**VISOKA ŠKOLA ZA INFORMACIJSKE TEHNOLGIJE**

**STRUČNI STUDIJ INFORMACIJSKE TEHNOLOGIJE**

**ZAGREB**

**DIPLOMSKI RAD**

**IZRADA WEB STRANICE**

**Stjepan Žugčić**

**Zagreb, 2018. godine**

[POPIS SLIKA: 3](#_Toc505791958)

[POPIS TABLICA: 4](#_Toc505791959)

[1. UVOD: 5](#_Toc505791960)

[2. DRUGO POGLAVLJE: 7](#_Toc505791961)

[2.1 Kratka povijest Interneta: 7](#_Toc505791962)

[2.2 O web dizajnu: 8](#_Toc505791963)

[2.3 Pravila web dizajna: 8](#_Toc505791964)

[2.3 Formati slika i grafika: 10](#_Toc505791965)

[2.4 Prezentacijski jezik Hyper Text Markup Language-HTML: 11](#_Toc505791966)

[2.5 Jezik za definiranje stilova CSS: 14](#_Toc505791967)

[2.6 Uređivanje teksta korištenjem CSS-a: 17](#_Toc505791968)

[2.7 Uređivanje HTML elemenata korištenjem CSS-a: 18](#_Toc505791969)

[2.8 Div element, selektori class i ID: 19](#_Toc505791970)

[2.9 Model kutije: 20](#_Toc505791971)

[3. TREĆE POGLAVLJE: 21](#_Toc505791972)

[3.1 HTML 5 i CSS 3: 21](#_Toc505791973)

POPIS SLIKA:

[Slika 1 : Osnovna struktura HTML dokumenta 11](#_Toc505791974)

[Slika 2: Mogućnosti prijelaza u novi red prilikom pisanja paragrafa 12](#_Toc505791975)

[Slika 3: Pisanje naslova i podnaslova korištenjem <h1> i <h3> oznaka 13](#_Toc505791976)

[Slika 4: Neuređena i uređena lista 13](#_Toc505791977)

[Slika 5: Prikaz iz preglednika za neuređenu i uređenu listu 14](#_Toc505791978)

[Slika 6: Primjena inline CSS načina na paragraf 15](#_Toc505791979)

[Slika 7: Primjer pisanja koda unutar CSS datoteke kod external načina 15](#_Toc505791980)

[Slika 9: Primjena CSS internal načina 16](#_Toc505791981)

[Slika 10: Uređivanje paragrafa CSS svojstvima color, font-family i font-size 18](#_Toc505791982)

[Slika 11: Korištenje svojstava za uređivanje boje teksta, boje pozadine, margina i obruba 19](#_Toc505791983)

[Slika 12: Poziv ID i class selektora na <div> element 20](#_Toc505791984)

[Slika 13: Model kutije koji predstavlja jedan HTML element 20](#_Toc505791985)

POPIS TABLICA:

1. UVOD:

Početkom novog milenija i razvitkom računalnih tehnologija započinje ekspanzija jednog od najvećih izuma 20. stoljeća, Interneta. Svjedoci smo vremena u kojem je nezamislivo ne biti sudionikom društvenih mreža ili u svakom trenutku pratiti događaje iz cijelog svijeta. Ulaskom u novu eru eksponencijalno je počeo rasti broj korisnika Interneta, kako privatnih tako i poslovnih. Rast korisnika Interneta zahtijevao je razvitak tehnologija koje će omogućiti brzo i lako pronalaženje željenog sadržaja. Razvitkom tehnologija, Internet stranice (u daljnjem tekstu web stranice) su sve više počele biti slične aplikacijama na korisničkim računalima koje su ljudi ranije koristili bez upotrebe Interneta. Razvitak dinamičnih web stranica, odnosno stranica koje su imale svojstva i mogućnosti aplikacija, omogućio je da prosječan korisnik računala i Interneta bez učenja programskih jezika i usvajanja programerskih vještina, samostalno može mijenjati sadržaj web stranice ovisno o potrebi i vlastitim željama. Tako u današnje vrijeme postoje mnoge besplatne verzije programa koje omogućuju korisniku da samostalno izradi web stranicu i uredi je po vlastitom ukusu sa vlastitim sadržajem. Kod ovakvog pristupa korisnik nije u mogućnosti samostalno izrađivati cjelokupni dizajn, odnosno kako će stranica vizualno izgledati, već mora iskoristiti predloške koje će mu program sam ponuditi. Ukoliko korisnik ima vlastitu ideju kako bi želio da stranica izgleda te da samostalno po potrebi može mijenjati sadržaj poput slika i teksta, a tehnički je ograničen odnosno ne posjeduje znanja i vještine za izradu takvog web mjesta, prisiljen je potražiti pomoć kompetentne osobe koja će mu to omogućiti.

