

APLIKACJE INTERNETOWE PHP



TEMAT 2-01: Obsługa podstawowych kontrolek formularza.

Autor dokumentu: Wojciech Galiński

piątek, 7 października 2016 r.

351203 Technik informatyk

ŹRÓDŁA WIEDZY: <http://www.php.net/manual/pl>, <http://webmaster.helion.pl/index.php/kurs-php>, <http://pl.wikipedia.org/>, <http://pl.wikibooks.org/wiki/PHP>, <http://phpkurs.pl/>, <http://kursphp.com/>.

Zagadnienia obowiązkowe

1. **Obsługa formularza HTML w języku PHP** – aby kontrolki formularza mogły zostać prawidłowo przesłane do skryptu PHP, należy je zdefiniować w jeden z poniższych sposobów:

```
<form action="skrypt.php"> </form>
<form action="skrypt.php" method="get"> </form>
<form action="skrypt.php" method="post"> </form>
<form action="skrypt.php" method="post" enctype="multipart/form-data"> </form>
```

Pierwsza i druga definicja powodują przesyłanie danych z formularza do skryptu PHP metodą „HTTP GET” (zagnieżdżone w adresie skryptu PHP).

Przed wybraniem metody „HTTP GET” należy rozważyć jej dwa poważne ograniczenia:



- ➔ przesyłane dane są w pełni jawne (są widoczne w przeglądarce na pasku adresu);
- ➔ długość przesyłanych danych jest ograniczona (różnie dla poszczególnych przeglądarek).

Trzecia i czwarta definicja powodują przesyłanie danych z formularza do skryptu PHP metodą „HTTP POST”. Ostatnia definicja jest wymagana do przesyłania plików do serwera.

2. **Wartości domyślne formularzy i jego kontrolki** – stosować ich jak najmniej, ponieważ w różnych przeglądarkach, a nawet w różnych wersjach tych samych przeglądarek mogą obowiązywać inne wartości domyślne.

Unikając stosowania wartości domyślnych dbamy o to, żeby wszystkie przeglądarki internetowe traktowały tak samo kod HTML naszej strony internetowej.

3. **Obsługa kontrolki tekstowych** – oto krótka charakterystyka wymienionych kontrolki:


Kontrolka	Opis
<pre><input type="text" value="Anna" /></pre> 	Pole tekstowe pozwala wprowadzić użytkownikowi jednowierszowy tekst.
<pre><input type="hidden" name="n" value="v" /></pre>	Pole ukryte służy do przesyłania danych, których nie widzi zwykły użytkownik (generowane często przez inny skrypt PHP).
<pre><input type="password" value="Anna" /></pre> 	Pole hasła służy do przesyłania hasła – zamiast wprowadzanego tekstu, widać kropki.

- ➔ **przygotowanie formularza do wysłania danych oraz odebranie danych w skrypcie PHP** – kontrolki muszą znajdować się w ramach formularza HTML, a ich nazwy w ramach formularza muszą być unikalne.

PRZYKŁADY:

Kod formularza HTML (HTTP GET)	Kod formularza HTML (HTTP POST)	Kod skryptu PHP
<pre><form action="skrypt.php" method="get"> Podaj imię: <input name="imie" value="Anna" /> <button type="submit">OK</button> </form></pre>	<pre><form action="skrypt.php" method="post"> Podaj imię: <input name="imie" value="Anna" /> <button type="submit">OK</button> </form></pre>	<pre><?php print_r(\$_GET); print_r(\$_POST); echo 'imie='. \$_GET['imie']; echo 'imie='. \$_POST['imie']; ?></pre>
Kod formularza HTML (HTTP POST)		Kod skryptu PHP
<pre><form name="nazwa_formularza" action="#" method="post"> <input type="text" name="t" value="tekst" /> <input type="hidden" name="h" value="ukryte" /> <input type="password" name="p" value="hasło" /> <button type="submit">Prześlij dane</button> </form></pre>		<pre><?php \$t = (isset(\$_POST['t'])? \$_POST['t']: ''); \$h = (isset(\$_POST['h'])? \$_POST['h']: ''); \$p = (isset(\$_POST['p'])? \$_POST['p']: ''); echo "<p>text=\$t ,hidden=\$h, password=\$p</p>"; ?></pre>

4. Obsługa kontrolki obszaru tekstowego – oto krótka charakterystyka tej kontrolki:

Kontrolka	Opis
<pre><textarea name="opis"> Co by tu napisać...</textarea></pre> 	Obszar tekstowy pozwala wprowadzić użytkownikowi wielowierszowy tekst.

- ➔ przygotowanie kontrolki „<textarea></textarea>” do wysłania danych oraz odebranie tych danych w skrypcie PHP – w formularzy należy zdefiniować kontrolkę tego typu i nadać jej jakąś nazwę (za pomocą atrybutu „name”). Po prawidłowym przesłaniu danych z formularza, w skrypcie pojawią się one w tablicy „\$_POST” (albo w tablicy „\$_GET”, jeśli do przesłania danych użyto metody GET).

PRZYKŁADY:

Nr	Kod formularza HTML	Kod skryptu PHP
1	<pre><form action="#"> <textarea name="opis"></textarea> <input type="submit" value="Prześlij dane" /> </form></pre>	<pre><?php echo 'opis=' . \$_GET['opis']; ?></pre>
2	<pre><form name="nazwa_formularza" action="skrypt.php" method="post"> <textarea name="nazwa_ta"> Domyślny tekst</textarea> <input type="submit" value="Prześlij dane" /> </form></pre>	<pre><?php echo 'nazwa_ta=' . \$_POST['nazwa_ta']; ?></pre>

Zadania

1. Zdefiniuj formularz logowania (pola: nazwa użytkownika, hasło, przycisk wysyłania danych) i utwórz skrypt weryfikujący dane z formularza (sam wybierz prawidłowe hasło).
2. Utwórz 2 odsyłacze do formularzy rejestracji: nowego klienta oraz nowego pracownika (pola: nazwa, hasło, imię, nazwisko, adres e-mail). Rodzaj nowego użytkownika przechowuj w polu ukrytym, w którym będzie wpisane, że jest to: „pracownik” albo „klient”.
3. Utwórz formularz zawierający 2 pola dla liczb. Oblicz sumę, różnicę, iloczyn, iloraz oraz potęgę liczb z formularza. Jeśli nie wprowadzono liczb, poinformuj o tym.