koji nemaju definiranu vrijednost za neko svojstvo, koristiti će vrijednost iste preuzetu od roditeljskog elementa. To omogućuje da se osnovna svojstva kao boja, veličina ili vrsta fonta postave samo na jednom mjestu, umjesto da ih se posebno definira za svaki element.

```
body
{
        color: blue;
}
```

Ako element ***body,*** ima postavljenu boju teksta plavu, ona će biti plava na svim drugim elementima, bez da je treba eksplicitno postaviti, zbog toga što su svi elementi djeca elementa ***body***.

3.7 Komentari

Koriste se pri pojašnjavanju nekog pravila ili deklaracije. To je u biti proizvoljni tekst koji opisuje neku liniju ili blok koda.
Započinju slijedom znakova **/***, a završavaju slijedom ***/**. Mogu se nalaziti u jednom retku ili se protezati kroz više njih.

```
body
{
        /* definiraj boju */
    color:blue;
    font size: 12px;        /* definiraj veličinu fonta */
}
```

Često, programeri koriste komentare prilikom testiranja, kako bi privremeno isključili željeni dio koda:

```
body
{
    /* color:blue; */
    font size: 12px;
 }
```

Kod koji se nalazi unutar oznake komentara neće biti primijenjen, već će biti ignoriran.

3.8 Oblikovanje teksta

Postoje brojne mogućnosti pri oblikovanju teksta–osnovne, kao boja, veličina slova, font, pa do ne tako često korištenih kao prored, razmak između slova i slično.
Danas CSS nudi mogućnosti kao što je postavljanje sjene na slova, te korištenje fontova koje korisnik ne mora imati instalirane na svom računali, što je značajno unaprijedilo tipografiju na Internetu.

3.8.1 Boja

Boja teksta se postavlja pomoću svojstva ***color***. Postoji nekoliko načina za postavljanje toga svojstva .

3.8.1.1. Definiranje boje preko njezinog naziva

Vrijednost svojstva(*color*), postavlja se navođenjem engleskog naziva boje, na primjer:

color:blue;

Osnovne boje dostupne za korištenje navedene su u tablici;

Boja u CSS-u	Objašnjenje
aqua	svijetlo-plava
black	crna
blue	plava
fuchsia	svijetlo-ljubičasta
gray	siva
green	zelena
maroon	"trula višnja"
navy	"mornarski" plava
olive	maslinasto–zelena
orange	narančasta
purple	ljubičasta
red	crvena
silver	srebrna
white	bijela
yellow	žuta

Tab3.2 Lista osnovnih boja

Nazivi boja su jednostavni za korištenje i praktični, no ovakav način primjene ograničava izbor. Upravo zbog toga su i uvedeni sljedeći načini definiranja boja.

3.8.1.2.Definiranje boje primjenom heksadecimalnog koda

Ujedno jedan od najčešćih načina definiranja boje. Zapis ima oblik:

color: #f4521b;

Kod ima 6 heksadecimalnih znamenki ("tri para" sačinjavaju tri komponente), ispred kojih se navodi oznaka "**#**". Koriste se znamenke od 0 do 9 i slova a, b, c, d, e i f.
Navedene tri komponente definiraju:

- prve dvije znamenke koda boje predstavljaju vrijednost crvene
- druge dvije zelene,
- a posljednje dvije plave komponente boje

Veliki nedostatak korištenja takvoga zapisa je teška čitljivost, te je nemoguće iz samog CSS–a prepoznati o kojoj se boji radi.

3.8.1.2 Definiranje boje primjenom RGB i RGBa modela

U CSS–u je moguće postaviti boju izravnim navođenjem vrijednosti njezinih komponenti čiji je interval od 0 do 255, a definiranje ima zapis oblika:

color: rgb(200, 50, 30);

RGB model ima mogućnost proširenja na RGBa zapis, u kojem komponenta "a" predstavlja prozirnost. Moguće vrijednosti za prozirnost su 0 i 1, a predstavljaju postotak prozirnosti.

3.8.1.3 Zapis boje HSL i HSLa modelom

Treći način zapisivanja boje preko komponenti, a veoma sličan RGB modelu je upravo HSL model. Vrijednost komponente H (eng. hue) predstavlja nijansu, S (eng. saturation)

zasićenost, a L (eng. lightness) predstavlja svijetlost boje. Koristi se isključivo za dobivanje svjetlijih i tamnijih tonova iste boje. Definiranje ima zapis oblika:

color: hsl(5, 80%, 50%);

Isto kao i kod RGB modela moguće je definirati prozirnost boje dodavanjem komponente "a".

Vrijednosti komponenti elemenata RGB-a i HSL-a mogu biti kopirane iz grafičkih programa, ali pri tome treba voditi računa iz razloga što se u tim programima koriste varijacije navedenih modela, pa dobivena boja nije posve identična.

3.9 Font

Tekst kojim se ukrašavaju internetske stranice ima veliku ulogu u vizualnoj interpretaciji i estetici same stranice, pa je upravo iz toga razloga potrebno znati neke činjenice i pravila vezane uz fontove.

3.9.1 Vrsta fonta

Vrsta fonta se postavlja pomoću svojstva ***font–family***. Kao vrijednost se navodi naziv fonta:

font–family: Arial;

Pojedini fontovi u samom nazivu sadrže razmake ili specijalne znakove, pa ih je potrebno zapisati uz pomoć jednostrukih ili dvostrukih znakova navoda

font-family: **'** Times New Roman**.'**

Ako slučajno navedeni font nije instaliran na korisnikovu računalu, preglednik će prikazati svoj predefinirani font. Kako do toga ne bi došlo, može se navesti više zamjenskih fontova, koji će se upotrijebiti u slučaju da prvi font nije instaliran:

font–family: **'** Times New Roman **',** Arial;

Postoji pet osnovnih fonta:

Tip fonta	Objašnjenje
cursive	Font koji podsjeća na rukopis (Comic Sans)
fantasy	Font prikladan za naslove, vrlo dekorativan (Impact)
monospace	Font s jednakom širinom svih slova (Courier New)
serif	Font s serifima (Times New Roman)
sans-serif	Font bez sarifa (Arial)

Tab3.3 Lista osnovnih fontova

Vrsta fonta isto kao i boja, ako nije postavljena, bit i će naslijeđena od roditeljskog elementa.

