ACTIVITY ANSWER SHEET

Name	Rotoras, JerelRoi C.
Section:	BSIT – 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 to be executed if condition is true;`
	`}`

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

3. if…else if…else	`if (condition) {`
	`    code to be executed if this condition is true;`
	`} elseif (condition) {`
	`    code to be executed if first condition is false and this condition is true;`
	`} else {`
	`    code to be executed if all conditions are false;`
	`}`

4. switch…case	`switch (n) {`
	`    case label1:`
	`        code to be executed if n=label1;`
	`        break;`
	`    case label2:`
	`        code to be executed if n=label2;`
	`        break;`
	`    case label3:`
	`        code to be executed if n=label3;`
	`        break;`
	`    ...`
	`    default:`
	`        code to be executed if n is different from all labels;`
	`}`

5. for loop	`for (init counter; test counter; increment counter)`

	{
	    code to be executed for each iteration;
	}

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

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

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

9. break statement	break;
10. continue statement	continue;

11. try…catch	try {
	   // run your code here
	}
	catch (exception $e) {
	   //code to handle the exception
	}
	finally {
	   //optional code that always runs
	}

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

    $num = "1";

    if(is_int($num)){
        echo "$num is a number";
    }
    else {
        echo "$num is not a number";
    }


?>
```

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

Sample input: 0
Expected output: Positive & Even

Sample input: -1
Expected output: Negative and Odd

```
<?php
$num = 0;
if ($num >= 0){ $numtype1 = 'Positive';
}
else { $numtype1 = 'Negative';
} if ($num % 2 == 0){ $numtype2 = 'Even';
}
else { $numtype2 = 'Odd';
} echo $numtype1 ." & ". $numtype2;

?>
```

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

    $name = strtolower("Anna");


    if($name== strrev($name)){
        echo "palindrome";
    }
    else{
        echo "not 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

    $num = 4;
    $max = 4;
    for ($i = 1; $i < $max; $i++) {
      $num = $num * $i;
     } echo $num;



?>
```

`

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

$num = 3;

$start = 1;

for($x=1; $x <= $num; $x++){

    for($y = 1; $y <= $x; $y++){

    echo $start; $start++;

    if ($x == $y) {

        echo "<br>";

