

Aplikacje internetowe

Laboratorium 8

Cele:

- Obsługa formularzy PHP

Zadanie

Poznanie metod GET i POST, walidacja formularzy, bezpieczeństwo.

1. Prosty formularz

Przygotuj plik form.php zawierający:

```
<html>
<body>

<form action="welcome.php" method="post">
Imię: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>

</body>
</html>
```

2. Przygotuj obsługę formularza w pliku welcome.php, przetestuj w przeglądarce:

```
<html>
<body>

Witaj <?php echo $_POST["name"]; ?><br>
Twój adres e-mail to: <?php echo $_POST["email"]; ?>

</body>
</html>
```

3. Przygotuj analogicznie wersje z obsługą formularzy metodą GET w plikach form_get.php oraz welcome_get.php, przetestuj działanie, zwróć uwagę na wywoływany adres url.

4. Walidacja formularzy. Przygotuj formularz w pliku form_valid.php zawierający:

```
<!DOCTYPE HTML>
<html>
<head>
</head>
<body>

<?php
// zdefiniowanie zmiennych i ustawienie pustych wartości
$name = $email = $gender = $comment = $website = "";
```

```
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = test_input($_POST["name"]);
    $email = test_input($_POST["email"]);
    $website = test_input($_POST["website"]);
    $comment = test_input($_POST["comment"]);
    $gender = test_input($_POST["gender"]);
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>

<h2>Przykład walidacji formularza</h2>
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    Imię: <input type="text" name="name">
    <br><br>
    E-mail: <input type="text" name="email">
    <br><br>
    Strona www: <input type="text" name="website">
    <br><br>
    Komentarz: <textarea name="comment" rows="5" cols="40"></textarea>
    <br><br>
    Płeć:
    <input type="radio" name="gender" value="female">Kobieta
    <input type="radio" name="gender" value="male">Mężczyzna
    <br><br>
    <input type="submit" name="submit" value="Submit">
</form>

<?php
echo "<h2>Twoje dane:</h2>";
echo $name;
echo "<br>";
echo $email;
echo "<br>";
echo $website;
echo "<br>";
echo $comment;
echo "<br>";
echo $gender;
?>

</body>
</html>
```

5. Sprawdź do czego służą metody trim, stripslashes oraz htmlspecialchars.
6. Pola wymagane, walidacja poprawności wpisywanych wartości. Zaktualizuj poprzedni przykład dodając odpowiednią obsługę pól:

```

$nameErr = $emailErr = $genderErr = $websiteErr = "";
$name = $email = $gender = $comment = $website = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (empty($_POST["name"])) {
        $nameErr = "Imię jest wymagany polem";
    } else {
        $name = test_input($_POST["name"]);
    }

    if (empty($_POST["email"])) {
        $emailErr = "Email jest wymagany polem";
    } else {
        $email = test_input($_POST["email"]);
    }

    if (empty($_POST["website"])) {
        $website = "";
    } else {
        $website = test_input($_POST["website"]);
    }

    if (empty($_POST["comment"])) {
        $comment = "";
    } else {
        $comment = test_input($_POST["comment"]);
    }

    if (empty($_POST["gender"])) {
        $genderErr = "Płeć jest wymagany polem";
    } else {
        $gender = test_input($_POST["gender"]);
    }
}

```

7. Popraw nieco formularz dodając obsługę wymaganych pól:

```

<h2> Przykład walidacji formularza </h2>
<p><span class="error">* pole wymagane.</span></p>
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">
    Imię: <input type="text" name="name">
    <span class="error">* <?php echo $nameErr;?></span>
    <br><br>
    E-mail: <input type="text" name="email">
    <span class="error">* <?php echo $emailErr;?></span>
    <br><br>
    Strona www: <input type="text" name="website">
    <span class="error"><?php echo $websiteErr;?></span>
    <br><br>
    Komentarz: <textarea name="comment" rows="5" cols="40"></textarea>
    <br><br>
    Płeć:
    <input type="radio" name="gender" value="female">Kobieta
    <input type="radio" name="gender" value="male">Mężczyzna

```

```
<span class="error">* <?php echo $genderErr;?></span>
<br><br>
<input type="submit" name="submit" value="Submit">
</form>
```

8. Dodaj klasę stylu CSS: `.error {color: #FF0000;}`

9. Dodaj obsługę poprawności wpisywania imienia, adresu e-mail oraz adresu strony www, uzupełnij odpowiednio kod poprzedniego przykładu:

```
if (empty($_POST["name"])) {
    $nameErr = "Imię jest polem wymagany";
} else {
    $name = test_input($_POST["name"]);
    // sprawdzamy czy pole name zawiera tylko litery i spacje
    if (!preg_match("/^[a-zA-Z ]*$/",$name)) {
        $nameErr = "Tylko litery i spacje są dozwolone";
    }
}

if (empty($_POST["email"])) {
    $emailErr = "Email jest polem wymagany";
} else {
    $email = test_input($_POST["email"]);
    // sprawdzamy czy pole email zawiera poprawnie skonstruowany adres e-mail
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $emailErr = "Nieprawidłowy format adresu email";
    }
}

if (empty($_POST["website"])) {
    $website = "";
} else {
    $website = test_input($_POST["website"]);
    // sprawdzamy czy pole email zawiera poprawnie skonstruowany adres URL
    if (!preg_match("/\b(?:https?|ftp):\/\/\w+([a-z0-9+&@#\/%?~_!:\.,;]*[-a-z0-9+&@#\/%?~_!:\.,;])\/i",$website)) {
        $websiteErr = "Nieprawidłowy adres URL";
    }
}
}
```

10. Zapoznaj się z funkcjami `preg_match` oraz `filter_var`.

11. Zachowanie podawanych wartości w polach formularza. Zaktualizuj przykład uzupełniając odpowiednio kod:

```
Imię: <input type="text" name="name" value="<?php echo $name;?>">
<span class="error">* <?php echo $nameErr;?></span>
<br><br>
E-mail: <input type="text" name="email" value="<?php echo $email;?>">
<span class="error">* <?php echo $emailErr;?></span>
<br><br>
Strona www: <input type="text" name="website" value="<?php echo $website;?>">
<span class="error"><?php echo $websiteErr;?></span>
```

```
<br><br>
Komentarz: <textarea name="comment" rows="5" cols="40"><?php echo $comment;?></tex
tarea>
<br><br>
Płeć:
<input type="radio" name="gender" <?php if (isset($gender) &&
$gender=="female") echo "checked";?> value="Kobieta">Kobieta
<input type="radio" name="gender" <?php if (isset($gender) &&
$gender=="male") echo "checked";?> value="Mężczyzna">Mężczyzna
```

12. Na podstawie poprzedniego laboratorium, przygotuj bazę z tabelą przechowującą dane z powyższego formularza. Utwórz skrypty umożliwiające dodawanie informacji z formularza do bazy.