Bachelor Of Engineering
In
Information Technology

Semester Six, Third Year(Even semester)
6th June 2022
28th Offline Lecture

Padre Conceicao College of Engineering

Verna Goa 403722 India

Web Technology

RC 2019-20

Unit 3

Working with Functions,Arrays,Files, and Directories

Chapter 6

Introduction

- Functions and arrays are two useful programming concepts in any programming as well as scripting language. In simple terms, an array is a special variable with a unique name, which can hold more than one value at a time. On the other hand a function is a block of code with a unique name, which can be reused again and again. If required a function can be called by another function. A function comprises of statements, such as looping statements, conditional statements, and statements that define variables and arrays.

User-Defined Functions in PHP

- In PHP, there are more than 700 built-in functions. PHP also allows us to write our own functions. Reusability is the primary advantage of user-defined functions. Functions make it easy to use the same piece of code several times in your application without the need to write it again and again.
- A function is declared using function keyword, followed by the name of the function and the parenthesis that may contain variable names as arguments. The actual function code is enclosed in curly brackets. The last line of the code block should be a return statement, which returns control back to the called function.

Syntax to create User-Defined Functions in PHP

```
function<function-name>(argument 1, argument 2...., argument n)
{
//code to execute
return;
}
```

Naming conventions

- Some basic rules to be followed while providing names to functions are:

1. Function names are not case sensitive.Therefore,fname(),Fname() and FNAME() all refer to the same function.
2. Function names can contain only letters from the ASCII character set,digits,underscores.
3. Function names cannot begin with a digit.
4. Two functions cannot have the same name as PHP does not support function overloading
5. Reserved keywords cannot be used as function names.

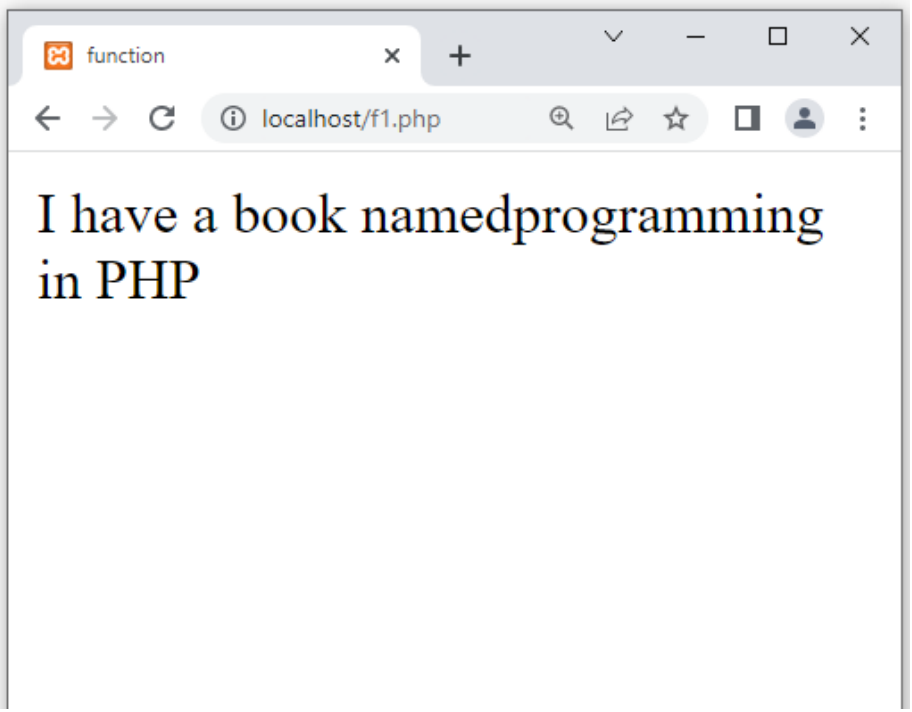
Creating and invoking a function

- You can create a function using function keyword and invoke or call it by its name. The name needs to be followed by a set of parenthesis that contain 0 or more values to be passed to the function.
- In the demo program,the function is named book_name(), that has a single argument $name and its body contains a single statement, which uses echo to print the name of the book

f1 - Notepad

File Edit Format View Help

```
<html>
<head><title>function</title></head>
<body>
<?php
function book_name($name)
{
echo "I have a book named".$name;
}
//function invocation
book_name("programming in PHP");
?>
</body>
</html>
```



