

Web programiranje

Vežbe 2 - Osnove PHP jezika

Podešavanje radnog okruženja

Dva su osnovna načina pristupa AMP (Apache-MySQL-PHP) bateriji aplikacija u zavisnosti gde je AMP instaliran:

- **lokalni** - AMP aplikacije su instalirane na lokalnom računaru, tako da nema potrebe za kopiranjem fajlova unutar projekta na udaljeni LAMP ili WAMP server. Ovakav način rada se preporučuje u fazi razvoja web aplikacije.
- **udaljeni** - Da bi se web aplikacija pokrenula, potrebno je prvo je iskopirati na udaljeni LAMP ili WAMP server, obično korišćenjem FTP ili SFTP protokola (koje alat *NetBeans* automatski podržava).

U daljem tekstu biće data tri postupka za podešavanje radnog okruženja, i to: (1) preko ličnog studentskog naloga na serveru spider.pmf.kg.ac.rs, (2) lokalna AMP instalacija na *Linux*-u, (3) lokalna AMP instalacija u *Windows*-u korišćenjem alata XAMPP.

Lični studentski nalog na IMI serveru

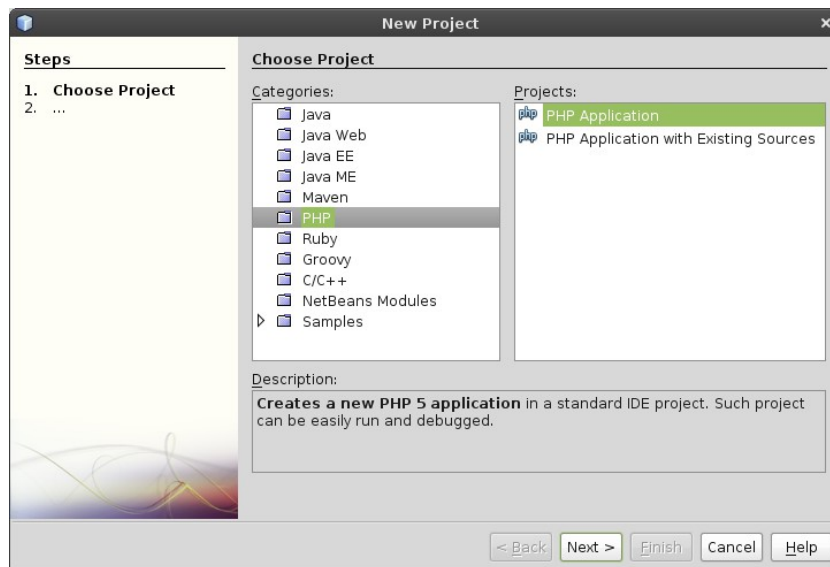
Za svakog studenta III godine otvoren je SSH nalog na fakultetetskom serveru *spider.pmf.kg.ac.rs*, tako da je samim tim aktivniran i SFTP protokol. Parametri konekcije su sledeći:

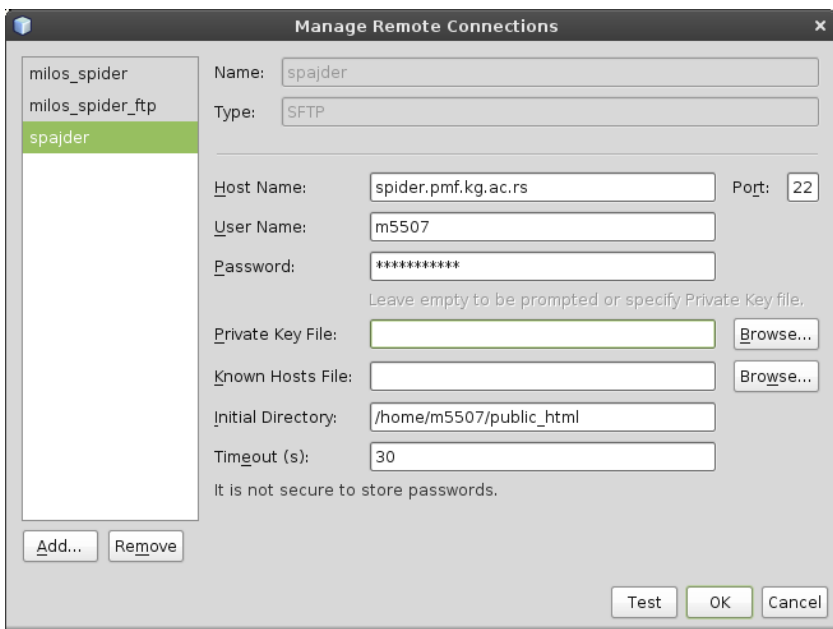
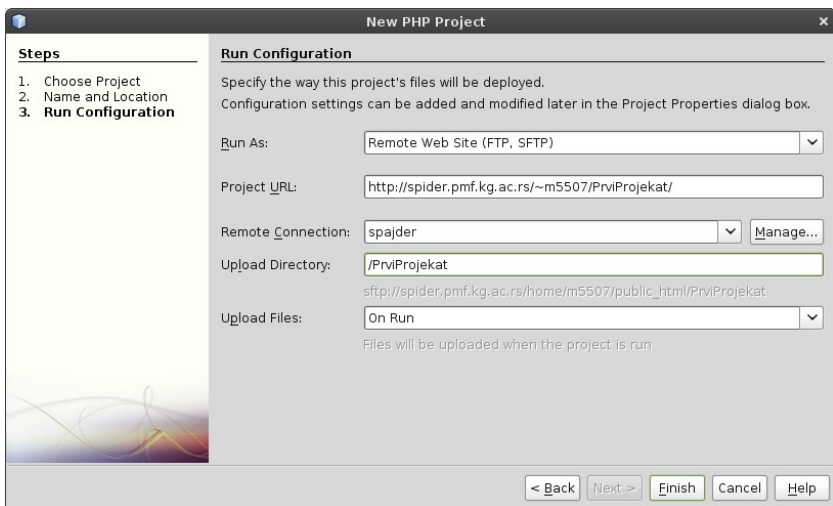
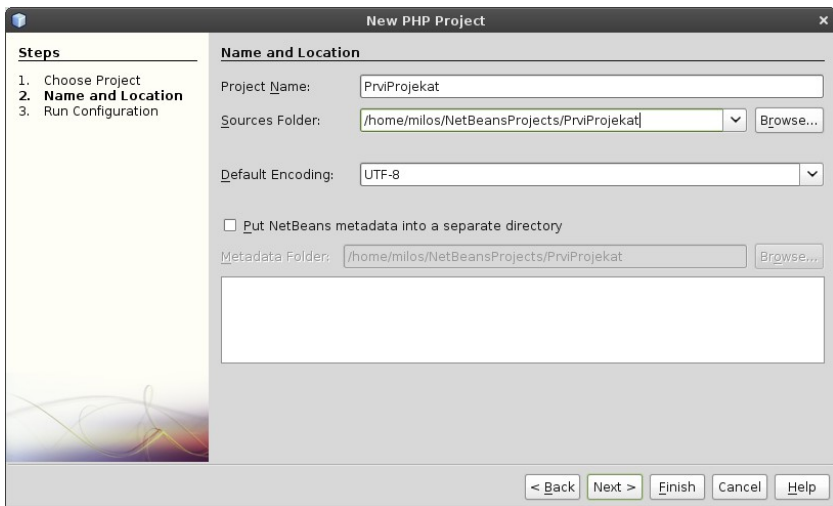
```
hostname: spider.pmf.kg.ac.rs
username: m{broj_indeksa}, na primer m5507
password: lozinka koju je podesio student
```

Prilikom prvog prijavljivanja na server, potrebno je napraviti direktorijum `~/public_html` (ukoliko on već ne postoji). Sve što se nalazi u direktorijumu `~/public_html` biće dostupno na web adresi:

```
http://spider.pmf.kg.ac.rs/~{username}
na primer:
http://spider.pmf.kg.ac.rs/~m5507/
```

Da bi se otvorio i podesio novi projekat u okviru NetBeans okruženja, potrebno je proći kroz *New project* wizard. Evo nekoliko snimaka ekrana (npr. za korisnika **m5507**) koji mogu pomoći prilikom podešavanja. Treba naglasiti da se podešena SFTP konekcija može iskoristiti i za neki novootvoreni projekat.