Prvi dio rada uvodi čitatelja u svijet Interneta i web dizajna. Ovdje se spominje kratka povijest Interneta i ukratko prikazuje kako je zapravo zapanjujućom brzinom u kratkom vremenu Internet dostigao granice o kojima su nekoć ljudi mogli samo maštati. Spomenute su i kratko opisane tehnologije za izradu web stranica koje se danas naširoko koriste. Također opisana je osnovna struktura Hyper Text Markup Language-HTML jezika za označavanje elemenata na web stranici te Cascade Style Sheet-CSS jezika za uređivanje istih tih elemenata. Kako je cilj ovog rada bio napraviti suvremenu web stranicu koja će se moći prikazivati na različitim uređajima koji danas postoje, za izradu je korištena Boostrap biblioteka koja značajno olakšava izradu web stranica i omogućava prikaz iste web stranice na različitim rezolucijama ekrana.

Ovdje još ide uvod…

2. DRUGO POGLAVLJE:

2.1 Kratka povijest Interneta:

Prije upoznavanja sa samim tehnologijama za izradu web stranica, važno je spomenuti kratku povijest globalne mreže Interneta.

Kada se govori o globalnim mrežama misli se na mrežu rasprostranjenu po čitavom svijetu. Osamdesetih godina 20. stoljeća takva ideja je bila svrstavana u područje znanstvene fantastike, no zbog razvitka mrežnih tehnologija ubrzo je postala stvarnost i nastala je mreža svih mreža Internet.

Povijest Interneta započinje šezdesetih godina 20. stoljeća u Americi kada je ministarstvo obrane pokrenulo projekt razvijanja računalnih mreža da računala mogu izmjenjivati podatke. Kako je to bilo doba hladnog rata, jedan od razloga je bio i ostvarivanje vojne nadmoći nad Sovjetskim savezom. Ministarstvo obrane osnovalo je Agenciju za napredne istraživačke projekte odnosno Advanced research project agency*-*ARPA. Mreža je dobila naziv Arpanet i koristila se uglavnom u vojne svrhe. U prvom koraku su bila spojena dva računala, da bi krajem 1969. godine bilo umreženo četiri računala. 1973. godine dvojica računalnih znanstvenika, Robert Kahn iz ARPE i Vincent Cerf, napisali su članak u kojem su iznijeli osnove arhitekture samog Interneta. Za komunikaciju između mreža je bio potreban standardan komunikacijski protokol, a nazivao se Transmission control protocol-TCP. 1978. Godine Vincent Cerf, Jon Postel i Steve Crocker podijelili su TCP protokol na dva dijela dodajući mu protokol unutar mreže, odnosno Internet protocol-IP te su na taj način stvorili TCP/IP protokol standard na kojem danas temelji Internet.

S vremenom je broj umreženih računala naglo porastao, pa je u vojnim krugovima bilo zaključeno da veliki broj umreženih računala može ugroziti nacionalnu sigurnost. Iz tog razloga 1983. godine odlučeno je da se mreža razbije u dva dijela, vojni sa nazivom Milnet i civilni sa nazivom Arpanet. Kako je ministarstvo odlučilo komercijalizirati internetsku tehnologiju, proizvođači računala su bili primorani uključiti TCP/IP u svoje mrežne protokole.

Prava revolucija u svijetu Interneta dogodila se početkom devedesetih godina prošlog stoljeća izumom servisa World Wide Weba-WWW. WWW je zamišljen kao mreža (engleski *web*) koja bi bila ispletena preko cijele zemaljske kugle. Tvorci WWW-a su engleski inženjer Tim Berners Lee te belgijski znanstvenik Robert Cailliau, a cijeli projekt je započeo u institutu za nuklearna istraživanja Conseil européen pour la recherche nucléaire–CERN. 1994. godine Berners Lee napušta CERN i osniva W3C konzorcij odnosno međunarodno tijelo za razvoj web standarda koje djeluje i danas.

1992. godine u mrežu je bilo priključeno oko milijun većih računala diljem svijeta te je osnovano društvo za Internet koje ga proglašava općim dobrom čovječanstva. U godinama koje su slijedile, velik broj informatičkih tvrtki se usmjeravao prema mrežnim aplikacijama i korisnicima donosio brojne mogućnosti i bogatstvo Interneta. U današnje vrijeme Internet koristi preko milijardu ljudi diljem svijeta te je Internet značajno utjecao i na život ljudi. Osim što se uz Internet može učinkovito poslovati, Internet nudi bezbroj informacija iz svih područja te se može slobodno reći da je Internet vrsta računalne mreže kojoj čak ni svemir nije granica.

2.2 O web dizajnu:

Web dizajn je vještina izrade web stranica korištenjem različitih tehnologija. Pod terminom tehnologija, podrazumijeva se korištenje programskih jezika i računalnih programa koji to omogućavaju. Prve web stranice nisu imale značajan grafički dio već su se samo sastojale od poveznica na druge stranice. Prvu web stranicu napisao je Tim Berners Lee 1990. godine u CERN-u, a zanimljiva je činjenica da stranica i danas postoji na Internetu. Napretkom tehnologije stranice su sve više dobivale grafički dio i počinjale su sve više sličiti stolnim aplikacijama na računalima.

