

Nastavni predmet	SKRIPTNI JEZICI I WEB PROGRAMIRANJE
Naslov cjeline	CSS – uređivanje web sjedišta
Naslov jedinice	Vježba 19: CSS izgled stranice (max-width, overflow, float, inline-block, align)

Maja Markovac i Sven Grgić

CILJ VJEŽBE

Učenik će demonstrirati i prikazati načine za poravnanje elemenata na stranici, odabrati i primijeniti način za poravnanje elemenata, odnosno njihovog sadržaja, prilagoditi vrstu poravnanja elementa, upotrebljavati svojstva za prikaz i poravnanje elemenata.

PRIPREMA ZA VJEŽBU

Prisjetite se i odgovorite pisano u bilježnicu:

- ▶ 1. Koja je osnovna razlika između inline i block-level elemenata?
- ▶ HTML se sastoji od različitih elemenata koji djeluju kao građevni blokovi web stranica. Svaki od tih elemenata spada u jednu od dvije kategorije - bilo elementa na razini bloka ili elementima koji se nalaze u nizu. Razumijevanje razlike između ove dvije vrste elemenata važan je korak u izgradnji web stranica. Pa što je element na razini blokova? Element na razini bloka je HTML element koji započinje novu liniju na web stranici i proteže cijelu širinu dostupnog horizontalnog prostora svog roditeljskog elementa. Ona stvara velike blokove sadržaja poput odlomaka ili odjeljaka stranice. U stvari, većina HTML elemenata su elementi na razini bloka.
- ▶ 2. Što je padding, a što margina?
- ▶ CSS svojstva padding-a koriste se za generiranje prostora oko sadržaja elementa, unutar definiranih obruba. CSS-om imate potpunu kontrolu nad ispunom. Postoje svojstva za postavljanje ispune za svaku stranu elementa (gore, desno, dolje i lijevo).
- ▶ U tipografiji, margina je područje između glavnog sadržaja stranice i rubova stranice. Margina pomaže definirati gdje red teksta počinje i gdje završava. Kada je stranica poravnata, tekst se raširi tako da bude u ravnini s lijevom i desnom marginom.

IZVOĐENJE VJEŽBE

CSS max-width

- block-level element uvijek zauzima punu raspoloživu širinu. Postavljanje širine block-level elementa spriječiti će da se on raširi do rubova svog spremnika. Nakon toga, margine se mogu postaviti na auto kako bi se vodoravno centriralo element unutar njegovog spremnika. Pri tom će element zauzeti specificiranu širinu, a preostali će prostor biti podijeljen jednako između dviju margina.
- U slučaju da je prozor preglednika manji od širine elementa, može se pojaviti problem sa elementom <div> u smislu da će preglednik dodati vodoravnu kliznu traku na stranicu.
- U takvoj situaciji upotreba max-width može poboljšati rukovanje preglednika malim prozorima, što je važno kod izrade sjedišta koja trebaju biti upotrebljiva i na manjim uređajima. Pogledajte i provjerite kako funkcionira primjer na [poveznici](#).

CSS overflow (preljev)

- CSS svojstvo overflow kontrolira što se događa sa sadržajem koji ne stane u neko područje. Ovo svojstvo određuje hoće li se odsjeci sadržaj ili će se dodati klizne trake u takvom slučaju. Funkcionira samo za blok elemente specificirane visine i ima slijedeće vrijednosti: visible – podrazumijevano, sadržaj koji se prelijeva ne odsjeca se i prikazuje se izvan okvira elementa, hidden – preljev se odsjeca, ostatak sadržaja će biti skriven, scroll – preljev se odsjeca i dodaje se klizna traka kako bi se video ostatak sadržaja, auto – slično kao scroll, ali dodaje kliznu traku samo ako je potrebno. Primjere pogledajte na [poveznici](#).
- Svojstva overflow-x i overflow-y određuju hoće li se promijeniti preljev sadržaja samo vodoravno, okomito ili oboje: overflow-x određuje što će biti sa lijevom/desnom rubom sadržaja, overflow-y određuje što će biti sa gornjim/donjim rubom sadržaja.
- Pogledajte i proučite primjere, te riješite tri vježbe zadane na stranici ([vježbe](#)).

CSS svojstva float i clear

- CSS svojstvo float određuje kako bi element trebao “plutati”. CSS svojstvo clear određuje koji elementi mogu plutati osim obrisanog elementa i na kojoj strani.
- Svojstvo float koristi se za pozicioniranje i oblikovanje sadržaja (npr. može omogućiti da slika pluta lijevo u odnosu na tekst u spremniku). To svojstvo može imati vrijednosti: left – element pluta na lijevom dijelu svojeg spremnika, right – element pluta na desnoj strani svojeg spremnika, none - element ne pluta (ispisuje se točno tamo gdje se u tekstu pojavljuje - podrazumijevano), inherit – element nasljeđuje vrijednost smještaja od roditeljskog elementa. Svojstvo float može se koristiti kako bi omotalo (wrap) tekst oko slika.
- U normalnoj situaciji div elementi biti će ispisani jedan iznad drugog, a ako koristimo float možemo dozvoliti elementima da plutaju jedan pored drugog.
- Pogledajte kako funkcioniraju primjeri na [poveznici](#).
- Svojstvo clear određuje koji elementi mogu plutati uz obrisani element te na kojoj strani. Ovo svojstvo može imati vrijednosti: none – omogućuje plutajuće elemente na obje strane (podrazumijevano), left – plutajući elementi nisu dozvoljeni na lijevoj strani, right – plutajući elementi nisu dozvoljeni na desnoj strani, both – plutajući elementi nisu dozvoljeni ni lijevo ni desno, inherit – vrijednost za clear element nasljeđuje od svojeg roditelja. Najčešći način za korištenje svojstva clear je nakon što se primijeni svojstvo float na neki element.