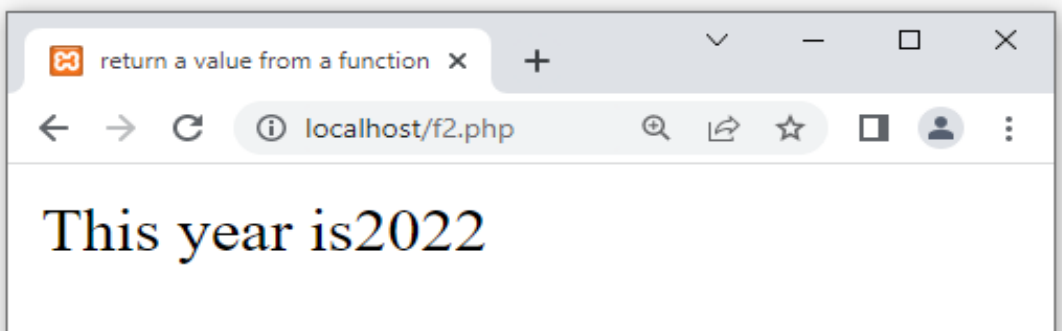
Returning Value from a Function

- You can return a value to the caller function, using the return $value statement. Execution control is transferred to the caller immediately after the return statement. If there are other statements in the function after the return statement,they will not be executed.

f2 - Notepad

File Edit Format View Help

```
<html>
<head><title>return a value from a function</title></head>
<body>
<?php
function get_Year()
{
$year=date("Y");
return $year;
//value returns to the calling function in our case echo function
}
//function invocation
echo "This year is".get_Year();
?>
</body>
</html>
```



Bachelor Of Engineering
In
Information Technology

Semester Six, Third Year(Even semester)
7th June 2022
29th Offline Lecture

Padre Conceicao College of Engineering

Verna Goa 403722 India

Understanding the Variable Scope

- A key concept related to user-defined function in PHP is variable scope; the extent of a variable’s visibility within the space of a PHP program. By default , variables used within a function are local. A local variable is restricted to the function space alone, and it can’t be viewed or manipulated outside the function in which it is declared. In the demo program the $score is defined in the main program, and the change_score() function contains code to change the value of this variable. After running this function , the value of $score remains at its original setting, because the changes made to $score within the change_score() function remain “local” to the function and do not reflect in the program. The global keyword helps use the same variable everywhere in your code, even inside the functions.

Passing Arguments by Reference

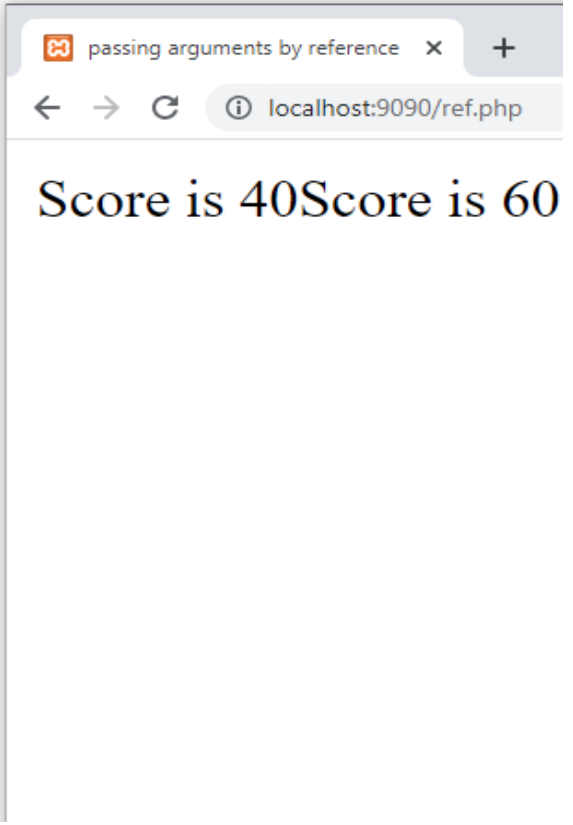
- An alternative to returning a result or using a global variable is to pass a reference to a variable as an argument to the function. This means that any change made to the variable within the function affects the original variable.

