

ACTIVITY ANSWER SHEET

Name	Kristene Mae Aganap
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	<pre>if (condition) { code to be executed if condition is true; }</pre>
2. if...else	<pre>if (condition) { code to be executed if condition is true; } else { code to be executed if condition is false; }</pre>
3. if...else if...else	<pre>if (condition) { code to be executed if this condition is true; } elseif (condition) { code to be executed if this condition is true; } else { code to be executed if all conditions are false; }</pre>
4. switch...case	<pre>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; }</pre>
5. for loop	<pre>for (init counter; test counter; increment counter) { code to be executed; }</pre>
6. do while loop	<pre>do { code to be executed; } while (condition is true);</pre>


```

        echo "Negative and Odd";
    }
}
}
}
$number = -32;
check($number)

?>

```

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

Sample input: Anna

Sample input: Bogart

Expected output: Palindrome

Expected output: Not a Palindrome

```

<?php
function palindrome($string) {
    if(strrev ($string) == $string) {
        return 1;
    }else{
        return 0;
    }
}

$word = "anna";
if(palindrome($word)) {
    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
$n = 4;
$x = 1;
for ($i=1; $i<=$n-1;$i++)

{
    $x*=( $i+1);
}

echo "The factorial of $n = $x"."\\n";

?>

```

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
echo "<pre>";
$num = 1;
for ($k = 1; $k <= 5; $k++) {
    for ($a = 1; $a <=$k; $a ++ ) {
        echo $num . "&nbsp;";
        $num++;