3.9.2 Veličina fonta

Definira se uz pomoć svojstva ***font-size***. Vrijednost tog svojstva može biti postavljena na više načina:

- pomoću ključnih riječi
- apsolutnih jedinica (pikseli ili točke)
- te pomoću relativnih jedinica (postotci, *em* i *rem* jedinice)

3.9.2.1 Ključne riječi

Definiranje fonta navođenjem ključne riječi, izgleda ovako

font-size; large;

Dostupne ključne riječi i njihovo objašnjenje prikazani su u sljedećoj tablici:

Ključna riječ	Objašnjenje
xx – small	Najmanja veličina slova
x – small	Vrlo mala veličina slova
small	Manja veličina slova
medium	Srednja veličina slova
large	Veća slova
x – large	Vrlo velika slova
xx – large	Najveća slova
larger	Za razinu veća slova od roditeljskog elementa
smaller	Za razinu manja slova od roditeljskog elementa

Tab3.4 Lista ključnih riječi za definiranje fonta

Veličina fonta ako nije eksplicitno postavljena, biti će naslijeđena od roditeljskog elementa.

3.9.2.2. Pikseli

Postavljanje veličine fonta uz pomoć piksela, definira se:

font-size: 16px;

Piksel, je mjerna jedinica koja je definirana veličinom pojedine točke na ekranu računala. Upotrebljava se prilikom preciznog definiranja veličine teksta, no definirana veličina teksta uvijek malo varira, s obzirom na preglednik koji se upotrebljava za prikazivanje internetskih stranice.

3.9.2.3 Točke

Postavljanje veličine fonta uz pomoć točaka , ima zapis:

font-size: 14pt;

Točka je mjerna jedinica koja se upotrebljava kod definiranja CSS–a za ispis internetskih stranica (nije preporučljivo koristiti ih za definiranje veličine teksta).

3.9.2.4 Postotci

Postavljanje veličine fonta, u postotcima, definira se kao:

font-size*:*60%*;*

Ovo je relativan način postavljanja veličine fonta u odnosu na roditeljski element. Na isti način se definira veličina fonta korištenjem ključnih riječi, s tim da se ovako povećava preciznost definiranja.
Najveća prednost kod relativnog definiranja veličine fonta jest u tome da je na jednom mjestu moguće promijeniti veličinu fonta za sve elemente.

3.9.2.5 Jedinica em

Na sljedeći način postavlja se veličina fonta na 0.60 (odnosno 60%), od veličine roditeljskog elementa:

font-size:0.75em;

Em je relativna mjerna jedinica za duljinu, koja se koristi slično kao i postotci. 1 *em* predstavlja veličinu fonta jednaku onoj roditeljskog elementa, o.5 *em* predstavlja upola manju, a 2 *em* predstavlja dva puta veću veličinu fonta od roditeljskog elementa. Potrebno je voditi računa, da veličinu fonta, uvijek treba zadati u odnosu na roditeljski element.

3.9.3 Poravnavanje teksta

Za poravnavanje teksta, koristi se svojstvo ***text-align*** sa sljedećim vrijednostima:

KLJUČNA RIJEČ	OBJAŠNJENJE
left	Poravnavanje uz lijevi rub elementa u kojem je sadržan
right	Poravnavanje uz desni rub elementa u kojem je sadržan
center	Test je centriran na sredini elementa u kojem je sadržan
justify	Tekst je rastegnut , tj. poravnat uz lijevi i desni rub elementa u kojem je sadržan

Tab3.5 Vrijednosti svojstva *text-align*

Ako tekst mora biti poravnat na desnu stranu, potrebno je napisati:

text-align: right;

Ako nije definirana, vrijednost ovoga svojstva nasljeđuje se od roditeljskog elementa.

3.9.4 Velika i mala slova

Ima slučajeva kada je potrebno čitav tekst u nekom elementu ispisati velikim slovima (naslovi, navigacija i slično). Da bi se to napravilo, nije potrebno ručno mijenjati tekst u HTML datoteci, već se može koristiti svojstvo ***text-transform***.

text-transform: none;

U sljedećoj tablici je dan popis svih raspoloživih ključnih riječi uz primjere:

Ključna riječ	***Objašnjenje***	***Primjer***
none	Običan tekst	Otišao sam u Zagreb.
capitalize	Svaka riječ započinje velikim slovom	Otišao Sam U Zagreb.
lowecase	Sva slova su mala	otišao sam u zagreb.
uppercase	Sva slova su velika	OTIŠAO SAM U ZAGREB.

Tab3.6 Vrijednosti svojstva *text-transform*

Ako nije definirana, vrijednost ovoga svojstva nasljeđuje se od roditeljskog elementa.

3.9.5 Ukrašavanje teksta

3.9.5.1 Crte

Uz pomoć CSS–a, tekst može biti podcrtan, precrtan i slično. Kako bi se to postiglo koristi se svojstvo ***text-decoration***, a zapis ima sljedeći oblik:

text-decoration: overline;

U tablici su prikazana raspoloživa svojstva i ključne riječi:

Ključna riječ	**Objašnjenje**
none	Običan tekst
underline	Podcrtani tekst
overline	Tekst s linijom iznad
line-through	Precrtani tekst

Tab3.7 Vrijednosti svojstva *text-decoration*

Ako nije definirana, vrijednost ovoga svojstva nasljeđuje se od roditeljskog elementa.

3.9.5.2 Sjena

Tekstu može biti dodana i sjena, a to se postiže pomoću svojstva ***text-shadow:***

text-shadow:silver 10px 10px 10px;

Definiraju se četiri parametra kod postavljanja ovog svojstva. Ti parametri redom su:
boja sjene–može se definirati pomoću imena boje, heksadecimalnog ili RBG koda
horizontalni pomak–određuje pomak sjene u odnosu na tekst (izraženo u pikselima)
vertikalni pomak–također određuje pomak sjene u odnosu na tekst
radijus zamućenja–definira duljinu za koju je tekst raširen, što stvara efekt zamućenja

U tekstu je moguće istovremeno postaviti više sjena, a to se radi pojedinačnim izdvajanjem pomoću zareza:

text-shadow: black 5px 5px,
gray 6px 6px
silver 7px 7px

U slučaju, kada se pomak za definiranje sjene uzima negativna vrijednost, sjena će biti smještena s lijeve, odnosno s gornje strane.