Passing Arguments by Reference

ref - Notepad

File Edit Format View Help

```
<html>
<head><title>passing arguments by reference</title></head>
<body>
<?php
//function definition
//change the value of $score
function change_score(&$score)  //$score as a reference argument
{
$score = $score +20;
}
//Define a variable in the main program
$score = 40;
echo "Score is ".$score;                //output:score is:60
//run the change_score() function
change_score($score);
//print $score again
echo "Score is ".$score;                //output:score is:60
?>
</body>
</html>
```



Built-in Functions in PHP

- PHP offers many built-in functions that can be used in your code. These functions are well documented and very helpful in achieving programming goals.
- String manipulations Functions
- Date and Time Functions in PHP
- Mathematical Functions

String manipulations Functions

1. Getting the length of a string
2. Splitting the string into an Array
3. Joining Array Elements into a Single String
4. Finding the position of a string in another string
5. Repeating the same string many times
6. Reversing a string

Getting the length of a string

- Strlen()-Returns the length of the string
- Syntax to define strlen() function

```
int strlen (string $string)
```

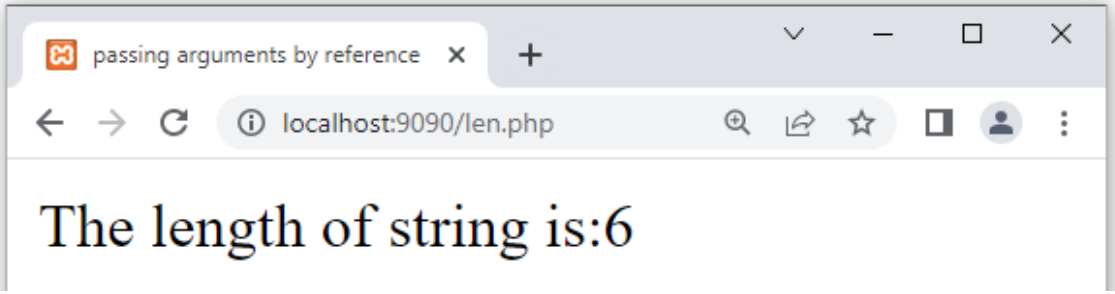
- Parameters – string – Required. The string being measured for length
- Return Value – The length of the string on success, and 0 if the string is empty.

Getting the length of a string

len - Notepad

File Edit Format View Help

```
<html>
<head><title>passing arguments by reference</title></head>
<body>
<?php
$string="kogent";
echo "The length of string is:".strlen($string);
//output:The length of string is:6
?>
</body>
</html>
```



Splitting the string into an Array

- Explode() – breaks a string into an array
- Syntax to define explode() function

```
array explode(separator,string $string,limit)
```

Parameters

1. Separator-Required . Specifies where to break the string.
2. String-Required. Specifies the string to be split.
3. Limit-Optional. Specifies the maximum number to return.

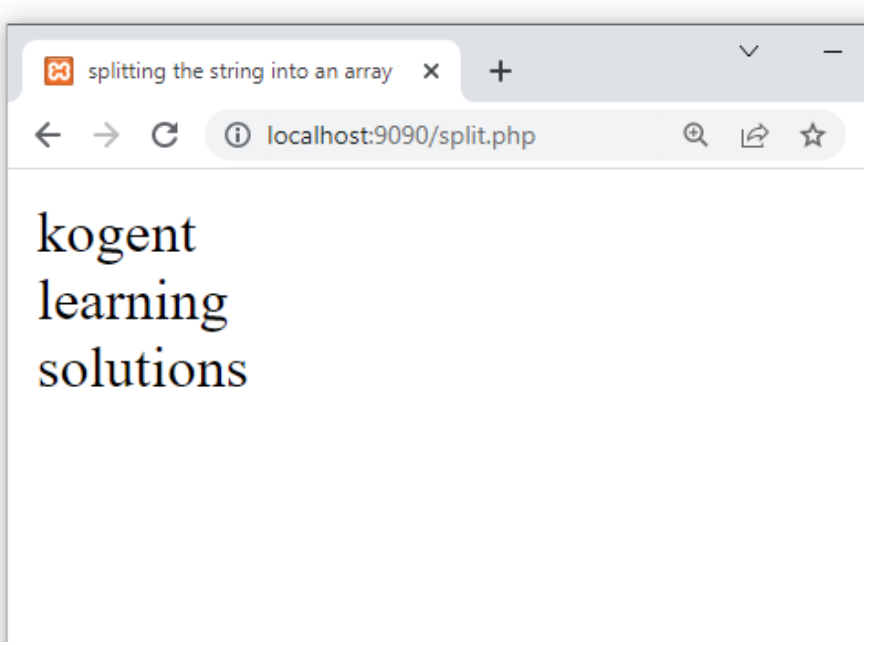
Return Value- If separator is an empty string (“”) explode() returns false. If separator is not contained in the string then an empty array is returned.

