



**MALAD KANDIVALI EDUCATION SOCIETY'S
NAGINDAS KHANDWALA COLLEGE OF COMMERCE,
ARTS & MANAGEMENT STUDIES & SHANTABEN NAGINDAS
KHANDWALA COLLEGE OF SCIENCE**
MALAD [W], MUMBAI – 64
(AUTONOMOUS)

**(Reaccredited 'A' Grade by NAAC)
(AFFILIATED TO UNIVERSITY OF MUMBAI)
(ISO 9001:2015)**

CERTIFICATE

Name: Ms. MENARIA KOMAL CHANDRAPRAKASH

Roll No:42 Programme: BSc IT Semester: II

This is certified to be a bonafide record of practical works done by the above student in the college laboratory for the course **Web Programming II** (Course Code:**2024UISPR**) for the partial fulfillment of Second Semester of BSc IT during the academic year 2020-2021.

The journal work is the original study work that has been duly approved in the year 2020-2021 by the undersigned.

External Examiner

**Subject-In-Charge
(Anisha Asirvatham)**

Date of Examination: (College Stamp)

Name: MENARIA KOMAL CHANDRAPRAKASH

Roll No: 42

Subject: Web ProgrammingII Practical (2024UISPR)

Class: F.Y.B.SC (IT)

SEMESTER-II

NO	DATE	TITLE	SIGN
1.	27/01/21	XML	
2.	03/02/21	PHP Basics-I	
3.	10/02/21	PHP Basics-II	
4.	17/02/21	String Functions and Arrays	
5.	24/02/21	PHP and Database	
6.	03/03/21	Write a program to demonstrate use of sessions and cookies.	
7.	10/03/21	Create a shopping cart using php and Mysql	
8.	17/03/21	Write a program to demonstrate XML parsing with php	
9.	24/01/21	Demostrate json with php	

5 PHP and database

Aim (a) :- write a PHP code to create :-
(i) create database college (ii) create
table Department (Dname, Dno, Number -
(f - Faculty))

Theory :-

- 1) A PHP script starts with <?php and ends with ?>
- 2) mysqli - connect() Function is used to connect with MySQL database
- 3) mysqli - close() Function is used to Disconnect with MySQL database
- 4) mysqli - query() Function is used to create database table.
- 5) mysqli - select_db() function is used to change the default database for the connection
- 6) die :- It is used to print message and exit from the current PHP script
- 7) mysqli - error() Function return the description of the error occurred during the last mysqli Function call

Code :-

```
<?php  
$servername = "localhost";  
$username = "root";  
$password = "";  
$conn = mysqli_connect($servername, $username,  
$password);  
  
if (!$conn)  
    die('could not connect : '. mysqli_error());  
if (mysqli_query($conn, "create database  
college"))  
  
echo "Database created successfully";  
else  
echo "Error creating Database : ". mysqli-  
error();  
mysqli_select_db($conn, "college");  
  
$query = "create table Department(dname-  
varchar(50), dno smallint, no-of-faculty-  
small int);  
  
if (mysqli_query($conn, $query))  
echo "Table created successfully";  
else
```

Expt. No.....

Page No..... Date

```
echo "Error creating table : ", mysqli_error();  
}
```

```
mysqli_close($conn);  
?>
```

Teacher's Signature :

Output :-

Database created successfully

Table created successfully

5 PHP and Database

Aim (b) :- Write a PHP program to create a database named "College". Create a table named "Student" with following fields (sno, name, percentage). Insert 3 records of your choice. Display the names of the students whose percentage is between 35 to 75 in a tabular format.

Theory :-

- 1] A php script starts with `?php` and ends with `?>`
- 2] mysqli_connect() Function is used to connect with MySQL data
- 3] mysqli_close() Function is used to disconnect with MySQL database
- 4] mysqli_query() Function is used to create database, table
- 5] mysqli_error() Function return the description of the error occurred during the last mysqli Function call.
- 6] The Insert Into statement is used to add new records to MySQL Table

- 7) The select statement is used to select data from one or more tables.
- 8) mysqli_num_rows() Function checks if there are more than zero rows returned.
- 9) mysqli_fetch_assoc() Function Fetches a result row as an associative array.
- 10) The Where clause is used to extract only those records that fulfill a specified condition.

