

Aplikacje WWW - laboratorium

PHP.

Celem ćwiczenia jest przygotowanie prostej aplikacji internetowej wykorzystującej technologię PHP. Aplikacja pokazuje takie aspekty, obsługa formularzy oraz zmiennych sesyjnych.

Ćwiczenia można wykonać na dowolnym komputerze, którym zainstalowano serwer HTTP (np. Apache z obsługą PHP). Rozwiązania ćwiczeń omawianych w poniższym zestawie zostały przygotowane z wykorzystaniem pakietu Krasnal Serv, który jest przygotowanym dla użytkowników systemu Windows zestawem programów pozwalających na szybkie rozpoczęcie pracy z skryptami PHP. W skład pakietu wchodzi między innymi następujące, wstępnie skonfigurowane narzędzia:

- Apache 1.3.31
- PHP 5.0.2
- MySQL 3.23.58

Skrypt PHP można umieścić w pliku o rozszerzeniu zależnym od konfiguracji serwera. Zwykle są to pliki **.php**. Fragment kodu, który ma zostać zinterpretowany jako skrypt można zaznaczyć na jeden z 5 sposobów (wybór także zależy od konfiguracji serwera):

```
<? kod_php ?>

<?php kod_php php?>

<? php kod_php ?>

<script language="php"> kod_php </script>

<% kod_php %>
```

Ćwiczenie 1

Zapoznanie się z konfiguracją serwera HTTP i interpretera skryptów PHP.

1. W katalogu umożliwiającym umieszczanie interpretowanych przez serwer skryptów PHP (zazwyczaj jest to ten sam katalog w którym umieszcza się dokumenty HTML, czyli dla serwera Apache ma on zazwyczaj nazwę `public_html`, `.homepage` lub zbliżoną, dla Krasnal Serv jest to podkatalog w katalogu: `c:\usr\krasnal\www`) utwórz podkatalog `php1` a w nim plik o nazwie `cw_1.php`. W pliku tym umieść poniższy kod, a następnie uruchom przeglądarkę i wywołaj w niej stronę generowaną przez ten kod (dla Krasnal Serv adres będzie miał postać zbliżoną do: http://127.0.0.1/~php1/cw_1.php). Przeanalizuj wyniki działania skryptu.

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

phpinfo() - Microsoft Internet Explorer

Plik Edycja Wzrost Ułubione Narzędzia Pomoc

Wstecz Wyszukaj Ulubione

Adres http://127.0.0.1/~php1/cw_1.php

Przejdź

PHP Version 5.0.2



System	Windows NT MASEL-ACER 5.1 build 2600
Build Date	Sep 24 2004 01:24:24
Configure Command	cscrip/nologo configure.js "--enable-snapshot-build" "--with-gd=shared"
Server API	Apache
Virtual Directory Support	enabled
Configuration File (php.ini) Path	C:\WINDOWS\php.ini
PHP API	20031224
PHP Extension	20040412
Zend Extension	220040412
Debug Build	no
Thread Safety	enabled
IPv6 Support	enabled
Registered PHP Streams	php, file, http, ftp, compress.zlib
Registered Stream Socket Transports	tcp, udp

Internet

Ćwiczenie 2

W tym ćwiczeniu zostanie utworzony skrypt wyświetlający w przeglądarce tekst powitalny i formularz logowania do serwisu.

1. Stwórz plik **cw_2.php** i umieść w nim poniższy kod (odwołanie do poprzedniego skryptu – **cw_1.php** jest celowe !!!).

```
<html>
<head>
<title>PHP - cwiczenie 2</title>
</head>
<body>

Witaj<p>

<form name="login_form" action="cw_1.php" method="get">
<p>
  nazwa:<input type="text" name="nazwa" maxlength="20" size="20"/>
</p>
<p>
  hasło:
  <input type="password" name="haslo" maxlength="20" size="20"/>
</p>
<p>
  <input type="submit" name="zaloguj" value="zaloguj"/>
</p>
</form>

</body>
</html>
```

2. Spróbuj go przeanalizować i wskazać fragmenty kodu w języku PHP (A może ich w nim nie ma? Dlaczego?).
3. Uruchom przeglądarkę, następnie wypełnij formularz danymi i naciśnij przycisk „zaloguj”. Przeanalizuj otrzymane wyniki. W jakich tablicach zastały umieszczone przesłane dane?
4. Zmodyfikuj kod, w taki sposób, aby teraz do przesyłania danych na serwer wykorzystywał metodę POST. Jak się zmieniły wyniki?

phpinfo() - Microsoft Internet Explorer

Plik Edycja Widok Ulubione Narzędzia Pomoc

Wstecz Wyszukaj Ulubione

Adres http://127.0.0.1/~php1/cw_1.php Przejdź

PHP Variables

Variable	Value
PHP_SELF	/~php1/cw_1.php
_REQUEST["nazwa"]	aaaa
_REQUEST["haslo"]	aaaa
_REQUEST["zaloguj"]	zaloguj
_POST["nazwa"]	aaaa
_POST["haslo"]	aaaa
_POST["zaloguj"]	zaloguj
_SERVER["COMSPEC"]	C:\WINDOWS\system32\cmd.exe
_SERVER["CONTENT_LENGTH"]	37
_SERVER["CONTENT_TYPE"]	application/x-www-form-urlencoded
_SERVER["DOCUMENT_ROOT"]	c:/usr/apache/httpd/html/
_SERVER["HTTP_ACCEPT"]	*/*
_SERVER["HTTP_ACCEPT_ENCODING"]	gzip, deflate
_SERVER["HTTP_ACCEPT_LANGUAGE"]	pl
_SERVER["HTTP_CACHE_CONTROL"]	no-cache
_SERVER["HTTP_CONNECTION"]	Keep-Alive
_SERVER["HTTP_HOST"]	127.0.0.1