Splitting the string into an Array

split - Notepad

File Edit Format View Help

```
<html>
<head><title>splitting the string into an array</title></head>
<body>
<?php
$string="kogent learning solutions";
$element =explode(" ",$string);
echo $element[0]."<br/>";
echo $element[1]."<br/>";
echo $element[2];
?>
</body>
</html>
```



Joining Array Elements into a Single String

- Implode () :Joins array elements with a string.
- Syntax to define implode() function

String implode (separator , array)

Parameters

1. Separator-Optional . Specifies what to put between the array elements, default is an empty string.
2. Array – Required. The array to join to a string.

Return Value- Returns a string containing a string representation of all array elements in the same order, with separator string between each element.

Joining Array Elements into a Single String

join - Notepad

File Edit Format View Help

```
<html>
<head><title>Joining array elements into a single
string</title></head>
<body>
<?php
$string=array('kogent','is','a','learning','solution');
echo implode("",$string);
?>
</body>
</html>
```

Finding the position of a string in another string

- Strpos() –Finds the position of first occurrence of a string in another string (case-sensitive).
- Syntax to define strops() function

int strops (string,find,start)

Parameters

String-Required . Specifies the string to search.

Find-Required. Specifies the string to find.

Start-Optional. Specifies where to begin the search.

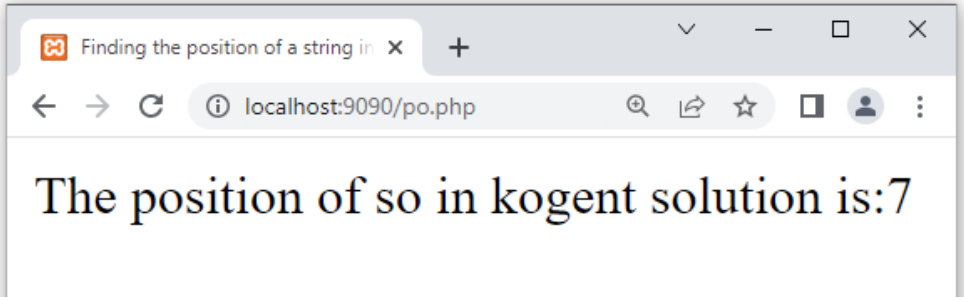
Return Value- Returns the position as an integer. If it is not found, strops() will return Boolean false.

Finding the position of a string in another string

po - Notepad

File Edit Format View Help

```
<html>
<head><title>Finding the position of a string in another string</title></head>
<body>
<?php
$string="kogent solution";
echo "The position of so in kogent solution is:".strpos($string,"so"); //output:7
?>
</body>
</html>
```



Repeating the same string many times

- str_repeat() – repeats a string.
- Following syntax is used to define the str_repeat() function

```
string str_repeat (string $string  , repeat)
```

Parameters

String-Required. Specifies the string to be repeated

Repeat-Required. Specifies the number of times the string is repeated. The number must be greater or equal to 0

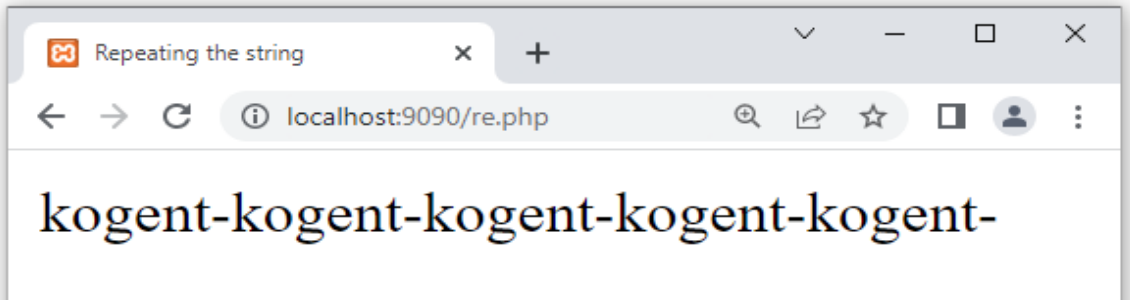
Return Value- Returns the repeated string.

