

## PHP

PHP is a server side scripting language that is commonly used in web development. It is a popular choice for building dynamic and interactive websites as it allows the developer to create server side scripts that can generate HTML, process form data & interact with the databases.

```
<!DOCTYPE html>
<html>
<head>
<title> PHP Example </title>
</head>
<body>
<h1> Welcome </h1>
<p> Current date and time :<br/>
<?php echo date ('Y-m-d H:i:s'); ?> </p>
</body>
</html>
```

### PHP Variable

variable is store and manipulate data  
PHP variable are dynamically typed i.e you are not require to declare the type of the variable.

eg \$name = "John";  
\$age = 25;  
\$greeting = "Hello, \$name!";  
echo \$greeting;

```
$message = "My name is ". $name . " and I am "
           . $age . " years old ";
echo $message;
```

```
$name = "John";
$age = 25;
$isStudent = true;
$grades = array (85, 90, 75)
```

### PHP Operators

Operators are used to perform various operations on variables and values.

#### 1. Arithmetic Operators

+ , - , / , \* , += , -= , \*=

#### 2. Assignment Operators

+= , \*= , -= , /= , \*= , /=

#### 3. Comparison Operators

== , != , <= , >= , > , <

#### 4. Logical Operators

88, 11, !

## 5. String Operators

• , • =

## 6 Ternary Operators

? :

### PHP Expressions

Expression is a combination of values, variables, operators and function calls that can be evaluated to produce single value

\$sum = 5 + 3; // arithmetic expression

\$message = "Hello" . "John"; // String concatenation Expression

\$isG = 10 > 5; // greater than comparison expression

\$result = calculateSum(5, 3); // function call Expression

### PHP Controls and Conditional Statements

Conditional statements are used to control the flow of the code based on certain conditions

if,  
if...else  
if...elseif...else  
switch

Controls are used to control the flow of execution in a program

1) Conditional statements

2) Loops

- while (condition) {

    }

- do {

    } while (condition);

- for (initialization; condition; iteration) { }

- foreach (\$array as \$value) { }

3) Jump statements

- break : terminates the execution of a loop

- continue : skip the current iteration and proceed to new iteration

- return : terminates the execution of a function

4) Control structures for Exception Handling

try {

    } catch (Exception Type \$exception) {

    } finally {

    }

- Q What is AJAX?
- Q Why to use AJAX and where it is used.
- Q XMLHttpRequest
- Q Synchronous V/s asynchronous
- Q How AJAX works.

Q What is PHP? Explain string handling in PHP. Enlist any five string handling functions of PHP using suitable example

This PHP is a server side scripting language that is commonly used for web development. It is a popular choice for building dynamic and interactive websites, as it allows developers to create server side scripts that can generate HTML, process form data & interact with the database.

Spiral

Date \_\_\_\_\_

A string is a sequence of characters like "Hello World". PHP supports a no of data types, including strings. Alphanumeric characters are allowed in string variables. When these conditions are met strings are generated.

String functions in PHP are used to modify or ~~do~~ a string or to take knowledge about a string.

The length function is the most basic example of a string function.

1) `strlen()` - returns the length of a string

```
<?php  
echo strlen("Hello World"); // outputs 12  
?>
```

2) `str_word_count()` - counts the no of words in a string

```
<?php  
echo str_word_count("Hello World"); // outputs 2  
?>
```

3) `strrev()` - reverse a string

```
<?php  
echo strrev("Hello world"); // outputs olleH dloollH  
?>
```

4) `strpos()` - search for a specific text & returns the index

```
<?php  
echo strpos("Hello World", "World");  
?>
```

5) `str_replace()` - replace some characters with some other

```
<?php  
echo str_replace("World", "Bolly", "Hello World");  
?>
```

PHP Math Functions

Q What is PHP Math ? enlist some functions

A The math function can handle values within the range of integer & float types. PHP Math functions are part of PHP core. No additional installation required.

PHP has a set of math functions that allows us to perform mathematical tasks on numbers.

1) `pi()` - returns the value of PI

```
<?php  
echo pi(); // returns 3.1415926535898  
?>
```

2) `min()` and `max()` - used to find lowest or highest values

```
<?php  
echo min(0, 150, 30, 20, -8, -200); // outputs -200  
echo max(0, 150, 30, 20, -8, -200); // outputs 150  
?>
```

3) `abs()` - returns absolute value of a no

```
<?php  
echo abs(-6.7); // outputs 6.7  
?>
```

4) `sqrt()` - returns the square root of a no

```
<?php  
echo sqrt(64); // outputs 8  
?>
```

5) `round()` - rounds a floating no to its nearest int

```
<?php  
echo round(0.60); // outputs 1  
?>
```

Can we require for security and integrity? whether a particular can be get missing.

