Practical: 2

Aim: Perform basic programs of PHP using operator, condition, Loop, Date & Time functions.

Q.1 Study the following functions & write a description for each: echo(), print(), phpinfo(), define(), var_dump(), date(), Time().

echo():

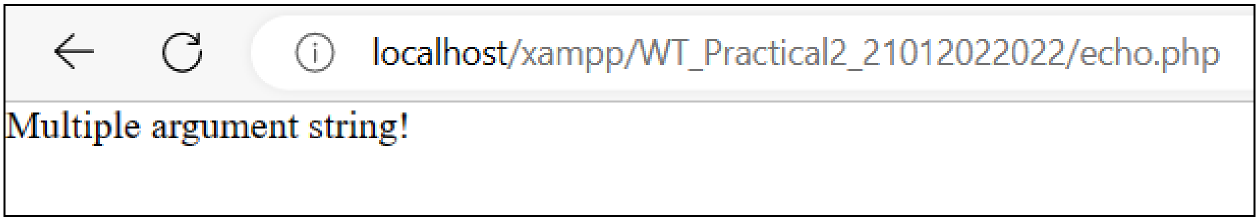
- This is a type of output string.
- Displaying Strings as multiple arguments: We can pass multiple string arguments to the echo statement instead of a single string argument, separating them by comma (‘,’) operator.
- For example, if we have two strings i.e. “Hello” and “World” then we can pass them as (“Hello”, “World”).
- It can't return any value.

Syntax
echo $arg1 , $arg2 ...;

Program:

```
<?php
   echo "Multiple ","argument ","string!";
?>
```

Output:



```
Multiple argument string!
```

print():

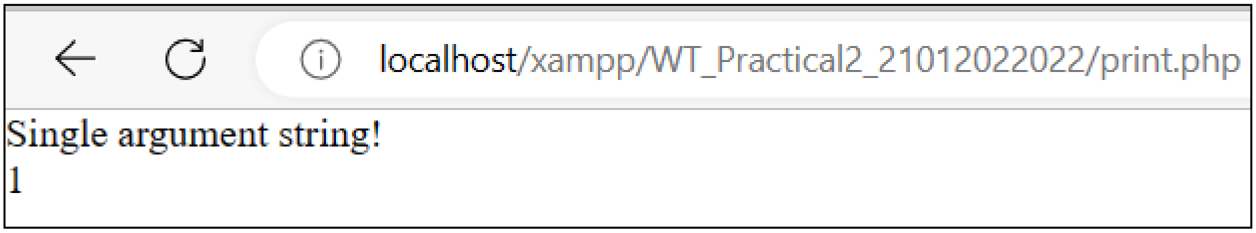
- This is also a type of output string.
- It can or cannot be written with parenthesis.
- You cannot pass multiple arguments.
- It returns 1 and 0.
  - '1' -> When line having no error.
  - '0' -> When line having error.

Syntax:
print ($arg);

Program:

```
<?php
   print print "Single argument string!<br>";
?>
```

Output:



```
Single argument string!
1
```

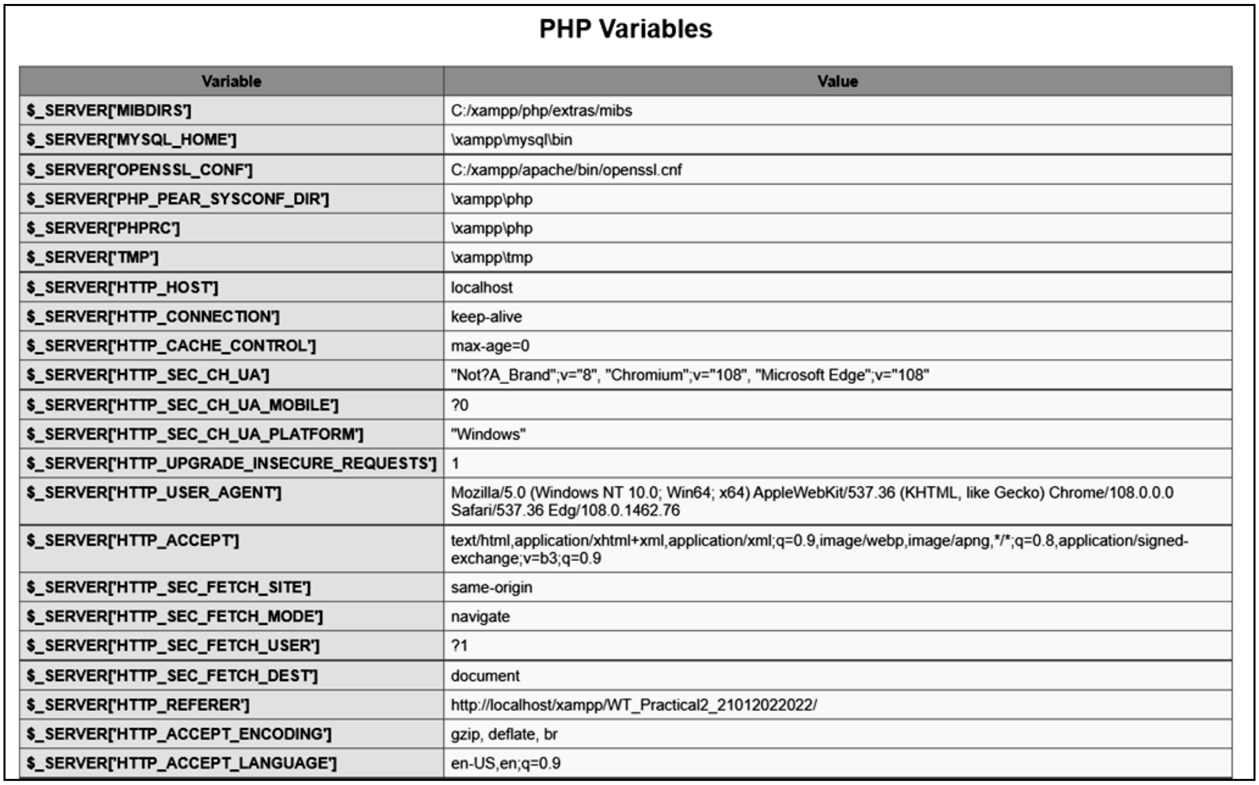
phpinfo():