Repeating the same string many times

re - Notepad

File Edit Format View Help

```
<html>
<head><title>Repeating the string</title></head>
<body>
<?php
$string="kogent-";
echo str_repeat($string,5);
?>
</body>
</html>
```



Reversing a string

- Strrev( )- Reverses a string.
- Syntax to define the strrev () function

  int strrev (string $string)
- Parameters- The string to be reversed
- Return Value – Returns the reversed string

Reversing a string

rev - Notepad

File Edit Format View Help

```
<html>
<head><title>Reversing a string</title></head>
<body>
<?php
$string = "kogent";
echo "The reversed string is:".strrev($string);
?>
</body>
</html>
```

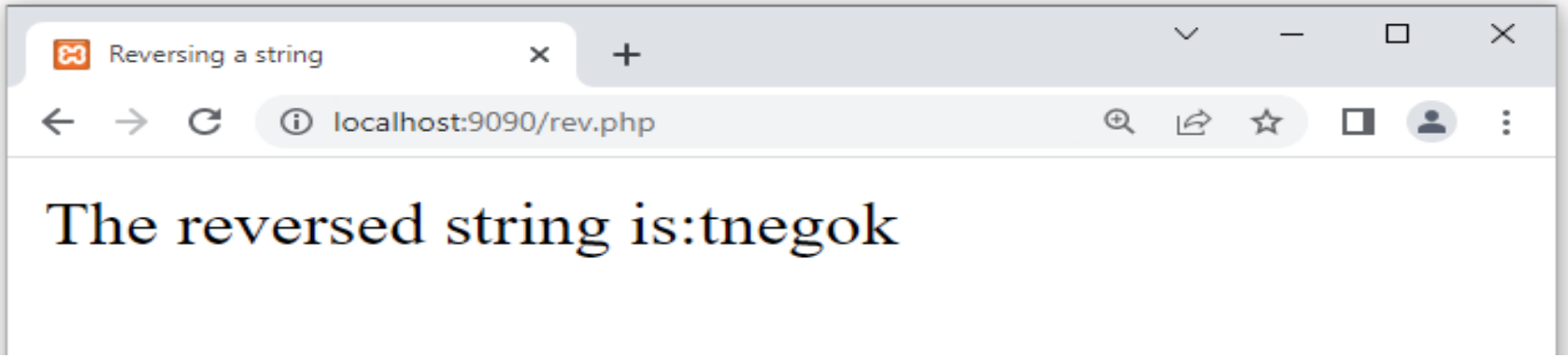


Table 6.1 String Manipulation Functions

Sr.No	Function	Description	Version
1	bin2hex()	Converts a string of ASCII characters to hexadecimal values	3
2	chr()	Returns a character from a specified ASCII value	3
3	chunk_split()	Splits a string into a series of smaller parts	3
4	c()	Converts a string from one Cyrillic character-set to another	3
5	count_chars()	Returns how many times an ASCII character occurs within a string and returns the information	4
6	echo()	Outputs strings	3
7	fprintf()	Writes a formatted string to a specified output stream	5
8	html_entity_decode()	Converts HTML entities to characters	4
9	htmlentities()	Converts characters to HTML entities	3

Table 6.1 String Manipulation Functions

Sr.No	Function	Description	Version
11	money_format()	Returns a string formatted as a currency string	
12	print()	Outputs a strings	
13	printf()	Outputs a formatted string	
14	rtrim()	Deletes whitespace from the right side of a string	
15	sscanf()	Parses input from a string according to a format	
16	str_ireplace()	replaces some characters in a string (case-insensitive)	
17	str_replace()	Replaces some characters in a string (case-sensitive)	
18	str_split()	Splits a string into an array	
19	str_word_count()	Count the number of words in a string	
20	strcasecmp()	Compares two strings (case-insensitive)	

