Name: Nzwinimana Ferdinand

Reg number: 21RP08847

Class: L6Y2IT A

Assignment 2

1. Explain php programing beyond definition?
   PHP (Hypertext Preprocessor) is known as a general-purpose scripting language that can be used to develop dynamic and interactive websites.

   It was among the first server-side languages that could be embedded into HTML,

   Making it easier to add functionality to web pages without needing to call external files for data.

2. Why do we need to use php programming?
   - PHP allows web developers to create dynamic content and interact with databases
   - It's easy to learn and use
   - It's open source (and therefore free!)
   - It enjoys strong community support it's fast and secure
   - It is well connected with databases
   - There's a lot of legacy code

3. What is the latest php version we have today and list the updated features for the latest 3 release?
   8.2
   Release Date2022-12-08
   PHP 8.2 is the latest PHP version which brings read only classes,
   DNF types, null, false, and true types, sensitive parameter redaction support,
   A new random extension, and several new features along with a few deprecations.
   8.1
   Release Date2021-11-25
   PHP 8.1, released in 2021, brings major new features such as Enums,
   Fibers, never return type, Intersection Types, read only properties, and more,
   While ironing out some of its undesired legacy features by deprecating them.
   8.0
   Release Date2020-11-26
   PHP 8.0, on the 25th year of PHP history, brings several important features such as Union Types, JIT, Constructor Property Promotion, Match Syntax, Named Parameters, and several more performance, syntax, and quality-of-life improvements.

4. What is different between new release vs stable release of a software product?

A 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 **while** a release is 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.

5. What are the main feature of php programing?

- 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

6. with help of examples explain why php is case sensitive?

PHP classes are a mix between variables and functions, so they are partially case-sensitive

PHP is a unique programming language in terms of case sensitivity.

In PHP, variables and constants are case sensitive, while functions are not case sensitive.

Let's see some practical example below:

```
// 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.

7. What and why do we use comments while writing php codes, with help of example explain different type of comment lines?

Reason: A comment in PHP code is a line that is not executed as a part of the program. Its only purpose is to be read by someone who is looking at the code. Comments can be used to: Let others understand your code.

One-line comments

The one-line comment is placed at the end of the line or at the current block.
A one-line comment starts with the pound (#) or double forward-slash (//). The rest of the text after the (//) is ignored by the PHP interpreter.

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

```
<? Php

$rate = 100;
$hours = 173;
$payout = $hours * $rate; // payout calculation
```

Code language: HTML, XML (xml)

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

```
<? Php
$title = 'PHP comment'; # set default title
```

Multi-line comments

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.
*/
```

In practice, you use the multi-line comment when you need to span comments multiple lines

8. Differentiate with real example the following php output functions
    a) echo() vs print()

The echo statement can be used with or without parentheses: echo or echo ().

Example

```
<?php
echo "<h2>PHP is Fun!</h2>";
echo "Hello world!<br>";
echo "I'm about to learn PHP!<br>";
echo "This ", "string ", "was ", "made ", "with multiple parameters.";
?>
```

The print statement can be used with or without parentheses: print or print()

Example

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

b) print() vs printf()

The PHP print

The print statement can be used with or without parentheses: print or print().

Display Text

The following example shows how to output text with the print command (notice that the text can contain HTML markup):

Example

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

PHP printf() Function

Example

Output a formatted string:

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

c) printf() vs print_r()

PHP printf() Function

Example

Output a formatted string:

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

PHP print_r() Function

Example

Print the information about some variables in a more human-readable way:

```
<?php
$a = array("red", "green", "blue");
print_r($a);

echo "<br>";

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

d) print_r() vs var_dump()

PHP print_r() Function

Example

Print the information about some variables in a more human-readable way:

```
<?php
$a = array("red", "green", "blue");
print_r($a);


echo "<br>";


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

PHP var_dump() Function

Example

Dump information about different variables:

```
<?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"
```

}

}

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

   And special datatypes.

A variable is said to be of the scalar type if it only accepts single values. In PHP, there are 4 scalar data types.

Boolean

Integer

Float

String

PHP Data Types: Compound Types

In contrast to Scalar data types, a variable is called compound if it holds multiples values within. There are 2 compound data types in PHP.

Array

Object

PHP Data Types: Special Types

Resource

NULL

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. It is important to know that variables in PHP are usually represented by a dollar sign ($) followed by the name of the variable

A variable can have a short name (like x and y) or a more descriptive name (age, carname, total volume).

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?

The PHP super global variables are:

- **$GLOBALS**: is a PHP super global variable which is used to access global variables from anywhere in the PHP scrip
- **$_SERVER:** is a PHP super global variable which holds information about headers, paths, and script locations.
- **$_REQUEST:** is a PHP super global variable which is used to collect data after submitting an HTML form**.**
- **$_POST**: is a PHP super global variable which is used to collect form data after submitting an HTML form with method="post". $_POST is also widely used to pass variables.
- **$_GET:** is a PHP super global variable which is used to collect form data after submitting an HTML form with method="get".
- **$_FILES:** is an associative array containing items uploaded via HTTP POST method. Uploading a file requires HTTP POST method form with enctype attribute set to multipart/form-data.
- **$_ENV:** is another super global associative array in PHP. It stores environment variables available to current script
- **$_COOKIE:** is used to retrieve a cookie value. It typically an associative array that contains a list of all the cookies values sent by the browser in the current request, keyed by cookie name.
- **$_SESSION**: is an associative array that contains all session variables. It is used to set and get session variable values**.**