- Phpinfo page in php gives details like platform information, server environment information, PHP details like version, license info, HTTP headers, and the PHP license. It also includes PHP compilation options, extensions, OS version information, etc.
- Whenever the phpinfo() function is called without any argument then it will all the information about the PHP configuration in the system

Program:

```
<?php
   phpinfo(INFO_VARIABLES);
?>
```

Output:



PHP Variables

Variable	Value
$_SERVER['MIBDIRS']	C:/xampp/php/extras/mibs
$_SERVER['MYSQL_HOME']	\xampp\mysql\bin
$_SERVER['OPENSSL_CONF']	C:/xampp/apache/bin/openssl.cnf
$_SERVER['PHP_PEAR_SYSCONF_DIR']	\xampp\php
$_SERVER['PHPRC']	\xampp\php
$_SERVER['TMP']	\xampp\tmp
$_SERVER['HTTP_HOST']	localhost
$_SERVER['HTTP_CONNECTION']	keep-alive
$_SERVER['HTTP_CACHE_CONTROL']	max-age=0
$_SERVER['HTTP_SEC_CH_UA']	"Not?A_Brand";v="8", "Chromium";v="108", "Microsoft Edge";v="108"
$_SERVER['HTTP_SEC_CH_UA_MOBILE']	?0
$_SERVER['HTTP_SEC_CH_UA_PLATFORM']	"Windows"
$_SERVER['HTTP_UPGRADE_INSECURE_REQUESTS']	1
$_SERVER['HTTP_USER_AGENT']	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/108.0.0.0 Safari/537.36 Edg/108.0.1462.76
$_SERVER['HTTP_ACCEPT']	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
$_SERVER['HTTP_SEC_FETCH_SITE']	same-origin
$_SERVER['HTTP_SEC_FETCH_MODE']	navigate
$_SERVER['HTTP_SEC_FETCH_USER']	?1
$_SERVER['HTTP_SEC_FETCH_DEST']	document
$_SERVER['HTTP_REFERER']	http://localhost/xampp/WT_Practical2_21012022022/
$_SERVER['HTTP_ACCEPT_ENCODING']	gzip, deflate, br
$_SERVER['HTTP_ACCEPT_LANGUAGE']	en-US,en;q=0.9

define():

The define() function defines a constant.

Constants are much like variables, except for the following differences:
-> A constant's value cannot be changed after it is set
-> Constant names do not need a leading dollar sign ($)
-> Constants can be accessed regardless of scope
-> Constant values can only be strings and numbers

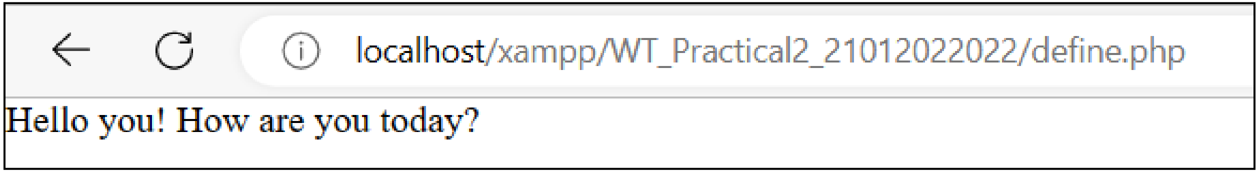
- name **(Required)**: Specifies the name of the constant
- value **(Required)**: Specifies the value of the constant.
- case_insensitive (Optional): Specifies whether the constant name should be case-insensitive. Possible values:
  - TRUE - Case-insensitive (deprecated in PHP 7.3)
  - FALSE - Case-sensitive (this is default)

Syntax:
define(name,value,case_insensitive)

Program:

```
<?php
        define("GREETING","Hello you! How are you today?");
        echo constant("GREETING");
?>
```

Output:



```
Hello you! How are you today?
```

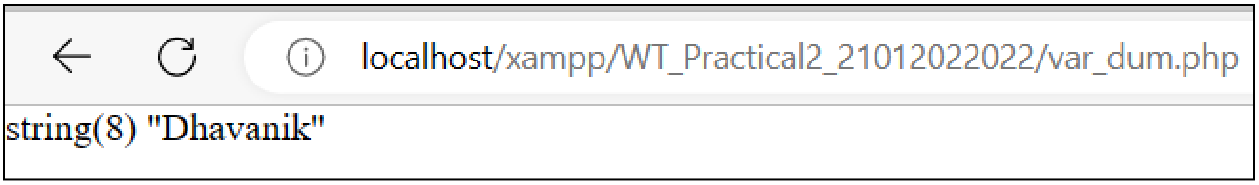
var_dump():

- PHP var_dump() function The var_dump() function is a built-in function of PHP that dumps the information about the variables.
- This information includes the data type and value of the variable.
- In case of string, it also includes the size of the string passed inside the function.

