

ՅՈՒՆԵՍԿՈ ԵՎ ԵՐԵՎԱՆԻ ՄԱՐԶԻ ԲԵՐԻՄԻՆԻՍՏԻԿԱԿԱՆ ԳԻՏԱԿԱՆ ԿԵՆՏՐՈՆ

ԹԻՖԼԻՍ

ՄԻԿՈՆԱՆԻ ՄԻԿՐՈՆ

Otvoreno računarstvo

- PHP: Hypertext Preprocessor (PHP)

Mario Žagar



HTML nije kraj, opet smo na početku...

```
<html>
  <head>
    <title>PHP - primjer1</title>
  </head>
  <body>
    <?php echo "Pozdrav od PHP-a!"; ?>
  </body>
</html>
```

PHP



- PHP: Hypertext Preprocessor
- Skriptni jezik opće namjene, otvorenog kôda
- Pogodan za razvoj Weba
- Dostupan na raznim platformama
 - Windows, Linux, Unix, OS X, ...
- Jednostavan za učenje
 - Sintaksa slijedi C, Javu, Perl
- Cilj:
Jednostavan jezik za **brzo** stvaranje dinamičkih stranica Weba



PHP, <http://www.php.net/>

Kratka povijest

- Prapovijest: niz skripta pisanih u Perlu za obradu podataka s obrazaca
- PHP/FI (Personal Home Page/Forms Interpreter)
 - Biblioteka funkcija pisanih u C-u
- PHP 1.0/2.0 – osnovna funkcionalnost kao i danas
 - Otvoren za proširenja
- PHP3 – razvoj nove jezgre
- PHP4 – opet nova jezgra (Zend Engine)
- PHP5 – proširen objektni model, aktualna verzija
- PHP6 – podrška za Unicode string tip, *čišćenje* jezika

PHP danas

- Četvrti najpopularniji programski jezik
 - Prema TIOBE indeksu (<http://www.tiobe.com>)
 - Iza Jave, C-a, Visual Basic-a
 - Ispred C++-a, Perl-a, Python-a, C#-a
- Najpopularniji dodatak za Apache web poslužitelj
 - Apache je najpopularniji poslužitelj
- U aktivnom razvoju, s velikom bazom korisnika
- Jedan od gradivnih blokova LAMP arhitekture
 - Jeftina, pouzdana, sigurna, skalabilna
 - Web aplikacije
 - Linux, Apache, MySQL, PHP (ili Perl/Python)

Pokretanje PHP-a

- Izvodi se na poslužitelju
- Više načina rada
 - Kao dio poslužitelja Weba (modul)
 - Kao CGI poveznica koju poslužitelj Weba poziva
 - Iz naredbene linije (CLI – *Command Line Interface*)
- U osnovi radi kao *filtar*
 - Preuzima datoteku s programom (PHP *skripta*)
 - Generira podatke (resurs) kao rezultat rada
- Namijenjen prvenstveno dinamičkim stranicama Weba
- Moguća izrada klasičnih aplikacija
 - Naredbena linija ili GUI (GTK+)

Osnovna svojstva

- Programski kôd
 - Uključen u HTML/XML/... dokument
 - Samostalan
- Blokovi PHP kôda označeni oznakama

<ul style="list-style-type: none"> • <code><?php ?></code> • <code><script language="php"> </script></code> • <code><? ?></code>, <code><?= ?></code> • <code><% %></code>, <code><%= %></code> 	}	Preporučeno
	}	Ovisno o postavkama
- Sve van oznaka se ignorira
 - Ne parsira se kao PHP kôd
 - Prosljeđuje se kao rezultat izvođenja
 - Moguće gniježđenje unutar HTML/XML/... dokumenta

Osnovna svojstva

- Komentari – nekoliko stilova
 - Više redova – `/* */`
 - Jedan redak – `//`
 - Jedan redak – `#`
- Sintaksa slična jeziku C
- Neki konstrukti posuđeni iz Perl-a
- Proceduralna paradigma (funkcije i *program*)
- Objektno-orijentirana paradigma
 - PHP4 osnovna OO
 - PHP5 proširena OO funkcionalnost

Osnovna svojstva – tipovi podataka



- Skalarni tipovi
 - *boolean*
 - *integer*
 - *float* (dvostruka preciznost)
 - *string* (8-bitni znakovi)
- Složeni tipovi
 - *array* (simboličko polje, *hash*)
 - *object* (OO funkcionalnost)
- Posebni tipovi
 - *resource* (pokazuju na vanjske resurse – npr. otvorene datoteke)
 - *NULL* (prazan tip, pridružen varijablama koje nemaju vrijednost)