Lokalna instalacija na LINUX-u

Kao što je pomenuto u ranijem materijalu, na primeru Ubuntu distribucije, dovoljno je instalirati sledeće pakete pozivom **apt** komande ili preko grafičkog paket menadžera (npr **Synaptic**):

```
apt-get install apache2 libapache2-mod-php5 php-pear mysql-server php5-mysql
```

Apache server se pokreće komandom koja, naravno, zahteva administratorske privilegije, pa se koristi prefiks **sudo**:

```
sudo /etc/init.d/apache2 restart
```

ili

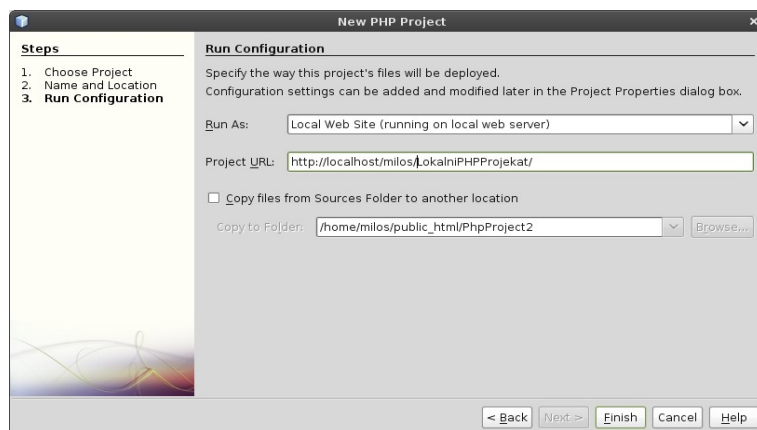
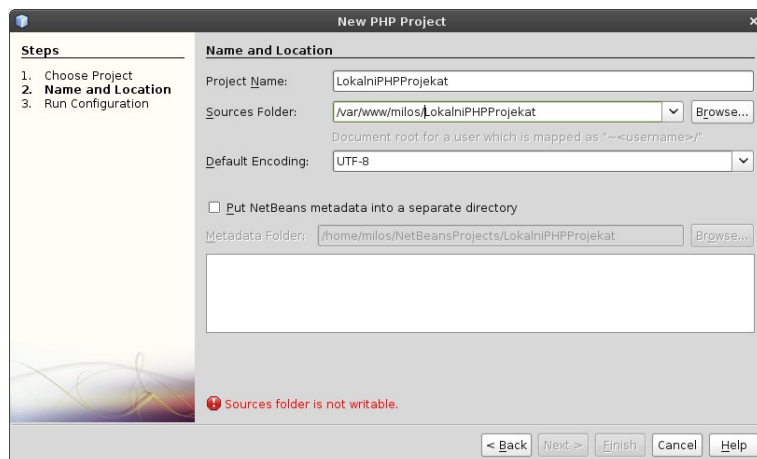
```
sudo service apache2 restart
```

Osnovni direktorijum za smeštanje HTML i PHP dokumenata je **/var/www**, koji je vlasništvo superusera **root**, pa je preporučljivo napraviti poddirektorijum i postaviti da njegov vlasnik bude sam korisnik koji razvija web aplikaciju (tj. onaj koji pokreće *NetBeans*):

```
sudo mkdir /var/www/milos
```

