[**http://www.pmf.ni.ac.rs/pmf/konferencije/ekofizika/webukratko/identifikovanje.htm**](http://www.pmf.ni.ac.rs/pmf/konferencije/ekofizika/webukratko/identifikovanje.htm)

[**http://pilprozor.wordpress.com/2011/06/01/kola-web-programiranja-css-1-nastavak/**](http://pilprozor.wordpress.com/2011/06/01/kola-web-programiranja-css-1-nastavak/)

**HTML**, skraćenica od **HyperText Markup Language**, je standarizovani [jezik za označavanje](http://bs.wikipedia.org/w/index.php?title=Jezik_za_ozna%C4%8Davanje&action=edit&redlink=1) koji se koristi pri strukturiranju tekstova i multimedijalnih objekata u [web stranicama](http://bs.wikipedia.org/wiki/Web_stranica) i [elektronskoj pošti](http://bs.wikipedia.org/wiki/Email). Kao modifikovanu i pojednostavljenu verziju [SGML](http://bs.wikipedia.org/w/index.php?title=SGML&action=edit&redlink=1) jezika, HTML standarizuje i održava [World Wide Web Consortium (W3C)](http://bs.wikipedia.org/wiki/World_Wide_Web_Consortium).

Ova web stranica, kao i mnoge druge, su potpuno ili djelimično označene HTML jezikom. Iako se HTML znakovi (*tags*) često nazivaju kodom, tehnički HTML nije kod jer računarski kod su instrukcije koje od računara traže da izvrši određenu operaciju. Svrha HTML znakova je da se se struktura dokumenta "označi" tako da bi korisnikov agent (*user agent*), tj. [Internet pretraživač](http://bs.wikipedia.org/wiki/Internet_pretra%C5%BEiva%C4%8D) mogao da prepozna strukturu dokumenta i ispravno je prikaže u prozoru Internet pretraživača kojeg korisnik koristi.

Za dodatno modifikovanje web stranica, do sada su izašli neki dodaci:

* [CSS](http://bs.wikipedia.org/wiki/CSS) ili Cascading Style Sheets za izgled i položaj prezentacije
* Skriptni jezici ([Javascript](http://bs.wikipedia.org/w/index.php?title=Javascript&action=edit&redlink=1), [VBScript](http://bs.wikipedia.org/w/index.php?title=VBScript&action=edit&redlink=1)) za omogućavanje dinamičnosti i interaktivnosti na web prezentacijama,
* DOM ili [Document Object Model](http://bs.wikipedia.org/w/index.php?title=Document_Object_Model&action=edit&redlink=1), koji označava vezu između skripte i elementa na stranici

Gore navedeni dodaci, u paketu sa HTML jezikom, ponekad se nazivaju [DHTML](http://bs.wikipedia.org/w/index.php?title=DHTML&action=edit&redlink=1) ili Dinamični HTML.

Pored HTML-a, na [Internetu](http://bs.wikipedia.org/wiki/Internet) danas postoje jezici koji se koriste da korisnikovom pretraživaču dostave HTML koji će zavisiti od toga šta je korisnik tražio. Kod se u tom slučaju izvršava na[serverima](http://bs.wikipedia.org/wiki/Server) i HTML koji je rezultat procesa biva dostavljen korisniku. Neki od ovih jezika su: [ASP](http://bs.wikipedia.org/wiki/ASP) (*Active Server Pages*) i [PHP](http://bs.wikipedia.org/wiki/PHP) (*Hypertext Preprocessor*).

**Osnove HTML-a**

*Autor: prof. Milan Popovic, milan.popovic@bbs.edu.yu*

***Sadržaj:***

* [Uvod u WWW](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Uvod u WWW (World Wide Web))
* [Šta je HTML?](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Sta je HTML)
* [Šta je URL ?](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Sta je URL)
* [Osnovne HTML komande](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Osnovne HTML komande)
  + [Osnovni koraci: korišćenje tagova](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Osnovni koraci)
    - [HTML tag](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Tag HTML)
    - [head tag](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Tag head)
    - [body tag](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Body tag)
    - [Title](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Title tag)
    - [Headers](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#h tagovi)
    - [Paragraphs](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Paragrafi)
    - [Preformatted text](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Pre'formatizovani tekst)
    - [Boldface and italics](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Bold i italik)
    - [Lists](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Liste)
    - [Blockquote](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Blok navod)
    - [Center](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Centriranje)
    - [Horizontal rule (line)](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Horizontalna linija)
    - [Addresses](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Adresa)
    - [Comments](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Komentari)
    - [Strike-through](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Precrtavanje teksta)
* [HTML komande za linkovanje](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#HTML komande za linkovanje)
  + [Linkovanje na druge dokumente](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Linkovanje na druge dokumente)
  + [Linkovanje na imenovane veze](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Linkovanje na imenovanu vezu)
  + [Apsolutno i relativno linkovanje](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#Apsolutno i relativno linkovanje)
* [HTML komande za slike ( images)](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#HTML komande za linkovanje)
* [HTML tabele](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm#HTML Tabele)
* [HTML forme](http://www.osnove-programiranja.com/prirucnici/osnove_html.htm# HTML Forme)

**Uvod u WWW (World Wide Web)**

World  Wide Web, ili skraćeno Web je danas jedan od najpopularnijih mehanizama za desiminaciju informacija među ljudima. Pomoću web-a možemo prezentirati informacije koje se sastoje od reči, slika, video i audio zapisa - i sve to bez dodatnog softvera koji korisnik mora da instalira na svom računaru  - dovoljno je da ima bilo koji web pretraživač (Internet Explorer, Firefox, Chrome, Opera i sl.)

Pored toga što putem Web-a ljudi mogu da primaju informacije, oni mogu i da kreiraju sopstvene web stranice i da na taj način oni postaju izvor informacija za druge. Kreiranje Web stranica je veoma jednostavno ako savladate nekoliko osnovnih HTML komandi. U ovom kratkom tutorijalu biće brikazane osnovne HTML komande.

Za kreiranje web stranica mogu se koristiti razni HTML editori kao što su Microsoft Visual Web Developer i FrontPage, Adobe Cold Fusion i Dreamweaver i sl. U ovom tutorijalu ćemo se baviti HTML komandama i pisanjem HTML koda bez upotrebe specijalizovanih editora. Za programiranje HTML stranica biće vam potreban samo neki prost editor teksta (NotePad ili WordPad, na primer)

**Šta je HTML ?**

HTML je slraćenica od Hyper Text Markup Language, što se na srpski može prevesti kao Jezik za označavanje hiper tekstova. Hiper tekstovi su tekstovi koji pored reči sadrže i sliku, video i audio zapise. HTML je podskup jednog šireg jezika, SGML-a (Standard General Markup Language) i koristi se za definisanje izgleda  World Wide Web dokumenata (stranica) kao i za uspostavljanje veza (linkova) među dokumentima (podrazumeva se da dokument sadrži tekst, sliku, zvuk, grafiku).

**Šta je URL (Uniform Resource Locator) ?**

Kada neko želi da vam da uputstvo kako da pristupite nekoj informaciji na internetu on vam tada saopštava elektronsku adresu na kojoj se ta informacija nalazi. Ta elektronska adresa je zapravo URL. Svaki dokument na internetu ima jedinstvenu adresu (URL) što nam omogućava da pomoću nje lako dodjemo do web stranice koja nas interesuje u moru drugih web stranica.

Kada znamo URL adresu, onda u našem pretraživaču unesemo tu adresu i posle kratkog vremena sadržaj web stranice će se pojaviti na našem ekranu.

Na primer, da pristupite web stranici koja sadrži ovaj tutorijal ptorebno je da znate sledeći URL:

<http://osnove-programiranja.com/prirucnici/osnove_html.htm>

**Osnovne HTML komande**

U ovom delu biće opisane osnovne HTML komande i koraci koje treba preduzeti kako bi se neki dokument pripremio za prezentaciju na Web-u.

**Osnovni koraci: upotreba tagova**

HTML koristi tagove da pomoću njih ukaže pretraživaču  kako neki tekst ili slika treba da budu prikazani na ekranu. Tagovi se stavljaju unutar  <> zagrada. U većini slučajeva, tagovi se postavljaju na početak nekog dela dokumenta, a na kraju tog dela dokumenta postavlja se tag završetka.

Na primer da napravite naslov dokumenta koristite sledeći opis:

<title>Moj prvi HTML Dokument</title>

Uočite ova dva taga, <title> i </title> između kojih se nalazi tekst naslova dokumenta. Vidite da se tag završetka razlikuje od taga početka po kosoj crti (/).

HTML tagovi nisu osetljivi na velika i mala slova ( URL jeste osetljiv). U većini slučajeva (osim kod formatiranog teksta) HTML ignoriše blanko znake i prazne linije. Međutim, pogodno je da kada pišete HTML kod stavljate prazne linije zbog lakše čitljivosti kada budete želeli da ažurirate HTML kod.

**Tag HTML**

<html>  tag označava početak HTML programa (koda).

<html> tag se obično postavlja na početak dokumenta, kao prva naredba za pretraživače. Na kraju dokumenta postavlja se tag završetka </html> . Između ova dva taga nalaziće se svi ostali tagovi kao i sam sadržaj dokumenta.

**Tag head**

<head> tag je tag koji služi za početak jednog posebnog dela dokumenta koje se naziva zaglavlje dokumenta. Zaglavlje se neće prikazivati na ekranu.Ono služi da se unutar njega definišu neke karakteristike dokumenta, kao to je naslov.

Zaglavlje se opisuje između tagova <head> i </head>.

Sada možemo dati primer dokumenta koji sadrži zaglavlje u kojem je definisan naziv dokumenta tagovima <title> i </title>:

<html>

<head>

<title>Moj prvi HTML dokument</title>

</head>

**Title tag**

title tag vam omogućava da definišete naziv vašeg dokumenta. To se postiže na sledeći način:

<title>Moj prvi HTML dokument</title>  
  
Naziv dokumenta, pošto se nalazi u zaglavlju, neće biti prikazan na ekranu, ali će se pojaviti u na raznim drugim mestima (na vrhu prozora, u bukmark-u stranica i sl.)

**Body tag**

<body> i </body>  tagovi definišu početak i kraj sadržaja vašeg dokumenta. Sav tekst, sve slike i linkovi na druge dokumente nalaziće se unutar ovih tagova.

<body> se nalazi odmah nakon zaglavlja (head tagova). Tipična web stranica može izgledati ovako:

<html>

<head>

<title>My First HTML Document</title>

</head>

<body>

Ovde se nalazi sadržaj vašeg dokumenta: tekst, slike, grafika itd.

</body>

</html>

**h (Header) tagovi**

h tag se koristi za iybor veličine slova koja će se koristiti za naslove i podnaslove u vašem dokumentu. Postoji šest nivoa veličine od h1 do h6. Najveća slova se dobijaju tagom h1  a najmanja tagom h6. Slede primeri ya svih 6 nivoa.

<h1>Ovoje naslog sa h1 tagom</h1>

**Ovo je naslov sa h1 tagom**

<h2>Ovoje naslog sa h2 tagom</h2>

**Ovoje naslog sa h2 tagom**

<h3>Ovoje naslog sa h3 tagom</h3>

**Ovoje naslog sa h3 tagom**

<h4>Ovoje naslog sa h4 tagom</h4>

**Ovoje naslog sa h4 tagom**

<h5>Ovoje naslog sa h5 tagom</h5>

**Ovoje naslog sa h5 tagom**

<h6>Ovoje naslog sa h6 tagom</h6>

**Ovoje naslog sa h6 tagom**

Tagovima za naslove se ne postiže samo promena veličine slova, već su naslovi i boldovani, a takođe ispred i iza naslova se dodaje po jedna prazna linija.

**Paragrafi**

U HTML, paragraf se označava tagom tag <p> i pistavlja se na kraj svakog paragrafa "normalnog" teksta (pod normalnim se podrazumeva tekst koji nema upotrebljene neke druge tagove).

<p> ovaj tag prouzrokuje postavljanje teksta u novu liiju uz dodavanje jedne prazne linije ispred

<br> ovaj tag je sličan prethodnom sa raylikom što se ne dodaje jedna prazna linija ispred

**Pre-formatizovani tekst**

Tag <pre> se koristi da označi da će vaš tekst zadržati formu u ubliku kako je kreiran sa svim blanko znacima, novim linijama, tabovima i sl. Drugim rečima biće onakav kakvog ste ga uneli.

