**Names: NIYOGISUBIZO Pacifique**

**RegnNo: 21RP02142**

**Class: Level 6 Y2 IT-B**

**Module Name:**

**Module Code:**

**Assignment of PHP**

1. **Explain php programing beyond definition?**

PHP is an open-source server-side scripting language that many devs use for web development. It is also a general-purpose language that you can use to make lots of projects, including Graphical User Interfaces (GUIs).

PHP is a script language and interpreter that is freely available and used primarily on Linux Web servers. PHP, originally derived from Personal Home Page Tools, now stands for PHP: Hypertext Preprocessor, which the PHP FAQ describes as a "recursive acronym."

PHP executes on the server, while a comparable alternative, JavaScript, executes on the client. PHP is an alternative to Microsoft's Active Server Page (ASP) technology. As with ASP, the PHP script is embedded within a Web page along with its HTML. Before the page is sent to a user that has requested it, the Web server calls PHP to interpret and perform the operations called for in the PHP script

What Can PHP Do?

* PHP can generate dynamic page content
* PHP can create, open, read, write, delete, and close files on the server
* PHP can collect form data
* PHP can send and receive cookies
* PHP can add, delete, modify data in your database
* PHP can be used to control user-access
* PHP can encrypt data

1. **Why do we need to use php programming?**

Why PHP?

* PHP allows web developers to create dynamic content and interact with databases.
* The use of PHP makes website pages load faster as compared to many other web development technologies.
* PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
* PHP is compatible with almost all servers used today (Apache, IIS, etc.)
* PHP supports a wide range of databases
* PHP is free. Download it from the official PHP resource: www.php.net
* PHP is easy to learn and runs efficiently on the server side

**3. What is the latest php version we have today and list the updated features for the latest 3release?**

today we are using

**1.PHP Version 8.2.0**

Released: 08 Dec 2022

**2. PHP Version 8.1.13**

Released: 24 Nov 2022

**3.PHP Version 8.0.26**

Released: 26 Nov 2022

**Updated features in PHP V8.2.0**

* read-only classes
* null
* false and true as stand-alone types
* deprecated dynamic properties
* performance improvements and more

4.A new release is the distribution of the final version or the newest version of a software application.

A stable release is a version of a software package that has been tested and verified. It is the latest (and sometimes final version) of a program that is considered safe for public use. This kind of release is also called a “Stable” release

**5. What are the main features of php programming?**

**The main features of php is**;

* **open source scripting language:** so you can free download this and use.
* **PHP is a server site scripting language**. It is open source scripting language. It is widely used all over the world. It is faster than other scripting language. Some important features of php are given below
  1. **Simple**

It is very simple and easy to use, compare to other scripting language it is very simple and easy, this is widely used all over the world.

* 1. **Interpreted**

It is an interpreted language, i.e. there is no need for compilation.

* 1. **Faster**

It is faster than other scripting language e.g. asp and jsp.

* 1. **Open Source**

Open source means you no need to pay for use php, you can free download and use.

* 1. **Platform Independent**

PHP code will be run on every platform, Linux, Unix, Mac OS X, Windows.

* 1. **Case Sensitive**

PHP is case sensitive scripting language at time of variable declaration. In PHP, all keywords (e.g. if, else, while, echo, etc.), classes, functions, and user-defined functions are NOT case-sensitive.

* 1. **Error Reporting**

PHP have some predefined error reporting constants to generate a warning or error notice.

* 1. **Real-Time Access Monitoring**

PHP provides access logging by creating the summary of recent accesses for the user.

* 1. **Loosely Typed Language**

PHP supports variable usage without declaring its data type. It will be taken at the time of the execution based on the type of data it has on its value.

**6. With a help of examples explain why php is case sensitive?**

**PHP classes** are a mix between variables and functions, so they are partially case-sensitive. As you can see in the example above, the variables $num and $NUM can have different values. But when you declare two functions with the same name, PHP produces a fatal error.

**7. What and why do we use comments while writing php codes, With a help of example explain different types of php comments?**

**PHP comments** are usually meant to help programmers understand and interpret the PHP code. A PHP comment can explain the purpose of a particular section of code to other programmers. This way, when a developer is viewing a PHP file for the first time, they can more easily understand the code they're looking at.

