

OGLAVLJE 1

Osnove PHP-a

Dobrodošli u PHP! Zvanično je PHP skraćenica za PHP: Hypertext Preprocessor, ali ga milioni ljudi i dalje znaju po originalnom nazivu - Personal Home Page. To je i suština ovog jezika. U pitanju je kreiranje interaktivnih Web strana na najlakši mogući način. Web strane više ne moraju da budu statičke i nepromenljive. Sada možete da korisnicima šaljete nove Web strane prilagođene onome što oni žele da vide. Sve se to dešava u realnom vremenu. Možete da rukujete dugmadima, poljima za potvrdu i radio dugmadima, a možete čak i da interaktivno crtate i šaljete podatke u bazu podataka. Sve je na Vama, treba samo otvoriti kutiju.

Ovo je mesto na kome će Vaše Web strane postati "žive".

Radićemo sa verzijom 5 PHP-a. Web sajt je www.php.net. PHP je specijalno projektovan za kreiranje Web strana u vrlo kratkom roku. U ovoj knjizi ćemo PHP koristiti za programe na serveru. PHP skriptove ćemo instalirati na serverskoj mašini. Korisnici će te skriptove moći da otvaraju u Web pretraživačima i moći će da vide sve što se očekuje od pravih Web strana. Tu se misli na polja za unos teksta, tabele sa podacima koji su "u letu" dobijeni iz baze podataka, kao i pokretnu grafiku, odnosno na sve ono što većina savremenih, profesionalnih Web strana danas ima.

Kod Web strana koje su napisane u običnom HTML jeziku Web server samo prosleđuje HTML koji se nalazi na strani nazad do pretraživača. Korisnik tako može da vidi slike i tekst, ali to je sve. Kada u PHP-u napišete svoj skript, Vi, u stvari, predočavate serveru šta želite da se desi, bilo da je reč o čitanju teksta koji je korisnik uneo u polje za tekst, bilo da želite da pogledate koja je polja potvrdio. Nakon toga, možete da se opredelite šta želite da se sledeće desi, tako da možete da "u letu" napravite Web stranu koja se šalje nazad do pretraživača.

PHP je postigao veliki uspeh. Danas ga koristi više od 15 miliona Web strana. Na sledećim stranama ćete videti kako Web server može da se natera da sa PHP-om radi ono što Vi želite.

Početak rada sa PHP-om

Prvi korak u kreiranju interaktivnih Web strana je da se dobije pristup do Web servera na kome PHP radi. Vaš Internet Service Provider (ISP) verovatno već podržava PHP. To možete proveriti ako pitate osoblje za podršku koje radi za tog ISP provajdera ili možete da sami isprobate da li to radi. Postoje dva načina da sami probate podršku za PHP.

Prvi način je da, ako možete, otvorite prozor za unos komandi i povežete se sa Web serverom preko nekog programa, kao što su Telnet, SSH ili SSH2 (nemojte brinuti ako ne znate kakve su to aplikacije, jer Vam u ovoj knjizi nije ni potrebno), otkucate `php -v` na komandnoj liniji (u knjizi ćemo kao oznaku za unos komande sa komandne linije koristiti znak za %). Ako je PHP instaliran, rezultat treba da izgleda ovako:

```
%php -v
PHP 5.0.0 (cli) (built: Jul 13 2004 21:39:58)
Copyright (c) 1997-2004 The PHP Group
Zend Engine v2.0.0, Copyright (c) 1998-2004 Zend Technologies
```

Drugi način da odredite da li je PHP instaliran je da probate neki PHP skript. To možete da uradite ako pročitate odeljak "Kreiranje prvog PHP skripta" u ovom poglavlju. Ako skript radi, PHP je instaliran i sve je podešeno.

NAPOMENA

Ako želite da vidite listu ISP provajdera koji podržavaju PHP, pogledajte adresu www.php.net/links.php#hosts..

Lokalna instalacija PHP-a

Nije loše da na svom računaru instalirate PHP tako da prilikom razvoja možete da isprobate svoje skriptove. U tom slučaju nećete morati da trošite vreme na prebacivanje skriptova na računar provajdera, da ih onda isprobate preuzimanjem u svom pretraživaču, pa da u slučaju grešaka sve radite iz početka.

Neki operativni sistemi, kao što su Linux i mnoge verzije Unixa, sada se isporučuju sa već instaliranim PHP-om. Kod drugih operativnih sistema, kao što su Windows ili Mac OSX, moraćete da sami preuzmete i instalirate PHP. Prvi korak je da proverite da li ste instalirali PHP u lokalnu. Primenite komandu `php -v` (ovo se radi u prozoru za unos komandi - na primer, na Windowsu u DOS prozoru). Ako radi, onda je već sve podešeno.

PHP možete sami instalirati. Postoje unapred pripremljene verzije za različite operativne sisteme, koje se mogu preuzeti. Postoje verzije za operativne sisteme Windows, Mac OSX, Novell NetWare, OS/2, RISC OS, SGI IRIX 6.5x i AS/400. Binarni paket sa instalacijom za Windows možete pronaći na adresi

POGLAVLJE 1 Osnove PHP-a

<http://www.php.net/downloads.php>. Tu su i linkovi na verzije za druge operativne sisteme.

NAPOMENA

Binarne verzije za Linux i Unix se više ne distribuiraju, pošto je PHP kod njih već instaliran. Ako želite, možete da od izvornog koda napravite svoju instalaciju PHP-a. Idite na adresu <http://www.php.net/downloads.php> i preuzmite izvorni kod za PHP.

Instrukcije za instalaciju PHP možete pronaći u dokumentaciji za PHP, koja se nalazi na adresi <http://www.php.net/docs.php> (kompletnu dokumentaciju možete da pruzmete sa adrese <http://www.php.net/download-docs.php>). Instrukcije za instalaciju naći ćete i u datoteci za instalaciju (obično datoteka pod nazivom `install.txt`), koja se dobija kada se dekompresuje ono što ste preuzeli. Pošto se instrukcije menjaju svaki put kada se promeni PHP i pošto postoji puno operativnih sistema i Web servera, trebalo bi da koristite najnovije instrukcije i da po njima postupate. Prikazivanje svih instrukcija ovde bi oduzelo 20 strana, a one bi ionako zastarele do trenutka kada dođu do Vas.

Evo kako bi "stvari" mogle da izgledaju za Windows XP (detaljnije informacije se mogu naći u instrukcijama za instalaciju koje skidate sa Interneta). Prvo Vam je potreban Web server, kao što su Apache ili Microsoftov Internet Information Server (IIS). Program za instalaciju Apache servera na Windows ((`apache_2.0.52-win32-x86-no_ssl.msi`) može se preuzeti na adresi <http://httpd.apache.org/download.cgi>. Kada ovaj fajl preuzmete i dva puta ga brzo kliknete, instaliraće se Apache. IIS možete da na Windows XP instalirate ako iz Control Panela izaberete dugme Add/Remove Programs, izaberete dugme Add/Remove Windows Components, a onda izaberete IIS.

PHP se na Windows može instalirati na dva načina. Ono što je potrebno za oba načina se može preuzeti sa adrese <http://www.php.net/downloads.php>. Postoji jedna posebna izvršna datoteka za instalaciju na operativnom sistemu Windows, `php-5.0.x-installer.exe`, sa osnovnom podrškom za PHP (tu su standardna proširenja, ali ne i od drugih proizvođača). Ova datoteka automatski konfiguriše servere kao što su IIS, PWS i Xitami, a tu su i instrukcije za ručno podešavanje drugih servera, kao što je Apache.

Drugi način instalacije je pomoću `-zip` datoteke. Reč je o datoteci `php-5.0.x-Win32.zip`, koja sadrži punu instalaciju PHP-a, zajedno sa spošaljnijim proširenjima (ova opcija je bolja ako želite da pokrijete sve što postoji u ovoj knjizi). Preuzmite i raspakujte ovu datoteku. U njoj se nalaze i instrukcije za instalaciju. U zavisnosti od operativnog sistema koji koristite, obično se datoteka `php.exe` kopira u određenu fasciklu.