3.10 Skraćeni način navođenja svojstva fonta

Neka od osnovnih svojstava za oblikovanje teksta moguće je postaviti korištenjem samo jednog svojstva, a to je ***font.*** Ta svojstva su:

- font–style
- font–size
- font–family
- font–variant
- font–weight
- font–height

Način i redoslijed postavljanja vrijednosti je

font: style variant weight size/line–height family

Prilikom definiranja, obavezna svojstva su ***font–size*** *i* ***font–family***. Jednostavan primjer navedenog je:

font: 14px Arial;

Ne navedene vrijednosti, biti će postavljene na vrijednost *normal.*

3.11 Oblikovanje elemenata

U ranije navedenom sadržaju o HTML–u, jasno je vidljivo kako su svi elementi ustvari pravokutnici, odnosno zauzimaju pravokutnu površinu unutar preglednika. Korištenjem CSS–a tim elementima moguće je definirati boju, veličinu obruba i druga vizualna svojstva.

3.11.1 Definiranje boja u pozadini

Kako bi HTML elementi bili vidljivi, potrebno im je definirati pozadinsku boju. Za postavljanje boja u pozadini, koristi se svojstvo ***background–color***, a vrijednosti mogu biti definirane:

-imenom boje – ***background–color:*** black;

-heksadecimalnim kodom **– *background–color:*** #f4321b;

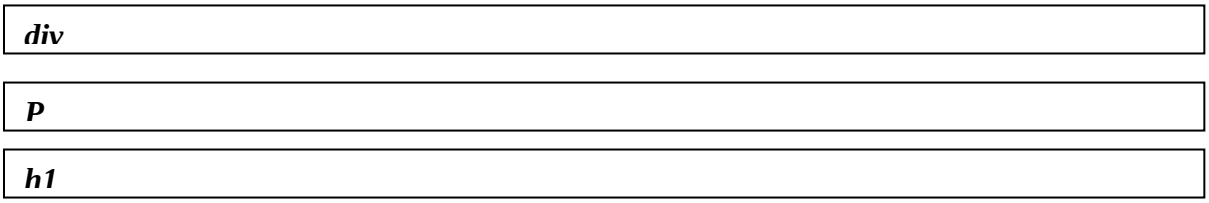
-RGB kodom**–*background – color:*** rgba (125, 40, 32, 0.9);

-HSL modelom **– *background–color:*** rgba (8, 85%, 40%);

3.11.2 Definiranje prikaza elemenata

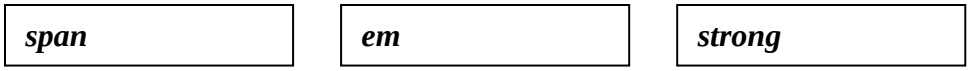
U HTML–u postoje dvije vrste elemenata. To su:

–blok–elementi (engl. block elements*)* zauzimaju cijeli redak u kojemu se nalaze. Moguće im je odrediti širinu i visinu. Svi elementi uneseni poslije njih morati će biti smješteni u novi red. To su elementi tipa ***div,p*** *i* ***h(1,2,…)***



Sl. 3.24 Primjer blok-elemenata

-linijski–elementi (engl. inline elements)*:* to su elementi prikazani unutar trenutačne linije teksta. Ne zauzimaju cijeli red već samo onoliko prostora koliko zauzima njihov sadržaj. To su elementi tima ***span, em*** *i* ***strong***



Sl. 3.25 Primjer linijskih-elemenata

Koristeći CSS svojstvo ***display***, elementima je moguće promijeniti tip prikaza. Ako se blok–elementima navedeno svojstvo postavi kao *inline*, oni će se i ponašati kao da su linijske elementi, a postavlja se:

```
div, p, h1
{
display: inline;
}
```

Isto je moguće napraviti linijskim–elementima.

Vrijednost svojstva ***display*** može biti postavljena kao ***inline–block***, čime dolazi do promjene ponašanja blok–elementa, tako da isti više neće zauzimati cijeli redak. Korištenjem ovoga svojstva, blok-elementima je moguće definirati visinu i širinu, što je veoma bitno prilikom rada u ograničenom prostoru.

U tablici su prikazane osnovne vrijednosti svojstva ***display:***

Ključna riječ	Objašnjenje
none	element neće biti vidljiv
block	elementi se prikazuju kao blok – element
inline-block	element zadržava svojstva blok – elementa (moguće mu je postaviti visinu i širinu), ali ne zauzima cijeli redak
flex	element se prikazuje kao blok – element u načinu pozicioniranja *flexbox*
inline-flex	element se prikazuje kao linijski – element u načinu pozicioniranja *flexbox*
inline	element se prikazuje kao blok – element
list-item	element se prikazuje ka element *li*
table-caption	element se prikazuje kao element *caption*
table.cell	element se prikazuje kao element *td*
table-column	element se prikazuje kao element *col*
table-column-group	element se prikazuje kao element *colgroup*
table-footer-group	element se prikazuje kao element *tfoot*
table.header.group	element se prikazuje kao element *thead*
table-row	element se prikazuje kao element *tr*
table-row-group	element se prikazuje kao element *tbody*
table	element se prikazuje kao element *table* (slično kao blok – element)
inline-table	element se prikazuje kao element *table* (s obilježjima linijskog – elementa)

Tab. 3.8 Osnovne vrijednosti svojstva *display*

3.12 Visina i širina

Ako se parametri elemenata unutar HTML–u ne postave eksplicitno, isti će isključivo ovisit o sadržaju ili roditeljskom elementu unutar kojeg se nalaze.

3.12.1 Visina

Definira se samo prilikom korištenja blok–elemenata pomoću svojstva ***height***, a izražava se u mjernim jedinicama za duljinu ili u postotcima.

Zapis u mjernim jedinicama za duljinu ima oblik:

height:123px;

Drugi način zapisa, odnosno zapis u postotcima, ima oblik:

height:75%;

Ako se vrijednost visine definira u postotcima, visina će biti izračunata kao postotak visine roditeljskog elementa, te će se na taj način se postići da visina bude prilagođena količini raspoloživog prostora.

Minimalna i maksimalan visina:

Minimalna visina se postavlja onda i samo onda kada postoji dinamički sadržaj čija visina varira. U takvim slučajevima se ne ograničava sama visina elementa, nego da se ista automatski prilagođava visini sadržaja.
Definira se pomoću svojstva ***min-height:***

min-height:50px;

Vrijednost svojstva, može se postaviti mjernim jedinicama duljine ili u postotcima. Inicijalna vrijednost je 0.

Maksimalna visina jeste ona visina koju element smije zauzeti. Definira se uz pomoć svojstva ***max-height***:

max-height:700px;

Vrijednost se može postaviti u mjernim jedinicama za duljinu ili u postotcima. Inicijalna vrijednost je "*none*".