- Prilikom brisanja svojstva float, trebao bi se clear podudarati sa float: ako je element plutajući lijevo, treba primijeniti clear na lijevoj strani. Plutajući element će nastaviti plutati, ali obrisani element pojaviti će se ispod njega na mrežnoj stranici.
- Ako je element viši od elementa koji ga sadrži te ako je plutajući, on će se prelijevati izvan spremnika, kako bi prevladali taj problem možemo dodati overflow: auto; sadržanom elementu:

```
.clearfix {
  overflow: auto;
}
```

- overflow: auto clearfix radi dobro u slučajevima u kojima je moguće kontrolirati margine i odmak (padding), inače se mogu vidjeti klizne trake. Novi i moderniji clearfix hack je sigurniji za upotrebu, pa je primijenjen na većini mrežnih stranica:

```
.clearfix::after {
  content: "";
  clear: both;
  display: table;
}
```

- Primjere pogledajte na [stranici](#).
- Na [stranici](#) pogledajte primjere za float:
 - Mrežu okvira sa sadržajima jednake širine
 - Niz slika smještenih jedna uz drugu
 - Okviri jednake visine
 - Primjer za flexbox
 - Navigacijski izbornik
 - Kompletan izgled stranice

CSS inline-block

- Usporedi li se sa display: inline, glavna razlika je da display: inline-block omogućava postavljanje širine i visine elementa. Osim toga, sa display: inline-block, poštuju se gornje i donje margine/padding što nije slučaj sa display: inline.
- Usporedi li se sa display: block, glavna razlika je da display: inline-block ne dodaje novu liniju nakon elementa, tako da se element može smjestiti uz druge elemente.
- display: inline-block se uobičajeno koristi za ispis liste vodoravno umjesto okomito (npr. kod stvaranja navigacijskih poveznica). Pogledajte primjere na [poveznici](#).

CSS align (vodoravno i okomito poravnanje)

- Kako bi vodoravno centralizirali block element (npr. <div>), koristi se margin: auto; Postavljanje širine elementa spriječit će njegovo rastezanje izvan rubova njegovog spremnika. Element će tada zauzeti specificiranu širinu, a preostali će prostor biti podijeljen jednako između dviju

marginu (centrirano poravnanje nema učinka ako svojstvo width nije postavljeno ili je postavljeno na 100%):

```
.center {
margin: auto;
width: 60%;
border: 3px solid #73AD21;
padding: 10px;
}
```

- Kako bi samo centrirali tekst unutar elementa, koristimo text-align: center;
- Kako bismo centrirali sliku, potrebno je postaviti lijevu i desnu marginu na auto i napraviti block element:

```
img { display:
block; margin-left:
auto; margin-right:
auto; width: 40%;
}
```

- Jedna mogućnost za poravnanje elemenata je korištenje: position: absolute:

```
.right {
position: absolute;
right: 0px; width:
300px;
border: 3px solid #73AD21;
padding: 10px;
}
```

- Apsolutno pozicionirani elementi pomaknuti su iz normalnog tijeka i mogu prekrivati druge elemente.
- Mogućnost za poravnanje elemenata je i korištenje svojstva float (pri tom treba na preporučeni način kontrolirati element koji je viši od elementa koji ga sadrži):

```
.right {
float: right;
width: 300px;
border: 3px solid #73AD21;
padding: 10px; }
```

- Okomito centriranje može se u CSS-u riješiti na više načina. Jednostavno rješenje je koristiti top i bottom padding (npr. padding: 50px 0;). Kako bi se centriralo i okomito i vodoravno, može se upotrijebiti padding i text-align: center;
- Druga je mogućnost koristiti svojstvo line-height sa vrijednošću jednakoj svojstvu height (pogledajte primjer na [stranici](#)).
- Okomito centriranje može se izvesti pomoću position i transform. Ako padding i line-height nisu opcije, rješenje je koristiti pozicioniranje i svojstvo transform. Centriranje se može izvesti i pomoću flexbox (pogledajte primjere na [stranici](#)).
- Riješite dvije vježbe pri dnu [poveznice](#).

CSS Display/Visibility	✓
CSS Positioning	✓
CSS Z-index	✓
CSS Overflow	✓
CSS Align	✓

PONAVLJANJE I PRAKTIČNA VJEŽBA:

1. U online [editoru](#) ili editoru koji uobičajeno koristite provježbajte naučeno.
2. Napravite mrežnu stranicu na kojoj ćete nakon kratkog uvodnog teksta o mrežnim kablovima navesti vrste parica (npr. ftp, stp i drugi). Tekst treba biti napisan tako da o svakoj vrsti ima napisan odlomak (ili dva) teksta, a uz svaki odlomak je slika. Slike trebaju biti pozicionirane naizmjenično lijevo-desno. Dakle, izgled stranice treba biti otprilike kao prikaz u nastavku (prije opisa pojedinih vrsta izvedbi potrebno je napraviti izbornik koji će omogućiti pomak stranice na odabranu vrstu – treba ih biti najmanje četiri). Svi odlomci teksta trebaju imati isto oblikovanje:

Parice za mreže				
Uvodni tekst...				
utp	ftp	stp
☺	Opis prve vrste			
Opis druge vrste				☺
☺	Opis treće vrste.....			

Napomena: ove praktične vježbe kad je to moguće izvodi svaki učenik samostalno.