



DEPARTMENT OF
COMPUTER SCIENCE AND ENGINEERING

Title: PHP form handling and Introduction of MySQL

WEB PROGRAMMING LAB
CSE 302



GREEN UNIVERSITY OF BANGLADESH

1 Objective(s)

- To implement form handling and validation in PHP
- To know the basic working knowledge of MYSQL database.

2 Problem analysis

The PHP superglobals are used to collect form-data. When the user fills out the form and clicks the submit button, the form data is sent for processing to a PHP file. The form data is sent with the HTTP GET or POST method.

3 Implementation in PHP

```
1 <!DOCTYPE HTML>
2
3 <html>
4
5 <head>
6
7     <style>
8
9         .error {color: #FF0000;}
10
11     </style>
12
13 </head>
14
15 <body>
16
17     <?php
18
19         $nameErr = $emailErr = $genderErr = $websiteErr = "";
20
21         $name = $email = $gender = $comment = $website = "";
22
23         if ($_SERVER["REQUEST_METHOD"] == "POST") {
24
25             if (empty($_POST["name"])) {
26
27                 $nameErr = "Please enter a valid name";
28
29             } else {
30
31                 $name = test_input($_POST["name"]);
32
33                 // check if name only contains letters and whitespace
34
35                 if (!preg_match("/^[a-zA-Z-' ]*$/",$name)) {
36
37                     $nameErr = "Only letters and white space allowed";
38
39                 }
40
41             }
42
43             if (empty($_POST["email"])) {
```

```

44
45     $emailErr = "valid Email address";
46
47 } else {
48
49     $email = test_input($_POST["email"]);
50
51     // check if e-mail address is well-formed
52
53     if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
54
55         $emailErr = "The email address is incorrect";
56
57     }
58
59 }
60
61 if (empty($_POST["website"])) {
62
63     $website = "";
64
65 } else {
66
67     $website = test_input($_POST["website"]);
68
69     // check if URL address syntax is valid
70
71     if (!preg_match("/\b(?:(:https?|ftp):\\/\|www\.)[-a-z0-9+&@#\\/%?=_~_
72         |!:,.;]*[-a-z0-9+&@#\\/%?=_~_]/i",$website)) {
73
74         $websiteErr = "Enter a valid Webiste URL";
75
76     }
77
78 }
79
80 if (empty($_POST["comment"])) {
81
82     $comment = "";
83
84 } else {
85
86     $comment = test_input($_POST["comment"]);
87
88 }
89
90 if (empty($_POST["gender"])) {
91
92     $genderErr = "Please select a gender";
93
94 } else {
95
96     $gender = test_input($_POST["gender"]);
97
98 }
99
100

```

```

101     function test_input($data) {
102
103         $data = trim($data);
104
105         $data = stripslashes($data);
106
107         $data = htmlspecialchars($data);
108
109         return $data;
110     }
111
112     ?>
113
114     <h2>PHP Form Validation Example</h2>
115
116     <p><span class="error">* required field</span></p>
117
118     <form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]
119         );?>">
120
121         FullName: <input type="text" name="name">
122
123         <span class="error">* <?php echo $nameErr;?></span>
124
125         <br><br>
126
127         E-mail address: <input type="text" name="email">
128
129         <span class="error">* <?php echo $emailErr;?></span>
130
131         <br><br>
132
133         Website: <input type="text" name="website">
134
135         <span class="error"><?php echo $websiteErr;?></span>
136
137         <br><br>
138
139         Comment: <textarea name="comment" rows="2" cols="10"></textarea>
140
141         <br><br>
142
143         Gender:
144
145         <input type="radio" name="gender" value="female">Female
146
147         <input type="radio" name="gender" value="male">Male
148
149         <span class="error">* <?php echo $genderErr;?></span>
150
151         <br><br>
152
153         <input type="submit" name="submit" value="Submit">
154
155     </form>
156
157     <?php

```

```

158
159     echo "<h2> Final Output:</h2>";
160
161     echo $name;
162
163     echo "<br>";
164
165     echo $email;
166
167     echo "<br>";
168
169     echo $website;
170
171     echo "<br>";
172
173     echo $comment;
174
175     echo "<br>";
176
177     echo $gender;
178
179     ?>
180
181 </body>
182
183 </html>

```

4 Input/Output

Output of the program is given below.

PHP Form Validation Example

* required field

FullName: *

E-mail address: *

Website: Enter a valid Webiste URL

Comment:

Gender: ☐ Female ☐ Male *

Final Output:

Arpita Chowdhury
 arpita@cse.green.edu.bd
 somewhere
 female

Figure 1: PHP form validation

5 Discussion & Conclusion

Based on the focused objective(s) to understand about PHP form handling and validation, the additional lab exercise made me more confident towards the fulfilment of the objectives(s).

6 Lab Task (Please implement yourself and show the output to the instructor)

1. Implement form validation using PHP.

6.1 Problem analysis

Create a PHP validation file considering string validation, email validation and password validation.

7 Lab Exercise (Submit as a report)

- Implement form validation in PHP.

8 Policy

Copying from internet, classmate, seniors, or from any other source is strongly prohibited. 100% marks will be *deducted* if any such copying is detected.