Program:

```
<?php
   $dk = "Dhavanik";
   var_dump($dk);
?>
```

Output:



```
string(8) "Dhavanik"
```

date():

- PHP date function is an in-built function that simplify working with date data types.
- The PHP date function is used to format a date or time into a human readable format.

Syntax:

date(format,[timestamp]);

- "date(…)" is the function that returns the current timestamp in PHP on the server.
- "format" is the general format which we want our output.

Format:

Key	Description
d	The day of the month (from 01 to 31)
D	A textual representation of a day (three letters)
j	The day of the month without leading zeros (1 to 31)
l (lowercase 'L')	A full textual representation of a day
N	The ISO-8601 numeric representation of a day (1 for Monday, 7 for Sunday)
S	The English ordinal suffix for the day of the month (2 characters st, nd, rd or th. Works well with j)
w	A numeric representation of the day (0 for Sunday, 6 for Saturday)
z	The day of the year (from 0 through 365)
W	The ISO-8601 week number of year (weeks starting on Monday)
F	A full textual representation of a month (January through December)
m	A numeric representation of a month (from 01 to 12)
M	A short textual representation of a month (three letters)
n	A numeric representation of a month, without leading zeros (1 to 12)
t	The number of days in the given month
L	Whether it's a leap year (1 if it is a leap year, 0 otherwise)
o	The ISO-8601 year number
Y	A four digit representation of a year
y	A two digit representation of a year
a	Lowercase am or pm
A	Uppercase AM or PM
B	Swatch Internet time (000 to 999)
g	12-hour format of an hour (1 to 12)
G	24-hour format of an hour (0 to 23)
h	12-hour format of an hour (01 to 12)
H	24-hour format of an hour (00 to 23)
i	Minutes with leading zeros (00 to 59)
s	Seconds, with leading zeros (00 to 59)
u	Microseconds (added in PHP 5.2.2)
e	The time zone identifier (Examples: UTC, GMT, Atlantic/Azores)

I (capital i)	Whether the date is in daylights savings time (1 if Daylight Savings Time, 0 otherwise)
O	Difference to Greenwich time (GMT) in hours (Example: +0100)
P	Difference to Greenwich time (GMT) in hours:minutes (added in PHP 5.1.3)
T	Time zone abbreviations (Examples: EST, MDT)
Z	Time zone offset in seconds. The offset for time zones west of UTC is negative (-43200 to 50400)
c	The ISO-8601 date (e.g. 2013-05-05T16:34:42+00:00)
r	The RFC 2822 formatted date (e.g. Fri, 12 Apr 2013 12:01:05 +0200)
U	The seconds since the Unix Epoch (January 1 1970 00:00:00 GMT)

Get a Time
Here are some characters that are commonly used for times:

H	24-hour format of an hour (00 to 23)
h	12-hour format of an hour with leading zeros (01 to 12)
i	Minutes with leading zeros (00 to 59)
s	Seconds with leading zeros (00 to 59)
a	Lowercase Ante meridiem and Post meridiem (am or pm)

PHP Time parameters

- “r” Returns the full date and time

```
echo date("r");
```

- “a”,”A” Returns whether the current time is am or pm, AM or PM respectively

```
echo date("a");
echo date("A");
```

- “g”,”G” Returns the hour without leading zeroes [1 to 12], [0 to 23] respectively

```
echo date("g");
echo date("G");
```

- “h”,”H” Returns the hour with leading zeros [01 to 12],[00 to 23] respectively

```
echo date("h");
echo date("H");
```

- “i”,”s” Returns the minutes/seconds with leading zeroes [00 to 59]

```
echo date("i");
echo date("s");
```

Day parameters

- “d” Returns the day of the month with leading zeroes [01 to 31]

echo date("d");

- “j” Returns the day of the month without leading zeroes [1 to 31]

echo date("j");

- “D” Returns the first 3 letters of the day name [Sub to Sat]

echo date("D");

- “l” Returns day name of the week [Sunday to Saturday]

echo date("l");

- “w” Returns day of the week without leading zeroes [0 to 6] Sunday is represent by zero (0) through to Saturday represented by six (6)

echo date("w");

- “z” Returns the day of the year without leading spaces [0 through to 365]

echo date("z");

Month Parameter

- “m” Returns the month number with leading zeroes [01 to 12]

echo date("m");

- “n” Returns the month number without leading zeroes [01 to 12]

echo date("n");

- “M” Returns the first 3 letters of the month name [Jan to Dec]

echo date("M");

- “F” Returns the month name [January to December]

echo date("F");

