

6.8. Obsługa formularzy

Dynamiczna strona WWW to strona, która jest generowana na podstawie danych przekazanych między innymi przez użytkownika. Podstawowym sposobem przekazywania danych z przeglądarki internetowej do serwera jest korzystanie z formularzy HTML. Użytkownik za pomocą formularza wprowadza różne informacje, które są wysyłane do serwera. Dane te są przetwarzane na serwerze przy użyciu skryptów PHP.

6.8.1. Formularz HTML

Przykład 6.85

Prosty formularz HTML może mieć postać:

```
<!DOCTYPE HTML>

<html>

<head>

<title>Formularz rejestracyjny</title>

<meta charset="UTF-8">

<style>
```

```

p {
    font-weight: bold;
    font-size: 14pt;
}
#wyk {
    font-weight: bold;
    font-size: 11pt;
}
</style>
</head>
<body>
<form action="http://localhost/skrypt_form.php" method="post">
<p>Formularz kontaktowy:</p>
Nazwisko:<br>
<input type="text" name="nazw" value=" " size="30"><br>
Imię:<br>
<input type="text" name="im" value=" " size="30"><br>
Zawód:<br>
<input type="text" name="zaw" value=" " size="30"><br>
Adres e-mail:<br>
<input type="text" name="adr" value=" " size="30">

<p id="wyk">Wykształcenie:</p>
<input type="radio" value="podstawowe"
name="wykszt" checked> Podstawowe<br>
<input type="radio" value="średnie" name="wykszt"> Średnie<br>
<input type="radio" value="wyższe" name="wykszt"> Wyższe<br><br>

<input type="checkbox" name="opcje" maxlength="1">
Zgadzam się na przetwarzanie moich danych osobowych<br><br>
<input type="submit" value="Wyślij" name="wyslij">&nbsp; &nbsp;
<input type="reset" value="Wyczyść" name="zeruj">
</form>
</body>
</html>

```

Podany w przykładzie formularz składa się z czterech pól do wprowadzania tekstu, pola wyboru typu radio, pola wyboru typu checkbox oraz przycisków typu submit (*Wyślij*) oraz reset (*Wyczyść*) (rysunek 6.27).

Formularz kontaktowy:

Nazwisko:

Imię:

Zawód:

Adres e-mail:

Wykształcenie:

☒ Podstawowe
☐ Średnie
☐ Wyższe

☐ Zgadzam się na przetwarzanie moich danych osobowych

Rysunek 6.27. Przykładowy formularz

Po kliknięciu przycisku *Wyślij* dane z formularza powinny zostać przesłane na serwer za pomocą metody POST. Parametr `action` wskazuje adres serwera i nazwę skryptu PHP, który zostanie wywołany po zatwierdzeniu formularza. W tym przykładzie zostanie uruchomiony skrypt o nazwie *skrypt_form.php*, który znajduje się na serwerze lokalnym o adresie *http://localhost*. Dane wprowadzone do formularza zostaną przekazane do wybranego skryptu. Parametr `method` określa sposób (metodę) przesłania informacji do wskazanego skryptu. Może przyjąć jedną z dwóch wartości: `post` lub `get`. Wartością domyślną jest `get`, ale zaleca się stosowanie metody `post`.

6.8.2. Przekazywanie danych — metoda `get` i `post`

Każdy formularz zaczyna się od znacznika otwierającego `<form>` i kończy się znacznikiem zamykającym `</form>`. Pomiędzy tymi znacznikami znajdują się wszystkie polecenia dotyczące formularza. Jeżeli dane z formularza będą przesyłane na serwer, znacznik ten powinien mieć następujące atrybuty:

`action="adres"`. To atrybut, który ustala adres do wysyłki danych z formularza. Dane wprowadzone do formularza zostaną przekazane do podanego skryptu.

`method="metoda"` określa sposób (metodę) przekazania informacji do wskazanego skryptu lub na wskazany adres. Atrybut może przyjąć jedną z dwóch wartości: `post` lub `get`.