    }

}

}


?>
```

Activity 2: PHP Built-in Functions

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

Array	The array_diff_ukey() function compares the keys of two (or more) arrays, and returns the differences. The array_fill() function fills an array with values. The array_chunk() function splits an array into chunks of new arrays. The array_flip() function flips/exchanges all keys with their associated values in an array.

Calendar	cal_days_in_month(*calendar,month,year);*
	cal_from_jd(*jd,calendar);*
	cal_info(*calendar);*
	cal_to_jd(*calendar,month,day,year);*

Date	The date_sunrise() function returns the sunrise time for a specified day and location. The date_time_set() function sets the time. The date_diff() function returns the difference between two DateTime objects. The date_sub() function subtracts some days, months, years, hours, minutes, and seconds from a date..
Directory	The chdir() function changes the current directory. The rewinddir() function resets the directory handle created by opendir(). The closedir() function closes a directory handle. The readdir() function returns the name of the next entry in a directory. The getcwd() function returns the current working directory.
Error	The debug_backtrace() function generates a PHP backtrace. The debug_print_backtrace() function prints a PHP backtrace. The error_get_last() function returns the last error that occurred (as an associative array). The error_log() function sends an error message to a log, to a file, or to a mail account.

File System	The basename() function returns the filename from a path. The chgrp() function changes the usergroup of the specified file. The chmod() function changes permissions of the specified file. The chown() function changes the owner of the specified file. The clearstatcache() function clears the file status cache. The copy() function copies a file.
Filter	The filter_has_var() function checks whether a variable of a specified input type exist. The filter_id() function returns filter ID of a specified filter name. The filter_input() function gets an external variable (e.g. from form input) and optionally filters it. The filter_input_array() function gets external variables (e.g. from form input) and optionally filters them.
FTP	The ftp_alloc() function allocates space for a file to be uploaded to the FTP server. The ftp_cdup() function changes to the parent directory on the FTP server. The ftp_chdir() function changes the current directory on the FTP server. The ftp_chmod() function sets permissions on the specified file via FTP. The ftp_close() function closes an FTP connection.
Libxml	The libxml_clear_errors() function clears the libxml error buffer. The libxml_disable_entity_loader() function enables the ability to load external entities. The libxml_get_errors() function gets the errors from the the libxml error buffer. The libxml_get_last_error() function gets the last error from the libxml error buffer.
Mail	The mail() function allows you to send emails directly from a script. The ezmlm_hash() function calculates the hash value needed when keeping EZMLM mailing lists in a MySQL database.

Math	The abs() function returns the absolute (positive) value of a number. The acos() function returns the arc cosine of a number. The acosh() function returns the inverse hyperbolic cosine of a number. The asin() function returns the arc sine of a number.
Misc	The connection_aborted() function checks whether the client has disconnected. The connection_timeout() function checks whether the script has timed out. The constant() function returns the value of a constant. The define() function defines a constant.
MySQLi	The affected_rows / mysqli_affected_rows() function returns the number of affected rows in the previous SELECT, INSERT, UPDATE, REPLACE, or DELETE query. The autocommit() / mysqli_autocommit() function turns on or off auto-committing database modifications.
Network	The checkdnsrr() function checks DNS records for *type* corresponding to *host*. The closelog() function closes the connection of system logger. The dns_check_record() function is an alias of the checkdnsrr() function.
SimpleXML	The __construct() function creates a new SimpleXMLElement object. The __toString() function returns the string content of an element. The addAttribute() function appends an attribute to the SimpleXML element.

Stream	stream_context_get_params()
	stream_bucket_prepend()
	stream_context_get_default()
	stream_context_set_options()

String	The addcslashes() function returns a string with backslashes in front of the specified characters. The addslashes() function returns a string with backslashes in front of predefined characters. The chop() function removes whitespaces or other predefined characters from the right end of a string.
XML Parser	The utf8_decode() function decodes a UTF-8 string to ISO-8859-1. The utf8_encode() function encodes an ISO-8859-1 string to UTF-8. The xml_error_string() function returns the XML parser error description.
Zip	The zip_close() function closes a ZIP file archive opened by the zip_open() function. The zip_entry_read() function reads from an open directory entry. The zip_entry_open() function opens a directory entry in a ZIP file for reading.
Timezones	**PHP** | date_default_timezone_set() **Function** PHP | IntlChar getBidiPairedBracket() Function

Activity 3: Regular Expression

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

```
<?php

$name = "I am John doe";

if (preg_match("/doe/", $name)){
    echo "doe is found the string";
}
else{
    echo "doe is not found the string";
}
?>
```

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

$name = "The quick brown fox";

if (preg_match("/fox/", $name)){
    echo "fox is found the string";
}
else{
    echo "fox is not found the string";
}
?>
```

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
$name = "The quick brown fox";
echo preg_replace('/\W\w+\s*(\W*)$/', '$1', $name);
?>
```

c. Write a PHP script to remove nonnumeric characters except comma and dot.

Sample String: '/$123,34.00A#'
Expected output: 123,34.00

```
<?php

$name = "/$123,34.00A#";
echo preg_replace("/[^0-9,.]/", "", $name);
?>
```

d. Write a PHP script to extract text (within parenthesis) from a string.

Sample String: ‘The quick brown [fox].’
Expected output: Fox

```
<?php
$name = "The quick brown [Fox].";
preg_match('#\[(.*?)\]#', $name, $match_name);
```

```
print $match_name[1];
?>
```

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
$characters = 'abcde$ddfd @abcd )der]';
echo preg_replace("/[^A-Za-z0-9 ]/", '', $characters);
?>
```

Activity 4: Error Handling

1. List down the different PHP errors. Provide example code on how to handle these errors.

```
<?php
$denominator = 0;
echo 2 / $denominator;
?>
```

```
Warning: Division by zero in C:\xampp\htdocs\act.php on line 3
INF
```

```
<?php
$denominator = 0;
if ($denominator != 0) {
   echo 2 / $denominator;
} else {
   echo "cannot divide by zero (0)";
}
?>
```