Naredni korak je da instalaciju PHP-a povežete sa Web serverom. Ako ste raspakovali `-zip` datoteku na operativni sistem Windows XP i ako koristite Apache, trebalo bi da promenite i datoteku `httpd.conf`. Ovo je konfiguraciona datoteka Apache servera. Ako koristite IIS, njega možete konfigurisati koristeći konzolu za upravljanje (izaberite Start → Settings → Control Panel → Administrative Tools → Internet Services Manager) i pratite instrukcije za instalaciju (više detalja možete naći u instrukcijama za instalaciju - u njima je detaljno "pokriven" svaki korak).

Podlašavanje razvojnog okruženja

Da biste mogli da pravite PHP strane, potreban Vam je neki editor teksta. PHP datoteke su mešavina HTML-a i PHP koda. Postoje različiti editori, koji se koriste na različitim operativnim sistemima, kao što su, na primer, vi, emacs, pico, BBEedit ili SimpleText za Macintosh ili Notepad i WordPad za Windows. Podrazumevana ekstenzija PHP datoteka je .php (na primer, myBigTimeWebPage.php).

Tekst se unosi kao običan tekst, ali predstavlja mešavinu HTML-a i PHP-a. Kao što se vidi na slici 1.1, da biste napravili PHP stranu, morate da prvo unesete tekst prema slici i da ga upamtite u datoteci sa ekstenzijom .php. Ovaj primer je uglavnom HTML kod. PHP deo je skript koji se nalazi između oznaka `<?php i ?>`. Kada ovaj dokument pogledate u svom pretraživaču, server koji podržava PHP čita dokument, pronalazi PHP deo i automatski ga izvršava. U ovom slučaju PHP kod je samo jedna linija `phpinfo()`, koja će prikazati HTML tabelu sa informacijama o instalaciji PHP-a na serveru, kao što ste videli u primeru "Kreiranje prvog PHP skripta".

```

<HTML>
  <HEAD>
    <TITLE>
      Mixing HTML and PHP!
    </TITLE>
  </HEAD>

  <BODY>
    <H1>
      Mixing HTML and PHP!
    </H1>
    <?php
      phpinfo();
    ?>
  </BODY>
</HTML>
  
```

Slika 1.1
Kreiranje PHP strane sa HTML kodom u njoj

POGLAVLJE 1 Osnove PHP-a

NAPOMENA

WordPad editor u Windowsu ima dosadnu naviku da nazivu datoteke dodaje ekstenziju .txt. Ovo se dešava u slučaju da ne prepoznaje ekstenziju. Ako tekstualni dokument pokušate da upamtite sa ekstenzijom .PHP, WordPAD će napraviti ekstenziju .PHP.txt. Ovo možete da izbegnete ako u trenutku zapisivanja datoteke naziv stavite pod dvostruke navodnike, na primer "logbook.php". Takođe treba da imate na umu da WordPAD podrazumevano datoteke pamti u rtf ili doc (Microsoft Word) formatu, u zavisnosti od verzije Windowsa koju koristite. Da biste bili sigurni da su Vaši XML dokumenti u običnom tekstualnom formatu, iz padajuće liste Save as type, koja se nalazi u okviru za dijalog Save As, izaberite opciju Text Document.

Čak možete i da programe za obradu teksta, kao što je Microsoft Word, koristite za kreiranje PHP dokumenata. Potrebno je samo da datoteku zapamtite kao običnu tekstualnu datoteku, a ne u nekom drugom formatu, kao što je .doc (na primer, u Wordu možete da iz padajuće liste Save as type, u okviru za dijalog Save As, izaberete opciju Text Only). Ipak je mnogo bolje da radite sa običnim editorom teksta, a ne sa programom za obradu teksta, pošto programi za obradu teksta dodaju neke karakteristike i dekoraciju koji nisu potrebni.

Za kreiranje PHP datoteka možete da koristite i neko integrisano razvojno okruženje za PHP (IDE). U ovakvim okruženjima imate na raspolaganju alate kojih nema kod običnih tekstualnih editora. Tu mislim na automatsku proveru onog što ste napisali da biste bili sigurni da je PHP kod ispravan i automatsko isticanje sintaktičkih elemenata (na primer, ključne reči PHP-a se prikazuju u drugoj boji, tako da lako možete da vidite šta ste uradili). Tu je i automatsko postavljanje, jer IDE može da prebaci Vaše PHP strane kod Vašeg ISP provajdera ako kliknete odgovarajuće dugme ili iz menija izaberete odgovarajuću stavku.

Evo liste IDE okruženja koja podržavaju PHP, a koja možete preuzeti sa Interneta. Treba da imate na umu da većina ovih okruženja nije besplatna.

- Komodo (www.activestate.com/Products/Komodo) radi na Windowsu i Linuxu.
- Maguma (www.maguma.com) radi samo na Windowsu.
- PHPEdit (www.phpedit.com/products/PHPEdit) besplatan je, ali radi samo na Windowsu.
- Zend Studio (www.zend.com/store/products/zend-studio.php) radi na Windowsu i Linuxu. Ovo je okruženje koje su napravili isti ljudi koji su napravili Zend softver mašinu, koja radi kao jezgro samog PHP-a.

Ako radite sa nekim ISP provajderom, potreban Vam je i neki način za transport PHP strana na njegov računar. Isto je važi i za obične HTML strane. Možete koristiti FTP (File Transfer Protocol) program ili Web interface, ako takav postoji. Ako ranije niste prebacivali Web strane na računar provajdera, konsultujte se sa provajderovim osobljem za podršku. Ti ljudi će Vas verovatno uputiti na neki FTP program ili Web interfejs koji su napravili. PHP strane se prebacuju i pokreću isto kao i obične HTML strane, naravno ako server podržava PHP.

Kreiranje prvog PHP skripta

Sve počinje kreiranjem PHP skripta. Sa PHP-om koji radi na serveru možete da radite mnogo štošta što ranije niste mogli: da vadite podatke iz baze podataka, da proverite nečiju lozinku, da odštampate tekst dobrodošlice, da koristite "kolačice", da pravite knjigu posetilaca, da pravite interaktivne strane, da računate porez, ili čak da pravite svoje aplikacije za rad sa potrošačkom korpom. Sve to može da radi nezavisno 24 časa dnevno, čak i kada Vi mirno spavate u svom krevetu.

Vaša prva PHP strana će biti jednostavna, sa jednostavnim PHP skriptom. Na istoj Web strani možete da mešate PHP i HTML, ali morate da imate na raspolaganju neki način da ih držite odvojeno. Ovo se radi tako što se PHP kod piše u okviru specijalnih tagova `<?php i ?>`:

```
<?php
.
. Your PHP goes here....
.
?>
```

NAPOMENA

Ovo se još može skratiti ako u datoteci `php.ini` uključite opciju `short tags` (skraćene oznake). To je datoteka koju PHP koristi za konfiguraciju. U tom slučaju možete da koristite samo oznake `<? i ?>`. Ja to ne bih preporučio, pošto će Vaši skriptovi često biti upotrebljavani i na drugim serverima, pored onog koji ste Vi podesili, tako da može doći do konflikta sa drugim skript jezicima.

Unutar ovih oznaka se nalaze PHP skriptovi, koji se sastoje od PHP iskaza, koji završavaju znakom tačka-zarez. Taj znak je bitan, pošto on "saopštava" PHP-u da je tekući red završen. Možda ste radili sa drugim skript jezicima, kao što je JavaScript, kod kojih je upotreba znaka tačka-zarez opcionalna, kod PHP-a ona je obavezna.

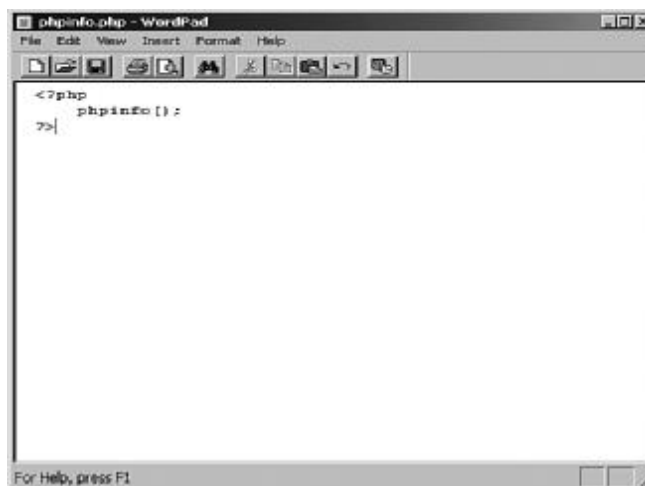
Naš prvi PHP skript je bio standardan. Koristili samo samo jedan iskaz - `phpinfo()`. Kao što ćete videti, PHP iskazi mogu da rade različite stvari. Ovaj koristi funkciju `phpinfo`, koja prikazuje informacije o verziji PHP-a koja se koristi. Kada takvu funkciju pozovete po nazivu, izvršavaju se svi iskazi koji se u njoj nalaze. Funkcija `phpinfo` je jedna od mnogih funkcija koje su ugrađene u PHP i koje su spremne za upotrebu. Funkcije mogu da sadrže stotine redova koda, a sve što mi treba da uradimo je da tu funkciju pozovemo preko njenog naziva.