Osnovna svojstva – nizovi znakova



- Tip podatka *string* sadrži 8-bitne znakove
 - Nacionalni znakovi kodirani u 8 bita
 - UTF-8, ISO8859-2, Win1250
- Podrška za Unicode znakove (više okteta) direktno u tipu podatka *string* – tek od PHP6
- Za primjenu u okviru Weba zadovoljavajuće
 - Biblioteke za konverziju kodnih stranica
 - Sučelja za rad s bazama podataka (i sâme baze) obavljaju sve potrebne konverzije

Osnovna svojstva – varijable

- Ime varijable – kao u C-u
 - Počinje sa *slovom* ili `_`, slijedi proizvoljan broj *slova*, brojeva ili `_`
- Pristupanje varijabli
 - `$ime_varijable`
 - Prema vrijednosti ili referenci (PHP4+)
 - `$broj = 43;`
 - `$referenca = &$broj; $referenca = 42; // $referenca=$broj=42`
- Tip varijable
 - Ne može se eksplicitno odrediti
 - PHP interno određuje tip prema kontekstu

Osnovna svojstva – izrazi

- Skup operatora sličan C-u
 - Spajanje stringova – operator .
 - Operator identiteta === - jednakost vrijednosti i tipa
- Upravljanje tokom programa
 - *if ... elseif ... else*
 - *while, do ... while*
 - *for, foreach*
 - *break, continue*
 - *switch*
 - *return*
- Funkcije
 - *function fx(\$x) { return \$x; }*

Primjer



```
<?php
```

```
$istina = true;  
if($istina)  
    print("Istina!");  
else  
    print("Laž");
```

; za kraj naredbe

```
for($i=0;$i<10;$i++)  
{  
    $polje[i] = $i;  
}
```

Pristupanje elementima polja

```
print("Duljina polja je " . count($polje));
```

Spajanje nizova znakova

```
foreach($polje as $vrijednost)  
{  
    print("Vrijednost: " . $vrijednost);  
}
```

foreach – iteriranje po polju

```
?>
```

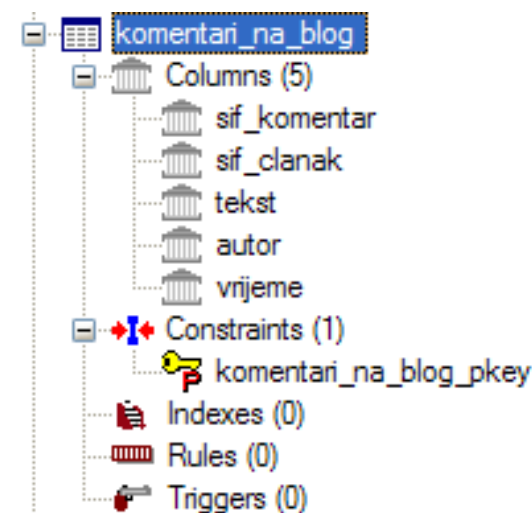
Što sve može - API

- Skup ugrađenih funkcija se može proširiti
- Velik broj postojećih proširenja (*extension*)
- Podrška za
 - Rad sa stringovima (slično C-u)
 - Rad s datotekama (slično C-u)
 - XML (DOM, libxml), XSL, XSLT, XML-RPC
 - Baze podataka (MySQL, PostgreSQL, Oracle, Informix,...)
 - Flash
 - PDF
 - LDAP, LotusNotes
 - HTTP, FTP, SMTP, SOAP, SSL, SSH, ...
 - ZIP, RAR, Zlib, ...

Primjer – podaci u bazi podataka



```
CREATE TABLE komentari_na_blog
(
  sif_komentar serial NOT NULL,
  sif_clanak int4 NOT NULL,
  tekst varchar(1024),
  autor varchar(128),
  vrijeme timestamp DEFAULT now(),
  CONSTRAINT komentari_na_blog_pkey PRIMARY KEY (sif_komentar)
)
```



	sif_komentar [PK] serial	sif_clanak int4	tekst varchar	autor varchar	vrijeme timestamp
1	1	10	Jako dobar član	Posjetitelj	2008-03-26 20:51:49.8518
2	2	10	Totalne gluposti	Komentator	2008-03-26 20:52:08.5915
3	3	10	Oprostite ne radi	Mali Pero	2008-03-26 20:52:56.1805
*					

Primjer



- Iz baze podataka dohvaćamo komentare na članak

```
<?php
```

```
// Spajanje na bazu - pg_??? za PostgreSQL
$db_conn = pg_connect("host=stroj dbname=baza user=korisnik password=");

// Identifikator clanka primljen u upitu
$sifClanak = $_REQUEST['clanak'];
$res = pg_query(
    "SELECT * FROM komentari_na_blog WHERE sif_clanak = '". $sifClanak . "'");

if (pg_numrows($res) > 0) // Ima rezultata
{
    $sve = pg_fetch_all($res); // Dohvati sve rezultate

    foreach($sve as $komentar) // Po svim rezultatima
    {
        print($komentar['autor'] . " je rekao '"
            . $komentar['tekst'] . "' . "<br/>"); // Ispisi
    }
}

?>
```

