ACTIVITY ANSWER SHEET

Name	Aldeza, Pedro Miles T.
Section:	BS-IT-3R1

Instructions:

1. Push your output on your GITHUB repository.
2. Use the answer sheet provided save it as PDF file then push it to your GitHub.
3. Answer the ff. problems write it on the answer sheet.
4. Late submissions will no longer be accepted.
5. Caught copying outputs of others will be given sanctions.
6. Failure to follow these instructions will be given sanctions.

Activity 1: Control Structures

1. Write down the syntax in PHP for the ff.

1. if	if (condition) {
	  code is executed if the condition is true;
	}

2. if…else	if (condition) {
	  code is executed if the condition is true;
	}
	else {
	  code is executed if the condition is false;
	}

4. switch…case	switch (v) {
	  case name1:
	  code is executed if v = name1;
	  break;
	  case name2:
	  code is executed if v = name2;
	  break;
	  .
	  .
	  .
	  default:
	  Code is executed if v is different from all names;
	}

5. for loop	for(init counter; test counter; increment counter) {
	  code is be executed for each iteration;
	}

6. do while loop	do { code is executed; } while (condition is true);

7. while loop	while (condition is true){
	  code is executed;
	}

8. foreach loop	foreach ($array as $value) {
	  code is to be executed;
	}

9. break statement	break; //or break name1;

10. continue statement	continue; //or continue name1;

11. try…catch	try {
	tryCode - block of code to try
	}
	catch (err) {
	catchCode - block of code to handle errors
	}
	finally {
	finallyCode - block of code to be executed regardless of the try/catch result
	}

2. Solve the ff. problem using PHP.

a. Write a program that checks if value is a number (integer).

Sample input: '1' Sample input: 1

Expected output: Not a number Expected output: A number

```php
<?php

$var1 = 1;



if(is_int($var1)){

    echo 'is a number';

}

else {

    echo 'not a number';

}

?>
```

```php
<?php

$var1 = '1';



if(is_int($var1)){

    echo 'is a number';

}

else {

    echo 'not a number';

}

?>
```

b. Write a program that checks if a value is positive or negative and odd or even.

Sample input: 0 Sample input: -1

Expected output: Positive & Even Expected output: Negative and Odd

```php
<?php


$var = 0;

//$var = -1;

    if($var % 2 == 0){

        echo "Positive and Even";

    }

    else{

        echo "Negative and Odd";

    }

?>
```

```php
<?php


//$var = 0;

$var = -1;

    if($var % 2 == 0){

        echo "Positive and Even";

    }

    else{

        echo "Negative and Odd";

    }

?>
```

c. Write a program that checks if a value is palindrome.

Sample input: Anna
Expected output: Palindrome

Sample input: Bogart
Expected output: Not a Palindrome

```php
<?php

$mystring = "Anna";

//$mystring = "Bogart";


if($mystring == strrev($mystring)){

    echo 'palindrome';

}

else {

    echo 'not a palindrome';

}

?>
```

d. Write a program to calculate and print the factorial of a number using a for loop.
Sample input: 4
Expected output: 24

```php
<?php

$var1 = 4;

$factorial = 1;



for ($x=$var1; $x>=1; $x--)

{

    $factorial = $factorial * $x;

}



echo "The factorial of $var1 is $factorial";

?>
```

e. Write a PHP program to generate and display the first n lines of a Floyd triangle.
Sample input: 3
Sample output:

1
2 3
4 5 6

```php
<?php

$len = 3;

$num = 1;



for($i = 1; $i <= $len; $i++){

    for($j = 1; $j <= $i; $j++){

        echo $num;

        $num++;

        if ($j == $i){

            echo "<br>";

        }

    }

}
```

Activity 2: PHP Built-in Functions

Write down the functionalities of the ff. built-in functions in PHP.

Array	The array() function is an inbuilt function in PHP which is used to create an array.
Calendar	The calendar extension contains functions that simplifies converting between different calendar formats.
Date	The date/time functions allow you to get the date and time from the server where your PHP script runs. You can then use the date/time functions to format the date and time in several ways.
Directory	The directory functions allow you to retrieve information about directories and their contents.
Error	The error functions are used to deal with error handling and logging. The error functions allow us to define own error handling rules, and modify the way the errors can be logged. The logging functions allow us to send messages directly to other machines, emails, or system logs. The error reporting functions allow us to customize what level and kind of error feedback is given.
File System	The filesystem functions allow you to access and manipulate the filesystem.
Filter	This PHP filters is used to validate and filter data coming from insecure sources, like user input.
FTP	The FTP functions give client access to file servers through the File Transfer Protocol (FTP). The FTP functions are used to open, login and close connections, as well as upload, download, rename, delete, and get information on files from file servers. Not all of the FTP functions will work with every server or return the same results. The FTP functions became available with PHP 3.
Libxml	The libxml functions and constants are used together with SimpleXML, XSLT and DOM functions.
Mail	The mail() function allows you to send emails directly from a script.
Math	The math functions can handle values within the range of integer and float types.
Misc	The misc. functions were only placed here because none of the other categories seemed to fit.
MySQLi	The MySQLi functions allows you to access MySQL database servers.

