RWANDA POLYTECHNIC

IPRC TUMBA

Integrated Polytechnic Regional College

Regno:21RP 03811

Level6year2

Class :B

Learning unit 2

Assignment of PHP

1.Explain php programing beyond definition?

PHP programing is an open-source server-side scripting language that many devices 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 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.

PHP (Hypertext Preprocessor) a general-purpose scripting language that can be used to develop dynamic and interactive websites

2.Why do we need to use php programming?

Why to use PHP? PHP can actually do anything related to server-side scripting or more popularly known as the backend of a website. For example, PHP can receive data from forms, generate dynamic page content, can work with databases, create sessions, send and receive cookies, send emails etc.

Advantages:

- ✓ Platform Independent
- ✓ Open source and dynamic Library support
- ✓ Organized
- ✓ Database Connectivity

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

- the latest php version used to day is **PHP 8.2**
- **list the updated features for the latest 3 release?**

Version	PHP (*)	Release
7.2 - 8.0	March 3rd, 2020	
7.3 - 8.1	September 8th, 2020	
8.0 - 8.2	February 8th, 2022	
8.1 - 8.2	February 7th, 2023	

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. Different between new release vs stable release of a software product:

new release of software:

the distribution of the final version or the newest version of a software application. A software release may be public or private and generally signifies the unveiling of a new or upgraded version of the 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..
- ✓ **stable release**: is a version that has been tested as thoroughly as possible and is as reliable as we can make it. It does not have all the new features of a beta release and it does not have the latest fixes for problems.

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

a. **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.

b. **Interpreted**

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

c. **Faster**

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

d. **Open Source**

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

e. **Platform Independent**

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

f. **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.

g. **Error Reporting**

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

h. **Real-Time Access Monitoring**

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

i. **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?

- **EX 1:** // you can create two variables like this:

```
$num = 99;
$NUM = 20;

echo $num; // 99
echo "\n".$NUM; // 20

//  but you can't have two functions like this:
function greetings (){
   echo "Hello World!";
}

//  Fatal error: Cannot redeclare GREETINGS ()
function GREETINGS (){
   echo "Hello World!";
}
```

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: cannot redeclare the function.

But class properties are case sensitive, so you can create two different properties with the same name, but different cases:

EX 2:

```
   public $name = "Nathan";
   public $NAME = "Jack";
}

$h = new HUMAN();
echo $h->name;      // Nathan
echo "\n".$h->NAME; // Jackth the same name, but different
```

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?

Why do we need comments in php codes

- ✓ Comments Are a Lightweight Way to Let You Experiment
- ✓ Comments Let You Exhibit Without Getting in Your Way
- ✓ You Can Generate Code Documentation with Comments
- ✓ Comments Explain Why You Wrote Something

With examples types of comments in php code

1.Single -line comments: are useful for short notes before a code block or for explaining a single line of code slashes (**//)**

```
<!DOCTYPE html>
<html>
<body>

<?php
// This is a single-line comment

# This is also a single-line comment
?>

</body>
</html>
```

2.multiple-line comments: allows for comments that span multiple lines, in case you want to comment out a larger section of code or leave a more descriptive comment.

```
<!DOCTYPE html>
<html>
<body>

<?php
/*
This is a multiple-lines comment block
that spans over multiple
```

```
lines
*/
?>


</body>
</html>
```

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

?>
```

- 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

Print_r vs var_dump() The var_dump() function displays structured information about variables/expressions including its type and value. Whereas The print_r() displays information about a variable in a way that's readable by humans.

9.Different datatype we have in php by categorizing them in scalar, compound, and special datatype

Scalar Types

- **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.
- **Boolean**: Booleans are the simplest data type works like switch. It holds only two values: TRUE (1) or FALSE (0).

- ✓ **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.

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:

- ✓ A variable starts with the $ sign, followed by the name of the variable

- ✓ A variable name must start with a letter or the underscore character
- ✓ A variable name cannot start with a number
- ✓ A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and)
- ✓ 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 an associative array containing items uploaded via HTTP POST method.
- ✓ **$_ENV** is another super global associative array in PHP. It stores environment variables available to current script.
- ✓ **$A cookie** is a variable which is stored in a user's web browser.
- ✓ **$Session** variables are stored in associative array called $_SESSION.

Bibliography

https://www.geeksforgeeks.org/how-to-declare-a-global-variable-in-php/

https://www.google.com/search?q=why++We+need+to+use+php+programing&sxsrf=AJOqlzW5icCQUO8xZEEkJkFvKRMSF7ultA%3A1674741385284&source=hp&ei=iYbSY9DnDrju1sQPy8qsoAU&iflsig=AK50M

_UAAAAAY9KUmfJLH5auiEhfXTOwO3SFOr5qR4bf&ved=0ahUKEwiQ0O7qseX8AhU4t5UCHUslC1QQ4dUDCAg&uact=5&oq=why++We+need+to+use+php+programing&gs_lcp=Cgdnd3Mtd2l6EAMyBQghEKABOgQIIxAnOgUIABCRAjoECAAQQzoKCAAQsQMQgwEQQzoLCAAQgAQQsQMQgwE6BQgAEIAEOgYIABAKEEM6CAguEIAEENQCOgUILhCABDoHCCEQoAEQCjoECCEQClAAWI-vAWDUxwFoAHAAeACAAdICiAHYMJIBBjItMjIuMZgBAKABAaABAg&sclient=gws-wiz

https://www.sitesbay.com/php/php-features-of-php

https://www.quora.com/How-many-data-types-are-there-in-PHP

https://www.php.net/releases/index.php

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