Date \_\_\_\_\_

### Q include vs require in PHP

It is possible to ~~insert~~ the content of one PHP file into another PHP file with the include or require statement.

The include (or require) statement ~~takes~~ all the text/code/markup that exists in a specified file and copies it to the file that uses the include statement.

Including files is very useful when you want to ~~use~~ include the same PHP, HTML or text multiple pages of a website.

The include and require statements are identical, except upon failure

- require will produce a fatal error and terminates the script
- include will produce a warning and script will continue

eg <html>  
  <body>  
    <h1> Welcome </h1>  
<?php include 'nofileexist.php';  
  echo "Hello";  
>  
</body>  
</html>

Hello will get print generating a warning

<html>  
  <body>  
    <h1> Welcome </h1>  
<?php require 'nofile.php';  
  echo "Hello";  
>  
</body>  
</html>

echo  
  will not be executed because the script dies after the require statement gets executed returning a fatal error.

### Q what is a COOKIE in PHP? perform some operation

Any cookie is a small bit of textual information that the server embeds on the user's computer. Each time the user (same computer) requests a page it will send a cookie too.

Create and retrieve cookie created

```
<?php  
$cookieName = "user";  
$cookieValue = "GuestUser";  
setcookie($cookieName, $cookieValue, time() + 86400 * 30, "/");  
<?>  
<html>  
<body>  
<?php  
if (!isset($_COOKIE[$cookieName])) {  
  echo "Cookie name '$cookieName' is not set";  
} else {  
  echo "Cookie '$cookieName' is set<br>";  
  echo "Value is: '$cookieValue'";  
<?>  
</body>  
</html>
```

Delete cookie set setcookie() with an expiration date in past

```
<?php  
$cookieName = "user";  
$cookieValue = "GuestUser";  
setcookie($cookieName, $cookieValue, time() - 3600);  
<?>
```

Spiral

```

<html>
<body>
<?php
echo "Cookie !". &lt;?php if ($cookie_name != "disabled") :?>
?> and
<?php
<?php>

```

If cookie enabled

```

<?php
setcookie("car", "Ferrari", time() + 3600, '/');
?>
<html>
<body>
<?php
if (!isset($_COOKIE)) {
    echo "Cookie enabled";
} else {
    echo "Cookie disabled";
}
?>
</body>
</html>

```

Q what is session ? perform some operation

Session variable hold the information of one user and is available to all pages in one application

Session is started with the session\_start() function  
Session variable are set with the \$HTTP global variable  
\$\_SESSION

Spiral

```

<?php
session_start();
?>
<html>
<body>
<?php
$SESSION["fawcolor"] = "yellow";
print_r($SESSION);
?>
// remove all session variable
session_unset();
// destroy the session
session_destroy();
?>
</body>
</html>

```

Q Explain arrays in php with its types with the help of an eg

An array stores multiple values in one single variables  
It can hold more than one value at a same time  
array() function is used to create array

Three types of arrays

- 1) Indexed arrays - arrays with a numeric index

```

$cars = array ("volvo", "BMW", "Toyota");
or
$cars[0] = "Volvo";
$cars[1] = "BMW";
$cars[2] = "Toyota";

```

Spiral

Date \_\_\_\_\_

⇒ Associative arrays - arrays with named keys

`$age = array ("Peter" => "35", "Harry" => "47");`

or

`$age ["Peter"] = "35";`

`$age ["Harry"] = "47";`

⇒ Multidimensional arrays - array containing one or more arrays

`array()`

`$cars = array ("Volvo", 22, 18);`

`array ("BMW", 15, 13),`

`array ("Toyota", 5, 2));`

`sort()` - sort arrays in ascending order

`asort()` - " " " descending "

`arsort()` - Sort associative arrays in ascending order, acc. to value

`krsort()` - " " " descending " " " key

`arsort()` - " " " descending " " " value

`krsort()` - " " " " " key

In addition to this answer you can show the  
eg to display the arrays.

Q What is form validation? Create a user login  
form with fields name and password. The name  
field can't be empty and password field can't be  
less than 8 characters.

The PHP superglobals `$_GET` and `$_POST` are used to  
collect form data

Date \_\_\_\_\_

Form validation is the necessary process to be done  
before the data is submitted to the database. This  
is done to avoid unnecessary errors.