Code :-

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$database = "college";
$conn = mysqli_connect($servername, $username,
$password, $database);

if (!$conn)
die("connection Failed: ". mysqli_connect_error());
$query = "Create table Student (.sno int(6)
primary key, .sname Varchar (20), percentage
decimal (7,2));";
```

```
if (mysql_query($conn, $query))
echo "Table created successfully";
else
echo "Error creating table : ".mysql_error($conn);
$query = "Insert into student values (107,
'Allena', 48)";
if (mysql_query($conn, $query))
echo "Record 1 inserted successfully";
else
echo "Error inserting record 1 : ".mysql_error();
$query2 = "Insert into student values (108,
'Smythi', 48)";
if (mysql_query($conn, $query2))
echo "Record 2 inserted successfully";
else
echo "Error inserting record 2 : ".mysql_error();
$query3 = "insert into student values (109,
'scott', 63.2)";
if (mysql_query($conn, $query3))
echo "Record 3 inserted successfully";
else
echo "Error inserting record 3 : ".mysql_error();
```

```
$sql = "Select * From student where
percentage >= 35 and
percentage <= 35";
```

```
$result = mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0)
echo "<table border = \"1\">
<tr>
<th>sno </th>
<th>sname </th>
<th>percentage </th>
</tr>";
```

```
while ($row = mysqli_fetch_assoc($result))
```

```
{
echo "<tr>";
echo "<td>". $row['sno']. "</td>";
echo "<td>". $row['sname']. "</td>";
echo "<td>". $row['percentage']. "</td>";
echo "</tr>";
}
```

```
echo "</table>";
```

```
}
```

```
else
```

```
{ echo "Table is empty"; }
```

```
mysqli_close($conn); } }
```

Teacher's Signature : _____

Output :-

Table created successfully
Record 1 inserted successfully
Record 2 inserted successfully
Record 3 inserted successfully

Sno	Sname	Percentage
107	Allena	48.00
108	Smithi	48.00
109	scott	63.2

5 PHP and Database

Aim (C) :- Design a PHP page for authenticating a user.

\$ POST global array length before executing the authentication code block.

Once the User authentication form is submitted, then this global array will contain the value of the form input fields.

Code :- <Html>

<head>

<Title> Login Form <Title>

<head>

<body>

<form name="frmUser" method="post" action="">

<div class="message"> <? PHP if

(bmessage != "") { echo \$message; } ?>

```
<table border = "0" cellpadding = "10"
  cellspacing = "1" width = "500" align =
  "center" class = "tblLogin" >
  <tr class = "tableheader" >
    <td align = "center" colspan = "2" >
      Enter Login Details </td>
    </tr >
  <tr class = "tablerow" >
    <td >
      <input type = "text" name = "userName"
        placeholder = "User Name" class = "login-input" >
    </td >
    <td >
      <input type = "password" name = "password"
        placeholder = "Password" class = "login-input" >
    </td >
  </tr class = "tablerow" >
  <tr class = "tableheader" >
    <td align = "center" colspan = "2" >
      <input type = "submit" name = "Submit"
        value = "Submit" class = "btnSubmit" >
    </td >
  </tr >
</table>
</form>
```

? PHP

\$message = " ";

if (count(\$_POST) > 0) {

\$conn = mysqli_connect ("localhost", "root",
 "phppot-example");

\$result = mysqli_query (\$conn, "Select *
From users Where user-name = '".\$_POST["UserName"]."' and password='".\$_POST["password"]."');

\$count = mysqli_num_rows (\$result);

if (\$count == 0) {

\$message = " Invalid Username or
Password! "; }
else {

\$message = " You are successfully
authenticated! "; }

}

?>
</body>
</html>

Enter Login Details

User Name

Password

Submit

5 PHP and Database

(d) Write a program to send email with attachment

Aim :- creating a page using php

Theory :-

\$to :- Recipient email address.

\$from :- Sender email address.

\$fromName :- Sender name.

\$Subject :- Subject of the email

\$file :- Relative path of the file that you want to attach with the email.

\$htmlContent - Body content of the email

Code :-

<?PhP

