## PHP

With pure **HTML web pages**, the server simply serves **static HTML** that only display content. PHP is used to turn the web sites into interactive web application.

  



Fig 1: Web Server 🡪 Static HTML File 🡪 Client Web Browser

With **PHP** in the mix, the web server is able to **dynamically generate HTML** web pages

   





Fig 2: Web Server 🡪 PHP 🡪 MySQL 🡪 PHP 🡪 Client Web Server

**PHP**: **Personal Home Pages**, it was later changed **PHP Hypertext Processor**.

**PHP**: **Server-side programming language** – it runs on a web server.

### Simple Form(Form.html)

<!doctype html>

<**html**>

<**head**>

<**meta** charset = "UTF-8"/>

<**title**> Simple Form </**title**>

</**head**>

<**body**>

<**p**> Please fill out the form </**p**>

<**img** src = "database.png" width = "100" height = "175"

alt = "database picture"/><**br**/>

<**form** method = "post" action = "khan.photon@gmail.com">

<**label** for = "firstname"> First Name: </**label**>

<**input** type = "text" id = "filename" name = "firstname"/><**br**/>

<**label** for = "lastname"> Last Name: </**label**>

<**input** type = "text" id = "lastname" name = "lastname"/><**br**/>

<**label** for = "email"> Email: </**label**>

<**input** type = "email" id = "email" name = "email"/><**br**/>

<**label** for = "gender">Gender: </**label**>

Male <**input** type = "radio" id ="gender" name = "gender" value = "male">

Female <**input** type = "radio" id = "gender" name = "gender"

value = female"><**br**/>

<**label** for = "description"> Description: </**label**>

<**textarea** name = "description"> Max word 500 words </**textarea**><**br**/>

<**input** type = "submit" value = "Submit" name = "submit"/>

</**form**>

</**body**>

</**html**>

Code 1 (Form.html): Simple Form HTML Code

**Tags:** <**form**></**form**>, <**input></input>,** <**label></label>**

**Attributes**: action, type, id, method, name, class, value

### Breakdown(Form.html)

Basic HTML Format

<!doctype html>

<**html**>

<**head**>

<**meta** charset = "UTF-8"/>

<**title**> Simple Form </**title**>

</**head**>

<**body**>

</**body**>

</**html**>

|  |  |
| --- | --- |
| **Labels/Tags** | **Description** |
| <!doctype html> | A document type declaration |
| <**html**> | Tells the browser that this is an HTML document |
| <**head**></**head**> | Includes title for the document, scripts, styles, meta information |
| <**meta** charset="UTF-8"/> | Specifies what character set is our website written with |
| "UTF-8" | Universal Character Set + Transformation Format 8-bit |
| <**body**></**body**> | This is where all the HTML contents are written |
| <**p**></**p**> | Paragraph |
| <**img** src="database.png"/> | Include Image, Attribute src is the image src path |
| alt="database picture" | Alternatively, text will show up if no image found |
| width="100" height="175" | Fixing the width and height of the image |

<**form** method = "post" action = "khan.photon@gmail.com">

Form action attribute should be changed

<**form** method = "post" action = "report.php"></**form**>

Form tag causes the PHP script to run on the server.

|  |  |
| --- | --- |
| **Labels/Tags** | **Description** |
| <**input><**/**input>** | Input tag is the input field where user can enter data |
| type="text | Attribute type is the value received and saved from the user |
| id="filename" | Attribute id is the unique identifier; used for manipulating the element |
| name="firstname" | Attribute name is the value received and saved from the user |

<**label** for = "gender">Gender: </**label**>

Male <**input** type = "radio" id ="gender" name = "gender" value = "male">

Female <**input** type = "radio" id = "gender" name = "gender" value = "female">

Label tag allows user clicks on the text within the <**label>** element, it toggles the control.

For attribute should be equal to the id attribute of the related element to bind together

<**textarea** name = "description"> Max word 500 words </**textarea**><**br**/>

Textarea tag allows users to type multi-lines, description type situations

<**input** type = "submit" value = "Submit" name = "submit"/>

Type attribute allows user to make a submit button

Value attribute allows user to change the text in the submit button

<**br**/>

It allows the display go to next line. Single line break.