Kada se govori o dizajnu, za sam dizajn odnosno vizualni dio koji će biti prikazan, zadužen je web dizajner, dok je za tehnički dio odnosno pisanje koda kojim će biti prikazana stranica zadužen web programer. Sam grafički dio moguće je nacrtati u nekom od programa za crtanje, a zatim se odgovarajućim programskim kodom sve zajedno ukomponira u cjelovitu web stranicu. Danas postoje brojni programi koji omogućavaju izradu statičkih web stranica bez potrebnih vještina programiranja, a na način da se za izradu koriste već gotovi predlošci koje program sam ponudi. Pod statičkim se prvenstveno misli na web stranice na kojima je moguće samo pregledavati sadržaj. Iako je ovakav pristup prihvatljiv i web stranica može u potpunosti služiti svrsi, mana je što program može proizvesti višak koda zbog kojeg se osjetno može smanjiti brzina pristupa, a osim toga osoba ili institucija kojoj je stranica namijenjena može biti dosta ograničena što se grafičkog aspekta tiče.

2.3 Pravila web dizajna:

Kod izrade web stranice postoji nekoliko bitnih aspekata kojih bi se trebalo pridržavati. Jedni od najbitnijih su jednostavnost korištenja navigacijeisadržaj koji se nalazi na stranici**.** Razlog tome je što prosječni korisnik kada ciljano posjećuje neku stranicuu što kraćem vremenu želi doći do informacije koja ga zanima, a upravo će mu jednostavnost korištenja navigacije to omogućiti. U današnje vrijeme na Internetu postoji velik broj stranica, točnije preko jedne milijarde što je zaista znatan broj, a statistike govore da većina ljudi u svijetu posjećuje iste stranice. Najbolji primjer je Google čija je prepoznatljiva početna stranica zaista jednostavnog izgleda, a sam sadržaj koji ona nudi te brzina kojom je moguće dohvatiti bilo koje informacije su uistinu zapanjujući. Idući bitan faktor kod izrade web stranice jepodrška za sve preglednike. Danas su od preglednika najzastupljeniji Google Chrome, Mozilla Firefox i Internet Explorer u starijim te Microsoft Edge u novijim verzijama Microsoft Windowsa.

Početkom 21. stoljeća i razvojem mobilnih telefonskih uređaja, uređaji su počeli dobivati mogućnost spajanja na Internet i pretraživanje web stranica. Iako je u početku pristup Internetu bio dosta skroman, vrlo brzo mobilni uređaji su davali mogućnost pristupa Internetu kao i na računalu. To je bio dosta veliki izazov za programere koji su morali izgled stranice prilagoditi veličini ekrana uređaja, a glavni razlog je jednostavnost korištenja navigacije koja je ranije spomenuta. Valja spomenuti i pojavu tablet računala koja također imaju manje ekrane od standardnih ekrana koja posjeduju prijenosna i stolna računala.

Prilikom pisanja teksta za neku web stranicu također treba obratiti pažnju na što više ključnih riječi koje se koriste jer će samim time stranica biti bolje pozicionirana na tražilicama.

Prije same izrade stranice potrebno je utvrditi odnosno isplanirati web sjedište. Web sjedište je direktorij na računalu ili poslužitelju koje sadržava skup stranica međusobno povezanim poveznicama koje zajedno čine jednu web stranicu. Planiranje web sjedišta obuhvaća različite korake koji osiguravaju da sama izrada bude jednostavnija te da se unaprijed utvrdi kakav sadržaj će stranica sadržavati što uvelike olakšava posao i dizajneru i programeru. Prvi korak u planiranju web sjedišta je definiranje ciljeva. U ovom koraku se utvrđuju ciljevi web sjedišta, odnosno koja će biti namjena web stranice. U idućem koraku se utvrđuju krajnji korisnici web stranice. Bitno je da se navede o kakvom web sjedištu je riječ, odnosno radi li se o web sjedištu neke tvrtke ili institucije ili se radi o forumu ili pak društvenoj mreži. Ovaj korak je vrlo bitan jer se moraju utvrditi i tehnički detalji poput spajanja na web sjedište. Ako krajnji korisnik koristi spore veze na Internet, previše multimedijskih sadržaja poput slika i videa bi usporilo pristup stranici. Iako danas većina korisnika koristi iste preglednike, vrlo važna činjenica je i da se utvrdi koji će se preglednici koristiti prilikom pristupa stranici, kako se ne bi koristile tehnologije koje ti preglednici ne podržavaju.

Idući korak pri planiranju web sjedišta je prikupljanje informacija**.** Ovdje se prije svega misli na prikupljanje slika, tekstova i multimedijskih sadržaja koje će stranica sadržavati. Nakon ovog koraka se klijentu odnosno naručitelju stranice predstavlja idejno rješenje kako bi stranica mogla izgledati te se ovisno o željama naručitelja stranica prepravlja i nadograđuje.

