

Nastavni predmet	SKRIPTNI JEZICI I WEB PROGRAMIRANJE
Naslov cjeline	CSS – uređivanje web sjedišta
Naslov jedinice	Vježba: CSS pozadine i uređivanje

CILJ VJEŽBE

Učenik će moći opisati i primjenjivati svojstva za oblikovanje pozadine stranice ili nekog elementa, odabrati i definirati pravilo za oblikovanje određenog svojstva elementa na stranici, dati primjer za oblikovanje, prilagoditi izgled elementa na stranici zadanoj ili zamišljenoj ideji, urediti elemente na mrežnoj stranici određivanjem njihove visine i širine, umetaka i margina te usporediti prikaz u pregledniku sa definicijama CSS pravila.

PRIPREMA ZA VJEŽBU

Odgovorite pisano u bilježnicu:

- ▶ 1. Što znate o RGBA bojama?
- ▶ 2. Na koje se načine mogu zadavati boje pri pisanju CSS pravila?

IZVOĐENJE VJEŽBE

CSS svojstva kod oblikovanja pozadine (*backgrounds*)

- za definiranje pozadinskih efekata koriste se:
 - **background-color** (određuje **boju pozadine nekog elementa**, npr. stranice (body), naslova (h1), odlomka (p) i drugo.),
 - **background-image** (određuje **sliku koja će se koristiti kao pozadina** određenog elementa, podrazumijevano slika se ponavlja i vodoravno i okomito tako da u potpunosti prekriva element),
 - **background-repeat** (opisuje **ponavljanje pozadinske slike**),
 - **background-attachment** (određuje treba li pozadinska slika *skrolati* ili biti nepomična prilikom *skrolanja* stranice. Ukoliko treba biti nepomična, koristi se vrijednost *fixed*, a ako se treba pomicati (skrolati) koristi se *scroll*),
 - **background-position** (određuje **položaj pozadinske slike** nekog elementa)

- Uz background-color može se koristiti i svojstvo *opacity* koje određuje neprozirnost, odnosno transparentnost elementa. Može poprimiti vrijednosti od 0.0 do 1.0, pri čemu **niža vrijednost određuje veću prozirnost**. Pri tome je važno napomenuti da kod primjene tog svojstva na neki element, svi elementi koje taj element sadrži također **nasljeđuju tu transparentnost** zbog čega se može dogoditi da tekst postane jako proziran, te se zbog toga teže čita.

- Primjer:

```
div { background-color: green; opacity: 0.3; }
```



- Ovo se može spriječiti tako da se za boje koristi RGBA vrijednosti.
- Primjer:**

```
div { background: rgba(0, 128, 0, 0.3) /* Zelena pozadina sa 30% neprozirnosti, prozirnost se ne odnosi na tekst. */ }
```



- Primjer za pozadinsku sliku:

```
body { background-image: url("paper.gif"); }
```

Kod ovakvog oblikovanja treba paziti da primjena pozadinske slike u kombinaciji sa tekстом ne učini tekst teško čitljivim.

- Ukoliko se želi ponavljanje pozadinske slike samo vodoravno ili samo okomito, potrebno je dodatno koristiti svojstvo **background-repeat**. Ono može imati vrijednost **repeat-x**, ako se slika ponavlja samo u vodoravnom smjeru, te **repeat-y** ako se ponavlja samo u okomitom smjeru. Ukoliko uopće nema ponavljanja to svojstvo treba imati vrijednost **norepeat**.
- Ako se pozadinska slika želi postaviti na određeno mjesto, npr. u gornji desni kut pozadine stranice, treba napisati:

```
body {
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
}
```

- Kako bi se skratio kod, moguće je specificirati sva svojstva pozadine kao jedno - background (tzv. **shorthand property**). Npr. za:

```
body {
  background-color: #ffffff;
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
}
```

može se kraće u jednoj deklaraciji napisati:

```
body {
  background: #ffffff url("img_tree.png") no-repeat right top; }
```

Pri tome je poredak vrijednosti svojstava **poredak vrijednosti svojstava** (vrijednosti ne moraju biti sve, ali poredak treba biti zadržan):

- `background-color`
- `background-image`
- `background-repeat`
- `background-attachment`
- `background-position`

- **Riješite pet vježbi zadanih na stranici [vježbe](#)**

CSS obrubi (borders)

- CSS svojstvo **border** omogućava određivanje stila, širine i boje obruba (granice) elementa na nekoj stranici. Element može imati obrub na svim svojim stranicama, može imati samo donji rub u određenoj boji, može imati zaobljene obrube, može imati podebljani obrub na lijevoj strani itd.
- CSS svojstvo **border-style** određuje koja će se vrsta obruba ispisati. Dozvoljene vrijednosti su:
 - `dotted`, `dashed`, `solid`, `double`
 - `groove`, `ridge`, `inset`, `outset` (3D obrubi, učinak ovisi o vrijednosti za `border-color`)
 - `none` (bez obruba)
 - `hidden` (skriveni obrub)
- CSS svojstvo **border-style** može imati od jedne do četiri vrijednosti za gornji, desni, donji i lijevi obrub. Ostala svojstva za obrube imaju učinak tek ako je postavljeno svojstvo `border-style`.
- CSS svojstvo **border-width** određuje širinu za četiri obruba. Može se postaviti kao specifična veličina u definiranim jedinicama (px, pt, cm, em...) ili korištenjem predefiniranih vrijednosti: `thin`, `medium` ili `thick`, npr.

```
p.two { border-style: solid; border-width: medium; }
```