Table 6.1 String Manipulation Functions

Sr.No	Function	Description	Version
21	strchr()	Finds the first occurrence of a string inside another string (alias of strstr())	3
22	strcmp()	Compares two strings (case-sensitive)	3
23	stripos()	Returns the position of the first occurrence of a string inside another string (case-insensitive)	5
24	strrpos()	Finds the position of the last occurrence of a string inside another string (case-sensitive)	3
25	strstr()	Finds the first occurrence of a string inside another string (case-sensitive)	3
26	strtolower()	Converts a string to lowercase letters	3
27	Strtoupper()	Converts a string to uppercase letters	3
28	substr_count()	Counts the number of times a substring occurs in a string	4

Table 6.1 String Manipulation Functions

Sr.No	Function	Description	Version
31	ucfirst()	Converts the first character of a string to uppercase	3
32	ucwords()	Converts the first character of each word in a string to uppercase	3
33	vfprintf()	Writes a formatted string to a specified output stream	5
34	vprintf()	Outputs a formatted string	4
35	wordwrap()	Wraps a string to a given number of characters	4

Date and Time Functions in PHP

- The Date and Time built-in Function in PHP provides the ability to read the system time in various formats and helps to manipulate the date information.

Some of the most common date and time functions are

1. Finding the current date and Time
2. Validating a Given Date
3. Returning the Unix Timestamp for a Date
4. Parsing English Textual Datetime Description into a Unix Timestamp

Finding the current date and Time

- date() – formats the local time and date.
- Syntax: date (format, timestamp)
- Return Value – Returns the formatted date and time
- Parameters – format – Required. Specifies how to return the result.

Parameters

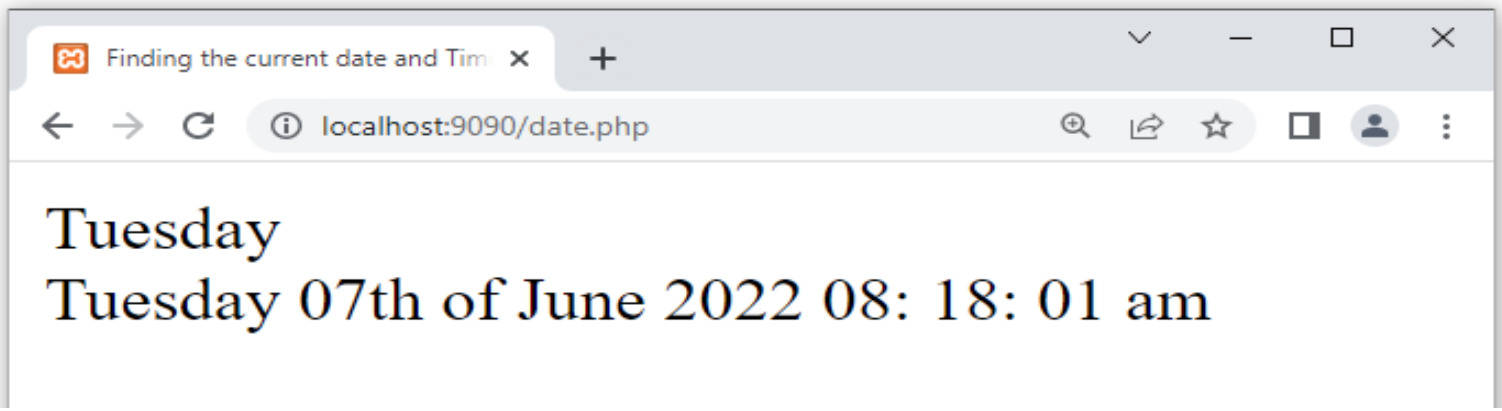
1. d -Specifies the day of the month (from 01 to 31)
2. D – Specifies a textual representation of a day (three letters)
3. F – specifies a full textual representation of a month (January to December)
4. t- Specifies number of days in the given month
5. g – specifies 12 hour format of an hour (1 to 12)
6. L – Specifies whether it is a leap year (1 if it is a leap year,0 otherwise)
7. l - specifies a full textual representation of a day
8. h - specifies 12 hour format of an hour (01 to 12)
9. i – specifies minutes with leading zero (00 to 59)
10. Y – specifies a four digit representation of a year
11. a – specifies a lowercase am or pm
12. S – specifies the English ordinal suffix for the day of the month (st,nd,rd or th)