POGLAVLJE 1 Osnove PHP-a

U ovom slučaju će funkcija `phpinfo` kreirati HTML tabelu u kojoj se nalaze informacije o Vašoj instalaciji PHP-a. Evo kako se funkcija poziva preko jednog iskaza:

```
<?php  
    phpinfo();  
?>
```

Unesite ovaj skript u svom editoru teksta, kao što je pokazano za slučaj WordPada na slici 1.2, i upamtite datoteku pod nazivom `phpinfo.php`. Nemojte zaboraviti da datoteku upamtite kao običan tekst. Ako ona nije upamćena na taj način, PHP će imati problema sa njom (jedan od načina da to proverite je da pogledate kompletan naziv datoteke u DOS prozoru pod Windowsom).



Slika 1.2
Kreiranje prvog PHP skripta

Čestitamo! Upravo ste kreirali svoj prvi PHP skript. Nije loše. Nakon što ste kreirali datoteku `phpinfo.php`, sledeći korak je da je upamtite na Web serveru, na mestu dakle server može da je pročita. Ako radite sa nekim ISP provajderom, prebacite datoteku `phpinfo.php` na mesto gde se smeštaju standardne Web strane. To ćete uraditi isto kao što prebacujete i obične HTML strane pomoću FTP programa ili Web interfejsa.

Ako radite u lokalu i ako ste na svom računaru instalirali PHP i Web server, datoteku `phpinfo.php` stavite na mesto gde server može da je pronađe. Kod Apache servera to je fascikla `htdocs`, koja se nalazi ispod fascikle na kojoj je instaliran Apache. Kod IIS servera to je fascikla `inetpub/wwwroot`. Na Linuxu to može biti fascikla `/var/www/html`. Nakon što ste proverili da li je Vaš Web server konfigurisan da radi sa PHP-om (prema instrukcijama za instalaciju koje ste preuzeli sa sajta www.php.net), pokrenite Web server.

PHP 5

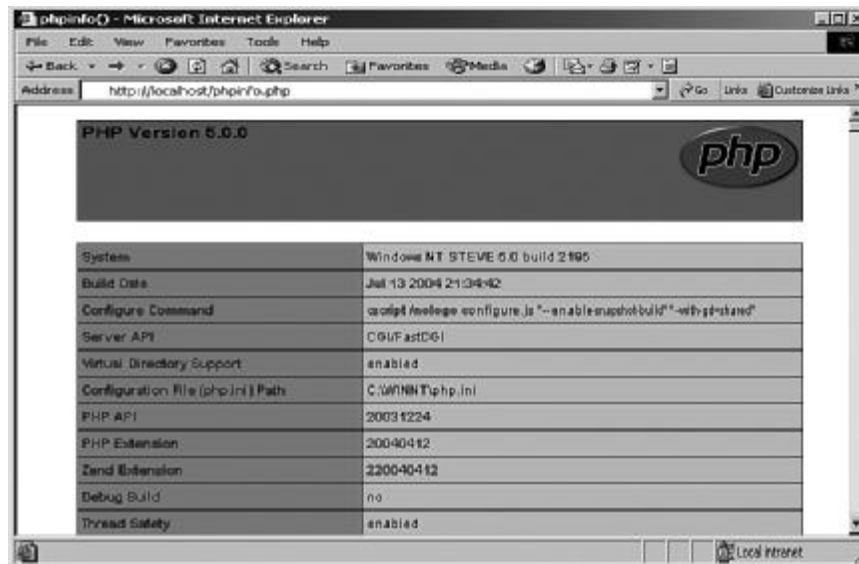
Izvršenje prvog PHP skripta

Da biste izvršili datoteku `phpinfo.php`, treba samo da je otvorite u svom pretraživaču, kao i bilo koju drugu Web stranu. To znači da je potrebno da pretraživač usmerite na URL adresu za `phpinfo.php` (na primer `http://www.vasisp.com/korisnickoime/phpinfo.php`). Ako PHP koristite lokalno, potražite adresu `http://localhost/phpinfo.php`.

NAPOMENA

Nemojte da datoteku `phpinfo.php` otvarate direktno u pretraživaču preko menija File → Open, pošto će se time ona otvoriti bez prolaza preko Web severa, tako da ni PHP skript neće raditi. Treba da budete sigurni da ste uneli pravi URL.

Ako je sve u redu, trebalo bi da se prikaže tabela sa informacijama o PHP-u, slično slici 1.3. Čestitamo - postali ste PHP programer! Ovde je prikazano više tabela. Ako krenete naniže, možete da pronađete informacije o Vašoj konfiguraciji PHP-a. Ako ste se ikad pitali koja je verzija PHP-a kod Vas instalirana, sada je trenutak da to proverite.



Slika 1.3

Rezultat prvog PHP skripta

Šta učiniti ako ovo ne radi?

Nažalost, štošta može da "pođe naopako" kada prvi put pokušavate da pokrenete PHP. Ako ne ide sve po planu, nemojte paničiti. Potrebno je samo da utrošite malo dodatnog vremena. Možda ste dobili praznu stranu, ili poruku da datoteka nije pronađena, ili se pojavila neka druga greška. Bez obzira šta se desilo, problem se može rešiti.

Prvo što treba da proverite, ako PHP radi u lokalnu, je da vidite da li je PHP podignut. Na komandnoj liniji u fascikli sa instalacijom PHP-a i otkucajte `php -v`. Ako se prikaže verzija PHP-a, onda on radi. Ako možete da komandnu liniju otvorite i na računaru svog ISP provajdera, na primer, preko Telnet-a ili SSH2 aplikacije, možete da izvršite isti test.

Sledeće što treba da proverite je da li je PHP instaliran na pravi način. Ovo je problem koji dovodi do prikazivanja prazne strane i, kada u svom pretraživaču izaberete opciju view source, moći ćete da vidite PHP skript. To znači da Web server nije prosledio skript do PHP-a koji treba da ga izvrši. Ovo može biti problematično i to je razlog što su instrukcije na sajtu www.php.net tako obimne. Najbolja ideja je da sve pročitate još jednom, red po red, i tako proverite da li ste uradili sve što je tamo navedeno.

Nakon toga, treba da proverite da li je datoteka `phpinfo.php` tamo gde Web server očekuje da bude. Kao što sam pomenuo, kod Apache servera lokacija je fascikla `htdocs`, koja se nalazi u fascikli gde je Apache instaliran. Kod IIS-a to je fascikla `inetpub/wwwroot`. Na Linuxu to može biti `/var/www/html`. Prava fascikla može biti različita u zavisnosti od servera koji se koristi. Na PH serveru koji ja koristim prava fascikla je `/httpdocs/ROOT`. Ako ste `phpinfo.php` prebacili na uobičajeni direktorijum na računaru provajdera (tamo gde se stavljaju HTML strane) i ako to ne radi, pitajte ISP provajdera. Ponekad je potrebno da on definiše podršku za određene fascikle. Neki ISP provajderi traže da za svoje PHP skriptove koristite drugu ekstenziju, kao što je, na primer, `.php5`, u slučaju verzije PHP5.

NAPOMENA

Da li za PHP skriptove morate da imate specijalna podešavanja pristupa u slučaju da koristite Unix? Moraju li datoteke da budu definisane kao izvršne? Ne, jednostavna podešavanja 644 (ne 755) će biti dovoljna.

Ako koristite IIS, treba da u datoteci `php.ini` proverite da li stoji `cgi.force_redirect = 0`. Prema instrukcijama za instalaciju podešavanje se mora izvršiti na ovaj način. Ako tog reda nema u ovoj datoteci, dodajte ga.