### Confirmation(Report.php)

<?php

**if**(!empty($\_POST["submit"])):

$firstname = $\_POST["firstname"];

$lastname = $\_POST["lastname"];

$email = $\_POST["email"];

$gender = $\_POST["gender"];

$description = $\_POST["description"];

**echo** "First Name: ".$firstname."<br/>";

**echo** "Last Name: ".$lastname."<br/>";

**echo** "Email: ".$email."<br/>";

**echo** "Gender: ".$gender."<br/>";

**echo** "Description: ".$description."<br/>";

**endif**;

?>

Code 2 (Report.php): Receiving the user input information from form.html

### Breakdown(Report.php)

|  |  |
| --- | --- |
| **Labels/Tags** | **Description** |
| <?php ?> | It indicates that the PHP code is written in this block |
| **If**(condition): | If the condition is true, do the following statement |
| $\_POST["submit"] | $\_POST receives the value from the name attribute |
| $firstname | $ creates the variable |
| **echo** | Outputting information beyond the confines of the <?php ?> |

Web browser know nothing about PHP and, therefore, have no ability to run PHP scripts.

Web servers with PHP support are equipped to run PHP scripts and turn them into HTML web pages that browsers can understand.

Every PHP must end with a semicolon “;”.

Name the PHP file with “.php”

### PHP Information(info.php)

<?php

phpinfo();

?>

Code 3 (Info.php): Detailed information about the PHP installed in the PC

### Variable

* PHP variable names must begin with a dollar sign, and cannot contain spaces
* The first character after the dollar be a letter or an underscore “\_”
* Characters after the first character after that can be a **letter, an underscore, or a number**
* Spaces and special characters are not allowed in any part of a variable name
* Use all lowercase for variable name
* Separate words in a multi-word variable name with underscores

### Assigning value to the variable

$description = “I am the man.”;

Pieces of text(strings) must be enclosed by quotes, either single quotes or double quotes.

### Special Variable

$\_POST is a special variable that is known as a superglobal. It is a collection of storage locations used to hold data from a web form. It is also an **array**. This array is filled the values the user entered into the form.

### Concatenation

**echo** "Description: ".$description."<br/>";

The period allows us to stick multiple strings of text together as one. This process is known as concatenation.

<?php

$name = "Jack Stubbort";

$age = "17";

$place = "Netherland";

$msg = $name."is ".

$age."-years-old. "

."He works in ".$place.".";

**echo** $msg;

?>

Code 4 (Concatenation.php): Long line of PHP codes spanned across multiple lines

### Escape Characters

* Escape characters in PHP starts with a backlash “\”
* Escape characters can be escaped in double-quoted only
* Single-quoted string only allow ‘\’ but not ‘\\’

<?php

$name = "Allen Smith";

$age = "21";

$occupation = "Engineer";

**echo** '$name is $age-years-old. \r\n He is an \"$occupation.\"';

**echo** "<br/><br/>";

**echo** nl2br("**$name** is **$age**-years-old. **\n** He is an **\"$occupation**.**\"\n\n**");

**echo** "**\\** **\"**";

?>

Code 5 (DoubleSingle.php): Escape Characters.

For newline break, we need to use nl2br() function to go new line.

### Mailing

#### Initial Setup

We can send mail from localhost with sendmail package , sendmail package is inbuild in XAMPP. So, if we are using XAMPP then you can easily send mail from localhost.

For example: We can configure C:\xampp\php\php.ini and c:\xampp\sendmail\sendmail.ini for gmail to send mail.

In C:\xampp\php\php.ini find extension=php\_openssl.dll and remove the semicolon from the beginning of that line to make SSL working for gmail for localhost.