Na primer:

<pre>ovo je

jedan primer

pre-formatizovanog

teksta</pre>

On će se na ekranu pojaviti u sledećem obliku::

ovo je

jedan primer

pre-formatizovanog

teksta

**Bold i italik tagovi**

Možete posebnim tagovima označiti da vaš tekst bude u bold i/ili italic obliku.

Postoji i tag za podvlačenje teksta, ali se obično ne koristi jer se linkovi takođe prikazuju kao podvučeni (o tome nešto kasnije).

Ne možete koristiti istovremeno oba taga (za bold i za italic). Ako greskom to pokušate onaj poslednji unet će biti primenjen.

Bold i itali se javljaju u dva oblika: fizičkom i logičkom i možete koristiti oba prema vašoj želji. Upotreba ovih tagova vidi se u sledećim primerima:

**Fizički tagovi**

Ovo je <b>boldovano</b>

Ovo je **boldovano**

Ovo je <i>italic</i>  
Ovo je *italic*

**Logički tagovi**

Ovo je <strong>jako istaknuto</strong>

Ovo je **jako istaknuto**

Ovo je <em>istaknuto</em>

Ovo je *istaknuto*

Postoji mala razlika između fizičkog i logičkog taga. Kod fizičkog se menja samo oblik slova, a logički je zavisan od pretraživača i može uneti i druge promene (  recimo promenu boje slova ).

**Liste**

U HTML postoji jednostavan način za pravljenje lista sa nabrajanjem, bez nabrajanja ili za definisanje. Čak štaviše moguće je staljati listu unutar liste i na taj način postići hijerarhijsko nabrajanje.

Kada koristite liste prazna polja između "bulita" ili brojeva se automatski dodaju i mogu biti različiti od pretraživača do pretraživača.

**Liste bez rednih brojeva**

Liste bez rednog broja počinju tagom <ul> iza koga slede linije teksta koje se označavaju tagom <li>. Lista se završava tagom </ul>.

Evo jedne liste bez rednih brojeva:

<ul>

<li> linija 1

<li> linija 2

<li> linija 3

</ul>

A evo kako će to izgledati na ekranu:

* linija 1
* linija 2
* linija 3

**Liste sa rednim brojevima**

Evo kako napravili gornju listu, ali sada sa rednim brojem ispred svake linije:

<ol>

<li> linija 1

<li> linija 2

<li> linija 3

</ol>

A evo kako će to izgledati na ekranu:

1. linija 1
2. linija 2
3. linija 3

**Definicione liste**

Definicione lsite vam omogućavaju da pravite strukturirane liste bez "bulita" i rednih brojeva. Evo primera:

<dl>

<dt> Ovo je jedan pojam

<dd> Ovo je njegova definicija

<dd> ovo je druga definicija

<dt> Ovo je drugi pojam

<dd> Ovo je njegova definicija

</dl>

Ova lista će izgledati ovako na ekranu:

Ovo je jedan pojam

Ovo je njegova definicija

Ovo je druga definicija.

Ovo je drugi pojam

Ovo je njegova definicija

**Ugnježdene liste (lista u listi)**

Na kraju, evo jednog primera koji pokazuje složenu list, gde se unutar jedne liste nalaze druge liste.

<ul>

<li> podatak 1

<ul>

<li> ugnježden podatak 1

<li> ugnježden podatak 2

</ul>

<li> podatak 2

<ul>

<li> ugnježden podatak 1

<li> ugnježden podatak 2

</ul>

<li> podatak 3

<ul>

<li> ugnježden podatak 1

<li> ugnježden podatak 2

</ul>

</ul>

A evo kako će to izgledati na ekranu::

* podatak 1
  + ugnježden podatak 1
  + ugnježden podatak 2
* podatak 2
  + ugnježden podatak 1
  + ugnježden podatak 2
* podatak 3
  + ugnježden podatak 1
  + ugnježden podatak 2

**Blok navod**

Blok navod (blockquote) tag se koristi da se neki deo teksta izdvoji i prikaže kao navod koji će u odnosu na ostatak teksta biti i sa leve i sa desne strane dopunjen blanko znacima. To se postiže sledećim tagovima:

<blockquote>Tekst koji se želi prikazati kao navod</blockquote>

Da ponovo istaknemo, blok navodu će biti dodata leva i desna margina kako bi se razlikovao od ostatka teksta u dokumentu.

**Centriranje**

Vi možete centrirati (postaviti u centar ekrana) tekst, sliku, naslove pomoću tagova:

<center>Ova rečenica je centrirana</center>

što će na ekranu izgledati ovako:

Ova rečenica je centrirana

<center> tag automatski dodaje novu liniju na kraju teksta koji je centriran.

**Horizontalna linija**

Ako želite da razdojite delove vašeg dokumenta, to možete postići ubacivanjem horizontalne linije pomoću taga  <hr>.

Horizontalna linija će biti prikazana ovako:

**Adresa**

Tag <address> se obično postavlja na kraj dokumenta i najčešće koristi za davanje kontakt informacija autora dokumenta. Sve što se stavi unutar ovog taga biće prikazano u italic obliku.

Evo primera korišćenja address taga:

<address>

Osnove HTML-a / Milan Popovic / milan.popovic@bbs.edu.yu

</address>

A na ekranu će to izgledati ovako:

*Osnove HTML-a / Milan Popovic / milan.popovic@bbs.edu.yu*

**Komentari**

Moguće je u HTML dokumente uneti komentare koji se neće prikazivati na ekranu pretraživača. To je najbolji način da se postave upozorenja i dodatne instrukcije za svako buduće ažuriranje vašeg dokumenta.

Komentari imaju sledeći oblik:

<!-----Ovo je komentar koji se neće pojaviti u pretraživaču----->

Komentar se može nalaziti i u više linija kao u sledećem primeru:

<!----Ovaj komentar se takođe neće videti u pretraživačima

iako je postavlje u dve linije--->

**Precrtavanje teksta**

Ako vam bude potrebno možete koristiti <strike> tag da deo teksta prcrtate kao što je urađeno u sledećem primeru:

<strike>Ovaj tekst će biti precrtan</strike>

Na ekranu pretraživača to će izgledati ovako:

~~Ovaj tekst će biti precrtan~~

**HTML komande za linkovanje**

Prava moć HTML dokumenata  nalazi se u njihovoj sposobnosti da  se povezuju sa drugim dokumentima, delovima teksta, slikama, video i audio zapisima. Veze (linkovi) su obično podvučeni ili naglašeni u dokumentima da se istakne njihova pojava. Kada korisnik klikne na neki link automatski se otvara novi prozor sa prikazom dokumenta na koji link ukazuje.

**Linkovanje na druge dokumente**

Linkovanje na drugi dokument (drugi URL) postiže se na sledeći način:

<a href="http://www.osnove-programiranja.com/">Osnove programiranja</a>

Tekst koji se nalazi između <a> i </a> tagova ("Osnove programiranjal") je tekst na koji korsinik treba da klikne kako se otvorila nova URL adresa (<http://www.osnove-programiranja.com/> ).

Evo šta je značenje pojedinih elemenata pri linkovanju:

<a -- označava početak linka

href -- označava "hypertext reference"

<http://www.osnove-programiranja.com/> -- ovo je URL dokumenta na koji se povezujemo

Osnove programiranja -- the text that will appear and be clickable

/a> -- kraj linka

Dakle, sve što vam je potrebno da vaš dokument povežete sa nekim drugim dokumentom jeste URL adresa tog drugog dokumenta.

**Linkovanje na imenovanu vezu**

**Povezivanje unutar istog dokumenta**

Da se linkujete na neki poseban deo vašeg dokumenta, potrebno je da najpre definišete mesto za linkovanje (anchor - sidro), a zatim da napravite link do tog mesta. Na primer, ako želite da se sa vrha vašeg dokumeta na prvu rečenicu iz četvrtog paragrafa vašeg dokumenta, vi ćete tom vađem četvrtom paragrafu dati neko ime, na primer "vazno" i kreirati link na to ime.

Evo kako bi to psotigli:

Najpre na početku četvrtog paragrafa definisemo vezu (anchor, sidro):

<a name="vazno">Prva rečenica iz četvrtog paragrafa</a>

A onda na vrh našeg dokumenta posatvimo link

<a href="#vazno">Link na vaznu napomenu</a>

Znak # koji se pojavljuje u linku zapravo pokazuje pretraživaču da treba da ostvari vezu unutar istog dokumenta, a ne da ide na drugu URL adresu (kao kod povezivanbja sa drugim dokumentima).

**Imenovana veza u drugom dokumentu**

Mogu nastupiti situacije u kojima vi želite da se povežete na neki drugi dokument, ali ne na njegov početak, već na neki specifičan deo tog dokumenta. Za primer, pretpostavimo pa vi iz vašeg dokumenta želite da se povežete sa paragrafom 2 iz jednog drugog dokumenta čija je URL adresa "drugiDokument.html".

Kao prvo, potrebno je da postavite vezu (anchor, sidro) u dokument na koji se povezujete (drugiDokument.html). Da to uradite, idete na drugi paragraf tog dokumenta i napravite vezu, recimo ovako:

<a name="drugiparagraf">some text</a>

Sada, da bi kreirali link na ovu vezu u vašem prvom dokumentu, idete na mesto u dokumentu sa kojeg želite da ostvarite link napravite link na sledeći način:

<a href="drugiDokument.html#drugiparagraf">Vazna napomena</a>

Korisnik će primetiti da je tekst "Vazna napomena" naglašen i da predstavlja link. Ako na njega klikne dobiće na ekranu drugi paragraf iz drugog dokumenta.

**Apsolutno i relativno linkovanje**

Ako u vašoj ulici sretnete suseda kojeg do tada niste znali i koji vas pita gde stanujet, vi mu verovatno nećete iydiktirati celu vašu adresu, već ćete mu recimo reći "druga kuća odavde". To se naziva relativnim adresiranjem, jer ste mu saopštili poziciju (adresu) vaše kuće u odnosu na mesto na kojem se nalazite. Za razliku od toga, vaša puna poštanska sdresa, je dovolja da vas nađe bilo ko, sa bilo koje pozicije. To je vaša apsolutna adresa.

Isti se koncept koristi i za adresiranje na internetu pomoću URL adresa. Ako se dokumenti koje povezujemo nalaze u istom folderu (direktorijumu) nije potrebno da saooštavano celu adresu. Na primer:

<a href="drugi\_dokument.html">Idi na sledeću stranu</a>

Ovo će vas linkovati na drugi\_dokument.html koji se nalazi u istom folderu kao i dokument koji ga poziva.

Možete se linkovati i na druge susedne foldere, kao u ovom primeru:

<a href="../treci\_dokument.html">Idi na treću stranu</a>

Ovo će vas linkovati na dokument koji se nalazi u folderu za jedan nivo višem od trenutnog foldera.

Preporučuje se upotreba relativnog linkovanja kad god je to moguće, jer to olakšava manipulaciju vašim stranicama.

**HTML komande za slike (images)**

Verovatno se zapazili da mnogi dokumeti na internetu sadrže slike. To je zbog toga što su slikama mogu preneti mnoge informacije. Kako kaže kineska poslovica "Jedna slika vredi 1000 reči". Najčešće se koriste slike u .gif i .jpeg formatu, ali su mogući idrugi formati.

Prva stvar kada želite da koristite slike jeste da ih sami kreirate ili locirate na internetu. Postoji na hiljade "public domain" (slobodnih za korišćenje) na internetu. Vi ih mozete preuzeti, konvertovati u .gif format (ako su u nekom drugom formatu). Da vas podsetimo, možete koristiti samo slike koje ste sami napravili, ili za čije korišćenje imate dozvolu.

**Stavljanje slike u dokument**

Slike koje se pojavljuju u vašim dokumentima moraju imati kompletnu URL adresu (ili adresu foldera gde se nalaze). Za prikaz slike koriste se sledeći tagovi:

<img src="path or URL of image file">  
  
Na primer, <img src="http://osnove-programiranja.com/img/NET048.jpg">  
  
Ovde "img src" označava izvor slike sa URL adresom gde se slika nalazi. U ovom slučaju slika NET048.jpg se nalazi u folderu img na web serveru osnove-programiranja.comi

Ako se slika nalazi u istom folderu gde se nalazi i sam dokument, tada nije potrebno zadavati celu URL adresu, već samo ime fajla koji sadrži sliku.