Na kraju možete i da pogledate odeljak "Problemi" u instrukcijama za instalaciju. PHP FAQ (najčešće postavljena pitanja) na adresi www.php.net/FAQ.php mogu da pomognu pri rešavanju problema, a isto važi i za pitanja na adresi www.php.net/manual/faq.installation. Možete da proverite i u arhivama za instalaciju PHP-a, na adresi <http://marc.theaimsgroup.com/?l=php-install/r=1/w=2> ili u novostima na adresi `alt.php` ili `comp.lang.php`.

Mešanje sa HTML kodom

PHP datoteke najčešće predstavljaju mešavinu HTML-a i PHP skriptova. Postavljanje HTML-a i PHP skriptova u isti dokument nije problem, pošto se PHP iskazi nalaze između oznaka `<?php i ?>`, što znači da server može lako da ih prepozna.

Već znate kako HTML radi na Web stranama. Ako, na primer, ubacite neki HTML kod koji prikazuje tekst, tekst se prikazuje kada Web server stigne do reda na strani u kome se nalazi taj tekst. Slično tome, ako PHP skript kreira neki tekst koji treba da se ubaci na Web stranu i pošalje nazad do pretraživača, tekst se na stranu ubacuje na mestu gde je bio skript.

Pogledajte primer 1.1, `phphtml.php`. U datoteci se nalazi mešavina HTML-a i PHP-a. Kao što vidite, HTML definiše naslov dokumenta (koji se prikazuje u zaglavlju pretraživača), pri čemu se preko oznake `<H1>` prikazuje tekst `Mixing HTML and PHP`, i to velikim masnim slovima.

Primer 1.1 Mešanje PHP-a i HTML-a

```
<HTML>
  <HEAD>
    <TITLE>
      Mixing HTML and PHP!
    </TITLE>
  </HEAD>

  <BODY>
    <H1>
      Mixing HTML and PHP!
    </H1>
    <?php
      phpinfo();
    ?>
  </BODY>
</HTML>
```

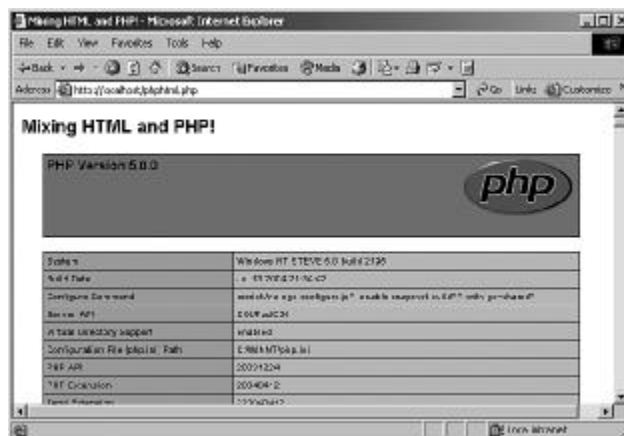
Nakon HTML elementa `<H1>` server nailazi na PHP skript, što znači da se izlaz iz funkcije `phpinfo` ubacuje na Web stranu, koja se šalje nazad do pretraživača. Izlaz se ubacuje baš na tom mestu. Kao što znate, funkcija `phpinfo` kreira HTML tabelu sa informacijama o instalaciji PHP-a, tako da rezultat treba da izgleda kao na slici 1.1. Prvo se prikazuje HTML zaglavlje, a onda sledi tabela sa informacijama o PHP-u.

Želite da PHP strana izgleda više zvanično? Na adresi "<http://www.php.net/download-logos.php>" se nalazi više logoa tipa "Rađeno u PHP-u". Samo ih preuzmite i ubacite na Web stranu pomoću elementa ``, kao u sledećem primeru:

POGLAVLJE 1 Osnove PHP-a

```
<HTML>
<HEAD>
  <TITLE>
    Mixing HTML and PHP!
  </TITLE>
</HEAD>

<BODY>
  <H1>
    Mixing HTML and PHP!
  </H1>
  <?php
    phpinfo();
  ?>
  <IMG SRC="php-power-white.gif">
</BODY>
</HTML>
```



Slika 1.4
Mešanje PHP-a i HTML-a

Na slici 1.5 možete videti primer sa skinutom slikom `php-power-white.gif`, koja se prikazuje na Web strani.



Slika 1.5
Ubacivanje PHP logoa

Kako ćete odštampati neki tekst?

Iskaz echo, koji na Web stranu ubacuje tekst, predstavlja jedan od iskaza koji se u PHP-u najviše koriste. Mi ćemo taj iskaz koristiti za prikazivanje nečeg što je urađeno u PHP-u. Počecemo odmah.

Iskaz echo se koristi jednostavno pomoću ključne reši echo, iza čega pod navodnicima sledi tekst koji treba da se prikaže. U sledećem skriptu možete videti kako se prikazuje tekst "Hello from PHP".

```
<HTML>
  <HEAD>
    <TITLE>
      Using the echo statement
    </TITLE>
  </HEAD>

  <BODY>
    <H1>
      Echoing some text:
    </H1>
    <?php
      echo "Hello from PHP.";
    ?>
    .
    .
    .
```

Pošto na HTML strani možete da imate više razbacanih skriptova, tekst možete da prikazujete na više mesta. To smo pokazali na primeru 1.2, echo.php.

Primer 1.2 rad sa iskazom echo

```
<HTML>
  <HEAD>
    <TITLE>
      Using the echo statement
    </TITLE>
  </HEAD>

  <BODY>
    <H1>
      Echoing some text:
    </H1>
    <?php
      echo "Hello from PHP.";
    ?>
    <H1>
      Echoing some more text:
```

POGLAVLJE 1 Osnove PHP-a

```
</H1>
<?php
    echo "Hello from PHP again!";
?>
<IMG SRC="php-power-white.gif">
</BODY>
</HTML>
```

Rezultat je prikazan na slici 1.6.



Slika 1.6

Upotreba iskaza echo

Tekst koji želite da prikazete možete da stavite pod jednostruke ili dvostruke navodnike. Preko ovog iskaza možete da prikazujete brojeve bez navodnika. To izgleda ovako:

```
Echo 111555;
```

Pošto ono za šta koristimo iskaz echo ide na Web stranu (osim ako PHP pozivate sa komandne linije), može biti korisno da na taj način šaljete i HTML oznake nazad do pretraživača. Na primer, ako želite da se svaka reč pojavi u drugom redu, možete da HTML iskaz `
` upotrebite na sledeći način:

```
echo "Hello<BR>from<BR>PHP.";
```

Dodatna snaga kod štampanja

Između prikazivanja teksta na komandnoj liniji i u pretraživaču postoji razlika. U pretraživaču možete da koristite HTML elemente, kao što su
 ili <P> i da tako formatirate svoj tekst. Kada tekst štamplate sa komandne linije, za formatiranje možete da koristite specijalne karaktere (ako tekst stavite pod dvostruke navodnike). Evo koji su karakteri specijalni:

<code>\n</code>	Novi red
<code>\r</code>	Return
<code>\t</code>	Tab
<code>\\</code>	Prikazuje se \
<code>\\$</code>	Prikazuje se \$.
<code>\"</code>	Prikazuje se ".
<code>\0 do \77</code>	Prikazuje se karakter koji odgovara heksadecimalnom (sa osnovnom 8) kodu.
<code>\x0 do \xFF</code>	Prikazuje se karakter koji odgovara heksadecimalnom (sa osnovom 16) kodu.

Na primer, iskaz `echo "Line 1\nLine 2"` prikazuje tekst "Line 1" u jednom redu i "Line 2" u sledećem (ako PHP radi na komandnoj liniji). U pretraživaču karakter `\n` ne znači ništa. Isti rezultat biste dobili ako biste napisali `"Line 1
 Line 2"`.

NAPOMENA

Ako tekst prikazujete u pretraživaču, morate ga formatirati pomoću HTML oznaka. Ako u tekst stavite nove redove, neće se ništa desiti, pošto pretraživač zanemaruje te nove redove, kao i na bilo kojoj Web strani. Ako želite da tekst formatirate na neki način, morate koristiti PHP skript i u pretraživač slati ispravan HTML kod.