2.3 Formati slika i grafika:

U današnje vrijeme kada je izrada web stranica dostigla zapanjujuću razinu, poprilično je nezamislivo da bilo koja web stranica ne sadrži multimedijske sadržaje poput slika, video uradaka ili zvukova. Kod odabira multimedijskih sadržaja treba pripaziti na veličinu u bajtima kako bi se stranica mogla što brže prikazati korisniku, a naročito ako korisnik koristi spore veze na Internet.

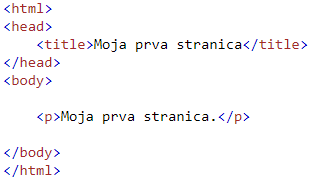
Kod odabira slika potrebno je napraviti kompresiju kako bi slike zauzimale što manje prostora na računalu. Kompresija je proces sažimanja podataka odnosno pretvaranje podataka u oblik koji zauzima manje prostora u memoriji računala. Postoje dvije vrste kompresija. Kompresija s gubitkom podataka lossy i kompresija bez gubitaka podataka lossless. Loosy kompresija odnosno kompresija sa gubitkom dijela podataka, podrazumijeva da se prilikom pretvaranja namjerno gubi dio podataka sa ciljem manjeg zauzeća u memoriji računala. Jedan od najčešće korištenih formata kod izrade web stranica, a kod kojeg se gubi kvaliteta prilikom kompresije je Joint Photographic Experts Group-JPEG format. Fotografije ovog formata mogu imati do 16 milijuna boja i velika većina programa podržava ovaj format. Najvažniji razlog korištenja ovog formata je što zauzima malo prostora na računalu, a ipak se kod kompresije ne primjećuje toliki gubitak kvalitete. Lossles kompresija odnosno kompresija kod koje se ne gube podaci, naziva se još i nezaboravna kompresija. Prilikom pretvaranja podataka svi podaci sadržani u izvornoj slici ostaju nepromijenjeni nakon kompresije. Drugim riječima slika ostaje pohranjena sa svim svojim dijelovima bez gubitka kvalitete. Najpoznatiji formati lossles kompresije, a koji se koriste prilikom izrade web stranica su Graphics Interchange Format-GIFiPortable Network Graphics–PNG. GIF format sadrži najviše 256 boja i najčešće se koristi za pohranu crteža i jednostavnijih slika. Bitne karakteristike GIF formata u odnosu na druge su prozirnost i animacija. . PNG format također sadrži 256 boja i najčešće je korišteni format za slike sa malo boja, slike sa slovima ili oštrim rubovima.

Svi ovi navedeni formati pripadaju rasterskoj grafici, no međutim valja spomenuti i vektorsku grafiku koja je najčešće korištena za prikaz geometrijskih oblika. Vektorskoj grafici pripada Scalable Vector Graphics–SVG, a sama osnova SVG-a je Extensible Markup Language–XML odnosno programski jezik za označavanje podataka. Samim time što je SVG baziran na XML-u moguće ga je kompresirati i pretraživati, a osim uređivanja u programima za crtanje, grafičke oblike u SVG formatu moguće je uređivati i u običnom uređivaču teksta.

2.4 Prezentacijski jezik Hyper Text Markup Language-HTML:

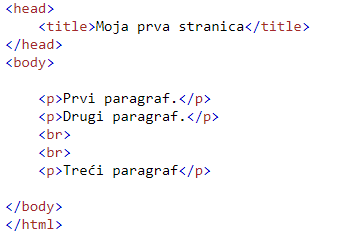
Da bi bilo koja stranica mogla biti prikazana u nekom od popularnih preglednika koji se danas koriste, mora biti napisana HTML jezikom. HTML je prezentacijski jezik odnosno jezik oznaka. Cijela osnova zasnovana je na hipertekstu. Hipertekst je tekstualna struktura koja se sastoji od međusobno povezanih čvorova. Osnovna razlika između običnog tekstualnog dokumenta i hipertekstualnog je što kod hipertekstualnog čitatelj sam odabire slijed čitanja, a upravo na taj način preglednik čita podatke i prikazuje ih korisniku točno kako su zapisani. Prve web stranice koristile su nekoliko HTML oznaka te nisu koristile značajniji grafički dio i sastojale i su se uglavnom od teksta i poveznica na neke druge stranice. Kao i kod većine programskih jezika, s vremenom se HTML jezik obogaćivao i nadopunjavao. Trenutna verzija koja se naširoko koristi je HTML 5.0. koja donosi brojne pogodnosti u odnosu na prijašnje verzije**.** Više o HTML 5.0. verziji biti će napisano na jednoj od slijedećih stranica u radu.

Kao i kod ostalih programskih jezika, prilikom pisanja HTML koda potrebno je pridržavati se osnovnih pravila. HTML dokument može biti napisan u bilo kojem uređivaču teksta, a mora završavati ekstenzijom .html. Dokument se sastoji od oznaka i važno je napomenuti da svaka oznaka koja se otvori, mora se i zatvoriti. Starije verzije HTML-a su dopuštale da se neke od oznaka ne trebaju zatvarati. Postoji nekoliko osnovnih oznaka koje sadrži svaki HTML dokument. Slika 1 prikazuje osnovnu strukturu HTML dokumenta.