```
$to = "Suryasen@gmail.com";
```

```
$subject = "Form";
```

```
$from = "Komalmenariapsm2003@gmail.com";
```

```
$message = "<pre>
```

```
<h3>Hello user!,</h3>
```

```
</pre>
```

(p) > Thanks for registering (p)
(p) > Your password is : (b) > surya sen@gmail.com (b)";

```
$headers = "MIME-Version: 1.0" . "\n";
$headers = "Content-type:text/html; charset
=iso-8859-1" . "\n";
$headers = "From:$from". "\n";
```

```
$success= mail($to, $subject, $message,
$headers);
```

```
if (!$success) {
```

```
echo "Mail to " . $to . " failed .";
```

```
}
```

```
else {
```

```
echo "Success : Mail was sent to . $to ;
```

```
}
```

```
?>
```

(b) 2 visitors
- 00 to 02
2 visitors bto 910

Output :-

Success! Mail was sent to suryaasen@gmail.com.

6 Write a program to demonstrate use of sessions and cookies.

Aim :- Creating a program which have sessions and cookies.

Theory :-

- 1) A session is started with session_start() function
- 2) Session variables are set with the PHP global variable :- \$SESSION
- 3) A cookie is created with the session_setcookie() Function
- 4) iset() function used to find out if the cookie is set
- 5) In this program creates cookie named "user" with value "Komal".
- 6) The cookies will expire after 15 days
- 7) The "1" means that the cookie is available in entire website.

Code :-

<?php

```
session_start();
if (!isset($_SESSION['views']))
{
    $_SESSION['views'] = $_SESSION['views']
        + 1;
}
else
{
    $_SESSION['views'] = 1;
}
echo "views = " . $_SESSION['views'];
?>
```

cookie code (setting and retrieving)

<?php

```
$cookie_name = "user";
$cookie_value = "Komal";
setcookie($cookie_name, $cookie_value, time()
    + (86400 * 15), "/");
?>
```

<html>

<body>

<?php

```
if (!isset ($cookie[$cookie_name]))  
{  
    echo "Cookie named \" ". $cookie_name . " \" is  
    not set! <br>";  
}  
  
else  
{  
    echo "Cookie " . $cookie_name . " is not set! <br>";  
    echo "Value is :- " . $cookie[$cookie_name];  
}  
  
</body>  
</html>
```

Output :-

cookie 'user' is set (trotzdem)
value is (Komal@23.12.1999)

[Komal@23.12.1999 ~] [Komal@23.12.1999 ~]

get (Komal@23.12.1999)

Cookie has gotten the cookie

(Komal@23.12.1999 ~)

(tutto)
(piedi)
qnd P)

7 Create a shopping cart using php and MySQL.

Aim :- design shopping cart using php & MySQL

Theory :-

mysqli_connect_error() function show the error in the code.

die :- It is used for if connection is not successful it will print the content

session_start() function :- Start the session of php code.

Code :-

config.php

L7php

```
$con = mysqli_connect('localhost', 'root', '', 'cart');
```

```
if (mysqli_connect_errno()) {
```

```
    die("Connection Fail : ". mysqli_connect_error());
```

```
}
```

index.php

<?php

session_start();

require_once 'config.php';

if (!empty(\$_GET['action'])) {
switch(\$_GET['action']) {

case 'add':

if (!empty(\$_POST['quantity'])) {
\$pid = \$_GET['pid'];

\$query = "SELECT * FROM products
WHERE id = " . \$pid;

\$result = mysqli_query(\$con, \$query);

while (\$product = mysqli_fetch_array(\$result))
\$itemArray = [

\$product['code'] => [

'name' => \$product['name'],

'code' => \$product['code'],

'quantity' => \$_POST['quantity'],

'price' => \$product['price'],

'image' => \$product['image']

]

];