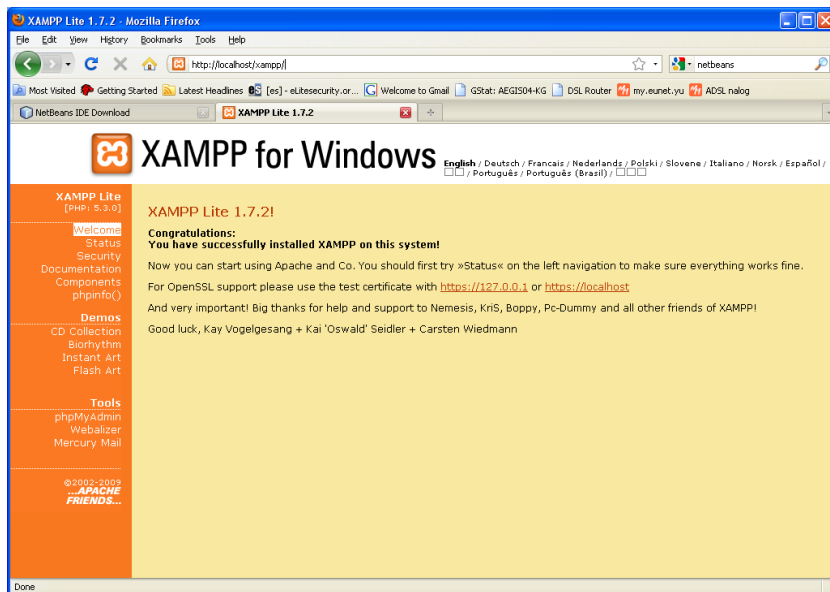
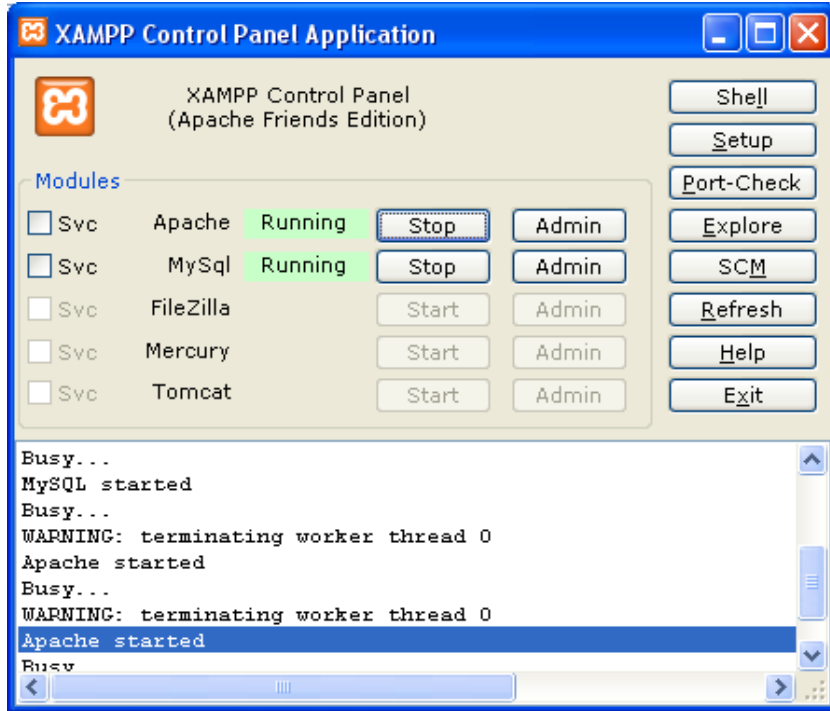
```
chown milos /var/www/milos
```