- “t” Returns the number of days in a month [28 to 31]

```
//echo date("t");
```

Year parameters

- ➢ "L" Returns 1 if it's a leap year and 0 if it is not a leap year

```
//echo date("L");
```

- ➢ "Y" Returns four digit year format

```
//echo date("Y");
```

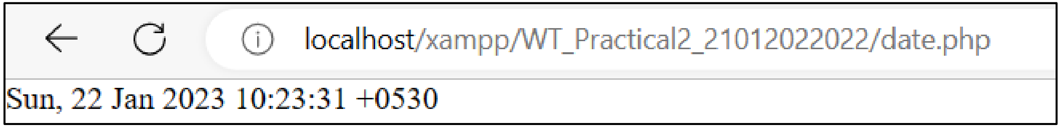
- ➢ "y" Returns two (2) digits year format (00 to 99)

```
//echo date("y");
```

Program:

```
<?php
//Time zones we can set by using date_default_timezone_set()
date_default_timezone_set("Asia/Kolkata");

echo date("r");

//"timezone_identifiers_list()" this function will return an array with list of time zones available.
// $x = timezone_identifiers_list();
// while (list ($key, $val) = each ($x))
// {
//     echo "$key -> $val <br>";
// }
?>
```

Output:



```
Sun, 22 Jan 2023 10:23:31 +0530
```

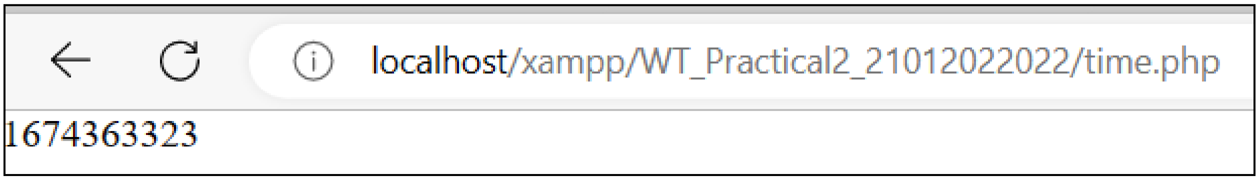
Time():

- The time() function is a built-in function in PHP which returns the current time measured in the number of seconds.
- Parameter: This function does not accept any parameters as shown above.
- Return Value: This function returns the current time measured in the number of seconds since the Unix Epoch.

Program:

```
<?php
// PHP program to demonstrate the use of current
// time in seconds since Unix Epoch

// variable to store the current time in seconds
$currentTimeinSeconds = time();

// prints the current time in seconds
echo $currentTimeinSeconds;
?>
```

Output:



localhost/xampp/WT_Practical2_21012022022/time.php

```
1674363323
```

Q.2 Demonstrate different ways to write a PHP code.

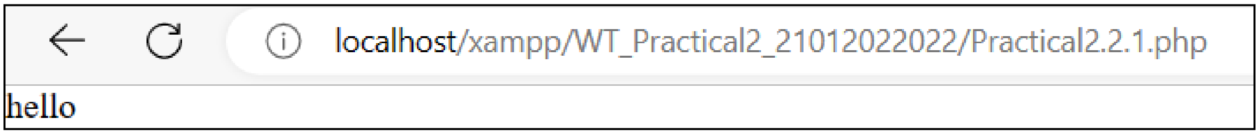
1. Without any HTML markups.

- It is use when interaction with database, api, session, cookie, filesystem, http request (GET,POST), email sending, json data, routing etc.

Program:

```
<?php
echo "hello";
?>
```

Output:



localhost/xampp/WT_Practical2_21012022022/Practical2.2.1.php

```
hello
```

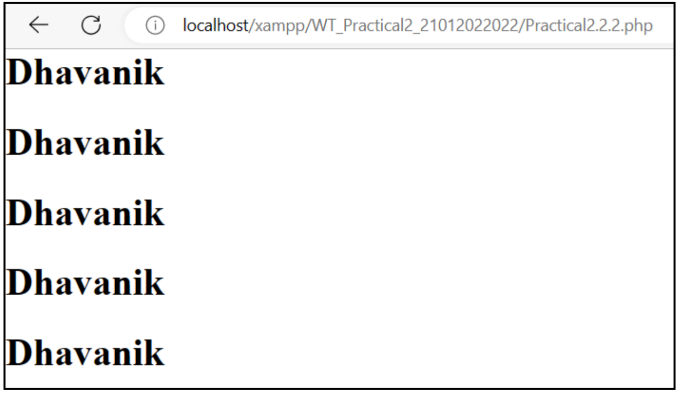
2. Embedding HTML markups in PHP code.

- It uses when you needed to make html control dynamic. Bester way use small HTML code inside PHP. But use large HTML code inside PHP is not batter way.