3.12.2 Širina

To je vrijednost koja se može postaviti samo blok elementima. Ako širina nije određena, blok elementi će zauzeti cijelu širinu roditeljskog elementa.
Blok elementi inicijalno zauzimaju cijelu širinu roditeljskog elementa. Za definiranje se koristi svojstvo ***width***, a postavlja se u jednoj od mjernih jedinica duljine (pikselima, točkama ili postotcima).

width:400px;

Prilikom korištenja postotaka kao mjerne jedinice, širina će biti izračunata kao postotak širine roditeljskog elementa. To se najčešće koristi prilikom automatskog prilagođavanja količini raspoloživog prostora.

Minimalna i maksimalna širina:

Uz pomoć svojstva ***min-width***, moguće je definirati minimalnu širinu koju element mora zauzimati:

min-width: 300px;

Ako nije određeno, inicijalna vrijednost za ovo svojstvo je 0.

Nasuprot minimalnoj širini, može se definirati maksimalna širina koju element smije zauzimati. Maksimalna širina se postavlja pomoću svojstva ***max-witdh.***

Vrijednost se također postavlja u mjernim jedinicama ili postotcima, a sama maksimalna širina nema inicijalnu vrijednost.

3.13 Padding

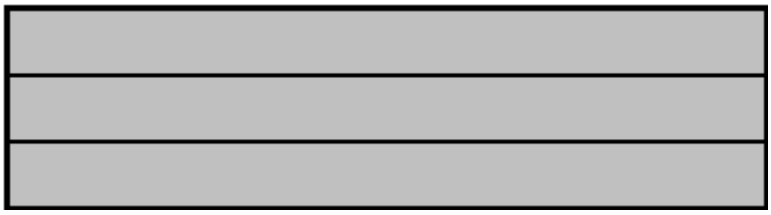
Predstavlja razmak između ruba elementa i teksta koji se nalazi unutar elementa. Može se postaviti linijskim i blok elementima, a inicijalno nije definiran.
Svojstvu se predaje vrijednost u mjernim jedinicama za duljinu ili u postotcima, a vrijednost može biti naslijeđena s roditeljskog elementa. Ako se pri definiranu navede samo jedna vrijednost, gornji, donji, lijevi i desni padding će imati istu tu vrijednost. S druge strane, ako je potrebno postaviti različite vrijednosti za padding , vrijednosti se navode u smjer kazaljke na satu, to jest gore, desno, dolje i lijevo, odnosno:

padding: 5px 10px 15px 20px;

U slučaju da se navedu tri vrijednosti od četiri, tada će se prva vrijednost opet odnositi na gornji , druga na desni, treća na donji položaj, a vrijednost koja nedostaje (lijeva), biti će jednaka nasuprotnoj vrijednosti.
Svojstvo padding ustvari je skraćeni način postavljanja vrijednosti za četiri odvojena svojstva, koja postoje za svaki pojedinačni položaj: ***padding-top, padding-right, padding-bottom*** i ***padding.left.*** Prilikom ovakvoga načina definiranja navedenog svojstava, položaji koji nisu definirani imati će inicijalnu vrijednost 0.

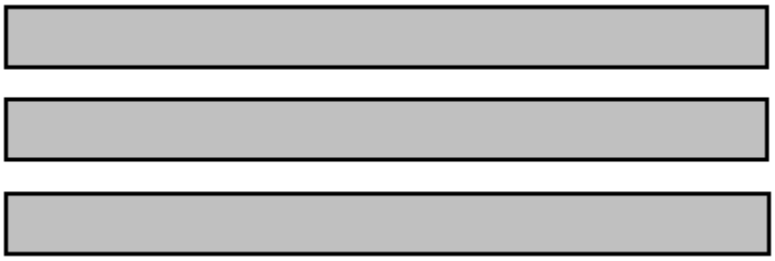
3.14 Margine

Među HTML elementima moguće je veoma precizno definirati margine, odnosno razmak (elementi tipa ***p*** i ***h*** imaju inicijalno postavljenu vrijednost margine, a elementi tipa ***div*** nemaju).
Margine se mogu postaviti linijskim i blok elementima, ali pod uvjetom da se kod linijskih elemenata može postaviti samo lijeva i desna margina, dok postavljanje gornje i donje margine neće imati nikakvoga učinka.
Ovako izgledaju elementi bez međusobno definiranih margina:



Sl. 3.26 Prikaz elemenata bez definiranih margina

Margine elemenata postavljaju se pomoću svojstva ***margin*** u mjernim jedinicama duljine ili postotcima (ako se koriste postotci, oni se odnose na širinu roditeljskog elementa).
Na slici su prikazani elementi sa definiranim marginama:



Sl. 3.27 Prikaz elemenata sa definiranim marginama

3.15 Dodavanje rubova

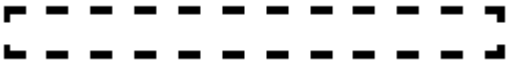
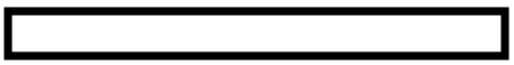
Uz pomoć CSS–a na HTML elementima moguće je iscrtati i oblikovati rub. Primjenjuje se na više svojstava, a definira se pomoću prefiksa ***border***.

3.15.1 Stilovi ruba

Stil ruba definira se korištenjem svojstva ***border–style,*** čija je vrijednost ključna riječ. Najčešće korištena ključna riječ je ***solid***, a zapis ima oblik:

border-style:solid,

Raspoložive ključne riječi prikazane su u tablici:

Ključna riječ	Objašnjenje
dashed	
double	
dotted	
groove	
hidden	rub nije vidljiv, ali se njegova širina računa kao dimenzija elementa
inset	
outset	
ridge	
solid	
none	bez ruba

Tab. 3.9 Najčešće korištene vrijednosti svojstva *border-style*

Vrijednost ovog svojstva ne može biti naslijeđena od roditeljskog elementa.
Kada se pomoću svojstva ***border-style*** postavi neka vrijednost ruba, ona će biti ista za sve rubove. Ako se kojim slučajem postave dvije vrijednosti, prva će se odnositi na gornji i donji, a druga na lijevi i desni rub.