Here we two types of comment

1. One-line comment

**The following example uses the // for a one-line comment:**

1. <?php
2. $rate = 100;
3. $hours = 173;

$payout = $hours \* $rate; // payout calculation

Code language: HTML, XML (xml)

**And the following example uses the # for a one-line comment:**

1. <?php

$title = 'PHP comment'; # set default title

b) Multi-line comment

A Multi-line comment start with /\* and end with \*/. For example:

<?php

/\*

This is an example of a multi-line comment,

which can span multiple lines.

\*/

**8. Differentiate with real example the following php output functions:**

* echo and print are more or less the same. They are both used to output data to the screen. The differences are small: echo has no return value while print has a return value of 1 so it can be used in expressions

**Example for echo**

<?php

$myTxt = "Hello World!";

$myNum = 123456789;

$myColors = array("Red", "Green", "Blue");

// Displaying variables

echo $myTxt;

echo "<br>";

echo $myNum;

echo "<br>";

echo $myColors[0];

echo "<br>";

example of print

<?php

print "Apple";

// (or)

print("Apple");

?>

and

<?php

printf('We are expected to score above %d%% for distinction', 85);

// Output: We are expected to score above 85%

// for distinction

?>

* he two PRINT procedures perform formatted output. PRINT performs output to the standard output stream (IDL file unit -1), while PRINTF requires a file unit to be explicitly specified

example for print

<?php  
print "<h2>PHP is Fun!</h2>";  
print "Hello world!<br>";  
print "I'm about to learn PHP!";  
?>

Example of printf

<?php  
$number = 9;  
$str = "Beijing";  
printf("There are %u million bicycles in %s.",$number,$str);  
?>

* The printf( ) function builds a formatted string by inserting values into a template. The print\_r( ) function is useful for debugging—it prints the contents of arrays, objects, and other things, in a more-or-less human-readable form

**Printf() example**

<?php  
$number = 123;  
printf("%f",$number);  
?>

**print\_r( ) examples**

<?php

$a = array("red", "green", "blue");

print\_r($a);

echo "<br>";

$b = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

print\_r($b);

?>

**d)** var\_dump() displays values along with data types as output. print\_r() displays only value as output.

**Example of var\_dump**

<?php

$a = 32;

echo var\_dump($a) . "<br>";

$b = "Hello world!";

echo var\_dump($b) . "<br>";

$c = 32.5;

echo var\_dump($c) . "<br>";

$d = array("red", "green", "blue");

echo var\_dump($d) . "<br>";

$e = array(32, "Hello world!", 32.5, array("red", "green", "blue"));

echo var\_dump($e) . "<br>";

Example: Say we have got the following array and we want to display its contents.

$arr = array ('xyz', false, true, 99, array('50'));

// Dump two variables

echo var\_dump($a, $b) . "<br>";

?>

var\_dump() function - Displays values and types

array(5) {

[0]=>

string(3) "xyz"

[1]=>

bool(false)

[2]=>

bool(true)

[3]=>

int(100)

[4]=>

array(1) {

[0]=>

string(2) "50"

}

}

print\_r() function - Displays human-readable output

Array

(

[0] => xyz

[1] =>

[2] => 1

[3] => 99

[4] => Array

(

[0] => 50

)

)

PHP print\_r() Function

**Print\_r() example**

**9. List and Describe different datatype we have in php by categorizing them in scalar, compound and special datatypes.**

PHP Data Types:

**Scalar Types**

It holds only single value. There are 4 scalar data types in PHP.

* **Boolean**: Booleans are the simplest data type works like switch. It holds only two values: TRUE (1) or FALSE (0).
* **Integer:** Integer means numeric data with a negative or positive sign. It holds only whole numbers, i.e., numbers without fractional part or decimal points
* **Float:** A floating-point number is a number with a decimal point. Unlike integer, it can hold numbers with a fractional or decimal point, including a negative or positive sign
* **String:** string is a non-numeric data type. It holds letters or any alphabets, numbers, and even special characters.

**Compound Types**

It can hold multiple values. There are 2 compound data types in PHP.

* **Array:** An array is a compound data type. It can store multiple values of same data type in a single variable.
* **Object:** Objects are the instances of user-defined classes that can store both values and functions. They must be explicitly declared

**Special Types**

* **Resource:** Resources are not the exact data type in PHP. Basically, these are used to store some function calls or references to external PHP resources. For example - a database call. It is an external resource.
* **NULL:** Null is a special data type that has only one value: NULL. There is a convention of writing it in capital letters as it is case sensitive**.**

**10. What is php variable, list the variable naming rules you have to obey while defining a variable in php?**

PHP variables are characters that stores value or information such as text or integers in your code.