```
if (isset ($SESSION['cart-item']) && !empty  
($SESSION['cart-item'])) {  
    if (in_array ($product['code'], array_keys  
($SESSION['cart-item']))) {  
        foreach ($SESSION['cart-item'] as $key =>  
            $value) {  
            if ($product['code'] == $key) {  
                if (!empty ($SESSION['cart-item'][$key]  
                            ['quantity'])) {  
                    $SESSION['cart-item'][$key]['quantity']  
                        = 0; }  
                $SESSION['cart-item'][$key]['quality']  
                    += $POST['quantity']; } } }  
    } else {  
        $SESSION['cart-item'] += $itemArray; } } }  
} else {  
    $SESSION['cart-item'] = $itemArray; } } }
```

```
break ;
case 'remove':
if (!empty($_SESSION['cart_item'])) {
    foreach ($_SESSION['cart_item'] as
        $key => $value) {
        if ($_GET['code'] == $key) {
            unset($_SESSION['cart_item'][$key]);
        }
    }
    if (empty($_SESSION['cart_item'])) {
        unset($_SESSION['cart_item']);
    }
}
break ;
case 'empty':
unset($_SESSION['cart_item']);
break ;
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
```

```
<title> Shopping Cart </title>
</head>
<body>
<div class = "container py-5">
  <div>
    <h3> Cart </h3>
    <a class = "btn" href = "index.php?> action = empty" > All Item Remove </a>
  </div>
  <div>
    <?php
      $total_quantity = 0;
      $total_price = 0;
    ?>
    <table>
      <tbody>
        <tr>
          <th> Name </th>
          <th> Code </th>
          <th> Quantity </th>
          <th> Item price </th>
          <th> Price </th>
          <th> Remove </th>
        <tr>
```

<?php

```
if (!isset($_SESSION['cart-item']) &&
    !empty($_SESSION['cart-item'])) {
```

```
foreach ($_SESSION['cart-item'] as $item)
```

}

```
    $item['price'] = $item['quantity'] *
```

```
    $item['price'];
```

?

<tr>

<td>

```
<img src="" alt="<?=$item['image']?>"
```

```
alt="" ?=> $item['name'] ?> with="100")
```

```
<?=$item['name']?>
```

<td>

```
<td> ?=> $item['code'] ?> <td>
```

```
<td> ?=> $item['quantity'] ?> <td>
```

```
<td> ?=> number_format($item['price'],
```

2) ?> <td>

```
<td> ?=> number_format($item['price'], 2)
```

?> </td>

<td>

```
<a href="index.php?action=remove"
```

```
?code => ?=> $item['code']?>">
```

X <td> <td>
</tr>

{?php

\$total_quantity += \$item['quantity'];
\$total_price += (\$item['price'] * \$item['quantity']);

}

if (!isset(\$_SESSION['cart_item'])) {
 if (!empty(\$_SESSION['cart_item'])) {
 ?>

<tr>

<td> Total : <td>

<td> <? = \$total_quantity ?> <td>

<td> </td>

<td> <? = number_format(\$total_price, 2); ?> </td>

<td> </td>

</tr>

{?php }

?>

```
</tbody>
</table>
</div>
```

```
<div>
<div>
<h1> Products List </h1>
<div>
<div>
<?php
$query = "SELECT * FROM products";
$product = mysqli_query($con, $query);
if (!empty($product)) {
while ($row = mysqli_fetch_array($product))
```

```
<?>
<form action="index.php?action=add"
      pid=<?>=$row['id']?>" method="post">
```

```
<div style="width: 18rem">
  <img alt="<?>=880w ['image']?>" width="150">
    alt="<?>=$row['name']?>"
```