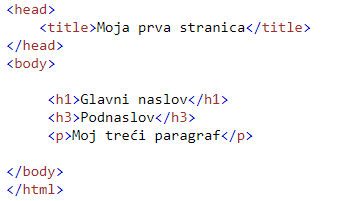
Slika 1 : Osnovna struktura HTML dokumenta

Prilikom pisanja HTML koda treba pripaziti na sintaksu i pridržavati se pravila pisanja. Svakako se preporuča ugnježđivanje elemenata radi lakše čitljivosti i kasnijeg održavanja koda. Ukoliko se dogodi sintaktička pogreška, preglednik će propustiti grešku što rezultira da će se stranica različito prikazati u različitim preglednicima, iako je prilikom kreiranja stranice ideja da stranica izgleda isto u svim preglednicima. Oznaka <html> je otvarajuća oznaka svakog HTML dokumenta, dok je </html> zatvarajuća oznaka. Unutar navedenih oznaka dolaze sve ostale oznake HTML dokumenta. Unutar oznaka <head> i </head> dolaze oznake <title> i </title> koje označavaju naslov stranice u pregledniku. U daljnjem nastavku rada će se još spomenuti neke od oznaka koje se ovdje još mogu navesti. Glavni sadržaj svakog HTML dokumenta se nalazi unutar <body> i </body> elemenata. Jedan od najosnovnijih je <p> element za označavanje teksta. HTML sve razmake, prijelaze u novi red i tabove prikazuje kao jedan razmak te ako se želi prijeći u novi red potrebno je to eksplicitno navesti, odnosno napisati još jedan <p> element. Također za prijelaz u novi red se može koristiti i oznaka <br>. Slika 2 prikazuje korištenje dva <p> elementa te oznaku <br>.



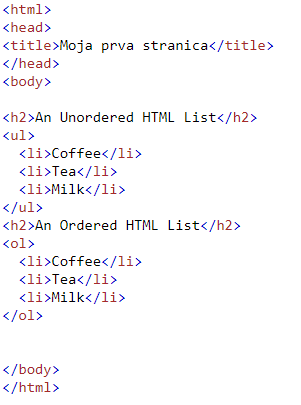
Slika 2: Mogućnosti prijelaza u novi red prilikom pisanja paragrafa

Kao i u običnom tekstualnom dokumentu svaki paragraf bi trebao imati naslov i podnaslove. U HTML-u se za pisanje naslova koriste oznake <h1> do <h6> gdje h1 označava najveću veličinu slova dok h6 označava najmanju veličinu slova. Slika 3 prikazuje pisanje naslova i podnaslova korištenjem oznaka <h1> i <h3>.



Slika 3: Pisanje naslova i podnaslova korištenjem <h1> i <h3> oznaka

Prilikom pisanja teksta kao i u običnom tekstualnom dokumentu, po potrebi se na web stranici mogu izrađivati liste. Postoje dvije vrste lista unutar HTML dokumenta, uređene liste odnosno neuređene liste. Neuređena lista se kreira korištenjem oznakom <ul>. Kod ovakvog prikaza se članovi liste prikazuju točkom, zvjezdicom ili kvadratićem. Uređena lista se kreira korištenjem oznaka <ol>. Članovi liste biti će prikazani brojem, slovom ili rimskim brojem ovisno o namjeni. Ovakav zapis se najčešće koristi kod prikaza liste gdje je bitan poredak. Za definiciju članova kod neuređene i uređene liste se koriste oznake <li>. Također liste mogu biti i ugniježđene jedna unutar druge. Izrada neuređenih i uređenih lista HTML-om prikazana je slikom 4.



Slika 4: Neuređena i uređena lista

Slika 5 prikazuje prikaz iz preglednika za primjer iz slike 4.



Slika 5: Prikaz iz preglednika za neuređenu i uređenu listu

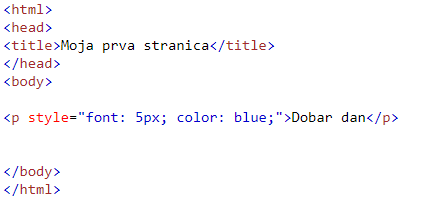
Sve dosad spomenute oznake odnose se samo na raspored teksta na samoj stranici. Za uređivanje teksta i ostalih elemenata na web stranici koristi se jezik za definiranje stilova na web stranici Cascade style sheet-CSS.