Program:

```
<?php
   echo "<html><body>";
   for($i=0;$i<5;$i++)
   {
      echo "<div><h1>Dhavanik</h1></div>";
   }
   echo "</body></html>";
?>
```

Output:



localhost/xampp/WT_Practical2_21012022022/Practical2.2.2.php

```
Dhavanik

Dhavanik

Dhavanik

Dhavanik

Dhavanik
```

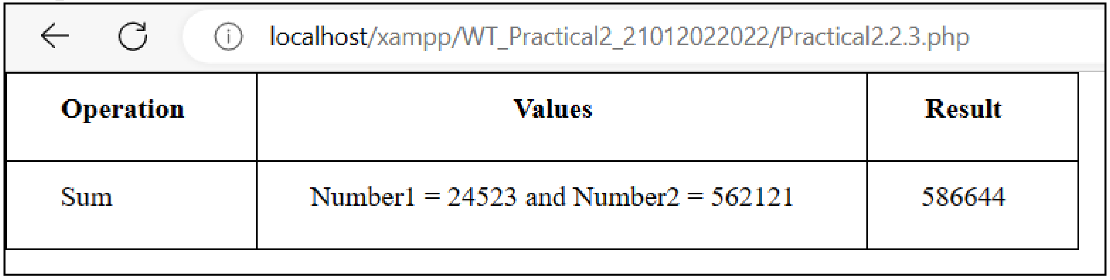
3. Embedding PHP code in HTML.

- You can use PHP code inside HTML for real time update data or make dynamic web page.
- Use when you need to perform operation and show result on web browser.

Program:

```
<?php
function sum ($num1, $num2)
{
   $num3=$num1+$num2;
   return $num3;
}
?>

<html>
   <head>
      <link rel="stylesheet" href="mystyle.css">
   </head>
   <body>
      <table border="1px solid">
         <tr>
            <th>Operation</th>
            <th>Values</th>
            <th>Result</th>
         </tr>
         <tr>
            <td>Sum</td>
            <td>Number1 = 24523 and Number2 = 562121</td>
            <td><?php echo sum(24523,562121); ?></td>
         </tr>
      </table>
   </body>
</html>
```

Output:



localhost/xampp/WT_Practical2_21012022022/Practical2.2.3.php

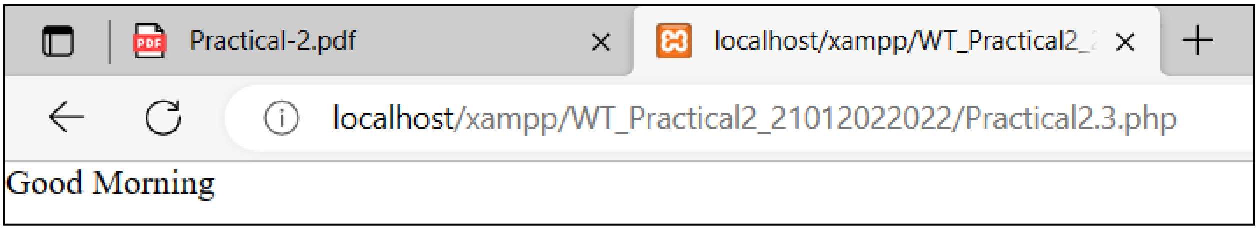
Operation	Values	Result
Sum	Number1 = 24523 and Number2 = 562121	586644

Q.3 Write a program that displays a different message based on time of day. [Note: For example page should display “Good Morning” if it is accessed in the morning.]

Program:

```
<?php
   date_default_timezone_set("Asia/Kolkata");
   $h = date("H");
   $m = date("i");
   if($h>=5 && $h <= 11 && $m>=0 && $m<=60)
   {
      echo "Good Morning";
   }
   elseif($h>=12 && $m>=0 && $h <= 14 && $m <= 60)
   {
      echo "Good Afternoon";
   }
   elseif($h>=15 && $m>=0 && $h <=19  && $m <= 60)
   {
      echo "Good Evening";
   }
   else
   {
      echo "Good Night";
   }
?>
```

Output:


Practical-2.pdf
localhost/xampp/WT_Practical2_
localhost/xampp/WT_Practical2_21012022022/Practical2.3.php

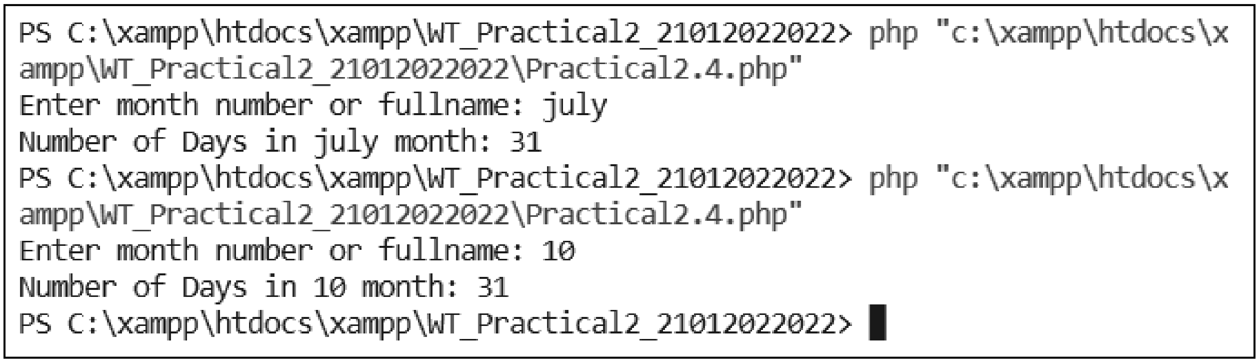
Good Morning

Q.4 Write a PHP function daysInMonth() that takes a month (between 1 and 12) as a parameter and returns the number of days in that month in a non-leap year. [For example daysInMonth(6) should return 30, because June has 30 days.]