```
div>
  (span>? = $row['name']); ?> /span)
```

```
  (span>?= number_format($row
  ['price'], 2)); ?> /span>
  /div)
```

```
div>
```

```
  input type="text" name="quantity"
  value="1" size="2">
```

```
  input type="submit" value="Add to
  cart" />
```

```
/div>
```

```
/div>
```

```
/form>
```

```
{?php}
```

```
? else {
```

```
echo "no product available";
```

```
?>
```

```
/div>
```

```
/div>
```

```
/div>
```

```
((div>
```

```
/body>
```

```
/html>
```

Cart

Name	Code	Quantity	Item Price	Price	Remove
Monitor Stand Monitor	122	1	₹ 5,000,00	₹ 5,000,00	<input checked="" type="checkbox"/>
Total :				₹ 5,000,00	

Product List

Acer Nitro Laptop
Acer Nitro Laptop
₹ 89,000,00
<input type="checkbox"/> <input type="button" value="Add to Cart"/>

Monitor stand
Monitor Stand
₹ 5,000,00
<input type="checkbox"/> <input type="button" value="Add to Cart"/>

{ Add To Cart }

Aim :- Write program to demonstrate
XML parsing
with PHP

Theory :-

- 1) PHP script start with `<?php` and end with `?>`
- 2) SimpleXML turns an XML document into a data structure you can iterate through like a collection of arrays and objects
- 3) `simplexml_load_string()` function is used to read XML data from a string

Code :-

`<?php`

1) my XML Data =

`<?xml version='1.0' encoding='UTF-8'?>`

`<note>`

`<to> IT and CS </to>`

`<From> MENTOR </From>`

`<heading> Reminder </heading>`

</body> ALL THE BEST </body>
</note> ");

\$xml = simplexml_load_string ("My
XML Data") or die ("Error: cannot
create object");
print_r (\$xml);

?>

Output :-

Simple XML Element Object [`I46`] =>

IT and CS the best days are

`[From]` => MENTOR [Heading] => Reminder
`[body]` => ALL THE BEST

10 Demonstrate json with php

Aim :- demonstrate json with php

Theory

JSON :- Javascript Object Notation

JSON is a syntax for storing and exchanging data.

JSON is text, written with Javascript object notation.

Code :-

```
?> <!DOCTYPE html>
<html>
<body>
```

```
<p id="demo"></p>
```

```
<script>
```

```
var myObj = x;
myObj = { name: "John", age: 30,
          city: "New York" };
x = myObj["name"];
```

document. getElementById ("dem"). inner
HTML = x;
• script
• body
• html

Output :-

John

Name :- Komal Menariya
Roll no :- 42
Class :- FY BSC IT

1 XML

a. Design a DTD corresponding XML document and display it in browser using CSS.

Aim :- Creating XML document
Code :-

XML file

Creating Books.xml as :-

```
<?xml version = "1.0" encoding = "UTF-8"?>
<?xml-stylesheet type = "text/css" href = "Rule.css" ?>
<books>
    <heading> Welcome to Bookshop </heading>
    <book>
        <title> Title - : Web Programming </title>
        <author> Author - : Chris Bates </author>
        <publisher> Publisher - : Wiley </publisher>
        <edition> Edition - : 3 </edition>
        <price> Price - : 300 </price>
    </book>
    <book>
        <title> Title - : Computer Networks </title>
        <author> Author - : Forouzan </author>
        <publisher> Publisher - : Mc Graw Hill </publisher>
        <edition> Edition - : 5 </edition>
        <price> Price - : 700 </price>
    </book>
</books>
```

Teacher's Signature :

CSS FILE :-

Creating Rule.css as :-

books {

color : white;
background-color : gray;
width : 100%;

}

heading { color : green;

font-size : 40px;

background-color : powderblue;

}

heading, title, author, publisher, edition,

price {

display : block;

}

title {

font-size : 25px;

font-weight : bold;

}

Output :-

Welcome to Bookshop

Title :- Web Programming

Author :- Chrisbates

Publisher :- Wiley

Edition :- 3

Price :- 300

Title :- Computer Networks

Author :- Forouzan

Publisher :- Mc Graw Hall

Edition :- 5

Price :- 700

Computer networks :- 916T (soft)

b] Design an XML document and display it in browser using XSL.

Aim :- Design an XML and XSL document
Code :-

(Creating Students.xml as :-

```
<?xml version = "1.0" encoding = "UTF-8"?>
<?xml-stylesheet type = 'text/xsl' href =
    Rule.xsl ?>
```

<student>

<s>

```
<name> Divyank Singh Sikarwar </name>
<branch> CSE </branch>
<age> 18 </age>
<city> Agra </city>
```

</s>

<s>

```
<name> Ankit Chayhan </name>
<branch> CSE </branch>
<age> 20 </age>
<city> Shahjahanpur </city>
```

</s>

<s>