Na primer ako imate sliku "mojaslika.gif" u istom folderu sa dokumentom dovoljno je da napišete:

 <img src="mojaslika.gif">

**Pozicioniranje teksta i slike**

**Atributi za img tag**

Tekst može da "pliva" na desno ili nalevo od slike. Atribut **align=left** kazuje da će tekst "plivati" nalevo i nadole od slike, a sa atributom **align=right** tekst će "plivati" nadesno i nadole.

**Na primer:**

<img src="http://www.osnove-programiranja.com/img/NET048.jpg" align=right>Ova slika

se koristi kao ilustarcija. Pošto ovde iammo align=right tag, tekst će se nalaziti

sa strane umesto ispod slike.<br clear=right>

**Ostavljanje praznina oko slike**

Vi možete povećavati prazninu (belinu) oko slike pomoću horiznontalnih i vertikalnih margina koje zadajete u piksolvima.

Na primer,dodavanjem atributa **hspace=20** biće dodato20 piksela levoj i desnoj margini slike.Atribut **vspace=** dodaje margine  na vrh i na dno slike .

**Primer:**

<img src="http://www.osnove-programiranja.com/img/NET048.jpg" align=left hspace=20>Ova slika

se koristi kao ilustarcija. Pošto ovde iammo align=right tag, tekst će se nalaziti

sa strane umesto ispod slike.<br clear=right>

Postoje još neki atributi za slike, ali ovde se njima nećemo baviti.

**Alternatvni tekst**

Ako se slika ne može pojaviti (iz bilo kog razloga) moguće je umesto nje dati altrenativni tekst pomoću **alt** modifikatora, kao u sledećem primeru:

<img align=middle alt="Ilustracija" src="http://www.osnove-programiranja.com/img/NET048.jpg">

**Linkovanje sa slike**

Može se praviti link bilo sa teksta bilo sa slike. Da sliku učinimo linkom dovoljno je da tag slike postavimo unutar taga za linkovanje.

Na primer:

<a href="http://www.osnove-programiranja.com/">

<img src="http://www.osnove-programiranja.com/img/NET048.jpg"></a>

Kao što vidite, <img src> tag je unutar <a href></a> taga. Kada je neka slika link, ona će imati oko sebe jedan plavi okvir kojim se pokazuje da je slika link. Kada user klikne na sliku povezaće se na željeni dokument, isto kao kada je kliknuo na tekst koji je označavao neki link.

**HTML Tabele**

Tabele se kreiraju pomoću <table> taga. Tabele se dele u redove (rows) pomoću taga <tr>, a svaki red se sastoji od jedne ili više ćelija podataka (data cells) pomoću taga <td>. Ćelije tabele mogu da sadrže tekst, sliku, listu, paragraf, formu, horizontalnu liniji, pa čak i novu tabelu (tabela u tabeli).

Evo jednog primera tabele:

<table border="1">

<tr>

<td>red 1, ćelija 1</td>

<td>red 1, ćelija 2</td>

</tr>

<tr>

<td>red 2, ćelija 1</td>

<td>red 2, ćelija 2</td>

</tr>

</table>

 Ovako bi gornja tabela izgledala u pretraživaču:

|  |  |
| --- | --- |
| red 1, ćelija 1 | red 1, ćelija 2 |
| red 2, ćelija 1 | red 2, ćelija 2 |

**HTML Forme**

HTML forma je deo dokumenta koji u sebi može da sadrži elemente forme kao što su razna polja za unos potataka, razne vrste dugmića za pokretanje akcije i sl.

Forma se kreira tagom <form> a pojedinačni elementi forme tagovima <input>.

Evo jednog jednostavnog primera forme koja sadrži dva elementa:

<form>

<input>

<input>

</form>

U pretraživaču bi to izgledalo ovako:

Top of Form

 

Bottom of Form

Tag <form> i tagovi <input> mogu imati niz atributa koji ih pobliže definišu.

Posmatrajmo slededeći primer:

<form>

Ime :

<input type="text" name="ime">

<br>

Prezime:

<input type="text" name="prezime">

</form>

U pretraživaču će to izgledati ovako:

Top of Form

Ime      :    
Prezime: 

Bottom of Form

Tag <input> se koristi i za definisanje dugmića (buttons) koji imaju različite oblike i upotrebu.

**Akciono dugme**

Akciono dugme se koristi za pokretanje neke akcije, recimo brisanje svih polja u formi, startovanje nekog javsript programa i sl.

Akciono dugme se kreira na način prikazan u sledećem primeru:

<form>  
<input type="button" name="dugme" value="Start" onClick="Akcija">  
</form>

Ovako bi to izgledalo u pretraživaču:

Top of Form

Bottom of Form

**Radio dugme**

Radio dugme se koristi kada želite da pravite izbor između malog broja opcija  kao u sledećem primeru.

<form>

<input type="radio" name="pol" value="muski"> Muski

<br>

<input type="radio" name="pol" value="yenski"> Yanski

</form>

Ovako to izgleda u pretraživaču::

Top of Form

 Muski  
 Zenski

Bottom of Form

Uočite da samo jedna od dve opcije može biti odabrana kada kliknete na dugme.

**Checkbox dugme**

Checkbox dugme se koristi kada želite da odaberete jednu ili više opcija iz nekog manjeg skupa kao što to pokazuje sledeći primer:

<form>

Šta sve imam?

Imam bicikl:

<input type="checkbox" name="vozilo" value="Bicikl">

<br>

Imam auto:

<input type="checkbox" name="vozilo" value="Auto">

<br>

Imam avion:

<input type="checkbox" name="vozilo" value="Avion">

</form>

A evo kako bi to izgledalo u pretraživaču:

Top of Form

Šta sve imam?  
Imam bicikl:   
Imam auto :   
Imam avion: 

Bottom of Form

Uočite da kod checkbox dumića možete da birate više od jedne opcije (za razliku od radio dugmića gde birate samo jednu opciju).

**Slanje sadržaja forme serveru**

Forme se najčešće koriste za slanje podataka ka serveru. Za slanje podatka se koristi dugme tipa "submit" a u tagu <form> se dodaje atribut "action" kojim se definiše koji će program na serverskoj strani preuzeti poslate podatke. Atributom "method" definiše se metod slanja podataka koji može biti "get" ili "post".

Evo primera:

<form name="input" action="http://osnove-programiranja.com/vezbe/login.php"

method="get">

Korisnik:

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

<input type="submit" value="Submit">

</form>

Ovako to izgleda u pretraživaču:

Top of Form

Korisnik:  

Bottom of Form

Ako sada u input polje ukucate niz slova i kliknete na dugme "Submit" pretraživač će poslati vaš podatak serveru na kojem će se aktivirati program login.asp koji će preuzeti poslati podatak i sprovesti odgovarajuću akciju.

CSS

[Datoteka:CSS.svg](http://sh.wikipedia.org/w/index.php?title=Posebno:Postavi_datoteku&wpDestFile=CSS.svg)

CSS

**CSS** ([eng.](http://sh.wikipedia.org/wiki/Engleski_jezik) *Cascading Style Sheets*) je jezik formatiranja pomoću kog se definiše izgled elemenata veb-stranice.

Prvobitno, [HTML](http://sh.wikipedia.org/wiki/HTML) je služio da definiše kompletan izgled, strukturu i sadržaj veb-stranice, ali je od verzije 4.0 HTML-a uveden CSS koji bi definisao konkretan izgled, dok je HTML ostao u funkciji definisanja strukture i sadržaja.

**Sadržaj/Садржај**

  [[sakrij/сакриј](http://sh.wikipedia.org/wiki/CSS)]

* [1 Istorija](http://sh.wikipedia.org/wiki/CSS#Istorija)
* [2 Jezik](http://sh.wikipedia.org/wiki/CSS#Jezik)
  + [2.1 Ciljni elementi](http://sh.wikipedia.org/wiki/CSS#Ciljni_elementi)
  + [2.2 Svojstva i vrijednosti](http://sh.wikipedia.org/wiki/CSS#Svojstva_i_vrijednosti)
* [3 Primjeri](http://sh.wikipedia.org/wiki/CSS#Primjeri)
* [4 Organizacija CSS koda](http://sh.wikipedia.org/wiki/CSS#Organizacija_CSS_koda)
* [5 Ograničenja](http://sh.wikipedia.org/wiki/CSS#Ograni.C4.8Denja)
* [6 Vidi još](http://sh.wikipedia.org/wiki/CSS#Vidi_jo.C5.A1)
* [7 Izvori](http://sh.wikipedia.org/wiki/CSS#Izvori)
* [8 Vanjske veze](http://sh.wikipedia.org/wiki/CSS#Vanjske_veze)

Istorija[[uredi - уреди](http://sh.wikipedia.org/w/index.php?title=CSS&action=edit&section=1)]

CSS je u određenoj formi postojao još u začecima [SGML](http://sh.wikipedia.org/w/index.php?title=SGML&action=edit&redlink=1)-a [1970ih](http://sh.wikipedia.org/wiki/1970-e) godina.

Kako je HTML postajao komplikovaniji, davao je sve više mogućnosti za definiciju izgleda elemenata, ali je istovremeno postajao nečitljiviji i teži za održavanje. Različiti [brauzeri](http://sh.wikipedia.org/w/index.php?title=Brauzer&action=edit&redlink=1) su prikazivali dokumente na različite načine, i postojala je potreba za dosljednom tehnikom definisanja prikaza elemenata na stranici.

Da bi se ovo postiglo, devet različitih metoda je predloženo na zvaničnom forumu [W3C](http://sh.wikipedia.org/w/index.php?title=W3C&action=edit&redlink=1)-a. Od devet, dvije metode su izabrane kao temelj onoga što je kasnije postalo CSS: CHSS ([eng.](http://sh.wikipedia.org/wiki/Engleski_jezik) *Cascading HTML Style Sheets*) i SSP ([eng.](http://sh.wikipedia.org/wiki/Engleski_jezik) *Stream-based Style Sheet Proposal*). Prvo je Hakon Vium Li (koji je sada šef tehničke službe kompanije [Opera](http://sh.wikipedia.org/w/index.php?title=Opera_(brauzer)&action=edit&redlink=1)) predložio CHSS u oktobru [1994](http://sh.wikipedia.org/wiki/1994), jezik koji je imao dosta sličnosti sa današnjim CSS-om. Bert Bos je radio na brauzeru po nazivu Argo, koji je imao sopstveni način definisanja stilova, SSP. Li i Bos su radili zajedno da bi osnovali CSS standard (slovo H je izbačeno iz skraćenice CHSS jer se CSS mogao odnositi i na druge jezike pored HTML-a).

Za razliku od postojećih jezika poput [DSSSL](http://sh.wikipedia.org/w/index.php?title=DSSSL&action=edit&redlink=1)-a i [FOSI](http://sh.wikipedia.org/w/index.php?title=FOSI&action=edit&redlink=1)-a, CSS je dozvoljavao da više opisa utiče na dokument, tj. jedna definicija stilova je mogla naslijediti osobine od druge.

Lijev prijedlog je postavljen na konferenciji "Veb mozaik" u [Čikagu](http://sh.wikipedia.org/wiki/Chicago) [1994](http://sh.wikipedia.org/wiki/1994). godine, i ponovo sa Bosovim prijedlogom [1995](http://sh.wikipedia.org/wiki/1995). Otprilike u ovo vrijeme je osnovan W3C, koji je preuzeo funkciju razvoja CSS-a. Do kraja[1996](http://sh.wikipedia.org/wiki/1996), CSS je bio spreman da se objavi kao standard, i CSS1 je objavljen u decembru.

Razvoj HTML-a, CSS-a i [DOM](http://sh.wikipedia.org/w/index.php?title=DOM&action=edit&redlink=1)-a se odvijao u jednoj istoj grupi, *HTML Editorial Review Board* (ERB). Početkom [1997](http://sh.wikipedia.org/wiki/1997). grupa ERB se podijelila na tri radne grupe: radna grupa za HTML, kojom je upravljao Den Konoli iz W3C-a, radna grupa za DOM, kojom je upravljao Loren Vud iz kompanije [Softkvod](http://sh.wikipedia.org/w/index.php?title=Softkvod&action=edit&redlink=1), i radna grupa za CSS, kojom je upravljao Kris Lili iz W3C-a.