Program:

```
<?php
   function daysInMonth($monthNumber)
   {
      if(is_int($monthNumber))
      {
         $arr = array(4,6,9,11);
         if($monthNumber>=1 && $monthNumber<=12)
         {
            if(in_array($monthNumber, $arr))
            {
               echo "Number of Days in $monthNumber month: 30";
            }
            elseif($monthNumber==2)
            {
               echo "Number of Days in $monthNumber month: 28";
            }
            else{
               echo "Number of Days in $monthNumber month: 31";
            }
         }
         else
         {
            echo "Enter valid Month Number";
         }
      }
      if(is_string($monthNumber))
      {
         $monthNumber = strtolower($monthNumber);
         $arr = array("april","june","september","november");
         if(in_array($monthNumber,$arr))
         {
            echo "Number of Days in $monthNumber month: 30";
         }
         elseif($monthNumber=="february")
         {
            echo "Number of Days in $monthNumber month: 28";
         }
         else{
            echo "Number of Days in $monthNumber month: 31";
         }
      }

   }
   $inputMonth = readline("Enter month number or fullname: ");
   daysInMonth($inputMonth);
?>
```

Output:



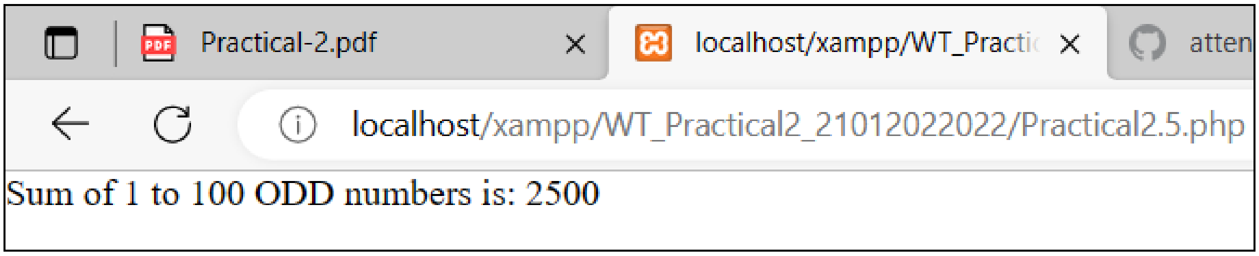
```
PS C:\xampp\htdocs\xampp\WT_Practical2_21012022022> php "c:\xampp\htdocs\x
ampp\WT_Practical2_21012022022\Practical2.4.php"
Enter month number or fullname: july
Number of Days in july month: 31
PS C:\xampp\htdocs\xampp\WT_Practical2_21012022022> php "c:\xampp\htdocs\x
ampp\WT_Practical2_21012022022\Practical2.4.php"
Enter month number or fullname: 10
Number of Days in 10 month: 31
PS C:\xampp\htdocs\xampp\WT_Practical2_21012022022> █
```

Q.5 Write a PHP program to make the sum of first 100 odd numbers.

Programs:

```
<?php
   $sum = 0;
   for($i=1;$i<=100;$i++)
   {
      if($i%2!=0)
      {
         $sum+=$i;
      }
   }
   echo "Sum of 1 to 100 ODD numbers is: ",$sum;
?>
```

Output:


Practical-2.pdf
localhost/xampp/WT_Practic
atten
localhost/xampp/WT_Practical2_21012022022/Practical2.5.php

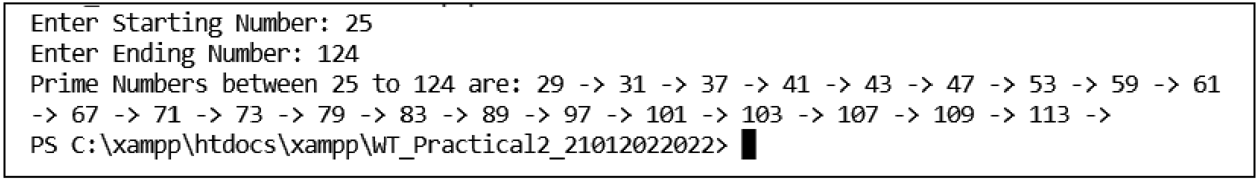
```
Sum of 1 to 100 ODD numbers is: 2500
```

Q.6 Write a PHP program to list out Prime numbers in given range.