```
<name> Simran Agarwal </name>
<branch> CSE </branch>
<age> 23 </age>
<city> Buland Shar </city>
```

Teacher's Signature : _____

</s>

<s>

<name> Abhay Chauhan </name>

<branch> CSE </branch>

<age> 17 </age>

<city> Shahjahanpur </city>

</s>

<s>

<name> Himanchal Bhatia </name>

<branch> IT </branch>

<age> 25 </age>

<city> Indore </city>

</s>

</student>

- XSLT Code:

(Creating Rule.xsl as :-

```
<?xml version = "1.0" encoding = "UTF-8"?>
<xsl:stylesheet version = "1.0"
    xmlns:xsl = "http://www.w3.org/1999/XSL/Format">
<xsl:template match = "/">
<html>
<body>
<h1 align = "center"> Student's Basic Details </h1>
<table border = "3" align = "center">
<tr>
```

</s>

</>

<name> Abhay Chauhan </name>

<branch> CSE </branch>

<age> 17 </age>

<city> Shahjahanpur </city>

</c>

</s>

<name> Himanshu Bhatia </name>

<branch> IT </branch>

<age> 25 </age>

<city> Indore </city>

</s>

</student>

- XSLT Code:

(Creating Rule.xsl as :-

<?xml version = "1.0" encoding = "UTF-8"?>

<xsl:stylesheet version = "1.0">

xmlns:xsl = "http://www.w3.org/1999/XSL/

<xsl:template match = "/">

<html>

<body>

<h1 align = "center"> Student's Basic Details </h1>

<table border = "3" align = "center">

<tr>

```
<th> Name </th>
<th> Branch </th>
<th> Age </th>
<th> City </th>
</tr>
<xsl:for-each select = "student/s">
<tr>
<td><xsl:value-of select = "name"/></td>
<td><xsl:value-of select = "branch"/></td>
<td><xsl:value-of select = "age"/></td>
<td><xsl:value-of select = "city"/></td>
</tr>
</xsl:for-each>
</table>
</body>
</html>
</xsl:template>
</xsl:stylesheet>
```

Output :-

Students' Basic Details

Name	Branch	Age	City
Divya Singh Sikarwar	CSE	18	Agra
Aniket Chauhan	CSE	20	Shahjahanpur
Simran Agarwal	CSE	23	Buland Shar
Abhay Chauhan	CSE	17	Shahjahanpur
Himanchal Bhatia	IT	25	Indore

c) Design XML Schema and corresponding XML document.

Code :-

Theory :-

- 1) A XML Schema is used to define the structure of an XML document.
- 2) $\langle xs:element name = "employee" \rangle$:- It defines the element name employee.
- 3) $\langle xs:complexType \rangle$:- It defines that the element 'employee' is complex Type.
- 4) $\langle xs:sequence \rangle$:- It defines that the complex type is a sequence of elements.
- 5) $\langle xs:element name = "Firstname" type = "xs:string" \rangle$:- It defines that 'Firstname' is of string / text type.
- 6) $\langle xs:element name = "lastname" type = "xs:string" \rangle$:- It defines that the element 'lastname' is of string / text type.
- 7) $\langle xs:element name = "email" type = "xs:string" \rangle$:- It defines that the element 'email' is of string / text type.

Teacher's Signature :

Code :-

employee.xsd

```

<?xml version = "1.0"?>
<xsd:schema xmlns:xsd = "http://www.w3.org/2001/XMLSchema">

<xsd:element name = "employee">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name = "Firstname" type =
        "xsd:string"/>
      <xsd:element name = "lastname" type = "xsd:
        string"/>
      <xsd:element name = "email" type = "xsd:
        string"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
</xsd:schema>
```

employee.xml