Radna grupa za CSS je počela da radi na problemima koji nisu bili obuhvaćeni CSS-om verzije 1, koji se tako razvio u CSS2, [4. novembra](http://sh.wikipedia.org/wiki/4._11.) [1997](http://sh.wikipedia.org/wiki/1997); objavljen je kao zvanična verzija [12. maja](http://sh.wikipedia.org/wiki/12._5.) [1998](http://sh.wikipedia.org/wiki/1998). CSS3, čiji je razvoj započet [1998](http://sh.wikipedia.org/wiki/1998). se još uvijek razvija.

Jezik[[uredi - уреди](http://sh.wikipedia.org/w/index.php?title=CSS&action=edit&section=2)]

CSS sintaksa se sastoji od *opisa* izgleda elemenata u dokumentu. Opis može da definiše izgled više elemenata, i više opisa može da definiše jedan element. Na taj način se opisi *slažu* jedan preko drugog da bi definisali konačni izgled određenog elementa (otuda naziv *Cascading* ([eng.](http://sh.wikipedia.org/wiki/Engleski_jezik) *cascade* - crijep) da bi se dočaralo slaganje jednog stila preko drugog u definisanju konačnog izgleda elementa)

Svaki opis se sastoji od tri elementa:

* definicija ciljnih elemenata
* svojstva
* vrijednosti

Nakon što definišemo ciljne elemente, tj. elemente na koje će se trenutni opis odnositi, nizom parova svojstvo-vrijednost definišemo izgled svakog ciljnog elementa. Sintaksa koja se pri tome definiše je sljedećeg oblika:

ciljni elementi {

svojstvo: vrijednost;

svojstvo: vrijednost;

...

}

CSS podržava i komentare, koji izgledaju kao u programskom jeziku [C](http://sh.wikipedia.org/w/index.php?title=Programski_jezik_C&action=edit&redlink=1), dakle navodeći se između znakova /\* i \*/.

**Ciljni elementi**[[uredi - уреди](http://sh.wikipedia.org/w/index.php?title=CSS&action=edit&section=3)]

Ciljni elementi se definišu na tri načina:

* navodeći HTML tag ciljnih elemenata
* navodeći *klasu* elemenata
* navodeći direktnu identifikacionu vrijednost (ID) elementa

Kada definišemo preko HTML taga, to znači da će opis uticati na sve elemente u dokumentu koji imaju ovaj tag. Definicija ciljnih elemenata se tada vrši preko direktnog upisa odgovarajućeg HTML taga:

body {

svojstvo: vrijednost;

...

}

Klasa HTML elementa je riječ koju stavimo kao vrijednost argumenta CLASS pri definiciji tog elementa. Definicija ciljnih elemenata (svih koji imaju određenu istu klasu) se vrši tako što upišemo znak tačke (.) a zatim naziv klase:

.imeKlase {

svojstvo: vrijednost;

...

}

Ovaj opis će, dakle, imati uticaja na sve elemente u dokumentu koji su definisani na sljedeći način:

**<p** class="imeKlase"**>**...**</p>**

**<div** class="imeKlase"**>**...**</p>**

...

Identifikaciona vrijednost elementa je vrijednost argumenta ID pri definiciji tog elementa. U jednom dokumentu identifikacione vrijednosti moraju biti jedinstvene, tj. može postojati samo jedan element sa određenom identifikacionom vrijednošću, što znači da ovakvi opisi mogu uticati samo na po jedan element u dokumentu. Definišu se pomoću znaka tarabe (#) a zatim identifikacionu vrijednost:

#identifikacionaVrijednost {

svojstvo: vrijednost;

...

}

Ovaj će, dakle, opis uticati na sve elemente koji u svojoj HTML definiciji imaju id="identifikacionaVrijednost".

**Svojstva i vrijednosti**[[uredi - уреди](http://sh.wikipedia.org/w/index.php?title=CSS&action=edit&section=4)]

Svojstva na koja želimo da utičemo u datom opisu se definišu preko niza ključnih riječi definisanih u [W3C](http://sh.wikipedia.org/w/index.php?title=W3C&action=edit&redlink=1) standardu, a koje se kategorišu u sljedeće grupe:

* definicija pozadine elementa
* ivica
* okvir
* prikaz
* dimenzije
* font
* generisani sadržaj
* margine
* unutrašnji prazan prostor
* pozicija
* izgled pripadajućeg teksta

Vrijednosti pojedinih svojstava se definišu na dva načina:

* predefinisanim ključnim riječima
* brojevnim vrijednostima

Predefinisane ključne riječi za vrijednosti svojstava se koriste u situacijama kada dato svojstvo ima ograničen broj mogućnosti. Tako na primjer svojstvo za definiciju pozadine elementa može biti samo scroll(da se kreće uvijek zajedno sa sadržajem elementa) i fixed (da stoji uvijek na istom mjestu bez obzira na sadržaj).

Brojevne vrijednosti se mogu zadavati na nekoliko načina:

* zadajući samo brojevnu vrijednost
* navodeći i jedinicu veličine skupa sa opisom (px, em, pt, ...)

Ako zadajemo samo brojevnu vrijednost, imamo mogućnost da je pišemo na dva načina:

* u dekadnom sistemu (709)
* u heksadekadnom sistemu (#a4fde3); ovaj sistem se najviše koristi u definiciji boje

Primjeri[[uredi - уреди](http://sh.wikipedia.org/w/index.php?title=CSS&action=edit&section=5)]

* Primjer za definiciju pozadine cijele stranice:

body {

**background-color**: **aqua**; */\* pozadina boje aqua \*/*

}

* Primjer za definiciju izgleda svih elemenata klase "citat":

.citat {

**font-family**: serif; */\* font sa kukicama, kao Times New Roman \*/*

**font-style**: italic; */\* iskošen tekst \*/*

**font-size**: 9pt; */\* veličina slova 9 tačaka \*/*

}

Organizacija CSS koda[[uredi - уреди](http://sh.wikipedia.org/w/index.php?title=CSS&action=edit&section=6)]

CSS kod se može zadavati na tri standardna mjesta:

* direktno u tagu, koristeći argument style
* u zaglavlju dokumenta unutar taga style
* u eksternoj datoteci, koja se u dokument uključuje tagom link

Za jako male dokumente, argument style taga može biti prihvatljiv, ali za veće dokumente ne može jer se na ovaj stil definiše za svaki element ponaosob. Definicija stilova se najčešće radi grupišući elemente u klase, pomoću argumenta class taga, a zatim definišući izgled tih klasa bilo u okviru elementa style u zaglavlju ili u eksternoj datoteci. Argument style taga se najčešće koristi ako neki element ne pripada nijednoj grupi i ima posebne zahtjeve u odnosu na sve ostale elemente.

Stilovi u eksternoj datoteci imaju jednu posebnu prednost u odnosu na preostala dva načina, a to je da na taj način možemo da utičemo na elemente više dokumenata, praktično na čitav jedan sajt. Dovoljno je da pisac bude dosljedan u definisanju klasa elemenata, da bi jedan fajl sa stilovima određivao izgled svih stranica. Na ovaj način jedna promjena u fajlu sa stilovima utiče na sve stranice te su stilovi centralizovani.

Ograničenja[[uredi - уреди](http://sh.wikipedia.org/w/index.php?title=CSS&action=edit&section=7)]

Poznata ograničenja "čistog" CSS-a uključuju u sledeća:

* Nedosljednost brauzera:

Različiti brauzeri će interpretirati CSS-kŏd na različite načine zbog svojih bagova ili zbog nepodržavanja određenih mogućnosti CSS-a. Na primjer, [Majkrosoftov](http://sh.wikipedia.org/wiki/Microsoft) [Internet Eksplorer](http://sh.wikipedia.org/w/index.php?title=Internet_Eksplorer&action=edit&redlink=1), čija ranija izdanja nisu podržavala mnoge mogućnosti CSS-a 2.1, su pogrešno interpretirala veliki broj bitnih opisa objekata, uključujući širinu, visinu i plutanje. Da bi se postigla dosljednost prikaza u brojnim popularnim brauzerima, moraju se koristiti raznorazne "vještine" i "štimanja". I pored toga, ponekad je nemoguće postići identičan prikaz dokumenta u različitim brauzerima.

* Složenost

Veliki sajtovi imaju problem veličine CSS dokumenata, koji vremenom postaju toliko složeni i dugački da održavanje i dodavanje novih svojstava postaje izuzetno težak posao.

* Opisi elemenata se ne mogu bazirati na drugim objektima

CSS ne podržava „nasljeđivanje“ - ne postoji način da se izabere opis nekog drugog elementa na kojem bi se bazirao trenutni element. Nasljeđivanje postoji, ali se bazira na unaprijed utvrđenom pravilu da se opisi nasljeđuju od elementa u kojem se dotični element nalazi, tj. od roditeljskog elementa. Radna grupa za CSS objašnjava da ne želi da uvede proizvoljno nasljeđivanje jer bi to otežalo i usporilo rad brauzera.

One block declaration cannot explicitly inherit from another

* Problemi pri vertikalnoj orijentaciji elemenata

Iako je prilično lako postići horizontalni tok elemenata, vertikalni tok je prilično teško, i ponekad nemoguće, postići na željeni način. Jednostavni prohtjevi, poput centriranja objekta po vertikali, ili prikaz podnožja dokumenta na samom dnu vidljivog dijela prozora, zahtijevaju prilično komplikovane opise, ili opise koje podržavaju samo određeni brauzeri. Ponekad je najlakše uposliti skriptni jezik poput[Javaskripta](http://sh.wikipedia.org/w/index.php?title=Javaskript&action=edit&redlink=1) da na osnovu tekućeg stanja klijenta prikaže element na odgovarajućoj poziciji, čime se opet gubi na kompatibilnosti jer određeni korisnici ne žele da koriste Javaskript.

* Nedostatak aritmetičkih operacija

Dosadašnje verzije CSS-a ne podržavaju ni najjednostavnije aritmetičke operacije (npr. margin-left: 10%-10px;). Neki pomaci su napravljeni, raspravom Radne grupe za CSS o uvođenju izraza calc(), a Internet Eksplorer od verzije 5.0 podržava izraz expression, sa sličnim mogućnostima.

* Neslaganje pojedinačnih opisa

Određena svojstva imaju sličnu svrhu, i često se ne slažu u potpunosti. Na primjer, svojstva position, float i display određuju lokaciju prikaza elementa i njihovo kombinovanje često prouzrokuje neočekivane rezultate a određene kombinacije su po pravilu zabranjene. Na primjer, ako određenom elementu dodijelimo svojstvo display: table-cell, onda mu ne možemo dodijeliti svojstvo float nitiposition: relative.

* Određeni elementi ne mogu sadržati plutajuće pod-elemente

Ne postoji svojstvo koje takve elemente forsira da podržavaju plutajuće pod-elemente. Ponekad dodjeljivanje svojstva position: relative rješava ovaj problem, ili definisanje i samog nad-elementa da ima plutajuće svojstvo, ali u situacijama kada to nije prihvatljivo za organizaciju dokumenta, to nije rješenje.

* Nemogućnost postavljanja više pozadinskih tema istom elementu

Složena grafička rješenja zahtijevaju više od jedne pozadinske slike po elementu, ali CSS podržava samo jednu. Zbog toga su dizajneri primorani da elemente slažu jedan preko drugog da bi dobili ono što su tražili, ili da odustanu od prvobitne ideje. Ovaj problem se planira riješiti u trećoj verziji CSS-a[[1]](http://sh.wikipedia.org/wiki/CSS#cite_note-1), a određena rješenja već postoje u brauzerima [Safari (brauzer)](http://sh.wikipedia.org/w/index.php?title=Safari_(brauzer)&action=edit&redlink=1) i [Konkveror](http://sh.wikipedia.org/w/index.php?title=Konkveror&action=edit&redlink=1).

* Kontrola oblika elemenata

CSS trenutno dopušta samo pravougaone elemente. Elementi zaobljenih uglova ili drugih nepravougaonih oblika zahtijevaju izlaženje iz opsega CSS-a ili korišćenje sličica.