Ako želite, možete da dugačak string koji je pod navodnicima u Vašem skriptu razložite na nekoliko redova. Ti novi redovi će se prikazati ako štamplate sa komandne linije. Ako štamplate sa Web strane, prekidi redova će se zanemariti.

```
<?php
echo "This text
spans
multiple
lines.";
?>
```

Stavke koje štamplate možete i da odvojite zarezima. Evo kako se to radi:

```
echo "Hello", "this", "is", "PHP.";
```

POGLAVLJE 1 Osnove PHP-a

Sve stavke koje ste na ovaj način odštampali štampaju se jedna iza druge:

```
HellothisisPHP.
```

Ako želite da između reči ubacite razmake, možete da uradite sledeće:

```
echo "Hello ", "this ", "is ", "PHP.";
```

Tada ćete dobiti:

```
Hello this is PHP.
```

Ako želite da odštampate neki specijalni karakter (kao što je `"`), a da pri tome ne naredite PHP-u da prekine tekst (pošto je `"` oznaka za kraj teksta), možete da koristite `\`. Evo kako se to radi:

```
echo "He said, \"I like ice cream.\"";
```

Ovo se naziva izvlačenje (escape) navodnika, tako da PHP prikazuje njega, a ne tretira ga kao oznaku kraja teksta.

Više posebnih nizova karaktera možete da u PHP-u spojite pomoću tačke (`.`). Evo primera:

```
echo "Hello " . "this " . "is " . "PHP.";
```

U ovom slučaju PHP uzima izraz `"Hello " . "this " . "is " . "PHP."` i sklapa reči zajedno (ovo se naziva konkatencija) u jedan niz karaktera. Taj string se nakon toga prosleđuje iskazu `echo`.

Evo nekoliko primera koji pokazuju kako se koristi iskaz `echo`:

<pre>echo 11115555;</pre>	prikazuje: 11115555
<pre>echo "Hello from PHP.";</pre>	prikazuje: Hello from PHP.
<pre>echo 'Hello from PHP.';</pre>	prikazuje: Hello from PHP.
<pre>echo "Hello", "from", "PHP.";</pre>	prikazuje: HellofromPHP.
<pre>echo "Hello " . "from " . "PHP.";</pre>	prikazuje: Hello from PHP.

Pored iskaza `echo`, možete da koristite i PHP iskaz `print`. On ima istu sintaksu: `print "Hello from PHP."`. Koja je razlika između iskaza `echo` i `print`? Nije velika. `print` je više PHP funkcija (pogledajte Poglavlje 4 o funkcijama), tako da on vraća vrednost koja je uvek 1. Kao i kod drugih funkcija, i ovde možete da tu vrednost pročitate, ali u ovom slučaju sa njom ne možete mnogo da radite. U većini praktičnih primera `echo` i `print` rade isto, tako da možete da koristite bilo koji od njih.

Štampanje dokumenata na licu mesta

Evo još jedne opcije za štampanje. Možete da kreirate dokumente na licu mesta, koji štampaju sve što čitaju, sve dok ne dođu do tokena za kraj (obično `END`). Ovaj token mora biti postavljen na početku poslednjeg reda. U primeru 1.3 je pokazano kako to izgleda:

PHP 5

Primer 1.3 Štampanje dokumenta na licu mesta

```
<HTML>
  <HEAD>
    <TITLE>
      Displaying here Documents
    </TITLE>
  </HEAD>

  <BODY>
    <H1>
      Displaying here Documents
    </H1>

    <?php
    echo <<<END
    This example uses
    "here document" syntax to display all
    the text until the ending token is reached.
    END;
    ?>

    <BR>
    <BR>
    <IMG SRC="php-power-white.gif">
  </BODY>
</HTML>
```

Rezultat ovog skripta je prikazan na slici 1.7.



Slika 1.7

Prikazivanje dokumenta na licu mesta

Izvršavanje PHP-a sa komandne linije

Pored toga što PHP možete da povežete sa Web serverom tako da taj server čita skriptove i izvršava ih, PHP može da se pokrene i sa komandne linije. To može biti dobro za testiranje pre nego što pređete na prebacivanje datoteka na računar ISP provajdera.

PHP je jezik koji se interpretira. To znači da u trenutku izvršenja PHP interpreter čita skript i interpretira svaki iskaz, konvertujući ga u kod koji računar može da izvrši. U verziji PHP 5.0 program koji interpretira komandnu liniju zove se Command Line Interpreter (CLI) i dato mu je ime `php`. Program koji radi na Web serveru zove se `php-cgi`.

PHP možete da sa komandne linije pozovete komandom `php`. Morate biti sigurni da računar može da pronađe `php`, što znači da morate biti sigurni da je `php` (fascikla bin PHP-a) u putanji na računaru na kojem `php` radi. Na primer, pod pretpostavkom da imate skript `echo.php`:

```
<?php
    echo "Hello from PHP.";
?>
```

Ako je `php` u Vašoj putanji, onda ovo možete da pokrenete sa komandne linije na sledeći način (% je oznaka za odziv na svim računarima, a u konkretnom primeru se podrazumeva da se nalazite u fascikli na kojoj je i program `echo.php`):

```
%php echo.php
```

Ako ovo funkcioniše, prikazaće se tekst

```
Hello from PHP.
```

Ako ne funkcioniše, možete da zadate tačnu lokaciju `php-a`, koji bi na Unixu ili Linuxu mogla biti:

```
$/usr/local/bin/php echo.php
```

A na Windowsu

```
$/usr/local/bin/php echo.php
```

Program CLI ima puno opcija koje se koriste na *komandnoj liniji*, koje možete koristiti za podešavanja njegovog rada. `Php` će Vam "reći" sve o opcijama koje postoje ako unesete `php -h`. Dobićete punu listu opcija:

PHP 5

```
%php -h
Usage:      php [options] [-f] <file> [--] [args...]
           php [options] -r <code> [--] [args...]
           php [options] [-B <begin_code>] -R <code> [-E <end_code>] [--] [args...]
           php [options] [-B <begin_code>] -F <file> [-E <end_code>] [--] [args...]
           php [options] -- [args...]

-a          Run interactively
-c <path>|<file> Look for php.ini file in this directory
-n          No php.ini file will be used
-d foo[=bar] Define INI entry foo with value 'bar'
-e          Generate extended information for debugger/profiler
-f <file>   Parse <file>.
-h          This help
-i          PHP information
-l          Syntax check only (lint)
-m          Show compiled in modules
-r <code>   Run PHP <code> without using script tags <?..?>
-B <begin_code> Run PHP <begin_code> before processing input lines
-R <code>   Run PHP <code> for every input line
-F <file>   Parse and execute <file> for every input line
-E <end_code> Run PHP <end_code> after processing all input lines
-H          Hide any passed arguments from external tools.
-s          Display colour syntax highlighted source.
-v          Version number
-w          Display source with stripped comments and whitespace.
-z <file>   Load Zend extension <file>.
```

Ako, na primer, želite da dobijete jednostavnu tekstualnu verziju informacija koje štampa funkcija `phpinfo`, možete koristiti opciju `-i`. To se radi ovako: `%php -i`. Već znate da verziju PHP-a koja se koristi možete dobiti pomoću opcije `-v` (primetićete da CLI na komandnoj liniji "govori" da se koristi CLI).

```
%php -v
PHP 5.0.0 (cli) (built: Jul 13 2004 21:39:58)
Copyright (c) 1997-2004 The PHP Group
Zend Engine v2.0.0, Copyright (c) 1998-2004 Zend Technologies
```

Na Linuxu i Unixu PHP skriptove možete koristiti tako što ćete uneti naziv skripta na komandnoj liniji. Ovo je moguće ako pomoću linije koja počinje sa `#!` ukažete na mesto gde se nalazi PHP (i skriptu treba da date dozvolu da se izvr{ava):

```
#! /usr/bin/php
<?php
    echo "Hello from PHP.";
?>
```

Komentari su skriptovima

Do sada smo na Web stranama koristili ili HTML ili PHP skriptove. Njih treba da čita računar. Postoji i komponenta koja je namenjena situaciji kada čitanje obavljaju ljudi. To su komentari.

Komentari su napomene koje dodajete na svoje PHP strane da biste nekom objasnili šta se dešava. Ovo je bitno zato što se, kada se na dugaćak i složen skript vratite posle duže vremena, verovatno nećete sećati šta se u njemu radi. Može se desiti i da svoj skript date nekom drugom. Tada su potrebni komentari. Pomoću komentara možete da opišete ponašanje skripta, tako da kasnije lako možete da shvatite o čemu je reč.