```

<?xml version = "1.0"?>
<employee xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance" xsi:type =
  "employee">
```

Expt. No _____

Page No. _____ Date _____

Schemalocation = "employee.xsd"

<Firstname> Komal </Firstname>

<Lastname> Menaria </Lastname>

<email> komalmenaria.cpm2003@gmail.com

</email>

</employee>

Output :-

Komal

Menariq

Komalmenariqpm2003@gmail.com

2 PHP Basics - IT

Aim :- Write a PHP program to accept a number from the user and print it Factorial

Theory :-

- 1) Form method :- The method attribute specifies how to send form-data (the form-data is sent to the page specified in the action attribute).
- 2) POST :- sends the form-data as an http post transaction
- 3) <input> :- input tag specifies an input field where the user can enter data
- 4) <input type = "text"> :- Defines a single line text field
- 5) <input type = "Submit"> :- Defines a submit button
- 6) A PHP script starts with <?php and end with ?>
- 7) echo - used to display the output
- 8) for - used to traverse set of code for the specified number of time
- 9) \$var is a normal variable with name var that stores any value like string, integer

Output :-

Enter a number :

Factorial

Factorial of 8 is 40320

2 The PHP Basics - II
(b)

Aim(b) :- write a PHP program to accept a number from the user and print whether it is prime or not

Theory :-

- 1] A PHP script starts with `?php` and ends with `?`
- 2] echo :- is used to display the output
- 3] \$var is a normal variable with name var that stores any value like string, integer, float etc.
- 4] for :- used to traverse set of code for the specified number of times.
- 5] If :- If is used to execute the block of code exist inside the if statement only if the specified condition is true.

Code.

Input. Html

```
<html>
<head>
<title> Prime Number </title>
</head>
<body>
```

```
<form method = "post" action = "checkprime.php">
Enter a number <input type = "text" name
= "n1" > <br>
<input type = "Submit" value = "check prime">
</form>
</body>
</html>
```

checkprime.php

```
<?php
```

```
$n1 = (int) $_POST ("n1");
```

```
$flag = 0;
```

```
for ($i = 2; $i <= $n1/2, +i++)
```

```
{ if ($n1 % $i == 0)
```

```
    $
```

```
    $flag = 1;
```

```
    break;
```

```
} }
```

```
if ($flag == 0)
```

```
echo "Number is prime";
```

~~else~~

```
echo "Number is not prime";
```

?>

(D) S (iteration)

Date _____
Page No. _____

Output :-

Entering a new number :- 18 ~~more~~

checkprime.

The Number is not prime.

3 PHP Basic - II

Aim (a) :- write a php code to find the greater of 2 numbers accept the no from the user

Theory :-

- 1) A PHP script starts with `<?php` and ends with `?>`
- 2) echo :- used to display the output
- 3) if... elseif... else statement executes different codes for more than two conditions.

Code :-

Input.htm

`<html>`

`<head>`

`<title> Greater of two no. </title>`

`</head>`

`<body>`

`<form method="post" action="check.php">`
1st number : `<input type="text" name="n1">` `
`

2nd number : `<input type="text" name="n2">` `
`

```
<input type="submit" value="check">
</form>
</body>
</html>
```

check.php

<?php

```
$n1 = (int) $_POST['n1'];
```

```
$n2 = (int) $_POST['n2'];
```

```
if ($1 > $2)
```

```
echo $n1. " is greater than ". $n2 ;
```

```
else if ($n2 > $n1)
```

```
echo $n2. " is greater than ". $n1 ;
```

```
else
```

```
echo "Both the no.s are equal";
```

```
?>
```

if both of above code is struck
and an exit from memory is

last line adds three digits

Output :- result of two

numbers

1st number :- 8

2nd number :- 3

Submit

B is greater than 3.

left is on left to return

right side is either "top" is bottom
"down" or not. right hand is median

means "left" > "right" is median

crds <

3 PHP Basics - II

Aim (b) :- Write a PHP program to display the following binary pyramid :-

```
1  
0 1  
1 0 1  
0 1 0 1
```

Theory :-

- 1] POST = sends the form data as an http post transaction
- 2] echo = used to display the output
- 3] for = used to traverse set of code for the specified number of times.
- 4] If-else :- It executes one block of code if the specified condition is true and another block of code if the condition is false.

Code :-