        if($a == $k ){
            echo "&nbsp;";
            echo "<br/>";
        }

    }
}
echo "</pre>";
?>
```

Activity 2: PHP Built-in Functions

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

Array	<p>An array is an special variable, which can hold more than one value at a time.</p> <p>Examples:</p> <ul style="list-style-type: none">• is_array()\$arr• in_array(\$var, \$arr)• print_r(\$arr)• array_merge(\$arr1, \$arr2)• array_values(\$arr)
Calendar	<p>The calendar extension contains functions that simplifies converting between different calendar formats.</p> <p>Examples:</p> <ul style="list-style-type: none">• cal_days_in_month().• cal_from_jd().• cal_info().• cal_to_jd().• Unixtojd().
Date	<p>The date/time functions allow you to get the date and time from the server where your PHP script runs. You can use the date/time functions to format the date and time in several ways.</p> <p>Examples:</p> <ul style="list-style-type: none">• checkdate().• date_add().• date_create().• date_date_set().• date_diff().

Directory	<p>The directory functions allow you to retrieve information about directories and their contents.</p> <p>Examples:</p> <ul style="list-style-type: none">• <code>chdir()</code>.• <code>chroot()</code>.• <code>closedir()</code>.• <code>dir()</code>• <code>getcwd()</code>.
Error	<p>The error functions are used to deal with error handling and logging. It allows us to define own error handling rules, and modify the way errors can be logged.</p> <p>Examples:</p> <ul style="list-style-type: none">• <code>debug_backtrace()</code>.• <code>debug_print_backtrace()</code>.• <code>error_get_last()</code>.• <code>error_log()</code>.• <code>error_reporting()</code>.
File System	<p>The filesystem functions allows you to access and manipulate the filesystem.</p> <p>Examples:</p> <ul style="list-style-type: none">• <code>basename()</code>.• <code>chgrp()</code>.• <code>chmod()</code>.• <code>chown()</code>.• <code>copy()</code>.
Filter	<p>The PHP filters is use to validate and filter data coming from insecure sources, like user input.</p> <p>Examples:</p> <ul style="list-style-type: none">• <code>filter_id()</code>.• <code>filter_input()</code>.• <code>filter_list()</code>.• <code>filter_var()</code>.• <code>filter_var_array()</code>.
FTP	<p>The FTP functions give client access to file servers through the File Transfer Protocol. It is used to open, login and close connections, as well as upload, download, rename, delete and get information on files from file servers.</p> <p>Examples:</p> <ul style="list-style-type: none">• <code>ftp_alloc()</code>.• <code>ftp_cdup()</code>.• <code>ftp_chdir()</code>.• <code>ftp_chmod()</code>.• <code>ftp_close()</code>.
Libxml	<p>The libxml functions and constants are used together with SimpleXML, XSLT and DOM functions.</p> <p>Examples:</p> <ul style="list-style-type: none">• <code>libxml_clear_errors()</code>.• <code>libxml_get_errors()</code>.• <code>libxml_get_last_error()</code>.• <code>libxml_set_streams_context()</code>.• <code>libxml_use_internal_errors()</code>.

Mail	<p>The mail() function allows you to send emails directly from a script.</p> <p>Examples:</p> <ul style="list-style-type: none">• mail().• ezmlm_hash().
Math	<p>The math functions can handle values within the range of integer and float types.</p> <p>Examples:</p> <ul style="list-style-type: none">• abs().• acos().• acosh().• asin().• asinh().
Misc	<p>The misc functions were only placed here because none of the other categories seemed fit.</p> <p>Examples:</p> <ul style="list-style-type: none">• define().• defined().• die().• eval().• exit().
MySQLi	<p>The MySQLi functions allows you to access MySQL database servers.</p> <p>Examples:</p> <ul style="list-style-type: none">• mysqli_close().• mysqli_commit().• mysqli_connect().• mysqli_debug().• mysqli_errno().
Network	<p>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.</p> <p>Examples:</p> <ul style="list-style-type: none">• closelog().• fsockopen().• gethostbyaddr().• gethostbyname().• gethostname().
SimpleXML	<p>SimpleXML is an extension that allows us to easily manipulate and get XML data.</p> <p>Examples:</p> <ul style="list-style-type: none">• addChild().• asXML().• attributes().• children().• count().
Stream	<p>Streams are the way of generalizing file, network, data compression, and other operations which share a common set of functions and uses. It is a resource object which exhibits streamable behavior.</p> <p>Examples:</p> <ul style="list-style-type: none">• stream_get_filters().• stream_get_line().• stream_is_local().• stream_isatty().

	<ul style="list-style-type: none">• stream_select().
String	<p>The PHP string functions are part of the PHP core. No installation is required to use these functions.</p> <p>Examples:</p> <ul style="list-style-type: none">• crc32().• crypt().• echo().• explode().• fprintf().
XML Parser	<p>The XML functions let you parse, but not validate XML documents.</p> <p>Examples:</p> <ul style="list-style-type: none">• utf8_decode().• utf8_encode().• xml_parse().• xml_parser_create().• xml_parser_free().
Zip	<p>The zip files functions allow you to read ZIP files.</p> <p>Examples:</p> <ul style="list-style-type: none">• zip_entry_name().• zip_entry_open().• zip_entry_read().• zip_open().• zip_read().e
Timezones	<p>If you need the time to be correct according to a specific location, you can set a timezone to use.</p>

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 a replace matching string within the context, 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
$mystring = 'This is a sample text only.';
$word = "text";
if(strpos($mystring, $word) !== false) {
    echo "$word is found in the string.";
}else{
    echo "$word is not found in 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
$mystring = 'This is a sample text only.';
echo preg_replace('/\W\w+\s*(\W*)$/','',$mystring). "\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
$mystring = '123,123.00 This is a sample text only';
echo preg_replace("/[^0-9,.]/", "", $mystring). "\n";
?>
```


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

Sample String: 'The quick brown [fox].'

Expected output: Fox

```
<?php
$my_string = 'This is a sampe [text] only.';
preg_match ('#\[(.*?)\]#', $my_string, $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: abcdeddfdf abcd der

```
<?php
$mystring = 'he@l&%lo w%o%rl@!d';
echo "Old string : ".$mystring." </br> ";
$newstr = preg_replace("/[^A-Z a-z 0-9 ]/", '', $mystring);
echo 'New string : '.$newstr."\n";
?>
```

Activity 4: Error Handling

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

- Parse error or syntax error

Example:

```
<?php
```

```
$x = "geeks";
```

```
y = "Information Technology";
```

```
echo $x;
```

```
echo $y;
```

```
?>
```

In the above program, \$ sign is missing inline 3 so it gives an error message.

- Fatal error

Example:

```
<?php
```

```
function add ($x, $y);
{
    $sum = $x + $y;
    Echo "sum = ".$sum;
}
$x = 0;
$y = 20;
```

```
diff($x, $y);
```

```
?>
```

In the above program, the diff function is called but the definition of function is not available, so it gives error.

- Warning errors

Example:

```
<?php
```

```
$x = "GeeksforGeeks";
```

```
include ("hello.php");
```

```
echo $x . "Computer science portal"
```

```
?>
```

The above program call an undefined file hello.php which are not available, so it produces error.

- Notice error

Example:

```
<?php
```

```
echo $x;
```

```
echo $geeks;
```

?>

The above program uses undeclared variable \$geeks so it gives an error message.