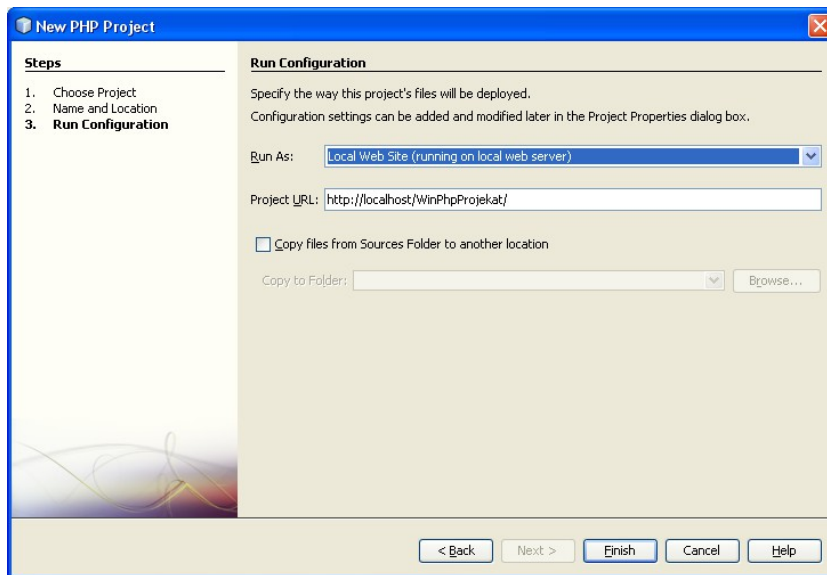
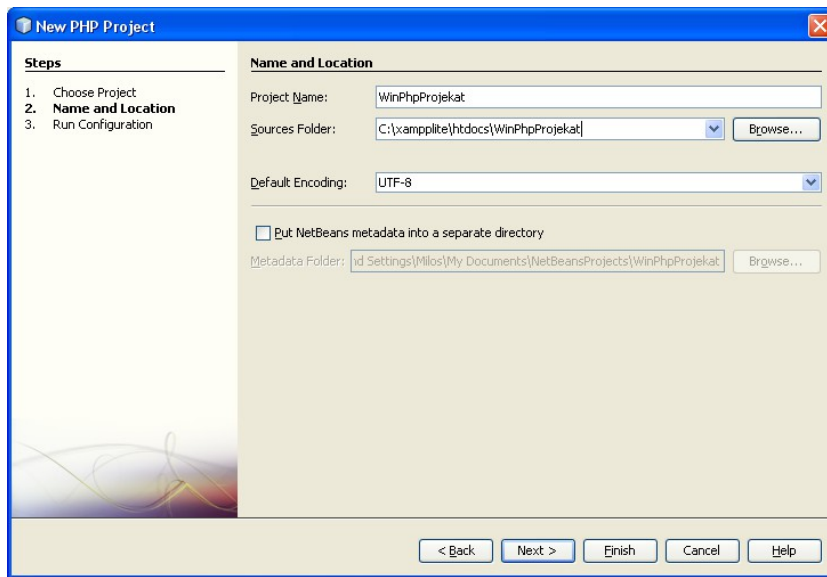
Dakle, sve web aplikacije koje korisnik milos razvija treba da se nalaze unutar direktorijuma **/var/www/milos**. Evo i postupka za otvaranje NetBeans projekta na lokalnom Linux sistemu:



Lokalna instalacija na Windows-u

Na Windows sistemu je daleko najjednostavnija varijanta instalacija AMP baterije korišćenjem neke "spakovane" instalacije sa grafičkim kontrolnim panelom, kao što je npr. XAMPP (<http://www.apachefriends.org/en/xampp-windows.html>). Instalacija je krajnje pojednostavljena, a evo nekoliko slika ekrana kako podesiti XAMPP i *NetBeans* projekat:





Treba obratiti pažnju da se projekat locira u DOCUMENT_ROOT XAMPP Apache instalacije, u prikazanom slučaju **c:\xampp\htdocs**.

Ugrađivanje PHP koda

Jedna od prednosti PHP-a je i mogućnost da se **PHP kod ugrađuje direktno u HTML strukture**. Međutim, potrebno je obratiti pažnju da se takve stranice identifikuju odgovarajućom ekstenzijom (tipično “.php”), kako bi ih web server pravilno interpretirao.