2.5 Jezik za definiranje stilova CSS:

Kako su prve web stranice bile relativno jednostavne, sam izgled stranice se uređivao direktno unutar samog HTML koda. Daljnjim razvojem web tehnologija kod je značajno porastao te se pojavila ideja da se HTML koristi samo za razmještaj elemenata na stranici, a da se CSS koristi za uređivanje istih tih elemenata. Korištenjem CSS-a znatno je olakšano održavanje izgleda stranice. Prva verzija specificirana je 1996. godine i omogućavala je promjenu fontova, boje teksta i pozadine te još nekoliko drugih mogućnosti. Kao i kod HTML-a, daljnjim razvojem omogućene su brojne mogućnosti CSS-a što je pridonijelo vizualnim napretkom web stranica. Najbitnija zadaća i svrha uvođenja CSS stilova je odvajanje dijela koda potrebnog za vizualno uređivanje stranice i dijela koda potrebnog za razmještaj elemenata na web stranici. Svaka CSS oznaka se sastoji od selektora, svojstva i vrijednosti svojstva. Selektor označava HTML element na koji će se određeni stil primjenjivati. Svojstvo određuje koje svojstvo će se urediti, ovisno o tipu HTML elementa. Jedna od osnovnih svojstava su color za boju teksta, font*-*size za veličinu teksta te background-color za boju pozadine elementa. U trenutnoj verziji CSS-a postoje brojna svojstva koja omogućavaju kreiranje zaista širokog spektra stilova. Vrijednosti svojstava ovise o kojem svojstvu je riječ. Postoje tri različita načina kako se CSS kôd može povezati sa HTML dokumentom:

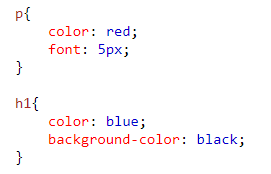
* Linijski (engleski Inline)
* Iz vanjske datoteke (engleski External)
* Unutar <style> oznaka (engleski Internal)

Linijski način omogućava uređivanje elemenata upisivanjem riječi style unutar elementa u samom HTML dokumentu. Nakon toga se unutar navodnika navedu sva svojstva koja se žele upotrijebiti. Na slici 6 je prikazan primjer korištenja inline načina.



Slika 6: Primjena inline CSS načina na paragraf

Ukoliko se za uređivanje elemenata koristi vanjska datoteka odnosno external način, datoteka mora imati ekstenziju .css. CSS datoteku je moguće napisati u bilo kojem uređivaču teksta. Pisanje datoteke se izvodi način da se CSS oznake napišu jedna ispod druge, a unutar njih, odnosno unutar vitičastih zagrada se navedu sva svojstva koja se žele primijeniti na odgovarajući element. Na slici 7 je prikazan primjer iz CSS datoteke.



Slika 7: Primjer pisanja koda unutar CSS datoteke kod external načina

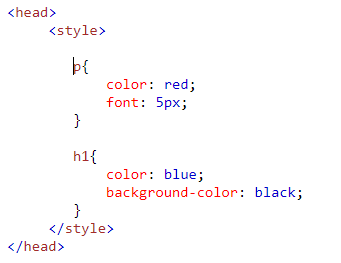
Nakon što su datoteci navedeni svi selektori i svojstva te vrijednosti svojstava koji se žele primijeniti na elemente, potrebno je datoteku povezati sa HTML dokumentom. Povezivanje CSS datoteke sa HTML dokumentom se izvodi na način da se unutar HTML dokumenta, a unutar <head> oznake navede putanja do CSS datoteke. Na slici 8 je prikazano povezivanje CSS datoteke sa HTML dokumentom.



Slika 8: Primjer povezivanja CSS datoteke sa HTML dokumentom

Oznaka linkdefinira vezu između HTML dokumenta i eksterne CSS datoteke. Oznaka typedefinira kojim načinom je kreirana datoteka, a u ovom slučaju je to tekst, dok oznaka rel definira odnos između dviju datoteka. U ovom primjeru je to stylesheet*,* odnosno stranica ili datoteka sa stilom koji će se primijeniti na odgovarajući HTML dokument.

Treći i posljednji način upotrebe CSS-a unutar HTML dokumenta je dodavanjem <style> oznake unutar <head> oznake. Unutar <style> oznake se popišu svi selektori sa svojstvima na isti način kao i kod external načina. Na slici 9 je prikazan primjer korištenja CSS internal načina.



Slika 9: Primjena CSS internal načina

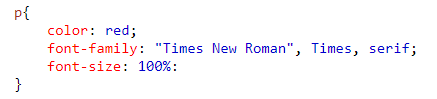
2.6 Uređivanje teksta korištenjem CSS-a:

Prilikom kreiranja web stranice, uređivanju teksta treba itekako posvetiti pažnju. Glavni i osnovni razlog je da sadržaj bude čitljiv i oku ugodan. Najprije je potrebno odabrati odgovarajući font kojim će biti prikazan tekst. Postoje dvije veće podjele fontova, a radi se o Serif i Sans-serif skupinama fontova. Serif skupini fontova pripadaju Times, Times New Roman, Courier, Courier New, Georgia i Garamond. Sve navedene podskupine imaju mala zaobljenja na rubovima i najčešće se koriste za pisanje naslova, podnaslova ili dijelova teksta koje je bitno naglasiti, a iz razloga što se kod manje veličine navedenih fontova gubi čitljivost teksta. Sans-serif podskupine fontova nemaju zaobljenja i samim time se kod manjih fontova ne smanjuje čitljivost i najčešće se koriste na većini stranica na Internetu. U skupinu Sans-serif fontova spadaju Arial, Tahoma, Verdana i Trebuchet.