Primjer



- Rezultat:

Posjetitelj je rekao 'Jako dobar članak'

Komentator je rekao 'Totalne gluposti'

Mali Pero je rekao 'Oprostite ne radi mi komentiranje, kako da pošaljem komentar'

U odnosu na CGI

- PHP prvenstveno namijenjen izradi dinamičkih stranica
 - **Automatsko** parsiranje upita (query string) i podataka dobivenih HTTP-om u varijable PHP-a
 - Moguće miješanje PHP i HTML kôda
 - **Automatsko** generiranje rezultata prema HTTP normi

Polje ime nije postavljeno u HTTP GET upitu (u obrascu)

```
<?php
if(!isset($_GET['ime']))
{
?>
    <h1>Niste unijeli ime!</h1>
<?php
}
else
    print("Dobrodošli, " . $_GET['ime']);
?>
```

Niste unijeli ime!

Nasljeđe CGI

- Globalne varijable definirane unaprijed
 - `$_ENV` – varijable okruženja
 - `$_ENV["HOME"]="/var/www"`
 - `$_ENV["HOSTNAME"]="orson"`
 - `$_ENV["HOSTTYPE"]="linux"`
 - `$_SERVER` – podaci o poslužitelju
 - `$_SERVER["REQUEST_METHOD"]="GET"`
 - `$_SERVER["QUERY_STRING"]="ime=Ivan"`
 - `$_SERVER["REQUEST_URI"]="/~korisnik/1.php?ime=Ivan"`
 - `$_REQUEST`, `$_GET`, `$_POST` – podaci iz zahtjeva
 - `$_REQUEST["ime"]="Ivan"`
 - `$_GET["ime"]="Ivan"`
 - `$_POST` – u ovom slučaju prazan, ponaša se na isti način

Logika programiranja

- PHP datoteka (skripta) = *filtar*
 - Prihvaća ulazne podatke
 - Generira rezultat
- Primjereno webu
 - Ne postoji trajna interakcija sa korisnikom (kao u GUI aplikacijama)
 - Ne postoji implicitno očuvanje stanja aplikacije
 - Aplikacija ≠ Proces
- Aplikacija razdvojena na niz skripti
 - Sve potrebne podatke potrebno eksplicitno prenijeti od jedne do druge skripte

Neki savjeti

- Ne postoji deklaracija tipa varijabli
 - Inicijalizacija na početnu vrijednost donekle pomaže
- Miješanje PHP i HTML kôda smanjuje čitljivost
 - Rješenje: sustavi predložaka
- Modularizacija
 - Moguće je u skriptu uključiti druge datoteke PHP-a
 - include, include_once, require, require_once
 - Definicija pomoćnih funkcija
 - Generiranje dijelova stranice (npr. navigacija-izbornik)
- Provjeravati podatke dobivene od korisnika
 - Nizovi znakova mogu sadržavati dijelove kojima je cilj oštetiti podatke u bazi
 - Npr. za ime korisnika 'Ivan\'; DROP DATABASE;'

Sustavi predložaka

- Sustavi predložaka (*templates*) razdvajaju prikaz (HTML) od logike (PHP)
 - Slično modelu *model-view-controller* (MVC)
 - Programaska logika (PHP) postavlja parametre (varijable)
 - Prikaz koristi podatke iz parametara za oblikovanje sadržaja
 - Postoje konstrukti kao što su
 - Operatori (formatiranje teksta, brojeva, datuma, ...)
 - Uvjetno izvođenje
 - Petlje
 - Iteratori
 - Predlošci se mogu gnijezditi (*include*)
 - Jednostavno razdvajanje stranice na zone i ponovno korištenje
 - Npr. poseban predložak za navigaciju, naslov stranice, tablični prikaz,...

Primjer: Smarty

- Sustav predložaka za PHP
- Koristi se kroz klasu Smarty
 - Postavljanje varijabli vidljivih u predlošcima
 - Prikazivanje predloška

PHP – programska logika

```
<?php

require('Smarty.class.php');

$smarty = new Smarty ();

$smarty->assign('ime','ivan');
$smarty->assign('prezime','ivić');

$smarty->display('index.tpl');
?>
```

Smarty – prikaz

```
<html>
<body>

<h1>Podaci o korisniku:</h1>

Ime: {$ime|capitalize}<br>
Prezime: {$prezime|capitalize}<br>

</body>
</html>
```