Podrazumevana sintaksa je da se ono što treba interpretirati kao PHP kod stavi između tagova:

```
<?php i ?>
```

Primer najjednostavnije dinamičke stranice u PHP-u bi glasilo:

```
<h3>Naslov</h3>
<?php
    echo "<p>Dinamicki izlaz</p>";
?>
<p>Staticki izlaz</p>
```

Moguće je izaći i ući u PHP onoliko puta koliko je to potrebno na datoj stranici. Recimo, sledeći kod savršeno funkcioniše:

```
<html>
  <head>
    <title><?php echo "Dobrodosli!";?></title>
  </head>
  <body>
    <?php
      $datum = "26 jul 2007";
    ?>
    <p>Today's date is <?php echo $datum; ?></p>
  </body>
</html>
```

Komentari

Dozvoljeno je koristiti 3 vrste komentara:

C++ stil:

```
<?php
  // Title: My first PHP script
  // Author: Jason
  echo "This is a PHP program";
?>
```

C stil:

```
<?php
  /*
    Title: My PHP Program
    Author: Jason
  */
```

```
Date: July 26, 2007
*/
```

```
?>
```

Shell stil:

```
<?php
# Title: My PHP program
# Author: Jason
echo "This is a PHP program";
?>
```

Ispis podataka u web pretraživač

PHP poseduje nekoliko komandi za ispis teksta u web browser:

```
int print(argument)
void echo(string argument1 [, ...string argumentN])
boolean printf(string format [, mixed args])
```

Kao što se vidi iz prototipa, osnovna razlika između ove dve naredbe je u broju argumenata i povratnoj vrednosti. Evo nekoliko primera:

```
<?php
    print("<p>I love the summertime.</p>");
?>
<?php
    $season = "summertime";
    print "<p>I love the $season.</p>";
?>
<?php
    $heavyweight = "Lennox Lewis";
    $lightweight = "Floyd Mayweather";
    echo $heavyweight, " and ", $lightweight, " are great fighters.";
?>
<?php
    printf("Bar inventory: %d bottles of tonic water.", 100);
    printf("%d bottles of tonic water cost $%f", 100, 43.20);
?>
```

Poslednji primer štampa sledeći tekst:

```
Bar inventory: 100 bottles of tonic water.
100 bottles of tonic water cost $43.20
```

Očigledno je da se koriste oznake formata identične onima poznatim iz jezika C. Sličnost sa C-om ogleda se i u prisustvu naredbe **sprintf**:

```
string sprintf(string format [, mixed arguments])
```

Evo i primera:


```
$cost = sprintf("$%.2f", 43.2); // $cost = $43.20
```

Skalarni tipovi

Logički tip (boolean)

Logički tip se u PHP-u ponaša slično kao u C-u, dakle nema stroge tipizacije kao u jeziku Java. 0 se tretira kao **false**, a sve ostale vrednosti kao **true**:

```
$alive = false; // $alive is false.
$alive = 1;     // $alive is true.
$alive = -1;   // $alive is true.
$alive = 5;    // $alive is true.
$alive = 0;    // $alive is false.
```

Celobrojni tip (Integer)

Zapisi celih brojeva mogu biti dekadni, heksadekadni i oktalni. U PHP6 uveden je i 64-bitni celobrojni tip

```
42          // decimal
-678900     // decimal
0755       // octal
0xC4E      // hexadecimal
```

Realni tip (Float)

Evo nekoliko načina zapisa:

```
4.5678
4.0
8.7e4
1.23E+11
```

String

Stringovi se u PHP-u tretiraju tradicionalno, kao nizovi karaktera. Mogu biti uokvireni jednostrukim ili dvostrukim navodnicima:

```
"PHP is a great language"
"whoop-de-do"
'*9subway\n'
"123$%^789"
```

```
$color = "maroon";
$parser = $color[2]; // Assigns 'r' to $parser
```

Znaci navođenja mogu biti dati unutar stringa ako se ispred njih stavi escape karakter “\”:

```
echo "Francuski kralj Filip IV \"lepi\"";
```