U PHP-u postoje tri tipa komentara. Prva vrsta omogućava da pišete komentare u više redova. Ovakvi komentari počinju sa `/*` i završavaju se sa `*/`. Evo kako to izgleda:

```
<?php
/* Start by displaying a
   message to the user */

    echo "Hello from PHP.";
?>
```

Možete i da svaku liniju posebno ogradite sa `/*` i `*/` i da tako napravite blok koji privlači više pažnje.

```
<?php
/* Start by displaying a */
/* message to the user */

    echo "Hello from PHP.";
?>
```

Ugnežđavanje komentara će "ugušiti" PHP, pa ga treba izbegavati. Sledeći kod ne bi trebalo koristiti:

```
<?php
/* Start by
   /* displaying a */
   message to the user */

    echo "Hello from PHP.";
?>
```

Ovo neće da funkcioniše pošto PHP traži oznaku za kraj komentara `*/`; kada je pronađe, pretpostavlja da je komentar završen. To predstavlja problem, pošto komentar nije stvarno završen, pa će PHP zaključiti da tamo gde očekuje PHP iskaze postoji neki običan tekst.

PHP 5

Drugi tip komentara su komentari u jednom redu. Oni sadrže tekst koji se nalazi samo u jednom redu. Za početak ovakvih komentara možete koristiti `//` ili `#`.

```
<?php
// Start by displaying a
# message to the user

    echo "Hello from PHP.";
?>
```

Ovakvi komentari su takođe korisni, pošto se mogu postaviti u liniju u kojoj postoji i kod. PHP će zanemariti sve što je iza oznake `#` ili `//`:

```
<?php
    echo "Hello from PHP.";           //Display a message
    echo "Hello from PHP again!";    #Display another message
?>
```

Ove komentare možete takođe koristiti za kreiranje blokova komentara:

```
<?php
// Start by displaying a
// message to the user

    echo "Hello from PHP.";
?>
```

Da biste napravili nešto što zaista privlači pažnju, možete da napišete:

```
<?php
#####
# Start by displaying a #
#/ message to the user #
#####

    echo "Hello from PHP.";
?>
```

U poslednje vreme preovlađuju komentari u jednom redu, pošto ih je lakše napisati, tako da ne morate da brinete o njihovom završetku. Komentari u više redova ipak i dalje imaju svoje mesto, pošto se često koriste na početku programa radi objašnjenja o čemu je u tom programu reč. Oni se takođe ponekad koriste kada se prave sopstvene funkcije, u kom slučaju komentar može da objasni čemu funkcija služi i kako se koristi.

Nije loše da u svom kodu koristite komentare koji razjašnjavaju šta se dešava. Sa druge strane, ako preterate, možete da zaklonite poentu, ali je, ipak, gore da komentara uopšte nema.

Rukovanje podacima: promenljive

Sve poruke koje smo do sada prikazivali bile su fiksne i nepromenljive.

```
<?php
    echo "Hello from PHP.";
?>
```

Ovim se samo prikazuje tekst i ništa više. Ako bi to bilo sve, PHP ne bi ni bio potreban - isti efekat bi mogao da se postigne korišćenjem HTML-a. Suština PHP-a je u dinamičkom rukovanju podacima. Za tu svrhu se koriste *promenljive*.

Promenljive su kontejneri podataka. Recimo, na primer, da prodajete šešire preko Weba i da želite da proverite ukupne zalihe u sva tri magacina, koji se nalaze u čikagu, Tokiju i Parizu. Da biste to uradili, morate da zajedno ubacite tri posebne vrednosti. PHP ima ugrađenu podršku za matematičke operacije nad podacima, uključujući i sabiranje. Prema tome, da biste sabirali vrednosti, možete da koristite operator `+`. Sledeći skript, na primer, štampa "I have 6 tomatoes" (Imam šest paradajza).

```
<?php
    echo "I have " , 1 + 2 + 3 , " tomatoes";
?>
```

Primitićete da smo ovde koristili brojeve, što je različito od običnog teksta. Pošto brojevi nisu tekst, nisu pod navodnicima. Rezultat je dobar, ali je još uvek statički, pošto smo u skript jednostavno stavili `1+2+3`. Kako ćemo sabrati brojeve koji pokazuju količine šešira u čikagu, Tokiju i Parizu?

To je trenutak kada "u igru" ulaze promenljive. U PHP-u one počinju oznakom za dolar `$`. Promenljive mogu da sadrže podatke; ako u promenljivim pod nazivima `$čikago`, `#tokyo` i `$paris` smestite broj šešira, evo kako biste mogli da to sabirate u vreme izvršenja:

```
<?php
    echo "I have " , $chicago + $tokyo + $paris , " party hats!";
?>
```

Ispravan naziv promenljive u PHP-u počinje slovom ili podvlakom, iza čega sledi proizvoljan broj slova, broja ili podvlaka. Naziv može imati proizvoljnu dužinu. Evo nekih ispravnih naziva promenljivih: `$pizza_temperature`, `$_number_of_tigers`, `$planet_number_9`.

Kao što vidite, promenljive se ponašaju kao skladište za podatke. Kako ćete te podatke smestiti u promenljive?

Dodeljivanje vrednosti promenljivim

Kada u PHP-u želite da kreirate promenljivu, podatke joj dodeljujete pomoću operatora dodele. Operatore dodele koji postoje u PHP-u ćemo detaljnije predstaviti u Poglavlju 2, a sada ćemo objasniti samo onaj koji se najviše koristi - operator jednako =. Evo jednog primera u kojem se koristi operator jednako za dodeljivanje vrednosti novim promenljivim (nakon što se ovaj program izvrši, promenljiva \$temperature će imati vrednost 69, vrednost \$pi iznosiće 3.14159265, itd).

```
$temperature = 69;
$number_of_earth = 1;
$pi = 3.1415926535;
$reassurance = "No worries.";
```

Primetićete da smo nekim promenljivim dodelili brojčane, a nekim tekstualne vrednosti. U pojedinim jezicima morate da zadate tip promenljive (na primer, string ili integer), ali u PHP-u to ne važi, pa je sve mnogo lakše.

NAPOMENA

Računar interno za skladištenje koristi tipove podataka, tako da treba da znate šta može da se desi ako podatke nepravilno unesete. Više detalja o tipovima podataka možete pronaći u odeljku "Rukovanje tipovima podataka" u ovom poglavlju.

Pogledajte primer `phpvariables.php`, koji smo dali u primeru 1.4. U ovom slučaju smo promenljivoj \$apples dodelili vrednost 1, a zatim smo prikazali vrednost koja je u toj promenljivoj.

```
echo "Setting number of apples to 1.<BR>";
$apples = 1;
echo "Number of apples: ", $apples, "<BR>";
.
.
.
```

Recimo da sada želite da povećate broj jabuka za tri. To možete uraditi ako vrednosti \$apples dodelite trenutnu vrednost \$apples, uvećanu za 3, nakon čega novi rezultat treba da se prikaže.

```
echo "Setting number of apples to 1.<BR>";
$apples = 1;
echo "Number of apples: ", $apples, "<BR>";
echo "Adding 3 more apples.<BR>";
$apples = $apples + 3;
echo "Number of apples now: ", $apples, "<BR>";
```

POGLAVLJE 1 Osnove PHP-a

Primer 1.4 Dodeljivanje vrednosti promenljivim

```
<HTML>
  <HEAD>
    <TITLE>
      Assigning values to variables
    </TITLE>
  </HEAD>
  <BODY>
    <H1>
      Assigning values to variables
    </H1>
    <?php
      echo "Setting number of apples to 1.<BR>";

      $apples = 1;

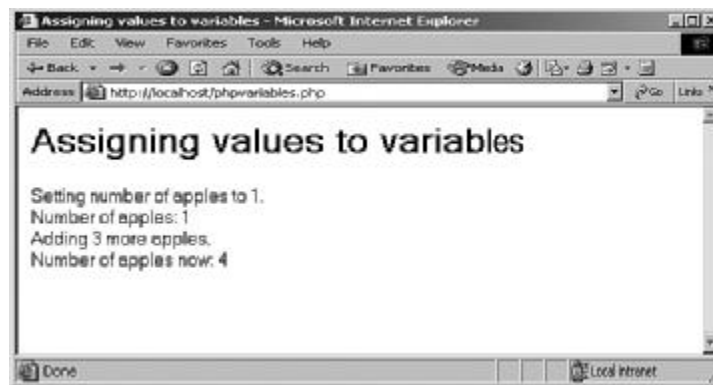
      echo "Number of apples: ", $apples, "<BR>";

      echo "Adding 3 more apples.<BR>";

      $apples = $apples + 3;

      echo "Number of apples now: ", $apples, "<BR>";
    ?>
  </BODY>
</HTML>
```

Rezultat je prikazan na slici 1.8. Sada vidite kako se u vreme izvršenja koristi matematika, kako se podaci smeštaju u promenljive i kako se manipuliše podacima.