```
?php  
for ($i=0 ; $i<4 ; $i++)  
{  
    for ($j=0 ; $j<=$i ; $j++)  
    {
```

Expt. No.....

Page No..... Date

if ((s[i]+s[j]) % 2 == 0)

 echo "#";

else

 echo "0";

}

 echo "nrbr";

}

?>

Teacher's Signature : _____

Output :-

1

0

0

1

0

1

1

0

1

Explanation :-
extra add. + right of bus = 010
and also for two's complement of bus \rightarrow no
need to add 1 to address of bus
as stated in starting it is also -H if
we want negative address at 8 bit
at 8 bit we have to add either to
left or to right

C++ is C MSB is 0 = id

C++ is C MSB is 0 = id

4 String Functions and arrays

Aim (a) :- write a PHP program to demonstrate different string functions

Theory :-

A php script starts with <?php and ends with ?>

- 1) strlen() :- Returns the length of a string
- 2) strchr () :- finds the first occurrence of a string inside another string
- 3) str_replace () :- Replace some character in a string
- 4) str_word_count :- Count the number of words in the string
- 5) Strpos() :- Returns the position of the first occurrence of a string inside another string
- 6) strtolower() :- convert the string into lower letters
- 7) strtoupper () :- convert the string into upper letters.
- 8) substr_count :- Count the number of times, a substring occurs in a string

- i) **Substr() :-** Returns a part of a string
- ii) **strcmp() :-** Compare two strings (case-sensitive)
- iii) **strcasecmp() :-** Compare two strings (case-insensitive)

Code :-

```
<?php
echo strlen ("Komal");
echo "<br>";
echo strchar ("Hello world!", "world");
echo "<br>";
echo str_replace ("world", "Komal", "Hello
world");
echo "<br>";
echo str_word_count ("Hello Komal");
echo "<br>";
echo strpos ("I like to read books; I
like singing too!", "like");
echo "<br>";
echo substr_count ("Hello world. The world is
nice", "world");
echo "<br>";
echo substr ("Hello world", 5);
echo "<br>";
echo strtolower ("HELLO - KOMAL");
echo "<br>";
echo strtoupper ("hello komal");
echo "<br>";
```

Expt. No.....

Page No..... Date

```
echo strcmp ("Hello Komal" , "Are you fine?");  
echo "Lb";  
echo strcasecmp ("Hello world ?" , "Hello  
world !");
```

?>

Output

5

world!

Hello komal!

2

2

2

world

hello komal

HELLO KOMAL

0

I

* blood loss at 0.5H T) log rt2 odg
; ("sdl1" < "log priril 0.1H
; " <rd3" odg
; blood loss at 0.5H T) +wo -rt2 odg
; ("blood" < "0.1H
; " <rd3" odg
; (2, "shock off") +rt2 odg
; " <rd3" odg
; ("IABP" 0.13H) +wo +rt2 odg
; " <rd3" odg
; ("komal off") +wo +rt2 odg
; " <rd3" odg

4 String Functions and arrays

Aim (b) :- Write a PHP program to create one dimensional array

Theory :-

- 1) A php script starts with <? php and ends with ?>
- 2) count() :- return the length of an array
- 3) Indexed arrays :- The index can be assigned automatically (index always starts at 0)
 - a) To loop through and print all the values of an indexed array we can use for loop.
- 4) Associative arrays :- Used named keys that we assign to them.

Code :-

```
<? PHP  
$Fruits = array ("apple", "mango", "orange");  
echo "I like ". $Fruits[0] . ", ". $Fruits[1].  
", ". "and ". $Fruits[2] . ", ".  
echo "<br>" ;
```

Expt. No.....

Page No..... Date

echo count (\$Fruits);
echo count ("2br");

sort (\$Fruits);

\$Length = count(\$Fruits)

for (\$x=0; \$x < \$array.length; \$x++)

{

echo \$arr[\$x];

echo "2br";

\$age = array ("Komal"=>"20", "devangi"=>
"25");

echo "devangi is ". \$age['devangi']. "years old";

?)

Teacher's Signature :

Output of Fig 9.19 is stored in file

I like apple, mango and orange.

apple

mango

orange

Devangi is 20 years old

No Harry has a fruit goal of

one to another set

goal of one set of goals for

and Harry has a prime with

card at option two task