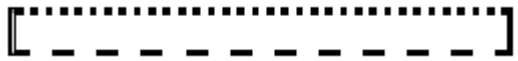
border-style: dotted solid;



Sl. 3.28 Primjer ruba definiranog ključnim riječima *dotted* i *solid*

Kao i kod definiranja nekih prijašnjih svojstava i ovom svojstvu se mogu predati četiri vrijednosti, odnosno jedna za svaki rub. Prva vrijednost odnositi će se na gornji, druga na desni, treća na donji a četvrta na lijevi rub.

border-style:dotted solid dashed double;



Sl. 3.29 Izgled ruba definiranog sa ključnim riječima *dotted solid dashed double;*

Drugi način definiranja je:

border-top-style: dotted;
border-right-style: solid;
border-bottom-style: dashed;
border-left-style: double;

Ne definirani rubovi poprimiti će inicijalnu vrijednost "*none*".

3.15.2 Širina ruba

Je vrijednost koja se postavlja pomoću svojstva ***border–width***. Kao vrijednost može se predati ključna riječ ili vrijednost u nekoj od jedinica duljine.

Ključne riječi koje se mogu rabiti su:

Ključne riječi	***Objašnjenje***
medium	
thin	
thick	

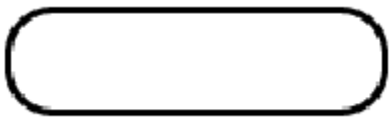
Tab. 3.10 Vrijednosti svojstva *border–width*

Vrijednost ovog svojstva se ne nasljeđuje s roditeljskog elementa.

3.15.3 Zakrivljenost rubova

Kao što je već navedeno, svi HTML elementi su originalno pravokutnici, kojima je iz estetskih razloga nekada potrebno zaobliti uglove. Za postavljanje zaobljenih uglova, koristi se svojstvo ***border-radius***, a vrijednost može bit postavljena korištenjem mjerne jedinice duljine ili postotcima.

border–radius: 20px;



Sl. 3.30 Primjer zakrivljenog ruba

Moguće je definirati vodoravni i okomiti radijus:

border–radius: 20px/10px;



Sl. 3.31 Primjer zakrivljenog ruba s definiranim vodoravnim i okomitim radijusom

Pomoću svojstva ***border–radius*** moguće je definirati različite radijuse na pojedinim rubovima, tako da se navedu četiri vrijednosti:

border–radius: 30px 5px 20px 10px;



Sl. 3.32 Primjer zasebno zakrivljenih ruba

Kao i kod prijašnjih slučajeva, zakrivljenost uglova može biti postavljena za svaki rub zasebno, ali zbog iznimno rijetke upotrebe, navedeno neće biti posebno objašnjeno.

4. Predlošci

U današnje se vrijeme na Internetu mogu pronaći predlošci za različite vrste i tipove dokumenata. Korisnik koji će koristiti neki od predložaka, mora u najmanju ruku biti upoznat s osnovnim elementima, zahtjevima i problematikom korištenja programa u kojem će "primjenjivati" taj predložak.

Takav je i slučaj kod gotovih predložaka za internetske stranice. Veoma je teško naći predložak, koji u potpunosti, zadovoljava sve tražene zahtjeve, pa je upravo iz toga razloga najbolje znati kako predložak samostalno izraditi, prema osobnim željama i zahtjevima.

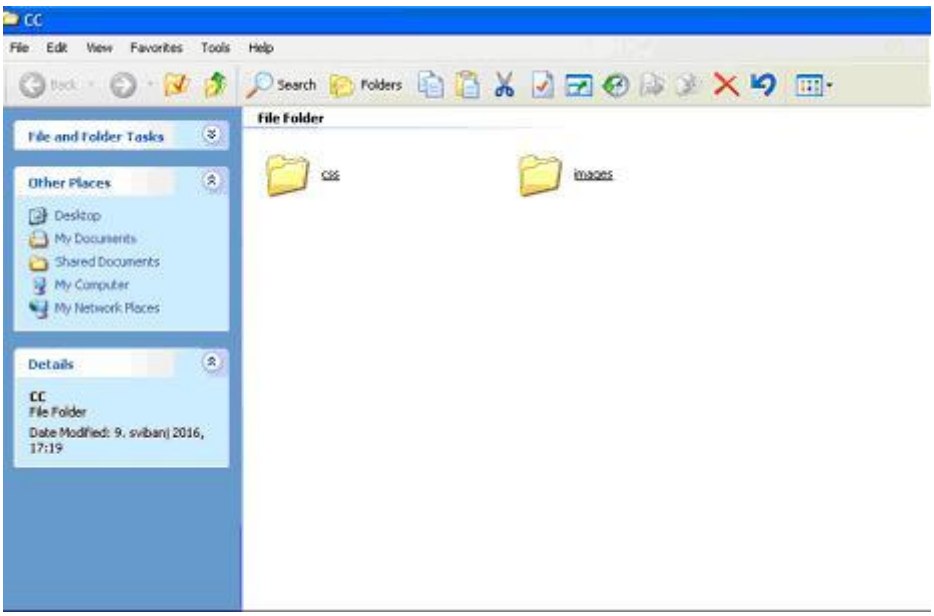
Osnovni cilj ovog rada jest objasniti izradu predloška, kako isti modificirati i primijeniti, no kako bih se izbjeglo beskrajno objašnjavanje svake pojedine linije koda i zato što su svi korišteni elementi objašnjeni u ranijim poglavljima o HTML–u i CSS–u, ovaj dio biti će čisto demonstracijski, a odnositi će se isključivo na primjenu i promjenu predložaka s obzirom na moguće zahtjeve.

4.1 Izrada predloška

Prvo i osnovno prilikom svakog stvaranja je definiranje redoslijeda osnovnih operacija, koje iznimno olakšavaju izradu kvalitetnog predloška. Te operacije su:

1. kreiranje izvorišne mape

Veoma je korisno iz razloga što znatno olakšava organizaciju dokumenata potrebnih za izradu predloška.