Postoji značajna razlika između uokviravanja stringova jednostrukim i dvostrukim navodnicima. Naime, string vrednosti u duplim navodnicima se automatski parsiraju u odnosu na specijalne karaktere i imena promenljivih:

```
<?php
$identity = 'James Bond';
$car = 'BMW';
// this would contain the string "James Bond drives a BMW"
$sentence = "$identity drives a $car <br />";
echo $sentence;
// this would contain the string "$identity drives a $car"
$sentencel = '$identity drives a $car';
echo $sentencel;
?>
```

Primer korišćenja PHP operatora

Primer štampa HTML tabelu sa originalnom cenom artikla, cenom sa uračunatim popustom, itd.

```
<html>
  <head>
  </head>
  <body>

    <?php
    // set quantity
    $quantity = 1000;
    // set original and current unit price
    $origPrice = 100;
    $currPrice = 25;
    // calculate difference in price
    $diffPrice = $currPrice - $origPrice;
    // calculate percentage change in price
    $diffPricePercent = (($currPrice - $origPrice) * 100)/$origPrice;
    ?>

    <table border="1" cellpadding="5" cellspacing="0">
      <tr>
        <td>Quantity</td>
        <td>Cost price</td>
        <td>Current price</td>
        <td>Absolute change in price</td>
        <td>Percent change in price</td>
      </tr>
      <tr>
        <td><?php echo $quantity ?></td>
        <td><?php echo $origPrice ?></td>
        <td><?php echo $currPrice ?></td>
        <td><?php echo $diffPrice ?></td>
        <td><?php echo $diffPricePercent ?>%</td>
      </tr>
    </table>

  </body>
</html>
```

Primer konkatencije stringova operatorom “.”

```
<?php
// set up some string variables
$a = 'the';
$b = 'games';
$c = 'begin';
$d = 'now';

// combine them using the concatenation operator
// this returns 'the games begin now<br />'
$statement = $a.' '.$b.' '.$c.' '.$d.'<br />';
print $statement;

// and this returns 'begin the games now!'
$command = $c.' '.$a.' '.$b.' '.$d.'!';
print $command;
?>
```

Obrada podataka iz formi

Izračunavanje matematičkih izraza i manipulacije stringovima su svakako korisni, ali to je nešto što se ne mora obavljati na strani servera, već se može uraditi i iz samog browser-a, pomoću jezika *JavaScript*. Serversko programiranje dolazi do izražaja onda kada je potrebno obrađivati podatke iz HTML formi, npr. unositi ih u bazu podataka.

PHP ima direktnu vezu sa podacima iz forme koji se prenose preko HTTP metoda POST ili GET:

forma.html:

```
<html>
<head></head>
<body>
<form action="message.php" method="post">
Enter your message: <input type="text" name="msg" size="30">
<input type="submit" value="Send">
</form>
</body>
</html>
```

message.php

```
<html>
<head></head>
<body>
```

```
<?php
// retrieve form data
```

```

$input = $_POST['msg'];
// use it
echo "You said: <i>$input</i>";
?>
</body>
</html>

```

Sledeći primer se ponaša slično kao prethodni, osim što se i forma i kod nalaze u jednom jedinom PHP fajlu i kao HTTP protokol se koristi protokol GET. Prilikom korišćenja GET protokola, vrednosti polja iz forme će se naći u *Address bar*-u web pretraživača. U primeru je takođe primetna upotreba asocijativnog niza (mape) `$_SERVER`, i to člana `$_SERVER["PHP_SELF"]`, koji daje adresu PHP fajla iz koga je pozvan:

```

<html>
  <head>
    <title>Primer forme za unos</title>
  </head>
  <body>
    <form action="<?php echo $_SERVER['PHP_SELF'] ?>" method="get">
      Ukucati poruku: <input type="text" name="poruka" size="30">
      <input type="submit" name="posalji" value="Posalji">
    </form>
  </body>

  <?php
  // Ako je kliknuto na "Posalji"
  if (isset ($_GET['posalji']))
  {
    // prihvatiti poruku iz forme
    $input = $_GET['poruka'];
    // Pisi poruku u italic stilu
    echo "Napisali ste: <i>$input</i>";
  }
  ?>
</html>

```