Finding the current date and Time

date - Notepad

File Edit Format View Help

```
<html>
<head><title>Finding the current date and Time</title></head>
<body>
<?php
echo date("l")."<br/>";
echo (date ("l dS \of F Y h: i: s a")." ");
?>
</body>
</html>
```



Validating a Given Date

- Checkdate ( ) – Validates a date

Syntax - checkdate (int month , int day, int year)

Parameters:

- month - required. Specifies the month
- day-required. Specifies the day
- year-required. Specifies the year

Return Value – Returns true if the specified date is valid, else return false

Validating a Given Date

val - Notepad

File Edit Format View Help

```
<html>
<head><title>Validating a Given Date</title></head>
<body>
<?php
var_dump(checkdate(12,31,2000));
var_dump(checkdate(2,29,2001));
?>
</body>
</html>
```

Returning the Unix Timestamp for a Date

mktime () – Returns the Unix timestamp for a date.

- Syntax int mktime (hour,minute,second,month,day,year,is_dst)

Parameters:

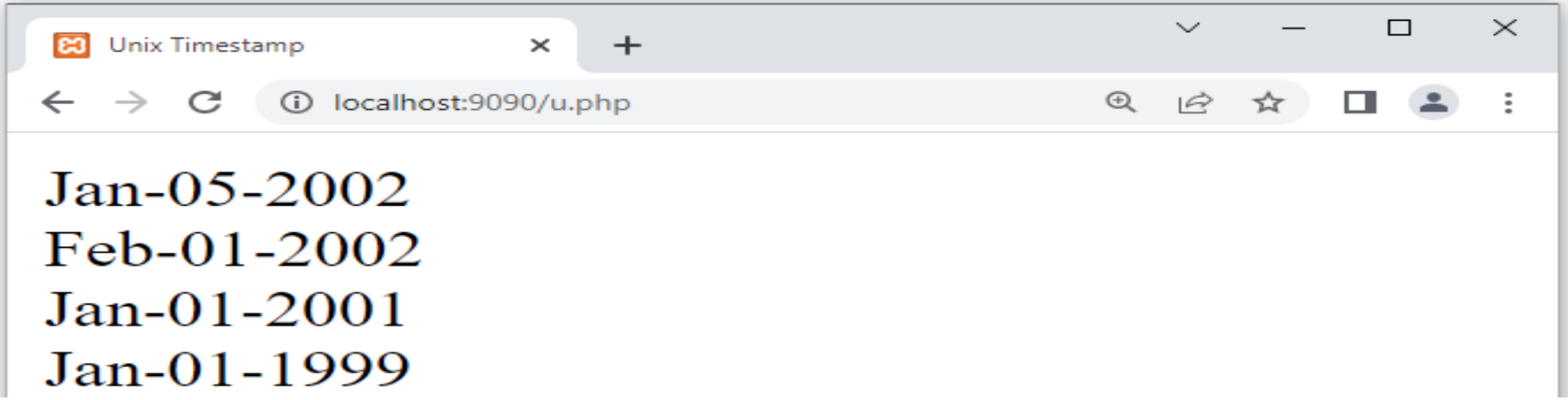
- Hour – Optional. Specifies the hour
- Minute - Optional. Specifies the Minute
- Second- Optional. Specifies the Second
- Month - Optional. Specifies the numerical Month
- Day - Optional. Specifies the Day
- Year - Optional. Specifies the Year
- Is_dst - Optional. Set this parameter to 1 if the time is during daylight savings time (DST)
- Return Value – Returns the Unix timestamp of the given Date

Returning the Unix Timestamp for a Date

u - Notepad

File Edit Format View Help

```
<html>
<head><title>Unix Timestamp</title></head>
<body>
<?php
echo(date("M-d-Y",mktime(0,0,0,12,36,2001))."<br/>");
echo(date("M-d-Y",mktime(0,0,0,14,1,2001))."<br/>");
echo(date("M-d-Y",mktime(0,0,0,1,1,2001))."<br/>");
echo(date("M-d-Y",mktime(0,0,0,1,1,99))."<br/>");
?>
</body>
</html>
```



Assignment 3

- Q4) **Write** a PHP program to Join array elements into a single string and display the output. (5 marks)

- *Assignment Announced to students : AA :07th June 2022*
- *Assignment to be Submitted by students : AS: 20th June 2022*