In PHP the script checks the data in respective fields  
based on the rules set by the developer and returns  
error if does not meet the requirements

`<?php`

`$name = $password = " ";`

`if ($SERVER["REQUEST_METHOD"] == "POST")`

`{`

`$name = $POST["name"];`

`$password = $POST["password"];`

`}`

`</html>`

`</head> </body>`

`<h2> Login Form </h2>`

`<form method="post">`

`Name: <input type="text" name="name" required>`

`Password: <input type="password" name="password" pattern="^(?=.{8,})[a-zA-Z0-9]*$>`

`pattern = "(?=.{8,})[a-zA-Z0-9]*$"`

`E8, 3'9"`

`</form>`

`</body>`

`</html>`

Spiral

Q Write a PHP program to develop student registration form and display all the submitted data on the page.

```
<?php  
$name = $age = $gender = $email = $password = " ";  
if ($SERVER["REQUEST_METHOD"] == "POST")  
{  
    $name = $POST["name"];  
    $age = $POST["age"];  
    $email = $POST["email"];  
    $gender = $POST["gender"];  
    $password = $POST["password"];  
}  
?  
<html>  
  <body>  
    <head> </head>  
    <body>  
      <h2> Student Registration Form </h2>  
      <form method="post">  
        Name: <input type="text" name="name" required>  
        age : <input type="number" name="age" >  
        gender: <input type="text" name="gender" >  
        email: <input type="email" name="email" >  
        password:<input type="password" name="password" >  
      </form>  
      <br>  
      <h2> details </h2>  
      </body>  
      <?php  
        echo "$name<br>";  
        echo "$age<br>";  
        echo "$email<br>";  
        echo "$gender<br>";  
      ?> </body> </html>
```

Q Explain the steps to successfully connect to MySQL database.

MySQLi extension ( i stands for improved) connection to MySQL using MySQLi

```
<?php  
$server  
$username = "localhost";  
$username = "root";  
$password = " ";  
$conn = new mysqli($server, $username, $password);  
if ($conn->connect_error)  
{  
    die("connection failed ". $conn->connect_error);  
}  
else  
    echo "connected successfully";  
?>
```

MySQL is an open-source relational database management system. It is the most popular database system with PHP.

Q PHP is considered as loosely typed.

PHP does not require the data type of a variable to be explicitly defined.

Date \_\_\_\_\_

Q Create a database in MySQL GEHU containing table course having three fields course name, courseid, and course duration write a PHP code to insert the values in the table course.

```
<?php  
$servername = "localhost";  
$username = "root";  
$password = "";  
  
$conn = new mysqli($servername, $username, $password);  
  
if ($conn->connect_error){  
    die("Connection failed: " . $conn->connect_error);  
}  
  
$sql = "CREATE DATABASE GEHU";  
if ($conn->query($sql) == TRUE) {  
  
    $sql1 = "CREATE TABLE Course ( cname VARCHAR(10)  
NOT NULL, cid INT(6) PRIMARY KEY, cduration  
VARCHAR(10) NOT NULL);";  
  
if ($conn->query($sql1) == TRUE) {  
    $sql2 = "INSERT INTO Course (cname, cid, cduration)  
VALUES ('BBA', '1', '7 years');";  
    $sql2 = "Insert into Course (cname, cid, cduration)  
VALUES ('BCOM', '2', '7 years');";  
  
if ($conn->multiquery($sql2) == TRUE) {  
    echo "Record entered successfully";  
} else {  
    echo "failed";  
}  
} else {  
    echo "Error";  
}  
}
```

Date \_\_\_\_\_

Q Use PHP and perform a Create, Read, Update, Delete operation

CRUD stands for Create, Read, Delete, Update  
Four basic operations we use to manage any database.  
To perform CRUD operations in PHP we'll use MySQLi extension

```
<?php
```

```
$servername = "localhost";  
$username = "root";  
$password = "";  
$dbname = "course";
```

```
$conn = new mysqli($servername, $username, $password,  
$dbname);
```

```
if ($conn->connect_error){  
    die("Connection failed: " . $conn->connect_error);  
}
```

— nextpage.

include 'create.php';  
\$sql = "Insert into course (cid, cname, cduration).  
VALUES ('BBA', '3', '4 years')";  
if (\$conn->query(\$sql) == TRUE) {  
 echo "success";  
} else {  
 echo "error";  
}

Spiral

// create

Date \_\_\_\_\_

\$res = \$conn->query ("Insert into course values ('DBBA', '3',  
'7 years')");

// read

\$rows = \$conn->query ("select \* from course");  
while (\$row = \$rows->fetchAssoc ()) {  
echo \$row['name'] . " " . \$row['cid'] . "  
" . \$row['cduration'];

// update

\$conn->execute ("update course set cname='BFD'  
where id=3");

// delete

\$conn->execute ("delete from course where  
cid=3");

}