Slika 1.8
Dodeljivanje vrednosti promenljivim

Ubacivanje promenljivih u nizove karaktera

Vrednosti promenljivih možete prikazati na sledeći način:

```
$apples = 1;
echo "Number of apples: ", $apples, ".";
```

Postoji i lakši način. Vrednosti promenljivih mogu da se ubace u niz karaktera ako ih stavite u niz karaktera omeđen dvostrukim (ne jednostrukim) navodnicima, što znači da se njihove vrednosti ubacuju direktno u string. Primenom ove tehnike naš prethodni primer izgleda ovako:

```
$apples = 1;
echo "Number of apples: $apples.";
```

Ovaj primer će odštampati Number of apples: 1. U primeru 1.5, `phpinterpolation.php`, dali smo kompletan prethodni primer. U njemu se posle dodeljivanja prikazuju vrednosti promenljivih, a koristi se i umetanje promenljivih u string.

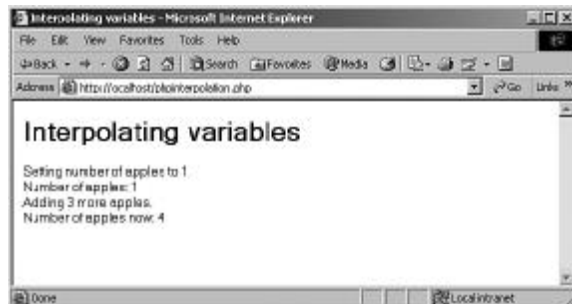
Primer 1.5 Ubacivanje promenljivih u nizove karaktera (stringove)

```
<HTML>
  <HEAD>
    <TITLE>
      Interpolating variables
    </TITLE>
  </HEAD>
  <BODY>
    <H1>
      Interpolating variables
    </H1>
    <?php
      echo "Setting number of apples to 1.<BR>";

      $apples = 1;
      echo "Number of apples: $apples <BR>";
      echo "Adding 3 more apples.<BR>";
      $apples = $apples + 3;
      echo "Number of apples now: $apples <BR>";
    ?>
  </BODY>
</HTML>
```

Rezultat je prikazan na slici 1.9.

POGLAVLJE 1 Osnove PHP-a



Slika 1.9
Ubacivanje vrednosti promenljivih u nizove karaktera

Ovo ubacivanje je brz način za prikazivanje sadržaja promenljive, ali šta učiniti ako promenljiva `$text` sadrži tekst "news", a Vi želite da na izlazu dobijete reč "newspaper"? Mogli biste da probate sledeće:

```
<?php
    $text = "news";

    echo "Where's the $textpaper <BR>";
?>
```

PHP ovo neće razumeti, pošto izgleda kao da se koristi promenljiva pod nazivom `$textpaper`. Greška koju ćete dobiti je:

```
PHP Notice: Undefined variable: textpaper in C:\php\t.php on line 4
```

Ispravan način da se ovo uradi je da se promenljive čiji se sadržaj ubacuje ograde velikim zagradama. U ovom slučaju to je promenljiva `$text` (primer 1.6).

Primer 1.6 Proširivanje promenljivih u nizovima karaktera

```
<HTML>
  <HEAD>
    <TITLE>
      Interpolating variables
    </TITLE>
  </HEAD>
  <BODY>
    <H1>
      Interpolating variables
    </H1>
    <?php
      $text = "news";

      echo "Where's the {$text}paper.";
    ?>
  </BODY>
</HTML>
```

Kreiranje promenljivih promenljivih

PHP omogućava da kreirate *promenljive promenljivih*. Promenljiva promenljiva je ona koja sadrži naziv promenljive. Evo kako to radi. Prvo se napravi promenljiva - na primer, pod nazivom \$apples:

```
<?php
    $apples = 4;
    .
    .
    .
?>
```

Nakon toga, kreirate novu promenljivu, čiji će naziv biti \$fruitname. U nju ćemo smestiti naziv promenljive \$apples.

```
<?php
    $apples = 4;
    $fruitname = "apples";
    .
    .
    .
?>
```

Promenljivoj \$apples možete da pristupite preko \$\$fruitname:

```
<?php
    $apples = 4;
    $fruitname = "apples";
    echo "Number of apples: ", $$fruitname;
?>
```

Ovaj skript prikazuje sledeći rezultat:

```
Number of apples: 4
```

Kada koristite ubacivanje pomoću dvostrukih navodnika, treba da budete izuzetno pažljivi, pošto PHP u tom slučaju ima problema sa izrazima kao što je \$\$fruitname. Ovo možete da rešite pomoću velikih zagrada: \${\$fruitname}.

Primer 1.7, phpvariables.php, pokazuje kako ovo radi:

Primer 1.7 Upotreba promenljivih promenljivih

```
<HTML>
  <HEAD>
    <TITLE>
      Using variable variables
    </TITLE>
  </HEAD>
```

POGLAVLJE 1 Osnove PHP-a

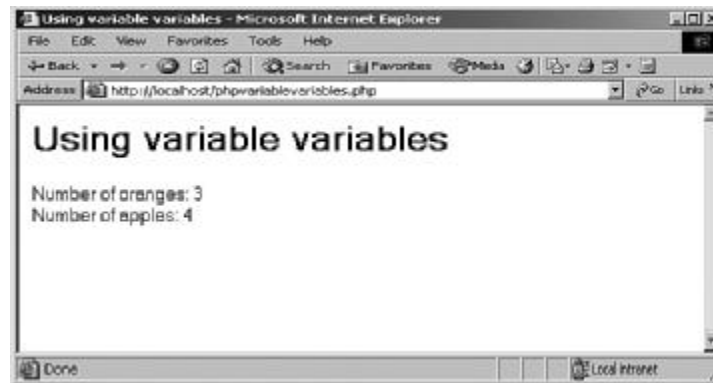
```
<BODY>
  <H1>
    Using variable variables
  </H1>
  <?php
    $apples = 4;
    $oranges = 3;
    $fruitname = "oranges";

    echo "Number of oranges: ${$fruitname} <BR>";

    $fruitname = "apples";

    echo "Number of apples: ${$fruitname} <BR>";
  ?>
</BODY>
</HTML>
```

Rezultat ovog primera je dat na slici 1.10.



Slika 1.10

Ubacivanje promenljivih promenljivih

Ako u prethodnom primeru ne biste koristili velike zagrade, dobili biste sledeći rezultat:

```
Number of oranges: $oranges <BR>
Number of apples: $apples <BR>
```

Vrlo je važno da shvatite promenljive promenljive. Možda Vam u ovom trenutku to izgleda samo kao neki kuriozitet. One su konkretno vrlo korisne kada se radi sa petljama i nizovima, kao što ćete videti u Poglavlju 3.

Kreiranje konstanti

Ponekad nećete želeći da promenljiva može da se menja - želećete da njena vrednost bude fiksna. Pretpostavimo, na primer, da imate promenljivu pod nazivom \$pi koja sadrži vrednost broja pi. Može se desiti da se takva vrednost nenamerno promeni, što nije dobro. Rešenje je da se napravi konstanta, čija vrednost ne može da se menja.

Konstante se kreiraju pomoću funkcije define. Konstanti se daju naziv i vrednost koju želite da joj dodelite: define ("pi", 3.14159265);. Naziv konstante je uvek pod navodnicima, dok je vrednost koja se dodeljuje pod navodnicima samo ako je reč o stringu (nizu karaktera). Pogledajte datoteku phpconstants.php u primeru 1.8. U ovom primeru se kreira konstanta pod nazivom pi, posle čega se prikazuje. Obratite pažnju da, kada koristite konstante, ne koristite prefiks \$. Rezultat primera je dat na slici 1.11.

