

FORM VALIDATION USING PHP

Forms:

HTML forms can have many attributes. Two of the most important attributes are “action” and “method”.

Action:

The action parameter tells the browser what script/site must be called when the visitor presses the submit button.

Method:

The method parameter tells the browser by which method to send the user submitted data to the web-server. The parameter values are either “**POST**” or “**GET**”.

Form Validation:

- Whenever you make a form you should not leave it alone without any form validation.
- Because there is no guarantee that the input is correct and processing incorrect input values can make your application give unpredictable result.
- You can validate the form input on two places, **client side** and **server side**.
- Client side form validation usually done with javascript.
- Server side form validation done with **PHP**.

Form Processing:

- Superglobal arrays are associative arrays predefined by PHP that hold variables acquired from user input, the environment or the web server and are accessible in any variable scope.
- The arrays `$_GET` and `$_POST` retrieve information sent to the server by HTTP get and post requests, respectively.
- These are super globals, which means that they are always accessible, regardless of scope and you can access them from any function.

- **Some useful superglobal arrays:**

Variable name	Description
<code>\$_SERVER</code>	Data about the currently running server.
<code>\$_ENV</code>	Data about the client's environment.
<code>\$_GET</code>	Data sent to the server by a get request.
<code>\$_POST</code>	Data sent to the server by a post request.
<code>\$_COOKIE</code>	Data contained in cookies on the client's computer.
<code>\$GLOBALS</code>	Array containing all global variables.

- The `$_SERVER["PHP_SELF"]` is a super global variable that returns the filename of the currently executing script.
- The **htmlspecialchars()** function converts special characters to HTML entities .That it will replace HTML characters like < and > with < and > .This prevents attackers from exploiting the code by injecting HTML or javascript code(cross-site scripting attacks) in forms.
- We will also do two more things when the user submits the form:
 - 1.Strip unnecessary characters (extra space,tab,newline) from the user input data (with the PHP **trim()** function).
 - 2.Remove backslashes(\) from the user input data(with the PHP **stripslashes()** function).
- **preg_match(regex,string)**-returns True if string matches regex.
Preg stands for perl regular expressions.
- **isset()**-This function returns True if the variable exists and is not NULL.

EXAMPLE:

//save with formvalid.php

```
<!DOCTYPE HTML>
<html>
<head>
<style>
.error {color: #FF0000;}
</style>
</head>
<body>
<?php
$nameErr = $emailErr = $genderErr = $websiteErr = "";
$name = $email = $gender = $comment = $website = "";
if ($_SERVER["REQUEST_METHOD"] == "POST")
{
    if (empty($_POST["name"]))
    {
        $nameErr = "Name is required";
    }
    else
    {
        $name = test_input($_POST["name"]);
        if (!preg_match("/^[a-zA-Z-' ]*$/", $name))
        {
            $nameErr = "Only letters and white space allowed";
        }
    }
    if (empty($_POST["email"]))
    {
        $emailErr = "Email is required";
    }
    else
    {
        $email = test_input($_POST["email"]);
        if (!filter_var($email, FILTER_VALIDATE_EMAIL))
        {
            $emailErr = "Invalid email format";
        }
    }
    if (empty($_POST["website"]))
    {
        $website = "";
    }
    else
    {
        $website = test_input($_POST["website"]);
```

```

        if (!preg_match("/^b(?:(:https?|ftp):\\V|www\\.)*[-a-z0-9+&@#\\%?~_!:,;]*[-a-z0-9+&@#\\%?~_]/i",$website))
        {
            $websiteErr = "Invalid URL";
        }
    }
    if (empty($_POST["comment"]))
    {
        $comment = "";
    }
    else
    {
        $comment = test_input($_POST["comment"]);
    }
    if (empty($_POST["gender"]))
    {
        $genderErr = "Gender is required";
    }
    else
    {
        $gender = test_input($_POST["gender"]);
    }
}

```

```

function test_input($data)

```

```

{
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}

```

```

?>

```

```

<h2>PHP Form Validation Example</h2>

```

```

<p><span class="error">* required field</span></p>

```

```

<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>">

```

```

Name: <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>

```

```

Website: <input type="text" name="website" value="<?php echo $website;?>">

```

```

<span class="error"><?php echo $websiteErr;?></span>

```

```

<br><br>

```

```

Comment: <textarea name="comment" rows="5" cols="40"><?php echo

```

```

$comment;?></textarea>

```

```

<br><br>

```

```

Gender:

```

```

<input type="radio" name="gender" <?php if (isset($gender) &&$gender=="female")
echo "checked";?> value="female">Female

```

```

<input type="radio" name="gender" <?php if (isset($gender) && $gender=="male")

```

```

echo "checked";?> value="male">Male
<input type="radio" name="gender" <?php if (isset($gender) && $gender=="other")
echo "checked";?> value="other">Other
<span class="error">* <?php echo $genderErr;?></span>
<br><br>
<input type="submit" name="submit" value="Submit">
</form>
<?php
echo "<h2>Your Input:</h2>";
echo $name;
echo "<br>";
echo $email;
echo "<br>";
echo $website;
echo "<br>";
echo $comment;
echo "<br>";
echo $gender;
?>
</body>
</html>

```

output:

Activities Google Chrome Apr 16 7:02 PM

localhost/form.php x +

localhost/form.php

PHP Form Validation Example

* required field

Name: * Only letters and white space allowed

E-mail: * Invalid email format

Website: Invalid URL

Comment:

Gender: ☐ Female ☐ Male ☐ Other * Gender is required

Your Input:

satya-209
s190209@rguktsklm
www.rguktsklm