Metoda GET jest używana, kiedy parametrów jest niewiele. W tej metodzie parametry są przekazywane za pomocą adresu URL. Schematyczna postać URL wygląda następująco:

http://www.nazwa_serwera.pl/strona.php?parametr1=wartość1¶metr2=wartość2

Długość adresu URL jest ograniczona, a przekazywane parametry są widoczne w pasku adresu przeglądarki, dlatego metoda ta jest stosowana do przesyłania niedużych ilości danych. Adres skryptu PHP od przesyłanych parametrów oddzielany jest znakiem zapytania. Przesyłane pary *parametr=wartość* są oddzielane znakiem ampersand (&).

Metoda POST do przekazywania parametrów wykorzystuje nagłówek strony. Metoda ta umożliwia przekazywanie większej liczby parametrów, a parametry nie są widoczne w pasku adresu przeglądarki.

Przesłane wartości parametrów są zapisywane w odpowiednich tablicach asocjacyjnych. Dane przesłane metodą GET są zapisywane w tablicy \$_GET, natomiast dane przesłane metodą POST w tablicy \$_POST. Tablice \$_GET i \$_POST są superglobalne. Oznacza to, że są widoczne w każdym miejscu kodu PHP bez konieczności użycia składni `globals`.

Indeksami tablicy \$_GET i \$_POST są nazwy pól formularza (wartość atrybutu name w formularzu HTML), do których się odwołujemy. Odwołanie do elementów tablicy ma postać:

```
$zmienna = $_GET['nazwa_pola'];
```

lub

```
$zmienna = $_POST['nazwa_pola'];
```

Przykład 6.86

W celu sprawdzenia poprawności działania skryptu z przykładu 6.85 zostanie utworzony skrypt PHP (*skrypt_form.php*) przetwarzający dane przesłane z formularza.

```
<?php
echo "Odpowiedź z PHP: <br>".$_POST['nazw'];
echo $_POST['im']."<br>";
echo $_POST['zaw']."<br>";
echo $_POST['adr']."<br>";
echo "Zostało wybrane wykształcenie: {$_POST['wyksz']}. ";
?>
```

Jeżeli podany kod zostanie zapisany w pliku *skrypt_form.php*, to po otwarciu formularza, wpisaniu danych (rysunek 6.28) i jego wysłaniu przeglądarka powinna wyświetlić przesłane dane (rysunek 6.29).

Formularz kontaktowy:

Nazwisko:

Imię:

Zawód:

Adres e-mail:

Wykształcenie:

☐ Podstawowe
☐ Średnie
☒ Wyższe

☒ Zgadzam się na przetwarzanie moich danych osobowych

Rysunek 6.28. Formularz z wypełnionymi polami

Odpowiedź z PHP:
Nowak Paweł
Malarz
np@gmail.com
Zostało wybrane wykształcenie: wyższe.

Rysunek 6.29. Dane przesłane za pomocą formularza

Funkcja `isset`

Podczas przetwarzania formularza można wykorzystać funkcję `isset()`. Pozwala ona stwierdzić, czy określone pole formularza zostało wypełnione, a dokładniej, czy do skryptu została przekazana wartość danego pola. Funkcja zwraca `true`, gdy wartość została przekazana, w przeciwnym razie zwraca `false`.

Przykład 6.87

W przykładzie 6.85 formularz oprócz pól tekstowych zawierał również pola typu `radio`. Są to trzy pola wyboru, w których można zaznaczyć tylko jedną opcję z grupy opcji. Kliknięcie wartości powoduje zaznaczenie opcji, ale równocześnie usunięte zostaje zaznaczenie innej opcji. Każde z pól tego typu ma taką samą wartość atrybutu `name`. Atrybut `value` w tym przypadku jest konieczny i musi być inny dla każdego pola. Przez wartość atrybutu `value` pola będą identyfikowane w skrypcie.