* Ne postoje striktna pravila o redoslijedu opisa i pripadajućih svojstava

Trenutno se opisi i svojstva mogu zadavati u bilo kom redu, ali ukoliko su u konfliktu, prednost ima posljednje zadati izraz.

* Nepostojanje promjenljivih

CSS ne podržava promjenljive ni u kakvom obliku. Promjenljive bi poslužile za centralizovani opis određene boje, veličine ili čitavog skupa svojstava, što bi obezbijedilo lakše održavanje i mijenjanje, ali i manje CSS dokumente. Sa trenutnim verzijama, pristupa se grupnom opisivanju nekoliko različitih klasa razdvojenim zarezom, da bi se umanjio problem nepostojanja promjenljivih.

* Nedostatak kolumna

Kolumne teksta, poput novinarskih, se najčešće rješavaju plutajućim elementima ili tabelama, ali različiti brauzeri, na monitorima različitih rezolucija, veličina i odnosa strana će prikazati različite rezultate. Dodavanje jednostavnih deklaracija za definisanje kolumna bi riješilo ove probleme.

ŠTA JE CSS

Članak napisan 21.07.2012.    Broj komentara 0 komentar/a     Broj pregleda 4540 pregleda[Štampaj članak](javascript:void(0))

[Pošaljite svom prijatelju link](javascript:void(0))

**CSS**(*Cascading Style Sheets*), a služi za definisanje stilova koji određuju izgled html elemenata (font, boje, pozadine, razmake...). Stilovi su definisani u formatu:

*sektor {   
    atribut1: vrednost1;   
    atribut2: vrednost2;   
    atribut3: vrednost3;   
    …   
}*

**SEKTORI**

Sektori mogu da se podele u tri grupe:

* HTML selektori (osnovni),
* ID selektori,
* Class selektori,

**HTML sektori**su osnovni sektori i predstavljaju nazive tagova koji se koriste u html/php dokumentu i na koje će biti primenjen. Primer:

*body {   
    color: #333333;   
    margin: 0 auto;   
    font-family: Verdana, Arial, Helvetica, sans-serif;   
}   
h1 {   
    color: #0000ff;   
}*

**ID sektori** nose prefiks # (tarabu). Koriste se ako je potrebno primeniti odrđeni stil na jedinstveni element u html/php dokumentu koji je obeležen određenim ID atributom. Primer:

*#zaglavlje {   
    color: #009900;   
}*  
  
a u samoj stranici (kodu stranice) biće prikazano kao:

*<p id=”zaglavlje”>Naziv firme</p>*

**CLASS sektori** nose prefiks . (tačku). Koriste se ako je potrebno primeniti odrđeni stil na element u html/php dokumentu koji je obeležen određenim CLASS atributom. Primer:

*.polozaj {   
    text-align: right;   
}*  
  
a u samoj stranici (kodu stranice) biće prikazano kao:

*<p class=”polozaj”>Potpis direktora</p>*

**KORIŠĆENJE**

CSS stilovi se mogu nalaziti u zasebnom (eksternom, **External**) fajlu sa ekstenzijom***.css***, ili se mogu nalaziti na samoj html/php stranici (interni). Interni stilovi mogu biti definisani u zaglavlju stranice (u okviru head taga,**Embeded**), ili biti u samom html elementu (**Inline**).

**Inline CSS** - Najjednostavniji i najmanje preporučljiv način za upotrebu CSS. Koristi se ako je potrebno na licu mesta (u samom HTML-u) izmeniti neki već postojeći CSS stil ili zadati određeni izgled bez kreiranja stila. Nije fleksibilan, jer kod eventualnog menjanja stila, potrebno je više vremena da se pronađe i zameni. Ukoliko na sajtu postoji vše stranica sa istim podešavanjem, utom vše posla. Primer:

*<p style=”color: #FF0000; text-align: center; ”>Ovo je neki pasus </p>*

**Embeded CSS** - CSS definisan u zaglavlju stranice je bolji način od prethodnog, ali i dalje može da predstavlja problem pri eventualnoj izmeni. U tom slučaju dizajner bi morao svaku stranicu “ručno” da proveri i izmeni stil i samim tim troši vše vremena. Ovaj način koristimo ako su nam potrebni stilovi za koje smo sigurni da ćemo ih koristiti samo na toj stranici. Primer:  *body,td,th {   
    font-family: Verdana, Arial, Helvetica, sans-serif;   
    font-size: 12px;   
    color: #333333;   
}   
a:link {   
    color: #006600;   
    text-decoration: none;   
}   
a:hover {   
    text-decoration: underline;   
}*

**External CSS** – Definisanje zasebnog CSS fajla je najpreporučljiviji način, jer pri eventualnoj izmeni sadržaja CSS fajla, stil će se automatski menjati i za sve stranice. Stranicu možete da povežete sa ***.css*** dokumentom, tako što head tagu unesete sledeći kod:

*<head>   
<link rel='stylesheet' type='text/css' href='putanja-do-css-a/stil.css' />   
</head>*

gde je *stil.css* eksterni css fajl.

PHP

[](http://bs.wikipedia.org/wiki/Datoteka:PHP-logo.svg)

[http://bits.wikimedia.org/static-1.23wmf10/skins/common/images/magnify-clip.png](http://bs.wikipedia.org/wiki/Datoteka:PHP-logo.svg)

**PHP** ili **P**ersonal **H**ome **P**age (ili *PHP: Hypertext Preprocessor*) je jedno od server-side [open source](http://bs.wikipedia.org/wiki/Otvoreni_softver) rješenja. PHP je reflektivni [programski jezik](http://bs.wikipedia.org/wiki/Programski_jezik)prvobitno dizajniran za pravljenje dinamičkih web stranica. [[1]](http://bs.wikipedia.org/wiki/PHP#cite_note-1) PHP se uglavnom koristi za izvršavanje skripti na [serverskoj](http://bs.wikipedia.org/wiki/Server) strani (*server-side scripting*), ali se može koristiti sa komandnog linijskog interfejsa ili preko samostalne grafičke aplikacije.

*Server side scripting* predstavlja vid izvršavanja [web](http://bs.wikipedia.org/wiki/Web) skripti na webu, a samo mu ime kaže da se izvršava na serveru (suprotno od [JavaScript](http://bs.wikipedia.org/w/index.php?title=JavaScript&action=edit&redlink=1)-a koji se izvršava na klijentskoj strani - [browseru](http://bs.wikipedia.org/wiki/Browser)). PHP je alternativa komercijalnim programskim jezicima ([Microsoft](http://bs.wikipedia.org/wiki/Microsoft) [ASP](http://bs.wikipedia.org/wiki/ASP)/[VBScript](http://bs.wikipedia.org/w/index.php?title=VBScript&action=edit&redlink=1)/[JScript](http://bs.wikipedia.org/w/index.php?title=JScript&action=edit&redlink=1), [Macromedia](http://bs.wikipedia.org/w/index.php?title=Macromedia&action=edit&redlink=1)[ColdFusion](http://bs.wikipedia.org/w/index.php?title=ColdFusion&action=edit&redlink=1), [Sun Microsystems](http://bs.wikipedia.org/w/index.php?title=Sun_Microsystems&action=edit&redlink=1) [JSP](http://bs.wikipedia.org/w/index.php?title=JSP&action=edit&redlink=1)/[JAVA](http://bs.wikipedia.org/wiki/Java_(programski_jezik))) i nekomercijalnom [CGI](http://bs.wikipedia.org/wiki/CGI)-u/[PERL](http://bs.wikipedia.org/w/index.php?title=PERL&action=edit&redlink=1)-u.

**Sadržaj**

  [[sakrij](http://bs.wikipedia.org/wiki/PHP)]

* [1 Historija](http://bs.wikipedia.org/wiki/PHP#Historija)
* [2 Upotreba](http://bs.wikipedia.org/wiki/PHP#Upotreba)
  + [2.1 Izvršavanje skripti na serverskoj strani](http://bs.wikipedia.org/wiki/PHP#Izvr.C5.A1avanje_skripti_na_serverskoj_strani)
  + [2.2 Izvršavanje skripti na komandnoj liniji](http://bs.wikipedia.org/wiki/PHP#Izvr.C5.A1avanje_skripti_na_komandnoj_liniji)
  + [2.3 Aplikacije grafičkog interfejsa na klijentskoj strani](http://bs.wikipedia.org/wiki/PHP#Aplikacije_grafi.C4.8Dkog_interfejsa_na_klijentskoj_strani)
* [3 Sintaksa](http://bs.wikipedia.org/wiki/PHP#Sintaksa)
  + [3.1 Tipovi podataka](http://bs.wikipedia.org/wiki/PHP#Tipovi_podataka)
  + [3.2 Objekti](http://bs.wikipedia.org/wiki/PHP#Objekti)
* [4 Resursi](http://bs.wikipedia.org/wiki/PHP#Resursi)
  + [4.1 Biblioteke](http://bs.wikipedia.org/wiki/PHP#Biblioteke)
  + [4.2 Ekstenzije](http://bs.wikipedia.org/wiki/PHP#Ekstenzije)
  + [4.3 Source code encoders, optimizers and accelerators](http://bs.wikipedia.org/wiki/PHP#Source_code_encoders.2C_optimizers_and_accelerators)
  + [4.4 Debuggers and profilers](http://bs.wikipedia.org/wiki/PHP#Debuggers_and_profilers)
  + [4.5 Templating engines](http://bs.wikipedia.org/wiki/PHP#Templating_engines)
  + [4.6 PEAR](http://bs.wikipedia.org/wiki/PHP#PEAR)
* [5 Support](http://bs.wikipedia.org/wiki/PHP#Support)
* [6 Criticism](http://bs.wikipedia.org/wiki/PHP#Criticism)
* [7 Važni datumi izdavanja](http://bs.wikipedia.org/wiki/PHP#Va.C5.BEni_datumi_izdavanja)
* [8 Dalji razvoj](http://bs.wikipedia.org/wiki/PHP#Dalji_razvoj)
* [9 Reference](http://bs.wikipedia.org/wiki/PHP#Reference)
* [10 Preporučena literatura](http://bs.wikipedia.org/wiki/PHP#Preporu.C4.8Dena_literatura)
* [11 Također pogledajte](http://bs.wikipedia.org/wiki/PHP#Tako.C4.91er_pogledajte)
* [12 Vanjski linkovi](http://bs.wikipedia.org/wiki/PHP#Vanjski_linkovi)

Historija[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=1) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=1)]

PHP je napisan kao skup [CGI](http://bs.wikipedia.org/w/index.php?title=Common_Gateway_Interface&action=edit&redlink=1) binarnih fajlova u [C programskom jeziku](http://bs.wikipedia.org/wiki/C_programski_jezik) od strane danskog/grenlandskog programera Rasmusa Lerdorfa [1994](http://bs.wikipedia.org/wiki/1994). godine, kako bi zamijenio mali skup [Perl](http://bs.wikipedia.org/wiki/Perl) skripti koje je koristio za održavanje lične stranice. [[2]](http://bs.wikipedia.org/wiki/PHP#cite_note-2)

Lerdorf je inicijalno napravio PHP kako bi prikazao svoj rezime i kako bi prikupio određene podatke, poput onih o broju posjeta na svojoj stranici. *Personal Home Page Tools* se pojavio u javnosti[8. juna](http://bs.wikipedia.org/wiki/8._juni) [1995](http://bs.wikipedia.org/wiki/1995). godine nakon što ga je Lerdorf kombinovao sa vlastitim *Form Interpreter*-om kako bi napravio *PHP/FI* (ovo izdanje se smatra drugom verzijom PHP-a).