**Rules for PHP variables:**

* 1. A variable starts with the $ sign, followed by the name of the variable
  2. A variable name must start with a letter or the underscore character
  3. A variable name cannot start with a number
  4. A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_ )
  5. Variable names are case-sensitive ($age and $AGE are two different variables)

11. List and explain at least 10 super global variables?

* **$GLOBALS** is a PHP super global variable which is used to access global variables from anywhere in the PHP script (also from within functions or methods).
* **$\_SERVER** is a PHP super global variable which holds information about headers, paths, and script locations.
* **PHP $\_REQUEST** is a PHP super global variable which is used to collect data after submitting an HTML form.
* **PHP $\_POST** is a PHP super global variable which is used to collect form data after submitting an HTML form with method="post". $\_POST is also widely used to pass variables.
* **PHP $\_GET** is a PHP super global variable which is used to collect form data after submitting an HTML form with method="get".
* **＄\_FILES** is a global constant or predefined variable in PHP that can be used to associate array items that are uploaded through the HTTP POST method.
* **The PHP $\_COOKIE** super global variable is used to retrieve a cookie value.
* **$\_SESSION[].** Session variables are stored in associative array called $\_SESSION[]. These variables can be accessed during lifetime of a session.
* **$\_ENV** is another superglobal associative array in PHP. It stores environment variables available to current script.

# Reference

# <https://www.freecodecamp.org/news/what-is-php-the-php-programming-language-meaning-explained/>

# <https://www.techtarget.com/whatis/definition/PHP-Hypertext-Preprocessor#:~:text=PHP%2C%20originally%20derived%20from%20Personal,JavaScript%2C%20executes%20on%20the%20client>.

# <https://www.w3schools.com/php/php_intro.asp>

# <https://www.google.com/search?q=Why+do+we+need+to+use+php+programming%3F&oq=Why+do+we+need+to+use+php+programming%3F&aqs=chrome..69i57j33i160l4.848j0j9&sourceid=chrome&ie=UTF-8>

# <https://www.php.net/releases/index.php>

# <https://www.google.com/search?q=which+php+version+8.2.0+updated+features&oq=which+php+version+8.2.0+updated+fe&aqs=chrome.1.69i57j33i160l3.20642j0j4&sourceid=chrome&ie=UTF-8>

# A stable release is a version of a software package that has been tested and verified. It is the latest (and sometimes final version) of a program that is considered safe for public use. This kind of release is also called a “Stable” releas

# <https://www.sitesbay.com/php/php-features-of-php>

# <https://www.phptutorial.net/php-tutorial/php-comments/>

# <https://www.google.com/search?q=Echo()+vs+print()&oq=Echo()+vs+print()&aqs=chrome..69i57.1238j0j4&sourceid=chrome&ie=UTF-8>

# <https://phppot.com/php/php-print-statements/#:~:text=print()%20%E2%80%93%20The%20print(),for%20creating%20a%20print%20statement>.

# <https://www.google.com/search?q=Print()+vs+printf()&oq=Print()+vs+printf()&aqs=chrome..69i57j0i22i30l9.930j0j4&sourceid=chrome&ie=UTF-8>

# <https://www.w3schools.com/php/php_echo_print.asp>

# <https://www.google.com/search?q=Print_r+vs+var_dump()&oq=Print_r+vs+var_dump()&aqs=chrome..69i57.1496j0j9&sourceid=chrome&ie=UTF-8>

# <https://www.javatpoint.com/php-data-types#:~:text=There%20are%204%20scalar%20data,string>

# <https://www.google.com/search?q=What+is+php+variable&sxsrf=AJOqlzU3Spx6_WE56pRDrXBqMOfW-hWMNQ%3A1674750864184&ei=kKvSY6fmCpz_7_UPraCKgAw&ved=0ahUKEwin7uKS1eX8AhWc_7sIHS2QAsAQ4dUDCA8&uact=5&oq=What+is+php+variable&gs_lcp=Cgxnd3Mtd2l6LXNlcnAQAzIECCMQJzIFCAAQgAQyBQgAEIAEMgUIABCABDIGCAAQFhAeMggIABAWEB4QDzIGCAAQFhAeMgYIABAWEB4yBggAEBYQHjIICAAQFhAeEA9KBAhBGABKBAhGGABQAFgAYNsYaABwAHgAgAHoAYgB6AGSAQMyLTGYAQCgAQHAAQE&sclient=gws-wiz-serp>

# <https://www.w3schools.com/php/php_variables.asp>

# <https://www.w3schools.com/php/php_superglobals_request.asp>