Gotowe Internet

Ćwiczenie 3

1. Zmodyfikuj kod poprzedniego skryptu w taki sposób aby wysyłała dane do skryptu: cw_3.php
2. Stwórz plik **cw_3.php** i umieść w nim poniższy kod

```
<html>
<head>
<title>PHP - cwiczenie 2</title>
</head>
<body>

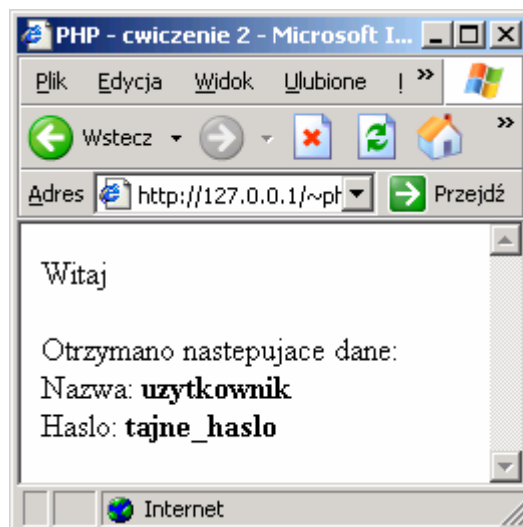
Witaj<p>

Otrzymano nastepujace dane:<br>

<?php
echo "Nazwa: <b>". $_POST["nazwa"]. "</b> <br>";
echo "Haslo: <b>". $_POST["haslo"]. "</b> <br>";

php?>

</body>
</html>
```



3. Uzupełnij powyższy kod o sprawdzenie, czy hasło i nazwa użytkownika są sobie równe i jeżeli tak jest, ustaw zmienną sesyjną **\$czy_zalogowany** na wartość **TAK**, wyświetl informacje o poprawnym zalogowaniu oraz wyświetl link pozwalający na skorzystanie z dalszej części serwisu. Jeżeli przekazane dane różnią się między sobą, to wyświetl informacje o niepoprawnym logowaniu oraz link pozwalający na podjęcie kolejnej próby.

```
<?php
session_start();
php?>
<html>
<head>
<title>PHP - cwiczenie 3</title>
</head>
<body>

Witaj<p>

Otrzymano następujące dane:<br>

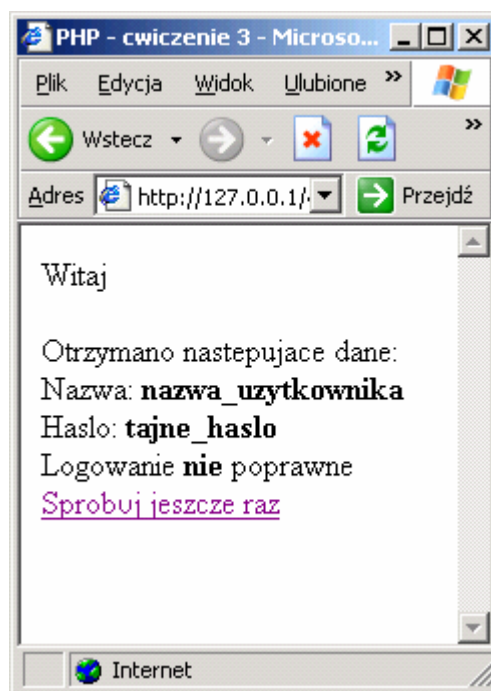
<?php
echo "Nazwa: <b>". $_POST["nazwa"]."</b> <br>";
echo "Hasło: <b>". $_POST["haslo"]."</b> <br>";

session_start();

if (($_POST["nazwa"]==$_POST["haslo"]) && isset($_POST["haslo"]) &&
($_POST["haslo"]<>''))
{
    echo 'Logowanie poprawne <br>';
    echo '<a href="cw_4.php">Dalej</a> <br>';
    $_SESSION['zalogowany'] = "tak";
    $_SESSION['zalogowany_uzytkownik'] = $_POST["nazwa"];
}
else
{
    echo 'Logowanie <b>nie</b> poprawne<br>';
    echo '<a href="cw_2.php">Spróbuj jeszcze raz</a> <br>';
    unset($_SESSION['zalogowany']);
    unset($_SESSION['zalogowany_uzytkownik']);
}

php?>

</body>
</html>
```



Ćwiczenie 4

Do samodzielnego wykonania. Stwórz plik **cw_4.php** następującej funkcjonalności:

1. Skrypt sprawdza, czy użytkownik jest zalogowany – ustawiona wartość zmiennej sesyjnej: zalogowany.
2. Zalogowanemu użytkownikowi wyświetlamy „spersonalizowane” powitanie: „Witaj nazwa_uzytkownika” oraz formularz będący cztero działaniowym kalkulatorem (dwa pola do wprowadzania liczb i lista pozwalająca wybrać działanie). Akcją dla tego formularza jest ten sam skrypt (wykorzystaj zmienną `$PHP_SELF`). Przed wyświetleniem formularza sprawdź jeszcze, czy wywołanie strony nie jest przypadkiem efektem wypełnienia formularza, wtedy po powitaniu wyświetl wynik działania a dopiero po nim formularz.
3. Jeżeli użytkownik nie jest zalogowany to korzystając z nagłówka w języku HTML: `REFRESH` – przekieruj go na stronę logowania (cw_2.php).