Sl. 4.33 Primjer izvorišne mape

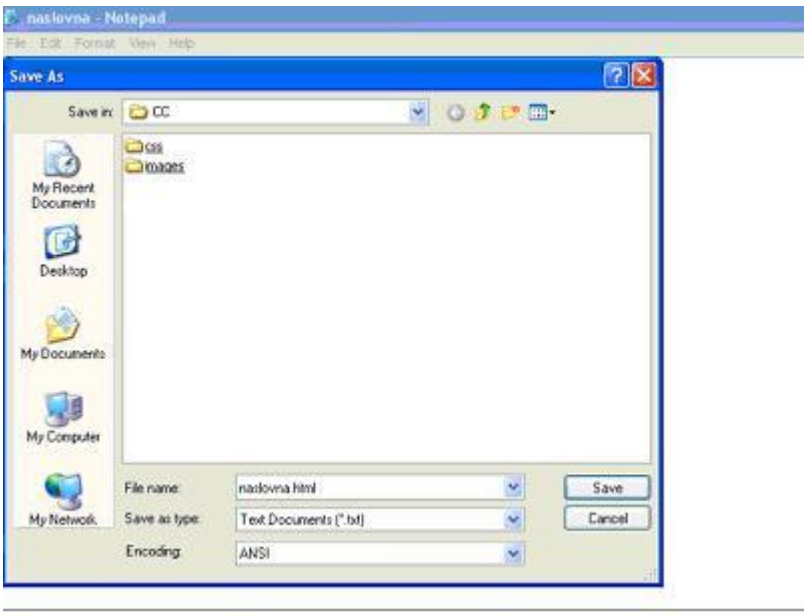
Izvorišna mapa ima proizvoljni naziv "CC", a u sebi sadrži pod–mape "css" i "images".
Prva pod–mapa u sebi posjeduje kaskadnu/kaskadne skripte s definiranim svojstvima i elementima, a druga slike koje će biti korištene prilikom izrade predloška.

NAPOMENA: ovo je samo ogledni primjer toga kako izvorišna mapa treba izgledati.

2. kreiranje "HTML" dokumenta i "CSS" skripte uz pomoć uređivača teksta

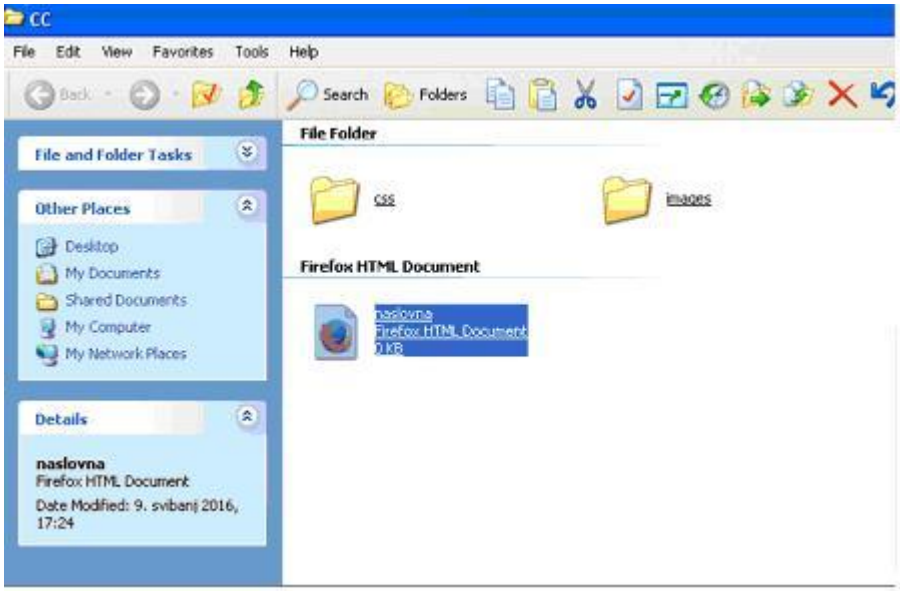
Prazni HTML dokument i CSS skripta kreiraju se na vrlo jednostavan način:

a) potrebno je otvoriti program za uređivanje teksta (NOTEPAD, WORDPAD ili NOTEPAD ++) i novo otvoreni dokument spremiti kao "naslovna.html" unutar izvorišne mape



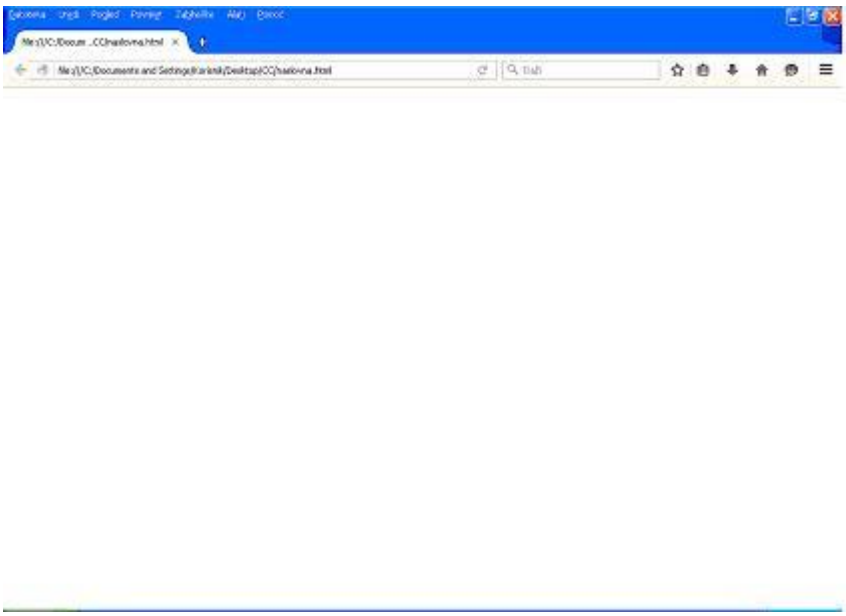
Sl. 4.34 Pretvaranje praznog tekstualnog dokumenta u internetsku stranicu

Kada je to urađeno, u izvorišnoj mapi se pojavljuje novi dokument s ikonom aktivnog Internet preglednika.



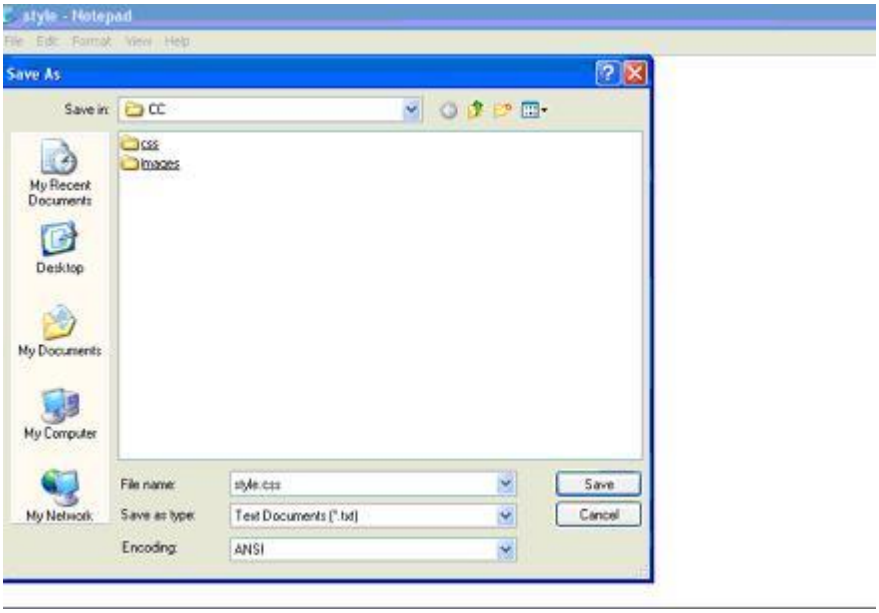
Sl. 4.35 Primjer Internet stranice s obzirom na aktivni preglednik