Primer 1.8 Kreiranje konstanti

```
<HTML>
  <HEAD>
    <TITLE>
      Using PHP constants
    </TITLE>
  </HEAD>

  <BODY>
    <H1>
      Using PHP constants
    </H1>
    <?php
      define ("pi", 3.1415926535);

      echo "The constant pi holds " , pi, "<BR>";

    ?>
  </BODY>
</HTML>
```



Slika 1.11
Upotreba konstanti

POGLAVLJE 1 Osnove PHP-a

Ako probate da promenite vrednost ove konstante (na primer, `pi = 3.14;`), PHP neće to prihvatiti. Čak se ni skript neće pokrenuti.

Pošto se ispred konstanti ne stavlja prefiks `$`, PHP može da se "zbuni" ako za naziv konstante upotrebite neku od rezervisanih reči. U listi koja sledi su date ključne reči:

Ključne reči u PHP-u:

<code>__CLASS__</code>	<code>__FILE__</code>	<code>__FUNCTION__</code>	<code>__LINE__</code>	<code>__METHOD__</code>
<code>and</code>	<code>default</code>	<code>endif</code>	<code>global</code>	<code>print</code>
<code>array</code>	<code>die</code>	<code>endswitch</code>	<code>if</code>	<code>require</code>
<code>as</code>	<code>do</code>	<code>endwhile</code>	<code>include</code>	<code>require_once</code>
<code>break</code>	<code>echo</code>	<code>eval</code>	<code>include_once</code>	<code>return</code>
<code>case</code>	<code>else</code>	<code>exception</code>	<code>isset</code>	<code>static</code>
<code>continue</code>	<code>elseif</code>	<code>exit</code>	<code>list</code>	<code>switch</code>
<code>class</code>	<code>empty</code>	<code>extends</code>	<code>new</code>	<code>unset</code>
<code>const</code>	<code>enddeclare</code>	<code>for</code>	<code>old_function</code>	<code>use</code>
<code>continue</code>	<code>endfor</code>	<code>foreach</code>	<code>or</code>	<code>while</code>
<code>declare</code>	<code>endforeach</code>	<code>function</code>	<code>php_user_filter</code>	<code>xor</code>

Pored ovih ključnih reči, postoji i određeni broj konstanti koje su unapred definisane i koje možete koristiti u svojim skriptovima. Evo primera:

<code>__LINE__</code>	Trenutni broj reda u datoteci
<code>__FILE__</code>	Puna putanja i naziv datoteke
<code>__FUNCTION__</code>	Naziv funkcije (ovo je dodato od verzije PHP 4.3.0)
<code>__CLASS__</code>	Naziv klase (ovo je dodato od verzije PHP 4.3.0)
<code>__METHOD__</code>	Naziv metoda klase (ovo je dodato od verzije PHP 5.0)
<code>PHP_VERSION</code>	Verzija PHPa
<code>PHP_OS</code>	Operativni sistem
<code>DEFAULT_INCLUDE_PATH</code>	Gde da PHP traži ono što mu je potrebno

Na primer, ako napišete `echo __LINE__` na određenoj poziciji u skriptu, prikazaće se trenutna linija koja se izvršava.

Rukovanje tipovima podataka

Vi ne možete da odlučujete o načinu na koji se Vaši podaci interno skladište. O tome odlučuje PHP. Iza scene, on podržava osam internih tipova podataka (sve tipove ćete upoznati u knjizi).

- boolean. Sadrži vrednosti true/false (tačno/netačno).
- integer. Sadrži cele brojeve, kao što su -1, 0, 5, itd.
- float. Sadrži brojeve sa pokretnim zarezom (tip double), kao što su 3.14159 ili 2.7218.
- string. Sadrži tekst, kao što je "Zdravo iz PHP-a".
- array (niz), Sadrži niz sa podacima.
- object. Sadrži objekte iz programa.
- resource. Sadrži resurse sa podacima.
- NULL. Sadrži vrednost NULL.

Obično ne morate da brinete o ovim tipovima podataka, pošto PHP određuje tip promenljive na osnovu podataka koje joj dodelite. Na primer, sledeći iskaz će promenljivoj \$variable dodeliti tip string:

```
$variable = "No worries.";
```

Sledeći iskaz će napraviti promenljivu tipa float:

```
$variable = 1.2345;
```

Sledeći iskaz će napraviti promenljivu tipa boolean:

```
$variable = TRUE;
```

Pošto PHP bira tip podatka na osnovu podatka koji dodelite promenljivoj, ovde nema problema. Problemi počinju kada počnete da mešate tipove - na primer, kada promenljivoj \$variable dodate vrednost pomoću operatora + (o njemu će biti reči u Poglavlju 3). Evo nekoliko primera:

```
<?php
$variable = "0"; // $variable je string podešen na 0.
$variable = $variable + 2; // $variable je sada ceo broj sa vrednošću 2.
$variable = $variable + 1.1; // $variable je sada tipa float sa vrednošću 3.1.
$variable = 2 + "8 apples"; // $variable je sada ceo broj sa vrednošću 10
?>.
```

POGLAVLJE 1 Osnove PHP-a

Ako želite da izbegnete potencijalne probleme sa tipovima podataka, nemojte mešati tipove. Čak i ako to uradite, PHP će skoro uvek "uraditi pravu stvar" (na primer, rezultat sabiranja celog i realnog broja će konvertovati u realni broj, što je upravo ono što treba da se uradi), ali, ako je potrebno da eksplicitno zadate tip promenljive, uvek možete koristiti iskaz *type cast*. Novi tip se stavlja između zagrada odmah ispred naziva promenljive čiji tip želite da zadate. Evo nekoliko primera:

```
$int_variable = (integer) $variable;  
$float_variable = (float) $variable;  
$string_variable = (string) $variable;
```

Evo nekoliko saveta u vezi sa mešanjem tipova podataka. Kada konvertujete u tip boolean, smatra se da su vrednosti tipa FALSE (o tome će biti više reči kasnije u ovoj knjizi):

- boolean FALSE
- integer 0
- float 0.0
- prazan string i string "0"
- Niz sa nula elemenata
- Objekat bez članova
- Specijalni tip NULL (uključujući i promenljive koje nemaju vrednost)

Svaki drugi tip se smatra da je TRUE (uključujući i resource). Kada konvertujete u tip integer:

- Boolean FALSE će dati 0 (nula), a Boolean TRUE će dati 1 (jedan).
- Vrednosti tipa float se zaokružuju prema nuli.

Kada konvertujete tip float, konverzija je ista kao kad je vrednost konvertovana u integer, a onda u float. Konverziju možete da vršite i iz integera u numeričke tipove, ali to ponekad može biti komplikovano (više detalja sadrži Poglavlje 3).

U narednom poglavlju ćete početi kontrolu svojih podataka u PHP skriptovima pomoću operatora i iskaza za kontrolu toka i petlje, što je veoma bitno za svakog PHP programera.

Zaključak

PHP omogućava da strane "ožive" na način na koji to nije moguće sa statičkim stranim. Evo nekoliko najbitnijih napomena za početak.

- Instrukcije za instalaciju PHP-a su različite od jednog do drugog operativnog sistema i i od jedne do druge verzije i mogu biti vrlo komplikovane. Da biste instalirali PHP, pratite instrukcije koje dolaze sa PHP-om koji ste preuzeli sa Interneta.
- Svoju instalaciju PHP-a ćete proveriti ako na komandnoj liniji otkucate `php -v`.
- Ako se prikažu informacije o PHP-u, možete da mu pristupite.
- Da biste radili sa PHP-om, morate da podesite razvojno okruženje, koje uključuje i neki editor teksta ili PHP IDE. Razvojno okruženje bi trebalo da sadrži i neki način za instalaciju PHP strana tamo gde server može da ih pročita.
- U PHP skriptovima se izvršni kod nalazi između oznaka `<? i ?>`. Na istu stranu možete da ubacite i HTML kod, sve dok je taj kod izvan ovih oznaka.
- Iskaz `echo` prikazuje tekst u pretraživaču.
- Dokumenti tipa `here` omogućavaju štampanje svake reči u tekstu dok se ne stigne do određenog tokena.
- U PHP-u postoje tri tipa komentara: `/* */`, `//` i `#`.
- Ispred naziva promenljivih stavlja se `$`. Ispravan naziv promenljive je onaj koji počinje slovom ili podvlakom, a slede slova, brojevi i podvlake.