Network	The Network functions contains various network function and let you manipulate information sent to the browser by the Web server, before any other output has been sent.
SimpleXML	SimpleXML is an extension that allows us to easily manipulate and get XML data. SimpleXML provides an easy way of getting an element's name, attributes and textual content if you know the XML document's structure or layout. SimpleXML turns an XML document into a data structure you can iterate through like a collection of arrays and objects.
Stream	Streams are the way of generalizing file, network, data compression, and other operations which share a common set of functions and uses. In its simplest definition, a stream is a resource object which exhibits streamable behavior. That is, it can be read from or written to in a linear fashion, and may be able to fseek() to an arbitrary location within the stream. A wrapper is additional code which tells the stream how to handle specific protocols/encodings.
String	The PHP string functions are part of the PHP core. No installation is required to use these functions.
XML Parser	The XML functions lets you parse, but not validate, XML documents. Expat is an event-based parser, it views an XML document as a series of events. When an event occurs, it calls a specified function to handle it. Expat is a non-validating parser, and ignores any DTDs linked to a document. However, if the document is not well formed it will end with an error message. Because it is an event-based, non validating parser, Expat is fast and well suited for web applications. The XML parser functions lets you create XML parsers and define handlers for XML events.
Zip	The Zip files functions allows you to read ZIP files.
Timezones	provides a complete list of the timezones supported by PHP, which are useful with

	several PHP date functions.

Activity 3: Regular Expression

1. Define Regular Expression (RegEx) and provide example programming scenario where you can use (RegEx). Provide example syntax in PHP.

Regular Expressions, commonly known as "regex" or "RegExp", are a specially formatted text strings used to find patterns in text. Regular expressions are one of the most powerful tools available today for effective and efficient text processing and manipulations. For example, it can be used to verify whether the format of data i.e. name, email, phone number, etc. entered by the user was correct or not, find or replace matching string within text content, and so on.

2. Solve the ff. problem using Regular Expressions.

a. Write a PHP script that checks if a string contains another string

Sample String: 'The quick brown fox'
Test input: 'Fox'
Expected output: Fox is found the string

```php
<?php

$str1 = 'The quick brown fox.';

if (strpos($str1,'fox') !== false)

 {

    echo 'The specific word is present.';

 }

else

 {

    echo 'The specific word is not present.';

 }

?>
```

b. Write a PHP script that removes the last word from a string.

Sample String: 'The quick brown fox'
Expected output: 'The quick brown'

```php
<?php

$str1 = 'The quick brown fox';

echo preg_replace('/\W\w+\s*(\W*)$/', '$1', $str1)."\n";

?>
```

c. Write a PHP script to remove nonnumeric characters except comma and dot.
Sample String: '/$123,34.00A#'
Expected output: 123,34.00

```php
<?php

$str1 = "$12,334.00A";

echo preg_replace("/[^0-9,.]/", "", $str1)."\n";

?>
```

d. Write a PHP script to extract text (within parenthesis) from a string.
Sample String: 'The quick brown [fox].'
Expected output: Fox

```php
<?php

$my_text = 'The quick brown [fox].';

preg_match('#\[(.*?)\]#', $my_text, $match);

print $match[1]."\n";

?>
```

e. Write a PHP script to remove all characters from a string except a-z A-Z 0-9 or " ".
Sample String: 'abcde$ddfd @abcd)der]'
Expected output: abcdeddfd abcd der

```php
<?php

$string = 'abcde$ddfd @abcd )der]';

$newstr = preg_replace("/[^A-Za-z0-9 ]/", '', $string);

echo 'New string : '.$newstr."\n";

?>
```

Activity 4: Error Handling

1. **Parse error**: syntax error, unexpected '$sum' (T_VARIABLE), expecting ',' or ';' in **C:\xampp\htdocs\index.php** on line **6**

```php
<?php
```

```
$num1=15;

$num2=30;

$sum= $num1+$num2;

echo "Sum: " $sum;

?>
```

Solution:

***always put a comma ‘,’ next to the double qoutes (“ “) when you wanna print out a variable ($sum).**

```
<?php

$num1=15;

$num2=30;

$sum= $num1+$num2;

echo "Sum: ", $sum;

?>
```