Taj dokument, predstavlja praznu internetsku stranicu, koja u sebi nema nikakav sadržaj, ali bez obzira na to može biti otvorena:



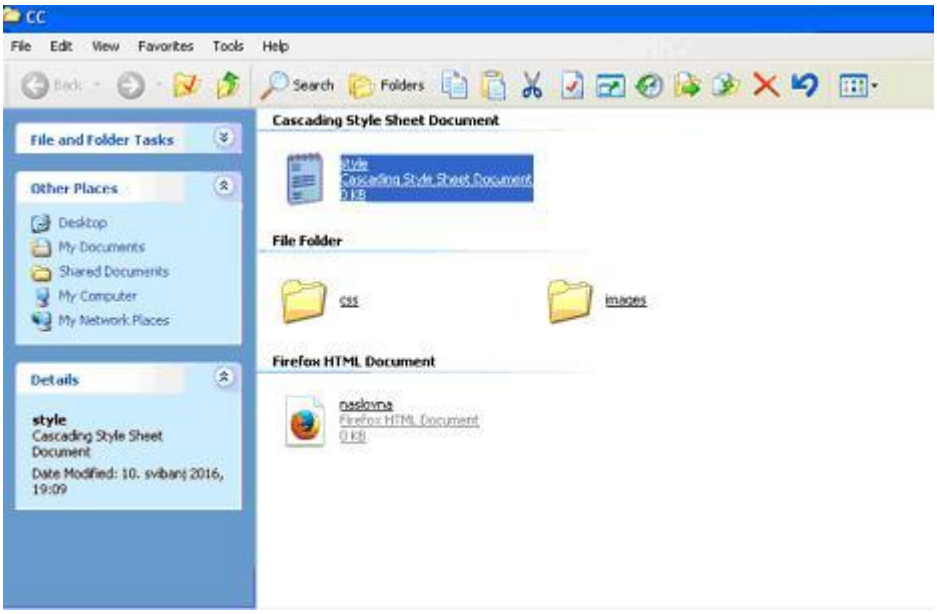
Sl. 4.36 Prazna Internetska stranica

b) na isti način koristeći uređivač teksta, kreira se i kaskadna skripta. Kako će ista u sebi sadržavati elemente vezane uz estetiku i izgled stranice, biti će joj dodijeljeno ime "style.css"



Sl. 4.37 Pretvaranje praznog tekstualnog dokumenta u kaskadnu skriptu

A unutar izvorišne mape biti će prikazana kao obični (prazni) tekstualni dokument, odnosno:



Sl. 4.38 Izgled i ikona skripte unutar izvorišne datoteke

Preostala dva koraka koja upotpunjuju cjelinu su:

3. definiranje elemenata i njihovih svojstava unutar kaskadne skripte

4. slaganje istih unutar HTML dokumenta prema utvrđenim zahtjevima i specifikacijama

Kao što je navedeno, u nastavku će biti izvršena demonstracija nekoliko "jednostavnijih" predložaka, koji će biti prikazani kroz direktnu primjenu, s naglaskom na usluge i proizvode specijalizirane auto-radionice.
Kako bi se izbjeglo objašnjavanje svake pojedine linije koda, korišteni su samo elementi i svojstva demonstrirana u ranijim poglavljima o HTML–u i CSS–u.

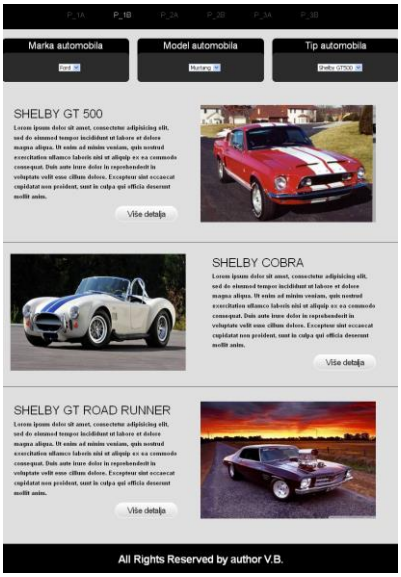
Na prvom predlošku, prikazane su usluge jedne specijalizirane auto-radionice.



Sl. 4.39 Izgled predloška s uslugama

Kao što je vidljivo na slici 39., elementi korišteni pri izradi stranice na vrlo jednostavan način prikazuju sve cijene i usluge koje ta radionica vrši.

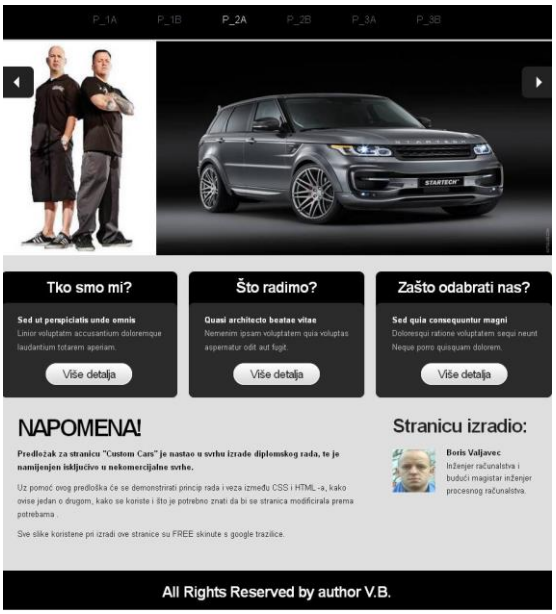
Korištenjem iste kao osnovnog predloška, uz nekoliko manjih izmjena, stranica s uslugama pretvara se u stranicu sa izdvojenim proizvodima iz ponude koji se nude potencijalnim klijentima odnosno kupcima.



Sl. 4.40 Predložak s izdvojenim modelima automobila iz ponude

Iz dosada demonstriranog, prilično je jasno da vrlo male promjene osnovne forme i premještanje elemenata, kao rezultat daju stranicu koja je u potpunosti drugačija od originalnog predloška

Drugi predložak je osmišljen i dizajniran kao naslovna stranica.