Zeev Suraski i Andi Gutmans, dva [izraelska](http://bs.wikipedia.org/wiki/Izrael) razvijača sa Technion IIT-a, su nanovo napisali [parser](http://bs.wikipedia.org/wiki/Parser) [1997](http://bs.wikipedia.org/wiki/1997). godine i stvorili bazu za PHP 3, promjenivši ime jezika u rekurzivni inicijalizam *PHP: Hypertext Preprocessor*. Razvojni tim je zvanično izdao PHP/FI 2 u novembru [1997](http://bs.wikipedia.org/wiki/1997). godine nakon mjeseci beta testiranja. Javno testiranje PHP 3 je otpočelo, a zvanično puštanje u upotrebu se desilo juna [1998](http://bs.wikipedia.org/wiki/1998). godine. Suraski i Gutmans su tada započeli novo prerađivanje PHP jezgre, proizvodeći Zend Engine [1999](http://bs.wikipedia.org/wiki/1999). godine.[[3]](http://bs.wikipedia.org/wiki/PHP#cite_note-3) Također su osnovali Zend Technologies u Ramat Gani (Izrael) koja se bavi upravljanjem razvoja PHP-a.

Maja 2000. godine je izdat PHP 4 sa Zend Engine 1.0. Najskoriji update isporučen od The PHP Group je za stariju seriju PHP 4 koja je do maja 2007. uključivala verzije do 4.4.7. PHP 4 će imati osigurano snabdijevanje sigurnosnim updateima do 31. decembra 2007.[[4]](http://bs.wikipedia.org/wiki/PHP#cite_note-4)

13. jula 2004, izdat je PHP 5 sa novom Zend Engine II. PHP 5 je uključio nove karakteristike poput:[[5]](http://bs.wikipedia.org/wiki/PHP#cite_note-5)

* Unapređena podrška za objektno-orjentisano programiranje
* PHP Data Objects nastavak, koji definiše *lightweight* konzistentni interfejs za pristup [bazama podataka](http://bs.wikipedia.org/wiki/Baza_podataka)
* Poboljšane performanse
* Bolja podrška za [MySQL](http://bs.wikipedia.org/wiki/MySQL)
* Ugrađena podrška za SQLite
* Integrisana SOAP podrška
* Podatkovni iteratori
* Rad s greškama preko izuzetaka

Posljednja stabilna verzija PHP-a je 5.2.3. koja je izdata 1. juna 2007. godine.

Upotreba[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=2) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=2)]

PHP se općenito izvršava na [web](http://bs.wikipedia.org/wiki/Web) [serveru](http://bs.wikipedia.org/wiki/Server), uzimajući PHP kod kao ulaz, kreirajući web stranice kao izlaz; međutim može se također koristiti za pravljenje [skripti](http://bs.wikipedia.org/w/index.php?title=Skripta&action=edit&redlink=1) sa komandne linije i aplikacije za korisnički interfejs na klijentskoj strani. PHP se može koristiti na većini web servera i na gotovo svakom [operativnom sistemu](http://bs.wikipedia.org/wiki/Operativni_sistem) i platformi bez ikakve novčane nadoknade. PHP Group također obezbjeđuje kompletan izvorni kod za korisnike koji grade, podešavaju ili nadograđuju kod za vlastitu upotrebu.

**Izvršavanje skripti na serverskoj strani**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=3) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=3)]

Originalno dizajniran za kreiranje dinamičkih web stranica, PHP se načelno fokusira na *server-side scripting* odnosno izvršavanje skripti na serverskoj strani. Izvršavanjem PHP parsera sa web serverom i web [browserom](http://bs.wikipedia.org/wiki/Browser), PHP model se može uporediti sa drugim server-side scripting jezicima kao što su Microsoftov [ASP.NET](http://bs.wikipedia.org/w/index.php?title=ASP.NET&action=edit&redlink=1) sistem, JavaServer Pages (Sun Microsystems) i Ruby on Rails [framework](http://bs.wikipedia.org/w/index.php?title=Framework&action=edit&redlink=1), pošto svi oni obezbjeđuju dinamički sadržaj klijentu sa web servera.

Kako bi se direktnije takmičio sa *framework* pristupom, Zend radi na Zend Frameworku - [skupu](http://bs.wikipedia.org/wiki/Skup) PHP gradivnih blokova i najboljih praksi u nastajanju (počevši od juna 2006); drugi PHP frameworkci sa istim osnovama uključuju CakePHP, PRADO i Symfony.

[LAMP arhitektura](http://bs.wikipedia.org/w/index.php?title=LAMP_arhitektura&action=edit&redlink=1) je postala popularna u [web](http://bs.wikipedia.org/wiki/Web) [industriji](http://bs.wikipedia.org/wiki/Industrija) kao način za plasiranje jeftinih, pouzdanih, skalabilnih i sigurnih web [aplikacija](http://bs.wikipedia.org/w/index.php?title=Aplikacija&action=edit&redlink=1). PHP se obično koristi kao *P* dio u ovoj kompoziciji zajedno sa [Linuxom](http://bs.wikipedia.org/wiki/Linux), [Apache](http://bs.wikipedia.org/wiki/Apache) serverom i [MySQL](http://bs.wikipedia.org/wiki/MySQL)-om, iako se *P* može odnositi i na [Python](http://bs.wikipedia.org/wiki/Python) [programski jezik](http://bs.wikipedia.org/wiki/Programski_jezik) ili [Perl](http://bs.wikipedia.org/wiki/Perl).

PHP se može koristiti sa velikim brojem relacionih sistema za upravljanje bazama podataka (RDBMS), izvršava se na svim najpopularnijim web serverima i dostupan je za mnoge različite operativne sisteme. Prilagodljivost znači da PHP ima široku instalacionu osnovu na Internetu; preko 19 miliona internetskih domena su udomljene na serverima sa instaliranim PHP-om. [[6]](http://bs.wikipedia.org/wiki/PHP#cite_note-6)

Primjeri, popularnih server-side PHP aplikacija uključuju: [phpBB](http://bs.wikipedia.org/w/index.php?title=PhpBB&action=edit&redlink=1), [WordPress](http://bs.wikipedia.org/w/index.php?title=WordPress&action=edit&redlink=1) i [MediaWiki](http://bs.wikipedia.org/wiki/MediaWiki).

**Izvršavanje skripti na komandnoj liniji**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=4) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=4)]

PHP također obezbjeđuje komandni linijski interfejs SAPI za razvoj shell i desktop aplikacija, demona, log parsiranja i drugih sistemskih administrativnih zadataka. PHP se sve više koristi na komandnoj liniji za zadatke koji su tradicionalno pripadali Perlu, Pythonu, awk-u ili shell scriptingu.

**Aplikacije grafičkog interfejsa na klijentskoj strani**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=5) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=5)]

PHP obezbjeđuje povezivanje sa GUI bibliotekama poput GTK+ (sa PHP-GTK), Qt (sa [PHP-Qt](http://www.php-qt.org/)) i bibliotekama za tekstualni način rada poput ncurses radi olakšanja razvoja šireg opsega aplikacija grafičkog interfejsa za cross platforme.

Sintaksa[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=6) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=6)]

PHP se primarno ponaša kao softverski filter. PHP program uzima informacije iz datoteke ili toka (stream) koji sadrže tekst i specijalne PHP instrukcije i izbacuje drugi tok podataka za prikazivanje. Od PHP 4, [parser](http://bs.wikipedia.org/wiki/Parser) [kompajlira](http://bs.wikipedia.org/wiki/Kompajler) ulaz za proizvodnju bytecode-a kojeg obrađuje Zend Engine, dajući poboljšanu performansu od [interpreterskog](http://bs.wikipedia.org/wiki/Interpreter) prethodnika. Zend Engine II je u srcu PHP 5.

Školski primjer programa Hello World za PHP izgleda na sljedeći način:

**<?php**

echo 'Hello, World!';

**?>**

PHP samo parsira kod unutar svojih delimitera, kao što su **<?php** i **?>**. Sve izvan toga se direktno šalje na izlaz i ne parsira se od strane PHP-a. Gornji primjer je ekvivalentan sljedećem tekstu na izlazu:

Hello, World!

Primarno korištenje ovoga je radi omogućavanja PHP iskazima da budu uklopljeni u [HTML](http://bs.wikipedia.org/wiki/HTML) dokumente, naprimjer:

**<?php**

*// PHP iskazi ovdje*

**?>**

Regularni HTML ovdje

**<?php**

*// Više PHP iskaza*

**?>**

Ispred [varijabli](http://bs.wikipedia.org/wiki/Varijabla) se dodaje simbol dolara ($), a primitivni tip se ne treba specificirati unaprijed. Za razliku od imena [funkcija](http://bs.wikipedia.org/wiki/Funkcija) i [klasa](http://bs.wikipedia.org/wiki/Klasa), imena varijabli su osjetljiva na veliko i malo slovo ([eng.](http://bs.wikipedia.org/wiki/Engleski_jezik): *case sensitive*).

Pomoću dvostrukih navodnika (""), kao i heredoc [stringova](http://bs.wikipedia.org/w/index.php?title=String&action=edit&redlink=1) je moguće uklopiti vrijednost varijable u string.

PHP nove linije koda tretira kao *whitespace*. Iskazi se završavaju tačkom-zarezom (;) osim u nekoliko specijalnih slučajeva.

PHP ima tri tipa [sintakse](http://bs.wikipedia.org/w/index.php?title=Sintaksa&action=edit&redlink=1) za komentare: /\* \*/ za blok komentare, i // kao i # za inline komentare.

Primjer jednostavne funkcije u PHP-u:

**<?php**

**function** imeStranice()

{

echo "Wikipedia - Slobodna Enciklopedija";

}

echo "Ovo je ";

imeStranice();

**?>**

Rezultat:

Ovo je Wikipedia - Slobodna Enciklopedija

**Tipovi podataka**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=7) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=7)]

PHP pohranjuje [cijele brojeve](http://bs.wikipedia.org/wiki/Cijeli_brojevi) u rangu koji zavisi od platforme. Ovaj rang je tipično za 32-bitne predznačene cijele brojeve ([eng.](http://bs.wikipedia.org/wiki/Engleski_jezik) *integer*).

Integer [varijablama](http://bs.wikipedia.org/wiki/Varijabla) se mogu dodijeliti vrijednosti pomoću decimalne (pozitivne i negatine), oktalne i heksadecimalne [notacije](http://bs.wikipedia.org/w/index.php?title=Notacija&action=edit&redlink=1). [Realni brojevi](http://bs.wikipedia.org/wiki/Realni_brojevi) se pohranjuju u rang specifičan za platformu. Mogu biti specificirani korištenjem floating point notacije ili pomoću dvije forme naučne notacije.

PHP ima urođeni [boolean](http://bs.wikipedia.org/w/index.php?title=Boolean&action=edit&redlink=1) tip, koji je tako i imenovan, slično boolean tipovima u [Javi](http://bs.wikipedia.org/wiki/Java_(programski_jezik)) i [C++](http://bs.wikipedia.org/wiki/C%2B%2B). Korištenjem pravila za konverziju boolean tipova, vrijednosti različite od nule se mogu interpretirati kao true, a nula kao false, kao u Perlu.

Null tip podataka predstavlja varijablu koja nema vijednost. Jedina vrijednost u null tipu podataka je *NULL*.

Varijable "resursnog" tipa predstavljaju reference na resurse iz vanjskih izvora. Tipično ih kreiraju [funkcije](http://bs.wikipedia.org/wiki/Funkcija) iz specifične ekstenzije, a mogu biti obrađivane samo od strane funkcija iz iste ekstenzije. Primjeri uključuju [datoteku](http://bs.wikipedia.org/wiki/Datoteka), sliku i resurse baze podataka.

[Nizovi](http://bs.wikipedia.org/w/index.php?title=Nizovi&action=edit&redlink=1) podržavaju i numeričke i [string](http://bs.wikipedia.org/w/index.php?title=String&action=edit&redlink=1) indekse, i heterogeni su. Nizovi mogu sadržavati elemente bilo kojeg PHP tipa, uključujući resurse, objekte, čak i druge nizove. Poredak se čuva u listama vrijednosti i u [hash tabelama](http://bs.wikipedia.org/w/index.php?title=Hash_tabela&action=edit&redlink=1) sa ključevima i vrijednostima, koji se mogu miješati.

