

**Борислав Панайотов Стоянов**

**ПРОГРАМИРАНЕ  
на PHP**



# УПРАЖНЕНИЯ

## 1. PHP и HTML форми

Много често се налага да се създава HTML форма, която потребителя може да попълни, а после данните събрани от формата подлежат на обработка с помощта на PHP. В този урок ще се симулира малък бизнес web сайт, в който има проста форма за поръчки.

Нека това е магазин за продажба на детски стоки. Продават се пъзели, топки и обръчи. За да събере информация за това какво иска да поръча посетителя на сайта от наша гледна точка трябва да създадем страница с HTML форма за попълване.

### 1.1. Създаване на HTML форма

Създава се HTML форма с възможност за потребителски избор на продукт. Кодът се запамята във файл `order.html`.

#### **order.html**

```
<html>
  <body>
    <h4>Магазин за детски стоки</h4>
    <form action="process.php" method="post">
      <select name="item">
        <option>Пъзел</option>
        <option>Топка</option>
        <option>Обръч</option>
      </select>
      Количество: <input name="quantity" type="text" />
      <input type="Поръчай" />
    </form>
  </body>
</html>
```

След като файл `order.html` е създаден е необходимо създаването на файл `process.php`, който да обработва информацията от HTML формата.

## 1.2. PHP обработка на HTML форма

Въведеното за `item` и `quantity`, определено в HTML форма трябва да се обработи. С помощта на асоциативен масив може да се вземе информацията от `$_POST` асоциативния масив.

Правилният начин за получаването на тази информация е да се създадат две нови променливи, `$item` и `$quantity` и да им се зададе тяхната стойност да е еквивалентна на стойността, която е „публикувана” (`posted`). Името на този файл ще е `process.php`.

### `process.php`

```
<html>
  <body>
    <?php
      $quantity = $_POST['quantity'];
      $item = $_POST['item'];
      echo "Вие поръчахте ". $item . ". Количество: " . $quantity . ".<br />";
      echo "Благодарим Ви за поръчката!";
    ?>
  </body>
</html>
```

Полето `name` в `$_POST['name']` отговаря на името, което е определено в HTML формата.

Сега качете файловете `order.html` и `process.php` на Web сървър с поддръжка на PHP и изпробвайте какво ще се случи. Ако някой е избрал предмета Обръч и е определил количество 6, `process.php` трябва да покаже следното:

```
Вие поръчахте Обръч. Количество: 6.
Благодарим Ви за поръчката!
```

## 2. Login форма

Често в web сайтовете се налага да се дава ограничен достъп до някои части от сайта, например до определени директории или страници. В този урок ще се изработи проста login форма за достъп до определени места от сайт, защитени с парола и потребителско име.

### `logform.html`

```
<html>
  <head>
    <title>Login форма</title>
```

```

<body>
  <h2 align="center">Login форма</h2>
  <form action="login.php" method="post">
    <table align="center" border="0">
      <tr>
        <td>Потребител: </td>
        <td><input type="text" name="potrebitel" /></td>
      </tr>
      <tr>
        <td valign="top">Парола: </td>
        <td><input type="password" name="parola" /><br />
          <input type="submit" value="Влез">
        </td>
      </tr>
    </table>
  </form>
</body>
</html>

```

Сега е необходимо във файл login.php да се изработи PHP скрипт, който ще обработи постъпила информация от потребител.

### login.php

```

<?php

$user = $_POST["potrebitel"];
$pass = $_POST["parola"];

if ($user == "user1" && $pass == "pass1")
{
  header("location: http://domain.com/secretpage.php");
}
else
{
  echo "<b>Грешно потребителско име или парола!</b>";
}

?>

```

### secretpage.php

```

<?php

echo "Секретна страница";

?>

```

В горния код чрез масива `$_POST` се взема от формата съдържанието на полетата за потребителско име и парола като в квадратните скоби се задават ключовете за тези стойности, т.е. имената на полетата, които са `potrebitel` и `parola`. Присвояват се като стойности съответно на променливите `$user` и `$pass`.

След това чрез конструкцията `if-else` се задава на скрипта следната задача:

Ако е изпълнено условието стойността на променливата `$user` да е `user1` и стойността на променливата `$pass` да е `pass1`, тогава отвори уеб адреса `secretpage.php`. Ако това условие не е изпълнено, тогава покажи съобщението "Грешно потребителско име или парола!".

Това означава, че ако използвате в `Login` формата потребителско име `user1` и парола `pass1` в браузъра ви ще се зареди страницата, до която желаете да има ограничен достъп. В примера тази страница е с название `secretpage.php`, а адреса и е `http://domain.com/secretpage.php`, където `domain.com` е вашия домейн.

Ако искате достъпът да е с друго потребителско име и парола трябва да се заменят `user1` и `pass1` с някакви други съчетания от букви и/или цифри.

За редиректване на браузъра се използва функцията `header()`, чиито синтаксис е

```
header(Location: URL)
```

На мястото на уеб адреса (URL) може да се изпише пълният адрес на страницата, която трябва да се зареди (както е в примера), а може да се укаже и само името на страницата, ако тя се намира в същата папка (директория), в която се намират формата и скрипта. Тогава редиректването ще е във вида:

```
header("location: secretpage.php");
```

В този пример са зададени само една парола и едно потребителско име за достъп до защитената страница. С помощта на конструкцията `switch` може да направим множество акаунти за различни потребители, които да имат достъп до различни части от сайта и да ползват за целта различни потребителски имена и пароли. В този случай скриптът `login.php` може да изглежда по следния начин:

## login.php

```
<?php
```

```
$user = $_POST["potrebite!"];  
$pass = $_POST["parola"];  
switch(true)  
{  
  case ($user == "user1" && $pass == "pass1"):  
    header("location: http://domain.com/secretdir1/secretpage1.php");break;  
  
  case ($user == "user2" && $pass == "pass2"):  
    header("location: secretdir2/secretpage2.php");break;  
  
  case ($user == "user3" && $pass == "pass3"):  
    header("location: anothersecretpage.php");break;  
  
  default: echo "<b>Грешно потребителско име или парола</b>";  
}  
?>
```

Чрез горния скрипт създадохме 3 акаунта за достъп с потребителски имена и пароли съответно user1/pass1, user2/pass2 и user3/pass3, които водят до различни страници от сайта.

## Литература

- 1.