Primjer



- Koristimo Smarty za ispis komentara iz baze

<?php

```
require('Smarty.class.php');  
$smarty = new Smarty();
```

} Uključivanje klase i inicijalizacija sustava predložaka

```
// Spajanje na bazu - pg_??? za PostgreSQL
```

```
$db_conn = pg_connect("host=stroj dbname=baza user=korisnik password=");
```

```
// Identifikator clanka
```

```
$sifClanak = $_REQUEST['clanak'];
```

```
$res = pg_query(  
    "SELECT * FROM komentari_na_blog WHERE sif_clanak = '". $sifClanak. "'";  
    );
```

```
if (pg_numrows($res) > 0) // Ima rezultata  
{
```

```
    $sve = pg_fetch_all($res); // Dohvati sve rezultate
```

```
    $smarty->assign('komentari', $sve);
```

```
    $smarty->display('komentari.tpl');
```

} Postavljanje podataka i prikaz predloška

```
}
```

?>

Primjer predložaka

- Predložak obično stvara samo dio stranice
- Jedan, glavni predložak uključuje ostale

```
<html>
  <body>
    {include file='zaglavlje.tpl'}

    <table>
      <tr><th>Autor komentara</th><th>Komentar</th></tr>
      {foreach from=$komentari item=komentar}
      <tr>
        <td>{$komentar.autor}</td><td>{$komentar.tekst}</td>
      </tr>
      {/foreach}
    </table>

    {include file='podnozje.tpl'}
  </body>
</html>
```

→ Iterator po polju

Polje sa simboličkim indeksom (*hash*) ponaša se kao objekt u programskim jezicima

Primjer predložaka

- Rezultat rada predloška
- Zaglavlje i podnožje stranice došli su iz drugih predložaka (zaglavlje.tpl i podnozje.tpl)

Dobrodošli na poslužitelj orson.rasip.fer.hr

Autor komentara

Komentar

Posjetitelj	Jako dobar članak
Komentator	Totalne gluposti
Mali Pero	Oprostite ne radi mi komentiranje, kako da pošaljem komentar

Sada je 21.49.28 26.03.2008.

Pogonjeno PHP-om

- PHP kao podloga za izgradnju
 - Sustava
 - Web stranica (...)
 - Web portala (FER web, PHPNuke, PHP-BB, ...)
 - Web 2.0 *social networking* sustava (Facebook, ...)
 - CMS sustava (CMS Made Simple, eZ Publish, ...)
 - Alata
 - Sustavi predložaka (Smarty, ...)
 - *Framework* biblioteka (Zend Framework, eZ components, Symphony, ...)
- PHP kao podloga za dogradnju
 - Sustavima cacheiranja (ehcache, mmcache, ...)
 - Optimizatorima (Zend Optimizer, ...)

Zašto PHP?

- Cilj:
 - Jednostavan jezik za brzo stvaranje dinamičkih stranica weba
- Sintaksa slična C-u
 - Široko poznata sintaksna struktura
- Velik broj funkcija
 - Velika vjerojatnost da je podržano sve što je potrebno
 - Velika broj korisnika sa istim problemima
- Podrжан kod ISP-ova
 - Podrška za postavljanje stranica/aplikacije
 - Mala veličina programa u odnosu na funkcionalnost

Prednosti i mane

- U odnosu na CGI
 - Najčešće se izvodi kao *modul* poslužitelja weba
 - Nema pokretanja dodatnog procesa *poveznice*
 - Manje zauzeće resursa (memorije, procesora)
 - Jednostavniji život administratora
 - Koristi mehanizme prijenose podataka ugrađene u HTTP
 - Koristi neke mehanizme opisane CGI-jem
 - Poslužitelj postavlja varijable okoline kao za CGI
 - Prijenos podataka između poslužitelja i modula nevidljiv korisniku
- U slučaju potrebe moguć rad i kao CGI poveznica
 - Za poslužitelje za koje ne postoji implementacija modula

Prednosti i mane

- U odnosu na druge programske jezike
 - Prednosti
 - Jednostavnost
 - Popularnost – velik broj korisnika, gotova rješenja
 - Mogućnost proširenja
 - Fokusiran na web
 - Mane
 - Sporost (kôd se ne kompajlira već interpretira u 2 faze)
 - Nedosljednost API-ja
 - Imenovanje funkcija ne slijedi jednu konvenciju
 - Proširenja API-ja unose nered
 - Nedostatci jezika
 - Potpuna podrška za Unicode tek u PHP6
 - Podrška za prostore imena (*namespaces*) tek u PHP5.3
 - Nepotpuni OO model (u odnosu na C++, Javu, .Net)

- Ostvaruje cilj – biti jednostavan
- Velikom broju korisnika *pokrenuo* web
- Relativno brz razvoj jednostavnijeg dinamičkog Weba
- Otvorenost kroz proširenja (*extensions*) i razvoj jezika
- Budućnost – Web stvaraju **korisnici** a ne programeri (developeri) – počinje danas
 - Web 2.0 – blogovi, sustavi za druženje, social networking, razmjenu multimedijalnih sadržaja, ...