Utworzymy skrypt PHP, który będzie rozpoznawał, jaka opcja pola typu `radio` została zaznaczona w formularzu (kod może zostać dołączony do pliku [skrypt_form.php](#)).

```

<?php
if (!isset($_POST['wykszt'])) {
    echo "<br>Proszę zaznaczyć pole <b>Wykształcenie</b>";
} else {
    echo "<br>Zostało wybrane wykształcenie: {$_POST['wykszt']}.";
}
?>

```

W podanym skrypcie do sprawdzenia, czy zaznaczone zostało pole opcji, zastosowano instrukcję warunkową `if` oraz funkcję `isset()`. W warunku `!isset($_POST['wykszt'])` został użyty operator logicznej negacji `!`, czyli gdy pole nie zostanie wypełnione, warunek zwróci wartość `true` i zostanie wysłany komunikat *Proszę zaznaczyć pole Wykształcenie*. W przeciwnym razie za pomocą składni `{$_POST['wykszt']}` zostanie odczytana z tablicy `$_POST` wartość opcji `wykszt` (będzie to wartość atrybutu `value` dla znacznika `<input type="radio">`).

Funkcja `empty`

Funkcja `empty()` sprawdza, czy istniejąca zmienna użyta jako argument jest pusta. Jeżeli zmienna jest pusta, to funkcja zwraca wartość `true`, w przeciwnym razie zwraca wartość `false`.

W formularzu może wystąpić pole typu `SELECT`, które służy do wyświetlania listy wartości i pozwala na wybranie jednej z nich lub kilku.

Przykład 6.88

```

<!DOCTYPE HTML>

<html>

<head>

<title>Formularz językowy</title>

<meta charset="UTF-8">

<style>

#wyb {

    font-weight: bold;

    font-size: 12pt;

}

</style>

</head>

<body>

<form action="skrypt_form1.php" method="post">

```

```

Nazwisko i imię:<br>
<input type="text" name="nazw" value=" " size="40"><br>

<p id="wyb">Wybór języka:</p>
<select name="języki[]" multiple>
<option value="Język angielski">Język angielski</option>
<option value="Język niemiecki">Język niemiecki</option>
<option value="Język francuski">Język francuski</option>
<option value="Język włoski">Język włoski</option>
<option value="Język rosyjski">Język rosyjski</option>
<option value="Język hiszpański">Język hiszpański</option>
</select>
<p><input type="submit" value="Wyślij" name="wyslij"></p>
</form>
</body>
</html>

```

Gdyby utworzona w formularzu lista wyboru wartości została zdefiniowana w postaci `<select name="języki" multiple>`, to skrypt obsługujący formularz miałby dostęp tylko do jednej wartości (`$_POST["języki"]`). Gdy kwadratowe nawiasy zostaną dodane w podany sposób: `<select name="języki[]">`, to elementy wybrane przez użytkownika w polu `języki[]` będą dostępne w tablicy `$_POST["języki"]`. Utworzony formularz został pokazany na rysunku 6.30.

Należy jeszcze dołączyć skrypt PHP (zapisany w pliku *skrypt_form1.php*) w postaci:

```

<?php
if (!empty($_POST['języki'])) {
    echo "<p><b>".$_POST['nazw']."'</b> zna:</p>";
    echo "<ul>";
    foreach ($_POST['języki'] as $wartosc) {
        echo "<li>$wartosc</li>";
    }
    echo "</ul>";
}

```

Rysunek 6.30.
Sposób wypełniania formularza

```

    } else {
        echo "<p><b>".$_POST['nazw']."</b> nie zna żadnego języka.</p>";
    }
?>

```

Instrukcja `if (!empty($_POST['języki']))` sprawdza, czy tablica `$_POST['języki']` jest pusta. Jeżeli nie, to nastąpi wyświetlenie zawartości każdego elementu tablicy (rysunek 6.31). W tablicy znajdują się wartości znacznika `<option value=...>` wybrane przez użytkownika.

Ta technika może być stosowana do różnych elementów, nie tylko do pól `SELECT`. Jeżeli kilku polom nadamy taką samą nazwę i nazwa pola będzie kończyła się nawiasem kwadratowym, to PHP połączy wszystkie wartości o tej samej nazwie pola w jedną tablicę.

Kowalski Artur zna:

- Język włoski
- Język hiszpański

Rysunek 6.31.