Program:

```
<?php
   function primeNumber($num)
   {
      for($i=2;$i<$num;$i++)
      {
         if($num%$i==0)
         {
            return 0;
         }
      }
      return 1;
   }
   $startNumber = readline("Enter Starting Number: ");
   $endNumber = readline("Enter Ending Number: ");
   echo "Prime Numbers between $startNumber to $endNumber are: ";
   for($i=$startNumber;$i<=$endNumber;$i++)
   {
      if(primeNumber($i)==1)
      {
         echo $i," -> ";
      }
   }
?>
```

Output:



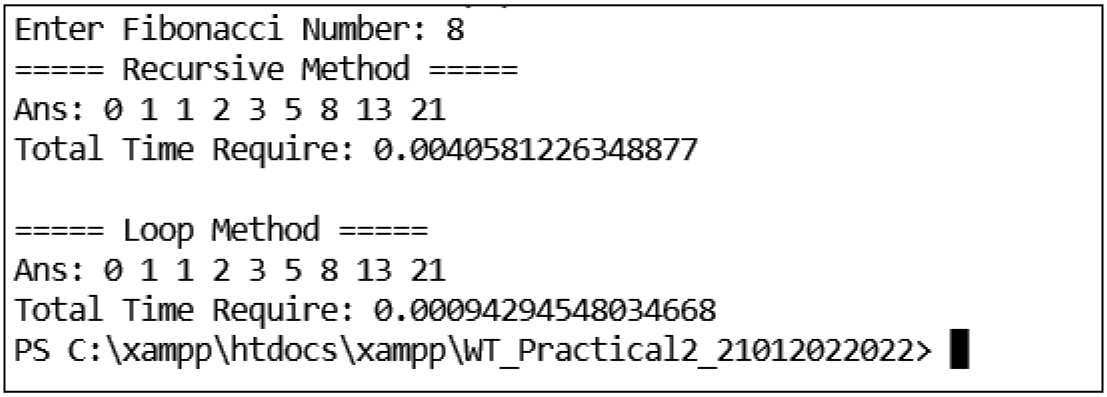
```
Enter Starting Number: 25
Enter Ending Number: 124
Prime Numbers between 25 to 124 are: 29 -> 31 -> 37 -> 41 -> 43 -> 47 -> 53 -> 59 -> 61
-> 67 -> 71 -> 73 -> 79 -> 83 -> 89 -> 97 -> 101 -> 103 -> 107 -> 109 -> 113 ->
PS C:\xampp\htdocs\xampp\WT_Practical2_21012022022>
```

Q.7 Write a PHP program to print fibonacci series with and without using recursion and check which method is efficient. [Note: microtime() function is an inbuilt function in PHP which is used to return the current Unix timestamp with microseconds.]

Program:

```
<?php
    function fibRecursiveMethod($memo,$num)
    {
        if($memo[$num]!=0)
        {
            return $memo[$num];
        }
        if($num==0 || $num==1)
        {
            return $num;
        }
        $memo[$num] = fibRecursiveMethod($memo, $num - 1) +
                              fibRecursiveMethod($memo, $num - 2);
        return $memo[$num];
    }
    function fibLoopMethod($num)
    {
        $arr = [];
        $arr[0] = 0;
        $arr[1] = 1;
        echo $arr[0], " ", $arr[1], " ";
        for($i=2;$i<=$num;$i++)
        {
            $arr[$i] = $arr[$i - 1] + $arr[$i - 2];
            echo $arr[$i], " ";
        }

    }
    $n = readline("Enter Fibonacci Number: ");

    //Recursive
    echo "===== Recursive Method =====\n";
    $start = microtime(true);
    $memo = [];
    for($i=0;$i<=$n;$i++)
    {
        $memo[$i] = 0;
    }
    echo "Ans: ";
    for ($i = 0; $i <= $n;$i++)
    {
        $ans = fibRecursiveMethod($memo,$i);
        echo $ans." ";
    }
    $end = microtime(true);
    $timeRequire = $end - $start;
    echo "\nTotal Time Require: ", $timeRequire;

    //Loop
    echo "\n\n===== Loop Method =====\n";
    echo "Ans: ";
    $start = microtime(true);
    fibLoopMethod($n);
    $end = microtime(true);
    $timeRequire = $end - $start;
    echo "\nTotal Time Require: ", $timeRequire;
?>
```

Output:



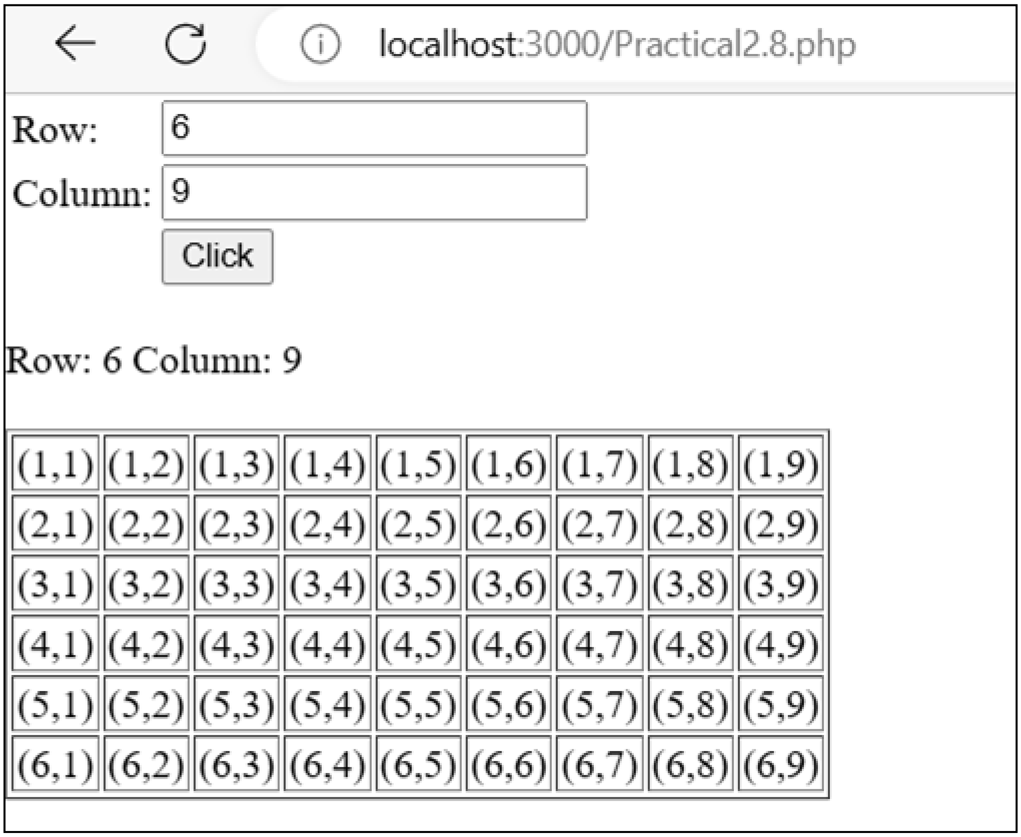
```
Enter Fibonacci Number: 8
===== Recursive Method =====
Ans: 0 1 1 2 3 5 8 13 21
Total Time Require: 0.0040581226348877

===== Loop Method =====
Ans: 0 1 1 2 3 5 8 13 21
Total Time Require: 0.00094294548034668
PS C:\xampp\htdocs\xampp\WT_Practical2_21012022022> 
```