Svojstvo **border-width** može imati od jedne do četiri vrijednosti (za svaki od četiri obruba), npr.

```
p.three { border-style: solid; border-width: 25px 10px 4px 35px; /* 25px gornji obrub, 10px desni, 4px donji i 35px lijevi */ }
```
- CSS svojstvo **border-color** koristi se za određivanje boje za četiri obruba. Ako to svojstvo nije postavljeno, nasljeđuje se boja elementa. Primjer:

```
p.one { border-style: solid; border-color: red; }
```
- Svojstvo **border-color** može imati jednu do četiri vrijednosti:

```
p.one { border-style: solid; border-color: red green blue yellow; /* crveni gornji obrub, zeleni desni, plavi donji i žuti lijevi obrub*/ }
```

- U slučaju potrebe da se definiira vrijednost za boju za svaku stranu pojedinačno, može se pisati:


```
p {
  border-top-style: dotted;
  border-right-style: solid;
  border-bottom-style: dotted;
  border-left-style: solid; } ili
skraćeno: p {
  border-style: dotted solid;
}
```
- Općenito, ako svojstvo **border-style** ima manje od četiri vrijednosti, definirano je ovako: u slučaju tri vrijednosti one se odnose na gornji, desni i lijevi zajedno, te donji obrub, u slučaju da su samo dvije vrijednosti odnose se na gornji i donji zajedno, te desni i lijevi zajedno. Osim za **border-style** svojstvo, to vrijedi i za **border-width** i **border-color**.
- Postoji i mogućnost skraćenog zapisivanja, odnosno korištenja svojstva **border**. Dakle, za **skraćivanje koda**, moguće je odrediti **sva individualna svojstva obruba u jednom svojstvu**. Svojstvo border je skraćeno (*shorthand*) svojstvo za slijedeća individualna: **border-width, border-style (obavezno) te border-color**, npr. :


```
p { border: 5px solid red;
}
```
- Također, moguće je odrediti sva individualna svojstva obruba za samo jednu stranicu okvira, npr.:


```
p {
  border-left: 6px solid red;
  background-color: lightgrey;
}
```
- CSS svojstvo **border-radius** koristi se za **dodavanje zaobljenih obruba** nekom elementu, npr.:


```
p {
  border: 2px solid red;
  border-radius: 5px;
}
```
- **Riješite četiri vježbe pri dnu [stranice](#).**

CSS margine (*margins*)

- CSS svojstvo **margin** koristi se za **stvaranje prostora oko elemenata**, izvan svih definiranih granica (obruba). CSS daje punu kontrolu nad marginama. Postoje svojstva za postavljanje margina za **svaku stranu** elemenata (gore, desno, dolje i lijevo): **margin-top, margin-right, margin-bottom i margin-left**.
- ta svojstva mogu imati slijedeće vrijednosti: **auto** (preglednik izračunava marginu), **length** (specificiranje margina u px, pt, cm i dr. Jedinicama), **%** (specificiranje margina u postotku širine sadržanog elementa) te **inherit** (specificiranje da margina treba biti naslijeđena od roditeljskog elementa). Dozvoljene su i negativne vrijednosti.

Primjer:

```
p { margin-top: 100px;
margin-bottom: 100px;
margin-right: 150px;
margin-left: 80px;
}
```
- Kako bi se skratio kod, moguće je navesti sva četiri individualna svojstva margina u jedno svojstvo, svojstvo **margin** pomoću:

```
p { margin: 25px 50px 75px 100px; }
```

- Ako svojstvo **margin** ima manje od četiri vrijednosti, tumači se po grupama, kao i kod svojstva **border-style**.
- Svojstvo **margin** može se postaviti na **auto** kako bi se **vodoravno centriralo element** unutar njegovog spremnika. Element će tada zauzeti navedenu širinu, a preostali će prostor biti podijeljen jednako između lijeve i desne margine, npr.:

```
div { width: 300px; margin: auto; border: 1px solid red; }
```
- U ovom primjeru ostavlja se da lijeva margina odlomka bude naslijeđena od roditeljskog **div** elementa:

```
div { border: 1px solid red; margin-left: 100px; }
p.ex1 { margin-left: inherit; }
```
- Gornja i donja margina nekog elementa ponekad se stapaju u jedinstvenu marginu jednaku većoj od dvije uključene margine. To se ne događa lijevim i desnim marginama. U primjeru:

```
h1 { margin: 0 0 50px 0; }
h2 { margin: 20px 0 0 0; }
```

obje se margine okomito stapaju u jednu od 50px.
- **Riješite četiri vježbe pri dnu [stranice](#).**