Ukoliko se kod kreiranja stranice odabere font koji krajnji korisnik nema instaliran na računalu, postoji mogućnost da neće vidjeti stranicu onako kako je dizajnirana. Iz tog razloga kod odabira fonta najsigurnije je koristiti sistemske fontove koji inicijalno dolaze sa većinom operacijskih sustava. Za odabir fonta zaduženo je svojstvo font-family.

Idući bitan faktor kod kreiranja stila kojim će biti prikazana web stranica je boja teksta. Poželjno je da se boja teksta prilagodi pozadini. Ukoliko je pozadina svijetlije boje tekst mora biti tamniji, a vrijedi i obrnuto ako je pozadina tamnije boje tekst mora biti svjetliji. Za odabir boje koristi se svojstvo *color*, a vrijednost svojstva moguće je podesiti na tri različita načina. Prvi i najjednostavniji način je da se upiše engleski naziv boje primjerice red za crvenu boju. Drugi način je upisivanjem heksadecimalne vrijednosti boje, a treći način je pomoću upisivanja Red Green Blue – RGB vrijednosti. RGB označava crvenu, zelenu i plavu boju, a vrijednosti mogu biti od (0,0,0) za crnu do (255,255,255) za bijelu boju.

Nakon odabiranja vrste fonta i boje potrebno je odabrati veličinu. Za promjenu veličine fonta koristi se svojstvo font-size, a vrijednost svojstva se može upisati na više različitih načina. Naj jednostavniji način**,** aujedno i najpopularniji je upisivanjem vrijednosti u pikselima. Idući način je upisivanjem engleskih naziva large ili small. Osim navedenih oznaka moguće je ispred upisati x ili xx što znači da će font biti manji odnosno veći od uobičajene veličine. Treći način je upisivanjem vrijednosti u postocima (%) gdje veličina fonta ovisi o veličini elementa unutar koje se nalazi tekst. Zbog različitih rezolucija na različitim uređajima koje danas postoje, odabir veličine u postocima (%) je najsigurniji. Slika 10 prikazuje uređivanje paragrafa sa spomenutim CSS svojstvima.



Slika 10: Uređivanje paragrafa CSS svojstvima color, font-family i font-size

2.7 Uređivanje HTML elemenata korištenjem CSS-a:

Kao što se boja teksta mora prilagoditi boji pozadine, tako se i boja pozadine mora prilagoditi boji teksta. Boja pozadine se određuje pomoću svojstva background-color, a vrijednost svojstva se upisuje na isti način kao i kod odabiranja boje teksta.

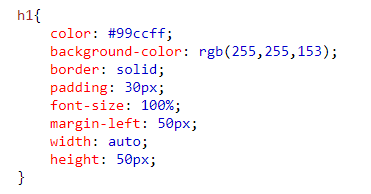
Osim odabiranja boje pozadine pojedinog elementa, elementu se kao pozadina može postaviti slika. Postavljanje pozadine se izvodi korištenjem svojstva background-image, a kao vrijednost potrebno je upisati putanju do direktorija gdje se slika nalazi. Korištenjem ključnih riječi *repeat*  slika će se ponavljati cijelom širinom ili visinom elementa, odnosno korištenjem ključne riječi *no-repeat* slika će biti prikazana samo jednom u originalnoj veličini.

Svaki element može imati obrube. Vrsta obruba se postavlja pomoću svojstva border. Postoje različite vrste obruba koje je moguće postaviti korištenjem svojstva border-style, a jedna od naj popularnijih je solid*.*Također moguće jeodabrati i debljinu obruba u pikselima. Svojstvo border moguće je koristiti samo na jednoj strani elementa korištenjem ključnih riječi top, bottom*,* left i right. Korištenjem samo svojstva margin bez ključnih riječi za pojedinu stranu elementa, ista vrijednost će biti primijenjena na sve četiri strane elementa. Po potrebi se može mijenjati boja obruba, a za tu namjenu se koristi svojstvo border-color.

Svaki element ima margine. Margina označava prazan prostor između elemenata. Ukoliko je prilikom kreiranja stranice potrebno razmaknuti elemente, to je moguće učiniti korištenjem svojstva margin te vrijednosti svojstva u postocima (%) ili pikselima (px). Također svojstvo margin moguće je koristiti samo na jednoj strani elementa korištenjem ključnih riječi top, bottom,left i right. Korištenjem samo svojstva margin bez ključnih riječi za pojedinu stranu elementa, ista vrijednost će biti primijenjena na sve četiri strane elementa.