Q.8 Write a PHP program to enter the numbers of rows and columns and in the next page generate the table with given rows and cols.

Program:

```
<html>
    <body>
        <table>
            <form method="post">
                <tr>
                    <td>Row: </td>
                    <td><input type="number" name="row"/>
                </tr>
                <tr>
                    <td>Column: </td>
                    <td><input type="number" name="column"/>
                </tr>
                <tr>
                    <td></td>
                    <td><input type="submit" name="submit" value="Click"/>
                </tr>
            </form>
        </table>
        <?php
            if($_POST)
            {
                $row = $_POST['row'];
                $column = $_POST['column'];
                echo "<br>Row: $row  Column: $column</br></br>";
                echo '<table border="1px solid black">';
                for($i=1;$i<=$row;$i++)
                {
                    echo '<tr>';
                    for($j=1;$j<=$column;$j++)
                    {
                        echo '<td>(',$i,",",$j,')</td>';
                    }
                    echo '</tr>';
                }
                echo '</table>';
            }
        ?>
    </body>
</html>
```

Output:



Row: 6

Column: 9

Row: 6 Column: 9

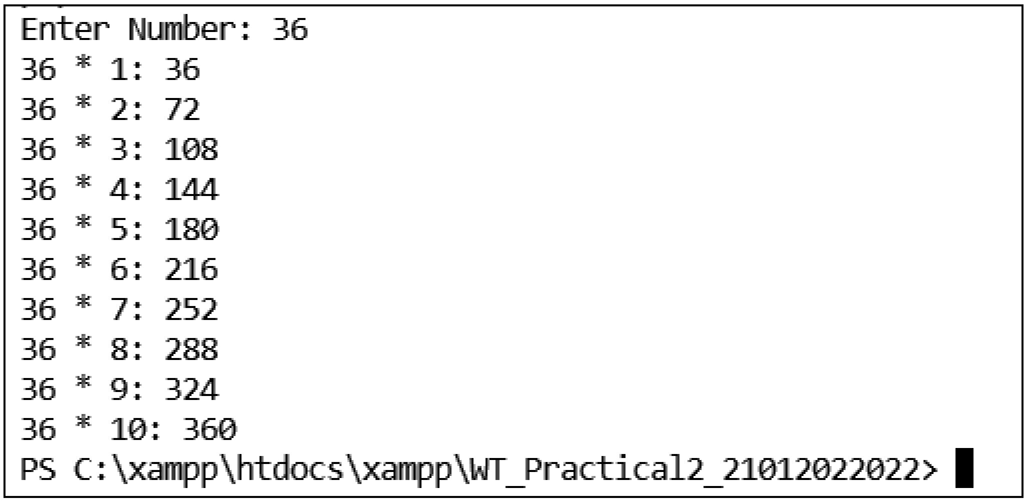
(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)	(1,8)	(1,9)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)	(2,8)	(2,9)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)	(3,7)	(3,8)	(3,9)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)	(4,8)	(4,9)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)	(5,8)	(5,9)
(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)	(6,8)	(6,9)

Q.9 Write a PHP program to print table of a number.
[For example:
9 * 1 = 9
9 * 2 = 18
….].

Program:

```
<?php
   $num = readline("Enter Number: ");
   for($i=1;$i<=10;$i++)
   {
      echo $num." * ".$i.": ".$num * $i."\n";
   }
?>
```

Output:



```
Enter Number: 36
36 * 1: 36
36 * 2: 72
36 * 3: 108
36 * 4: 144
36 * 5: 180
36 * 6: 216
36 * 7: 252
36 * 8: 288
36 * 9: 324
36 * 10: 360
PS C:\xampp\htdocs\xampp\WT_Practical2_21012022022>
```