Po wystaniu formularza przeglądarka wyświetli przesłane dane

Przykład 6.89

Po wprowadzonych zmianach kod zawierający formularz HTML będzie wyglądał tak:

```

<!DOCTYPE HTML>

<html>

<head>

<title>Formularz rejestracyjny</title>

<meta charset="UTF-8">

<style>

#fo {

    font-weight: bold;

    font-size: 14pt;

}

p.wyk {

    font-weight: bold;

    font-size: 11pt;

}

</style>

</head>

<body>

<form action="skrypt_form2.php" method="post">

<p id="fo">Formularz kontaktowy:</p>

```



```
Nazwisko:<br>
<input type="text" name="nazw" value=" " size="30"><br>
Imię:<br>
<input type="text" name="im" value=" " size="30"><br>
Zawód:<br>
<input type="text" name="zaw" value=" " size="30"><br>
Adres e-mail:<br>
<input type="text" name="adr" value=" " size="30">

<p class="wyk">Wykształcenie:</p>
<input type="radio" value="podstawowe"
name="wykszt" checked> Podstawowe<br>
<input type="radio" value="średnie" name="wykszt"> Średnie<br>
<input type="radio" value="wyższe" name="wykszt"> Wyższe<br><br>

<p class="wyk">Wybór języka:</p>
<select name="języki[]" multiple>
<option value="Język angielski">Język angielski</option>
<option value="Język niemiecki">Język niemiecki</option>
<option value="Język francuski">Język francuski</option>
<option value="Język włoski">Język włoski</option>
<option value="Język rosyjski">Język rosyjski</option>
<option value="Język hiszpański">Język hiszpański</option>
</select>

<p><input type="checkbox" name="opcje" maxlength="1">
Zgadzam się na przetwarzanie moich danych osobowych</p>
<input type="submit" value="Wyślij" name="wyslij">&nbsp; &nbsp;
<input type="reset" value="Wyczyść" name="zeruj">
</form>
</body>
</html>
```

A kod zawierający skrypt PHP (plik *skrypt_form2.php*) będzie wyglądał tak:

```
<?php
echo "Dziękujemy! Rejestracja przebiegła pomyślnie.<br>";
echo "Wprowadzone dane:<br>";
echo "Nazwisko: " . trim($_POST['nazw']) . "<br>";
echo "Imię: " . trim($_POST['im']) . "<br>";
echo "Zawód: " . trim($_POST['zaw']) . "<br>";
echo "Adres e-mail: " . trim($_POST['adr']) . "<br>";

if (!isset($_POST['wyksz'])) {
    echo "Proszę zaznaczyć pole Wykształcenie";
} else {
    echo "Wykształcenie: " . trim($_POST['wyksz']) . "<br>";
}

echo "<p>Znajomość języków:</p>";

if (!empty($_POST['języki'])) {
    echo "<ul>";
    foreach ($_POST['języki'] as $wartosc) {
        echo "<li>$wartosc</li>";
    }
    echo "</ul>";
} else {
    echo "<p>".$_POST['nazw']." nie zna żadnego języka.</p>";
}
?>
```

Po wprowadzeniu danych (rysunek 6.32) formularz zostanie wysłany, a w przeglądarce wyświetli się informacja o pomyślnym zarejestrowaniu użytkownika (rysunek 6.33).

Formularz kontaktowy:

Nazwisko:

Imię:

Zawód:

Adres e-mail:

Wykształcenie:

☐ Podstawowe
☐ Średnie
☒ Wyższe

Wybór języka:

Język angielski
Język niemiecki
Język francuski
Język włoski

☒ Zgadzam się na przetwarzanie moich danych osobowych

Rysunek 6.32. Formularz z wprowadzonymi danymi

Dziękujemy! Rejestracja przebiegła pomyślnie.

Wprowadzone dane:

Nazwisko: Marecka
Imię: Anna
Zawód: Lekarz
Adres e-mail: manna@gmail.com
Wykształcenie: wyższe

Znajomość języków:

Marecka nie zna żadnego języka.

Rysunek 6.33. Wynik pobrania danych z formularza

Ćwiczenie 6.16

Opracuj formularz przeznaczony do rejestrowania uczniów zapisujących się do szkoły. Określ informacje, które powinny znaleźć się w formularzu, oraz sposób ich wprowadzania. Utwórz skrypt przetwarzający dane przesłane z formularza. W skrypcie powinno być sprawdzane wypełnienie wszystkich pól.