In php.ini file find [mail function] and change

SMTP=smtp.gmail.com

smtp\_port=587

sendmail\_from = my-gmail-id@gmail.com

sendmail\_path = "\"C:\xampp\sendmail\sendmail.exe\" -t"

Now Open C:\xampp\sendmail\sendmail.ini. Replace all the existing code in sendmail.ini with following code

[sendmail]

smtp\_server=smtp.gmail.com

smtp\_port=587

error\_logfile=error.log

debug\_logfile=debug.log

auth\_username=my-gmail-id@gmail.com

auth\_password=my-gmail-password

force\_sender=my-gmail-id@gmail.com

Now you have done!! create php file with mail function and send mail from localhost.

PS: Don't forgot to replace my-gmail-id and my-gmail-password in above code. Also, don't forget to remove duplicate keys if you copied settings from above. For example comment following line if there is another sendmail\_path : sendmail\_path="C:\xampp\mailtodisk\mailtodisk.exe" in the php.ini file

Also remember to restart the server using the XAMMP control panel so the changes take effect.

For gmail please check https://support.google.com/accounts/answer/6010255 to allow access from less secure apps.

**Source: https://stackoverflow.com/questions/15965376/how-to-configure-xampp-to-send-mail-from-localhost**

### Mailing(Mail.php)

<?php

**echo** '<!doctype html>';

**echo** ' <html>';

**echo** ' <head>';

**echo** ' <meta charset = "UTF-8"/>';

**echo** ' <title> Simple Form </title>';

**echo** ' </head>';

**echo** ' <body>';

**echo** ' <img src = "images/database.png" width = "100" height = "100"';

**echo** ' alt = "database picture"/><br/>';

**echo** ' <p> Please fill out the form </p>';

**echo** ' <form method = "post" action = "';

**echo** $\_SERVER['PHP\_SELF']."**\"**>";

**echo** ' <label for = "firstname"> First Name: </label>';

**echo** ' <input type = "text" id = "filename" name = "firstname"/><br/>';

**echo** ' <label for = "lastname"> Last Name: </label>';

**echo** ' <input type = "text" id = "lastname" name = "lastname"/><br/>';

**echo** ' <label for = "email"> Email: </label>';

**echo** ' <input type = "email" id = "email" name = "email"/><br/>';

**echo** ' <label for = "gender">Gender: </label>';

**echo** ' Male <input type = "radio" id ="gender" name = "gender" value = "male">';

**echo** ' Female <input type = "radio" id = "gender" name = "gender" value = "female"><br/>';

**echo** ' <label for = "description"> Description: </label>';

**echo** ' <textarea name = "description"> Max word 500 words </textarea><br/>';

**echo** ' <input type = "submit" value = "Submit" name = "submit"/>';

**echo** ' </form>';

**echo** ' </body>';

**echo** ' </html>';

?>

Code 6 (Mail.php): First Section.

Form.html file has been converted to PHP file by adding echo.

$\_SERVER['PHP\_SELF']

The form sends the data to the same page and not directing to another PHP file.

<?php

**if**(!**empty**($\_POST["submit"])):

$firstname = $\_POST["firstname"];

$lastname = $\_POST["lastname"];

$email = $\_POST["email"];

$gender = $\_POST["gender"];

$description = $\_POST["description"];

**echo** "First Name: ".$firstname."<br/>";

**echo** "Last Name: ".$lastname."<br/>";

**echo** "Email: ".$email."<br/>";

**echo** "Gender: ".$gender."<br/>";

**echo** "Description: ".$description."<br/>";

$to = "khan.photon@gmail.com";

$subject = "Extra Description";

$message = $description;

mail($to, $subject, $message, 'From:'.$email);

*//Allow less secure apps on*

**endif**;

?>

Code 6 (Mail.php): Second Section.

It is exactly like the Report.php file only the mail($to, $subject, $message, 'From:'.$email) function has been added.

|  |  |
| --- | --- |
| **Variable** | **Description** |
| $to | Whom you are sending email to |
| $subject | Subject part of the email |
| $message | The actual message of the email |
| 'From:'.$email | Email Address of the sender |
| "From:".$email."\r\nCc:".$cc | Separate From and CC using (\r\n), Use double quotation |