Kako svaki element posjeduje obrube po potrebi se sadržaj unutar elementa može pomicati prema rubu elementa. To se postiže korištenjem svojstva padding. Ovo svojstvo je često korišteno jer osigurava preglednost i čitljivost. Postavljanje je isto kao i kod margina te se korištenjem ključnih riječi left, right, top i bottom**,** svojstvo paddingmože koristiti samo na jednoj strani elementa.

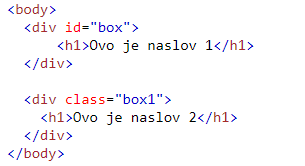
Kada se govori o visini i širini elementa odnosno o veličini, veličinu elementa moguće je mijenjati svojstvima height za visinu te width za širinu. Spomenuta svojstva imaju pred definirane vrijednosti. Inicijalna vrijednost za širinu elementa je auto što je maksimalna širina ekrana preglednika. Visina elementa se povećava sa povećanjem sadržaja unutra elementa. Na slici 11 je prikazano uređivanje <h1> elementa spomenutim svojstvima.



Slika 11: Korištenje svojstava za uređivanje boje teksta, boje pozadine, margina i obruba

2.8 Div element, selektori class i ID:

Osim već spomenutih elemenata koji su sastavni dio HTML-a poput <p> elementa za označavanje paragrafa teksta ili <h> elementa, u većini slučajeva će prilikom kreiranja web stranice biti potrebno napraviti vlastiti element sa vlastitim stilom. U tom slučaju potrebno je koristiti <div> elemente. Da bi bilo koje svojstvo moglo biti primijenjeno na neki <div> element, potrebno je definirati selektore koji će to omogućiti, a to su class i ID selektori. Definiranjem class selektora omogućava se da se isti stil može primijeniti na više različitih elemenata. Definiranjem ID selektora stil biti primjenjiv samo na jednom elementu, drugim riječima ID jednoznačno označava pojedini element. ID selektor se definira korištenjem znaka „ # “, dok se class selektor definira korištenjem znaka „ . “. Na sve <div> elemente je moguće primjenjivati sva ranije navedena svojstva. Nakon što su selektori definirani, u HTML dokumentu je potrebno pozvati ta svojstva. Na slici 12 je prikazano pozivanje selektora za <div> element.



Slika 12: Poziv ID i class selektora na <div> element

2.9 Model kutije:

Može se slobodno reći da svaki element unutar HTML-a predstavlja na neki način kutiju ili takozvani kontejner. Razmještajem tih istih kontejnera na stranici omogućava se lakša manipulacija sadržajem, a osim toga lakše je i održavanje same stranice. Model kutije (engleski box model) zamišljen je na način da se u samom centru kutije nalazi sadržaj (engleski content). Iza sadržaja dolazi ranije spomenuti paddingkoji predstavlja prazan prostor između sadržaja i obruba (engleski border). Zatim dolazi spomenuti border te nakon njega nevidljiva margina (engleski margin) koja predstavlja prazan prostor između elemenata odnosno kutija. Također kutija može sadržavati i pozadinsku boju i pozadinsku grafiku. Na slici 13 je prikazan model kutije.



Slika 13: Model kutije koji predstavlja jedan HTML element

3. TREĆE POGLAVLJE:

3.1 HTML 5 i CSS 3:

Za izradu web stranice ovog završnog rada, korištene su posljednje verzije HTML-a i CSS-a, HTML 5 te CSS 3. Dolaskom HTML 5 verzije dogodila se prava revolucija u svijetu web dizajna. Verzija omogućava kreiranje web stranice sa grafikama i pokretnim animacijama bez upotrebe nekih dodatnih programskih jezika kao što je bilo ranije potrebno. HTML 5 olakšava programerima posao jer se iste aplikacije mogu koristiti na različitim uređajima sa različitim rezolucijama. U HTML5 verziji dodano je mnoštvo elementa koji olakšavaju razmještaj elemenata na stranici, a i upotrebom tih elemenata će cjelokupna web stranica biti jednolična. Neki od tih elemenata su <header> za zaglavlje, <footer> za podnožje te <nav> za navigaciju unutar stranice. Uvođenjem HTML5 verzije olakšano je i pokretanje video i audio sadržaja jer su ranije na računalima morali biti instalirani i dodatni programi koji bi to omogućavali. Da bi bilo koja web stranica mogla imati funkcionalnosti HTML5 verzije, na samom početku pisanja koda treba napisati da se radi o HTML5 verziji kako bi to preglednik prilikom prevođenja mogao prepoznati.

Dolaskom CSS3 verzije također se dogodila revolucija u svijetu web dizajna. Uvedena su brojna svojstva koja omogućavaju vizualne promjene bez upotrebe dodatnih programa. Tako je moguće jednostavno zakriviti rubove elementa ili napraviti sjenu na sadržaj. Za pojedina nova svojstva iz CSS3 verzije potrebno je naglasiti za koje preglednike su namijenjena, odnosno ovaj način omogućava da se ista svojstva isto prikazuju na različitim preglednicima. Neke od oznaka za popularne preglednike su:

* -o- za Operu
* -moz- za Mozillu Firefox
* -webkit- za Chrome