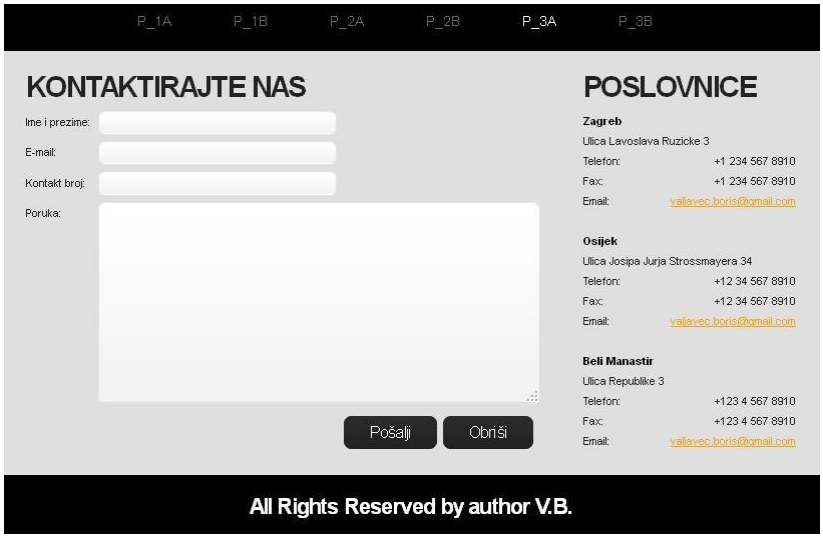
Sl. 4.41 Predložak naslovne stranice

Na isti način kao i ranije, naslovna stranica može biti promijenjena u predložak na kojem su prikazani detalji pojedinog proizvoda, koji se nalazi u ponudi.



Sl. 4.42 Predložak s definiranim detaljima automobila iz ponude

I na posljetku, uz pomoć elemenata koji su definirani u poglavlju ***2.11*** , moguće je dizajnirati predložak za jednostavni kontakt obrazac, kao ovaj prikazan na slici:



Sl. 4.43 Kontakt obrazac

Taj isti predložak se uz nekoliko manjih izmjena i prilagodbi može biti pretvoren u "blog" stranicu, koja je danas vrlo popularna, te služi kao osnovni oblik komunikacije između sadašnjih i budućih posjetitelja a isto tako vlasniku daje uvid u zadovoljstvo-nezadovoljstvo posjetitelja stranice.



Sl. 4.44 Izgled blog stranice

5.Zaključak

Teoretski i eksperimentalni dio dokazuju kako je moguće izraditi funkcionalnu Internet stranicu (predložak) pomoću HTML-a i CSS-a. Funkcionalna Internet stranica omogućuje korisniku jednostavno i intuitivno korištenje sadržaja kojim raspolaže.

Prilikom izrade ovoga rada to je postignuto uvođenjem različitih elemenata. Njihova zastupljenost na internetskim stranicama ovisi prvenstveno o informacijama koje posjeduju. Velika pozornost posvećena je tekstu koji se uzima kao neizostavna stavka svake internetske stranice, dok ostali elementi (kao na primjer multimedija) mogu, ali i ne moraju biti iskorišteni, odnosno ubačeni u stranicu. Ti elementi znatno pridonose prijenosu informacije. Osim što izazivaju veću zainteresiranost, korisnik će ih lakše zapamtiti, nego u slučaju kada su one interpretirane samo u tekstualnom obliku. S druge strane, za takve elemente na raspolaganju je mnoštvo različitih oznaka i formata zapisa, koji često mogu izazvati probleme. Upravo zbog toga je potrebno procijeniti kojim će se elementima željena informacija najbolje prenijeti i nakon toga, sukladno podršci preglednika, izabrati najprikladniji format, oznaku, te izraditi stranicu , odnosno predložak po želji.

Literatura

[1] Thomas A. Powell, McGraw- Hill Osborne; HTML & CSS: The Complete Reference, Fifth Edition; Media; 2010.

[2] Dan Thompson;: BASIC HTML - A STEP-BY-STEP Guide on How to Creating Your First Website from Begining to End!; 2010.

[3] Jennifer Niederst Robbins; Aaron Gustafson: Learning Web Design: A Beginner's Guide to (X)HTML, StyleSheets, and Web Graphics; 2007.

[4] Ian Lloyd : Build Your Own Web Site The Right Wa y Using HTML & CSS, 2nd Edition; 2008.

[5] Laura Lemay; Rafe Colburn : Sams Teach Yourself Web Publishing w ith HTML and CSS in One Hour a Day: Includes New HTML5 Coverage (6th Editi on); 2010.

[6] Meyer, Eric A.:Cascading Style Sheets: The Definite Guide, Second Edition, O'Reilly, 2004.

[7] J. Meloni, M. Morrison (2009). Sams Teach Yourself HTML & CSS in 24 Hours, Eighth Edition, Sams Publishing, Indianapolis

[8] E. Robson, E. Freeman (2012). Head First HTML and CSS, O'Reilly Media, Sebastopol

[9]W3C–World Wide Web konzorcij – organizacija zadužena za web standarde-www.w3.org

[10] Wikipedia – online enciklopedija-www.wikipedia.org

Sažetak:

Postupak izrade i primjene predloška za web stranicu

U ovom radu obrađena je izrada predloška za Internet stranicu, odnosno kako predložak izraditi, prilagoditi i primijeniti.

Rad je podijeljen na četiri cjeline. Prva, odnosno uvodna cjelina opisuje što je to Internet, nastanak prve Internet stranice, te kako se iste danas izrađuju. U drugoj i trećoj cjelini opisani su HTML i CSS, odnosno njihovi osnovni elementi te kako se uz pomoć njih može izraditi predložak (to jest Internet stranica). A u trećoj (ujedno i posljednjoj) cjelini opisana je sam izrada predloška – izrada, prilagodba i primjena.

Ključne riječi: HTML, CSS, Internet,

Abstract

Designing and applying template for web page

In this paper was treated the development of a template for a Web page, respectively how to create template, adapt and apply.

The work is divided into four parts. The first or introductory part describing what it is Internet, the emergence of the first Internet site, and how are same made now. The second and third part describe HTML and CSS, their basic elements and how with there help create a template (ie Web site). And in the third (also the last) part is described how to make template - development, customization and implementation.

Key words: HTML, CSS, Internet