**Objekti**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=8) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=8)]

Osnovne funkcionalnosti [objektno orijentisanog programiranja](http://bs.wikipedia.org/wiki/Objektno_orijentisano_programiranje) su dodate u PHP3. Upravljanje objektima je upotpunosti prerađeno za PHP 5, što se odrazilo na bolju performansu i više karakteristika. U prethodnim verzijama PHP-a, upravljanje objektima se obavljalo na isti načina kao i sa primitivnim tipovima. Mana ove [metode](http://bs.wikipedia.org/wiki/Metoda) je bila u tome što bi se cijeli objekt prekopirao kada bi se varijabla napunila ili bi se proslijedio metodi kao parametar.

|  |  |
| --- | --- |
| [Bih-usa.svg](http://bs.wikipedia.org/wiki/Datoteka:Bih-usa.svg) | **Ovaj članak nije preveden ili je djelimično preveden.** Ako smatrate da ste sposobni da ga prevedete, kliknite na link [**uredi**](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit) i prevedite ga vodeći računa o enciklopedijskom stilu pisanja i [pravopisu bosanskog jezika](http://bs.wikipedia.org/wiki/Pravilno-nepravilno_u_bosanskome_jeziku). |

In the new approach, objects are referenced by [handle](http://bs.wikipedia.org/w/index.php?title=Smart_pointer&action=edit&redlink=1), and not by value. PHP 5 introduced private and protected [member variables](http://bs.wikipedia.org/w/index.php?title=Member_variable&action=edit&redlink=1) and methods, along with [abstract classes](http://bs.wikipedia.org/w/index.php?title=Class_(computer_science)&action=edit&redlink=1) and [abstract methods](http://bs.wikipedia.org/w/index.php?title=Abstract_method&action=edit&redlink=1). It also introduced a standard way of declaring [constructors](http://bs.wikipedia.org/w/index.php?title=Constructor_(computer_science)&action=edit&redlink=1) and [destructors](http://bs.wikipedia.org/w/index.php?title=Destructor_(computer_science)&action=edit&redlink=1) similar to that of other object-oriented languages, such as C++, and an [exception handling](http://bs.wikipedia.org/w/index.php?title=Exception_handling&action=edit&redlink=1) model similar to that of other programming languages.

The static method and class variable features in Zend Engine 2 do not work the way some expect. There is no [virtual table](http://bs.wikipedia.org/w/index.php?title=Virtual_table&action=edit&redlink=1) feature in the engine, so the [static variables](http://bs.wikipedia.org/w/index.php?title=Static_variable&action=edit&redlink=1) are bound with a name at compile time instead of with a reference.

If the developer asks to create a copy of an object by using the reserved word *clone*, the Zend engine will check if a \_\_clone() method has been defined or not. If not, it will call a default\_\_clone() which will copy all of the object's properties. If a \_\_clone() method is defined, then it will be responsible to set the necessary properties in the created object. For convenience, the engine will supply a function that imports all of the properties from the source object, so that they can start with a by-value [replica](http://en.wiktionary.org/wiki/replica) of the source object, and only override properties that need to be changed.

Resursi[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=9) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=9)]

**Biblioteke**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=10) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=10)]

PHP includes a large number of free and open source libraries with the core build. PHP is a fundamentally [Internet](http://bs.wikipedia.org/wiki/Internet)-aware system with modules built in for accessing [FTP](http://bs.wikipedia.org/w/index.php?title=File_transfer_protocol&action=edit&redlink=1) servers, many database servers, embedded SQL libraries such as embedded [MySQL](http://bs.wikipedia.org/wiki/MySQL) and [SQLite](http://bs.wikipedia.org/w/index.php?title=SQLite&action=edit&redlink=1), [LDAP](http://bs.wikipedia.org/w/index.php?title=Lightweight_Directory_Access_Protocol&action=edit&redlink=1) servers, and others. Many functions familiar to C programmers such as those in the [stdio](http://bs.wikipedia.org/w/index.php?title=Stdio.h&action=edit&redlink=1) family are available in the standard PHP build.

**Ekstenzije**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=11) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=11)]

PHP allows developers to write [extensions](http://bs.wikipedia.org/w/index.php?title=Extension_(computing)&action=edit&redlink=1) in [C](http://bs.wikipedia.org/w/index.php?title=C_(programming_language)&action=edit&redlink=1) to add functionality to the PHP language. These can then be compiled into PHP or loaded dynamically at runtime. Extensions have been written to add support for the [Windows API](http://bs.wikipedia.org/w/index.php?title=Windows_API&action=edit&redlink=1), process management on [Unix-like](http://bs.wikipedia.org/w/index.php?title=Unix-like&action=edit&redlink=1) [operating systems](http://bs.wikipedia.org/w/index.php?title=Operating_system&action=edit&redlink=1), multibyte strings ([Unicode](http://bs.wikipedia.org/wiki/Unicode)), [cURL](http://bs.wikipedia.org/w/index.php?title=CURL&action=edit&redlink=1), and several popular [compression formats](http://bs.wikipedia.org/w/index.php?title=Compression_formats&action=edit&redlink=1). Some more unusual features include integration with [Internet relay chat](http://bs.wikipedia.org/w/index.php?title=Internet_relay_chat&action=edit&redlink=1), dynamic generation of images and [Adobe Flash](http://bs.wikipedia.org/w/index.php?title=Adobe_Flash&action=edit&redlink=1) content, and even [speech synthesis](http://bs.wikipedia.org/w/index.php?title=Speech_synthesis&action=edit&redlink=1). The [PHP Extension Community Library](http://bs.wikipedia.org/w/index.php?title=PHP_Extension_Community_Library&action=edit&redlink=1) (PECL) project is a repository for extensions to the PHP language.

**Source code encoders, optimizers and accelerators**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=12) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=12)]

As with many scripting languages, PHP scripts are normally kept as human-readable source code, even on production webservers. While this allows flexibility, it can raise issues with security and performance.

Encoders offer some source code security and enable [proprietary software](http://bs.wikipedia.org/w/index.php?title=Proprietary_software&action=edit&redlink=1) by hindering source code [reverse engineering](http://bs.wikipedia.org/w/index.php?title=Reverse_engineering&action=edit&redlink=1). Encoders fall broadly into two types; those that hide source code and those that compile code into optcode. The downside of this latter approach is that a special extension has to be installed on the server in order to run encoded scripts, however the approach of encoding compiled code and use of an extension offers typically the best performance, security and opportunity for additional features that may be useful for developers. Compiled code solutions may exploit the potential for increased security through the use of their own execution engine, although some simpler solutions rely on the regular PHP engine to execute the compiled code. The most commonly used packages for source code protection are from [Zend Technologies](http://bs.wikipedia.org/w/index.php?title=Zend_Technologies&action=edit&redlink=1) and [ionCube Ltd.](http://bs.wikipedia.org/w/index.php?title=IonCube_Ltd.&action=edit&redlink=1)

Code optimizers improve the quality of the compiled code by reducing its size and making changes that can reduce the execution time and improve performance. The nature of the PHP [compiler](http://bs.wikipedia.org/w/index.php?title=Compiler&action=edit&redlink=1)is such that there are often many opportunities for [code optimization](http://bs.wikipedia.org/w/index.php?title=Optimization_(computer_science)&action=edit&redlink=1).

Accelerators offer performance gains by [caching](http://bs.wikipedia.org/w/index.php?title=Caching&action=edit&redlink=1) the compiled form of a PHP script in [shared memory](http://bs.wikipedia.org/w/index.php?title=Shared_memory&action=edit&redlink=1) to avoid the overhead of [parsing](http://bs.wikipedia.org/w/index.php?title=Parsing&action=edit&redlink=1) and [compiling](http://bs.wikipedia.org/w/index.php?title=Compiling&action=edit&redlink=1) the code every time the script runs. They may also perform [code optimization](http://bs.wikipedia.org/w/index.php?title=Code_optimization&action=edit&redlink=1) to provide increased execution performance. Both commercial (e.g. Zend Platform) and open source accelerators (e.g. xcache, eAccelerator, APC) are available.

**Debuggers and profilers**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=13) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=13)]

[Debuggers](http://bs.wikipedia.org/wiki/Debugger) and [profilers](http://bs.wikipedia.org/w/index.php?title=Profiler&action=edit&redlink=1) allow developers to analyze running PHP code for potential and noted [software bugs](http://bs.wikipedia.org/w/index.php?title=Software_bug&action=edit&redlink=1) and bottlenecks. Examples of such software for PHP include [APD](http://bs.wikipedia.org/w/index.php?title=Advanced_PHP_Debugger&action=edit&redlink=1) and [Xdebug](http://bs.wikipedia.org/w/index.php?title=Xdebug&action=edit&redlink=1).

**Templating engines**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=14) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=14)]

Templating engines provide [macros](http://bs.wikipedia.org/w/index.php?title=Macro&action=edit&redlink=1) that allow PHP applications to uniformly identify common variables. One popular templating engine is [Smarty](http://bs.wikipedia.org/w/index.php?title=Smarty&action=edit&redlink=1). PHP itself makes a good templating engine.

**PEAR**[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=15) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=15)]

The [PHP Extension and Application Repository](http://bs.wikipedia.org/w/index.php?title=PHP_Extension_and_Application_Repository&action=edit&redlink=1) (PEAR) project aims to provide reusable libraries and components for PHP development. PEAR projects are usually written in PHP code using the[Object-oriented programming](http://bs.wikipedia.org/w/index.php?title=Object-oriented_programming&action=edit&redlink=1) paradigm.

Support[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=16) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=16)]

PHP has a formal development manual that is maintained by the [free software community](http://bs.wikipedia.org/w/index.php?title=Free_software_community&action=edit&redlink=1). In addition, answers to many questions can often be found by doing a simple internet search. PHP users assist each other through various media such as chat, forums, newsgroups and PHP developer web sites. In turn, the PHP development team actively participates in such communities, garnering assistance from them in their own development effort (PHP itself) and providing assistance to them as well. There are many help resources[[7]](http://bs.wikipedia.org/wiki/PHP#cite_note-7) available for the novice PHP programmer.

Criticism[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=17) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=17)]

Criticisms of PHP include those general criticisms ascribed to other [scripting languages](http://bs.wikipedia.org/w/index.php?title=Scripting_language&action=edit&redlink=1) and [dynamically typed languages](http://bs.wikipedia.org/w/index.php?title=Type_system&action=edit&redlink=1). This list includes criticisms that have been rectified in recent versions.

* PHP originally inserted data received over the network directly into the language namespace ("register\_globals"), leading to confusion between trusted and untrusted data, and unnecessary potential for security holes in PHP applications. This behavior was turned off by default from version 4.2.0 released in April 2002.[[8]](http://bs.wikipedia.org/wiki/PHP#cite_note-8) However, this feature is still being used by some legacy applications[[9]](http://bs.wikipedia.org/wiki/PHP#cite_note-9).
* PHP has traditionally used features such as "[magic\_quotes\_gpc](http://bs.wikipedia.org/w/index.php?title=Magic_quotes&action=edit&redlink=1)" and "magic\_quotes\_runtime" which attempt to escape apostrophes (') and quotes (") in strings in the assumption that they will be used in databases, to prevent [SQL injection](http://bs.wikipedia.org/w/index.php?title=SQL_injection&action=edit&redlink=1) attacks. This leads to confusion over which data is escaped and which is not, and to problems when data is not in fact used as input to a database. [[10]](http://bs.wikipedia.org/wiki/PHP#cite_note-10)
* PHP does not have complete native support for [Unicode](http://bs.wikipedia.org/wiki/Unicode) or multibyte strings. [[11]](http://bs.wikipedia.org/wiki/PHP#cite_note-11)
* PHP does not enforce the declaration of variables prior to their use, and variables which have not been initialized can have operations (such as [concatenation](http://bs.wikipedia.org/w/index.php?title=Concatenation&action=edit&redlink=1)) performed on them; an operation on an uninitialized variable raises an E\_NOTICE level error, but this is hidden by default.
* PHP has no [namespace](http://bs.wikipedia.org/w/index.php?title=Namespace_(computer_science)&action=edit&redlink=1) support, which leads to a very large amount of globally available functions that can easily number into the thousands.
* PHP's dynamic type conversion could potentially cause problems. Variable types in PHP, although they exist, are transparent to the programmer. Some may consider this a feature, as a variable can change from a [number](http://bs.wikipedia.org/w/index.php?title=Integer_(computer_science)&action=edit&redlink=1) to a [string](http://bs.wikipedia.org/w/index.php?title=String_(computer_science)&action=edit&redlink=1) and back again without extra lines of code. However, variable type errors are not detected at compile-time, and the dynamic-typing behavior lacks full predictability.
* The standard function library lacks internal consistency. Many functions perform relatively similar actions and have different name standards and argument orders. For example:
  + Argument consistency: strpos($haystack, $needle) vs. in\_array($needle, $haystack)
  + Naming convention: both of these work case-insensitively strcasecmp() vs. stristr() but the former indicates this with "case" while the later does with "i"
  + Function [name consistency](http://bs.wikipedia.org/w/index.php?title=Naming_conventions_(programming)&action=edit&redlink=1): strpos() vs. str\_replace()
* Functions are not [first-class objects](http://bs.wikipedia.org/w/index.php?title=First-class_object&action=edit&redlink=1). This requires referencing functions by strings and object methods as a two-element array of the object and method name as a string. Consequently,[anonymous functions](http://bs.wikipedia.org/w/index.php?title=Anonymous_function&action=edit&redlink=1) are also referenced by string.
* Lack of [late static binding](http://bs.wikipedia.org/w/index.php?title=Late_static_binding&action=edit&redlink=1) [[12]](http://bs.wikipedia.org/wiki/PHP#cite_note-12)
* Some portability issues with 32-bit and 64-bit integers, and sometimes unsigned integers get converted to signed values.[[13]](http://bs.wikipedia.org/wiki/PHP#cite_note-13)