CSS umetak (padding)

- CSS svojstvo **padding** koristi se za stvaranje **prostora oko sadržaja** elementa, **unutar bilo kakvih definiranih granica (rubova)**. CSS nudi potpunu kontrolu nad umecima, definirana su svojstva za postavljanje umetaka za svaku stranu elementa (gore, desno, dolje i lijevo: **padding-top**, **padding-right**, **padding-bottom** i **padding-left**).
- ta svojstva mogu imati slijedeće vrijednosti: **length** (određivanje margina u px, pt, cm i dr. jedinicama), **%** (određivanje umetaka u postotku širine sadržanog elementa) te **inherit** (određivanje da vrijednost za umetak treba biti naslijeđena od roditeljskog elementa). Negativne vrijednosti nisu dozvoljene. Primjer različitih umetaka za sve četiri stranice elementa <div>:

```
div {
padding-top: 50px;
padding-right: 30px;
padding-bottom: 50px;
padding-left: 80px;
}
```
- Kako bi se skratio kod, moguće je navesti sva četiri individualna svojstva umetaka u jedno svojstvo, svojstvo **padding**, pomoću:

```
p { padding: 25px 50px 75px 100px; }
```
- Ako svojstvo **padding** ima manje od četiri vrijednosti, tumači se po grupama, kao i kod svojstva **border-style** i **margin**, npr.:

```
div { padding: 25px 50px; }
```

 govori da će gornji i donji umetak biti 25px, a lijevi i desni 50.
- CSS svojstvo **width** određuje **širinu područja sadržaja elementa** (dijela unutar umetaka, rubova i margina elementa – **box model** ili **model okvira**). Dakle, ako element ima određenu širinu, umetak koji je dodan tom elementu bit će dodan ukupnoj širini elementa što često dovodi do neželjenog rezultata, npr.:

```
div { width: 300px; padding: 25px; }
```

 ovdje je stvarna širina elementa <div> 350px.
- Kako bi se zadržala stvarna širina elementa bez obzira na širinu umetaka, može se koristiti svojstvo **box-sizing**. U tom slučaju se kod povećanja *paddinga*, smanjuje prostor raspoloživ za sadržaj.

- **Riješite tri vježbe pri dnu [stranice](#).**

CSS postavljanje visine i širine (height and width)

- CSS svojstva **height** i **width** koriste se za **postavljanje visine i širine elementa**. Ta svojstva **ne uključuju umetak, rubove ni margine**. Ona postavljaju **visinu, odnosno širinu područja unutar umetaka, granica i margina nekog elementa**.
 - Svojstva **height** i **width** mogu imati slijedeće vrijednosti: **auto** (podrazumijevano, preglednik izračunava visinu i širinu), **length** (određivanje visine/širine u px, pt, cm i dr. jedinicama), **%** (određivanje visine/širine u postotku širine sadržanog bloka), **initial** (postavlja visinu/širinu na njihove podrazumijevane vrijednosti) te **inherit** (određivanje da će visina/širina biti naslijeđena od vrijednosti roditelja).
 - Svojstvo **max-width** koristi se za postavljanje maksimalne širine elementa. Svojstvo može biti specificirano kao **length** vrijednost (u px, pt, cm i dr. jedinicama) ili u postotku (**%**, postotku širine sadržanog bloka), te kao **none** (postavljeno u podrazumijevanu vrijednost što znači da nema maksimalne širine.).
 - **U slučaju da je prozor preglednika manji od širine elementa (npr. 500px) pojavit će se problem sa <div> elementom, pa će preglednik na stranicu dodati vodoravnu kliznu traku. U takvoj će situaciji korištenje max-width poboljšati rukovanje preglednika sa malim prozorima.**
- **Riješite dvije vježbe pri dnu [stranice](#).**

PONAVLJANJE I PRAKTIČNA VJEŽBA:

1. U online [editoru](#) provježbajte naučeno i bar jedan primjer prezentirajte u svojoj mapi do kraja vježbi (dovoljna je preslika ekrana).
2. Odaberite najmanje pet (5) citata, uredno ih i skladno uredite CSS-om prema mogućnostima koje ste naučili. Citati trebaju biti različiti, oblikovani na različite načine i smješteni na različite dijelove ekrana. Trebaju zauzimati različite širine, biti pisani različitim fontovima, stilovima pisanja, veličinama teksta, na različitim bojama podloga i različitim bojama teksta. Potrebno je za pozadinu koristiti boje, uzorke i slike, te varirati stil definiranja okvira za svaki pojedini citat. Koristite i prozirnost. Kompletan CSS za oblikovanje citata treba biti sadržan u vanjskoj CSS datoteci, te tako omogućiti dodavanje novih citata primjenom postojećih oblikovanja. Provjerite i kako bi vam stranica izgledala na drugim veličinama ekrana uređaja. Pripazite na usklađenost korištenih boja.

Napomena: ovu praktičnu vježbu izvodi svaki učenik samostalno.