Važni datumi izdavanja[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=18) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=18)]

|  |  |  |
| --- | --- | --- |
| **Verzija** | **Datum izdavanja** | **Najvažnije promjene** |
| PHP 1.0 | [8. juni](http://bs.wikipedia.org/wiki/8._juni) [1995](http://bs.wikipedia.org/wiki/1995). | Zvanično nazvan "Personal Home Page Tools (PHP Tools)". Prva upotreba imena "PHP". |
| PHP Version 2 (PHP/FI) | [16. april](http://bs.wikipedia.org/wiki/16._april) [1996](http://bs.wikipedia.org/wiki/1996). | Od strane svog tvorca smatran za "najbrži i najjednostavniji alat" za kreiranje dinamičkih web stranica. |
| PHP 3.0 | [6. juni](http://bs.wikipedia.org/wiki/6._juni) [1998](http://bs.wikipedia.org/wiki/1998). | Razvoj umjesto jednog uključuje više razvijača. Zeev Suraski i Andi Gutmans nanovo pišu osnovu za ovu verziju. |
| PHP 4.0 | [22. maj](http://bs.wikipedia.org/wiki/22._maj) [2000](http://bs.wikipedia.org/wiki/2000). | Dodan napredniji dvofazni parsiraj/izvrši *tag-parsing* sistem nazvan Zend engine. |
| PHP 4.1 | [10. decembar](http://bs.wikipedia.org/wiki/10._decembar) [2001](http://bs.wikipedia.org/wiki/2001). | Uvedeni superglobali ($\_GET, $\_SESSION, itd.) |
| PHP 4.2 | [22. april](http://bs.wikipedia.org/wiki/22._april) [2002](http://bs.wikipedia.org/wiki/2002). | Onemogućavanje register\_globals se podrazumijeva |
| PHP 4.3 | [27. decembar](http://bs.wikipedia.org/wiki/27._decembar) [2002](http://bs.wikipedia.org/wiki/2002). | Uveden CLI, kao dodatak za CGI |
| PHP 4.4 | [11. juli](http://bs.wikipedia.org/wiki/11._juli) [2005](http://bs.wikipedia.org/wiki/2005). |  |
| PHP 5.0 | [13. juli](http://bs.wikipedia.org/wiki/13._juli) [2004](http://bs.wikipedia.org/wiki/2004). | Zend Engine II s novim objektnim modelom. |
| PHP 5.1 | [25. novembar](http://bs.wikipedia.org/wiki/25._novembar) [2005](http://bs.wikipedia.org/wiki/2005). |  |
| PHP 5.2 | [2. novembar](http://bs.wikipedia.org/wiki/2._novembar) [2006](http://bs.wikipedia.org/wiki/2006). | Omogućavanje filterskih ekstenzija se podrazumijeva |

Dalji razvoj[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=19) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=19)]

PHP 6, u razvoju od jula 2007., ima za cilj neke od PHP 5 nedostataka.[[14]](http://bs.wikipedia.org/wiki/PHP#cite_note-14)

* Namespace support will be added
* podrška za izvorni Unicode će biti dodata
* opcija **magic\_quotes** će biti uklonjena
* **HTTP\_\*\_VARS** globalne varijable će biti uklonjene
* opcija **register\_globals** će biti uklonjena
* opcija **safe\_mode** će biti uklonjena

Reference[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=20) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=20)]

1. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-1) [Introduction](http://php.net/manual/introduction.php).
2. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-2) [Re: There ARE other scripting languages besides PHP](http://it.slashdot.org/comments.pl?sid=196655&cid=16113787).
3. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-3) [Zend Engine version 2.0: Feature Overview and Design](http://www.zend.com/zend/zend-engine-summary.php).
4. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-4) [php.net 2007 news archive](http://www.php.net/archive/2007.php)
5. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-5) <http://www.onlamp.com/pub/a/php/2004/07/15/UpgradePHP5.html>
6. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-6) <http://php.net/usage.php>
7. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-7) <http://php.net/support>
8. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-8) [PHP Manual: Chapter 29. Using Register Globals](http://php.net/register_globals)
9. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-9) <http://www.oscommerce.info/kb/osCommerce/Installation_and_Upgrades/46>
10. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-10) [Rasmus Lerdorf's thoughts about PHP6](http://news.php.net/php.internals/17883)
11. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-11) [Overview of PHP function which are currently Unicode compatible](http://www.php.net/~scoates/unicode/render_func_data.php) (in [CVS](http://bs.wikipedia.org/w/index.php?title=Concurrent_Versions_System&action=edit&redlink=1))
12. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-12) <http://www.digitalsandwich.com/archives/53-Late-Static-Binding-in-PHP.html> retrieved [March 28](http://bs.wikipedia.org/w/index.php?title=March_28&action=edit&redlink=1), [2007](http://bs.wikipedia.org/wiki/2007)
13. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-13) [Integers in PHP, running with scissors, and portability](http://www.mysqlperformanceblog.com/2007/03/27/integers-in-php-running-with-scissors-and-portability/).
14. [**Jump up^**](http://bs.wikipedia.org/wiki/PHP#cite_ref-14) <http://www.php.net/~derick/meeting-notes.html>

Preporučena literatura[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=21) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=21)]

* Kerner, Sean Michael. "[Is PHP The Cure For The 'Broken' Web?](http://www.internetnews.com/dev-news/article.php/3631831)", internetnews.com, [13. septembar](http://bs.wikipedia.org/wiki/13._septembar) [2006](http://bs.wikipedia.org/wiki/2006).
* Kerner, Sean Michael. "[Microsoft Opens PHP Door](http://www.internetnews.com/dev-news/article.php/3641101)", internetnews.com, [31. oktobar](http://bs.wikipedia.org/wiki/31._oktobar) [2006](http://bs.wikipedia.org/wiki/2006).
* Sweat, Jason E (2005). *Guide to PHP Design Patterns*, PHP|architect [ISBN 0-9735898-2-5](http://bs.wikipedia.org/wiki/Posebno:KnjizniIzvori/0973589825).
* Alshanetsky, Ilia (2005). *Guide to PHP Security*, PHP|architect [ISBN 0-9738621-0-6](http://bs.wikipedia.org/wiki/Posebno:KnjizniIzvori/0973862106).
* Shiflett, Chris (2005). *Essential PHP Security*, O'Reilly Media [ISBN 0-596-00656-X](http://bs.wikipedia.org/wiki/Posebno:KnjizniIzvori/059600656X).
* Ullman, Larry (2003). *PHP and MySQL for Dynamic Web Sites*, 1st Edition, Peachpit Press [ISBN 0-321-18648-6](http://bs.wikipedia.org/wiki/Posebno:KnjizniIzvori/0321186486).
* Vaswani, Vikram (2007). *PHP Programming Solutions*, McGraw-Hill [ISBN 0-07-148745-X](http://bs.wikipedia.org/wiki/Posebno:KnjizniIzvori/007148745X).

Također pogledajte[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=22) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=22)]

* [Drupal](http://bs.wikipedia.org/wiki/Drupal)

Vanjski linkovi[[uredi](http://bs.wikipedia.org/w/index.php?title=PHP&veaction=edit&section=23) | [uredi izvor](http://bs.wikipedia.org/w/index.php?title=PHP&action=edit&section=23)]

* [PHP](http://www.php.co.ba/) **(bs)**
* [PHP](http://yu.php.net/) **(sr)**
* [PHP](http://www.php-hr.net/) **(hr)**
* [Niz tutorijala namijenjenih uglavnom početnicima](http://www.personalmag.co.yu/php.htm) **(en)**
* [Obimni vodiči uglavnom namijenjeni početnicima](http://www.programiranje.net/vodici.php) **(bs)**
* [Par stvari o PHP-u](http://most-art.de/phptutorial/index.php) **(bs)**
* [Instalacija i konfiguracija IIS-a za rad sa PHP-om](http://www.beotel.rs/~steki/php_iis/index.html) **(sr)**
* [Zvanična PHP stranica](http://php.net/)**(en)**
* [Do You PHP? Rasmus Lerdorf](http://www.oracle.com/technology/pub/articles/php_experts/rasmus_php.html)**(en)**
* [PHP tutorijali na hrvatskom jeziku](http://www.dizajniranje.net/tutorials/?search=&cat=15)
* PHP (Hypertext preprocessor) je programski jezik raširen i popularan pri izradi web stranica. Izuzetno je jednostavan za učenje, a s naučenim se postiže mnogo.
* U trenutku kad korisnik posjeti vaše PHP stranice, vaš web server automatski obrađuje PHP kod na osnovu kojega određuje šta će prikazati korisniku.
* Sve ostalo kao npr. matematičke operacije, operacije s datotekama, varijable idr. ne prikazuje korisniku, te korisnik u Internet preglednik prima generiranu HTML stranicu bez PHP koda
* Večina sintakse je preuzeta iz C, Jave i Perla s nekonicinom jedinstvenih specifičnosti.  
  Sami smisao jezika je brzo i jednostavno pisanje dinamički generiranih stranica.
* Da biste koristili vašu PHP skriptu ili stranicu s PHP kodom, vaš web server mora podržavati PHP.   
  Osim nekih besplatnih, večina komercijalnih servera podržava PHP.
* Ako želite isprobavati PHP skripte na vašem računalu postoji i za to riješenje: download i instalacija programa [Xammp](http://www.apachefriends.org/en/xampp-windows.html)

## Osnove PHP-a

Šta je to sintaksa? Laički rečeno sintaksa predstavlja skup pravila koja svaki kod mora zadovoljavati.

PHP kod mora biti unutar ključnih riječi (tagova)

<?php  
/\*.....Ovdje dolazi kod....\*/  
?>

Ili skraćeno

<?  
/\*.....Ovdje dolazi kod....\*/  
?>

Svaka kodna naredba, definicija varijable mora se zaključiti s ;

Kako snimiti vaše PHP stranice

Ako želite umetati PHP kod u vaše HTML stranice i želite da server tj. internet preglednik ispravno interpretira vašu stranicu, morate umijesto standardne .html extenzije koristiti .php extenziju za vaše stranice   
npr. umijesto index.html treba biti index.php

Varijable

Varijabla služi za privremeno pohranjivanje određene vrijednosti (broja, teksta...), vrijednost pohranjena u varijabli se nakon toga može koristiti više puta kroz cijeli kod.

Varijable se definiraju na slijedeći način: $naziv\_varijable = vrijednost;

Znak dolar ispred naziva je obavezan. Napomena imena varijabli su osjetljiva na velika i mala slova npr. $naziv i $Naziv su dvije različite varijable.

Pravila koja morate poštivati prilikom imenovanja varijabli su slijedeći:

PHP varijable moraju započinjati s slovom ili "\_" znakom.

U imenu mogu imati samo znakove a-z, A-Z, 0-9, ili \_

Varijable s više od jedne riječi u nazivu radi preglednosti treba razdvajati s \_.   
Npr $moja\_varijabla   
Isto tako ih je u redu razdvajati koristeći velika ili mala slova. Npr $mojaVarijabla

http://phptutorial.odlican.net/osnovephpa.php