Roll No:- _____

Sonopant Dandekar Shikshan Mandali's

Sonopant Dandekar Arts,V.S.Apte Commerce,

M.H.Mehta Science college

DEPARTMENT OF COMPUTER SCIENCE

CERTIFICATE

Certified That Mr./Miss._____

of_____ has satisfactorily completed a course of necessary experiment in _____under my supervision in the FY.BSC Computer Science in the Year 2024 – 2025

Head of Department **Subject Teacher**

Date: / /2025

INDEX

SR NO.	Aim of Practical	Practical Date	Submission Date	Remarks
1.	Design a Web page that make use of :[Document Structure Tags , Text Formatting Tags, List Tags , Image and Image maps]			
2.	Design a Webpage that makes use of: [Table Tags , Form Tags , Embedded Multimedia Elements]			
3.	Design a Webpage that makes use of CSS properties to change the background of a Page, to change fonts and Text Styles and for positioning an element			
4.	Write JavaScript code for performing various mathematical operations.			
5.	Write JavaScript code for (Demonstrating different objects such as String, Math, Date, Demonstrating Browser objects.			
6.	Create an XML file with Internal / External DTD and display it using CSS & XSl.			
7.	Write PHP Scripts for retrieving Data From HTML form.			
8.	Write PHP Scripts for Working with Databases (Storing Records/Retrieving Records and Displaying Them.)			
9.	Dynamically fetch and Display server data using AJAX in a Wamp Environment Without Page Reloads.			
10.	Load and Display Server Data dynamically on hover using AJAX and jQuery in a Wamp.			

Practical No. 01

Aim: Design a Web page that make use of : a) Document Structure Tags
b) Various Text Formatting Tags
c) List Tags
d) Image and Image maps

---

a) Design a web page having Document Structured Tag.

```
<!DOCTYPE html>
<html>
 <head>
   <title> Document Structured Tag </title>
 </head>
<body>
  <h1>Heading Tag</h1>
  <p>Paragraph Tag</p>
</body>
</html>
```

OUTPUT:

b) Design a web page having text formatting tag.

```
<!DOCTYPE html>
<html>
<head>
<title> Text formatting Tag</title>
</head>
<body>
<p>This is normal text.</p>
<p><b>This is bold text.</b></p>
<p><strong>This is strong text with logical importance.</strong></p>
```

```
<p><i>This is italics text.</i></p>

<p><em>This is emphasized text.</em></p>

<p><mark>This is highlighted text.</mark></p>

<p>This is<sup>superscript</sup>text.</p>

<p>This is<sub> subscript</sub>text.</p>

<p>Small Font:<small> This is an example to show Smaller Text.</small></p>

<p>Big Font:<big> This is an example to show Larger Text.</big></p>

<p>Check out this example,<del>this text will be deleted.</del></p>

<p> Check out this example,<ins> to insert a new paragraph.</ins></p>

</body>

</html>
```

OUTPUT:

This is normal text.

This is bold text.

This is strong text with logical importance.

This is italics text.

This is emphasized text.

This is highlighted text.

This is superscript text.

This is subscript text.

Small Font: This is an example to show Smaller Text.

Big Font: This is an example to show Larger Text.

Check out this example, ~~this text will be deleted.~~

Check out this example, to insert a new paragraph.

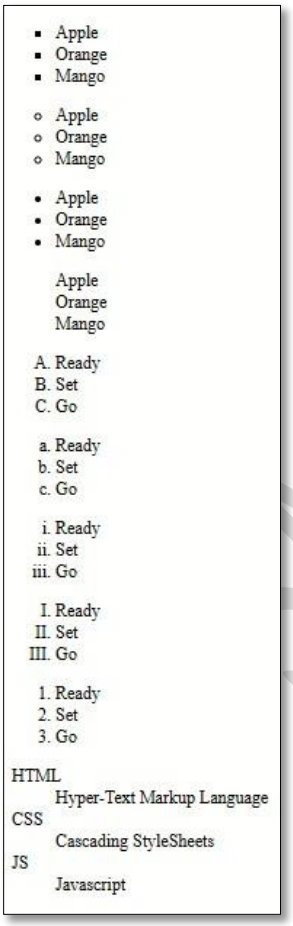
c) Design a web page having all list tag.

```
<!DOCTYPE html>
<html>
 <head>
   <title> List  Tag </title>
 </head>
<body>
 <ul style=" list-style-type:square;">
  <li>Apple</li>
  <li>Orange</li>
  <li>Mango</li>
</ul>

 <ul style=" list-style-type:circle;">
  <li>Apple</li>
  <li>Orange</li>
  <li>Mango</li>
</ul>

 <ul style=" list-style-type:disc;">
  <li>Apple</li>
  <li>Orange</li>
  <li>Mango</li>
</ul>

 <ul style=" list-style-type:none;">
  <li>Apple</li>
  <li>Orange</li>
  <li>Mango</li>
</ul>
```

```
<ol type="A">
 <li>Ready</li>
 <li>Set</li>
<li>Go</li>
</ol>

<ol type="a">
 <li>Ready</li>
 <li>Set</li>
<li>Go</li>
</ol>

<ol type="i">
 <li>Ready</li>
 <li>Set</li>
<li>Go</li>
</ol>

<ol type="I">
 <li>Ready</li>
 <li>Set</li>
<li>Go</li>
</ol>

<ol type="1">
 <li>Ready</li>
 <li>Set</li>
<li>Go</li>
</ol>
```

```
<dl>
 <dt>HTML</dt>
 <dd>Hyper-Text Markup Language</dd>
 <dt>CSS</dt>
 <dd>Cascading StyleSheets</dd>
 <dt>JS</dt>
 <dd>Javascript</dd>
</dl>
</body>
</html>
```

OUTPUT:



d) Design a web page having image and image map.

File Name: Image map.html

```
<!doctype html>
<html>
<title> Image Map </title>
</head>
<body>
<img src="workplace.jpg" alt="Workplace" usemap="#workmap">
<map name="workmap">
<area shape="rect" coords="77,20,193,167" alt="Laptop" href="laptop.html">
<area shape="circle" coords="231,139,50" alt="Coffee" href="coffee.html">
</map>
</body>
</html>
```

File Name: Coffee.html

```
<!doctype html>
<html>
<title> Coffee </title>
</head>
<body>
<br>
<p><b>Coffee</b>contains antioxidants and other active substances that may reduce internal
inflammation and protect against disease </p>
<img src="Cofee1.jpg" alt="Coffee" height="300px" width="300">
</body>
</html>
```

File Name: Laptop.html

```
<!doctype html>
<html>
<title> laptop </title>
</head>
```

```
<body bgcolor="Grey">

<center>

<font color=" White">

<font size="25">

<p>A <i><b><big><u>laptop</u><b></i></big> computer or notebook computer, also known as a laptop or notebook, is a small, portable personal computer (PC)</p>

<img src="laptop.jpg" alt="Laptop" height="300px" width="500">

</center>

</body>

</html>
```

OUTPUT:

A *laptop* computer or notebook computer, also known as a laptop or notebook, is a small, portable personal computer (PC)

Coffeecontains antioxidants and other active substances that may reduce internal inflammation and protect against disease

Practical No. 2

Aim: Design a Webpage that makes use of:

1. Table Tags
2. Form Tags
3. Embedded Multimedia Elements

---

a)Table Tag :

```
<!DOCTYPE html>
<html>
<head>
<title>HTML Table Example</title>
  <style>
     table {
 border-collapse: collapse;
width: 50%;
}       th, td
{
          border: 1px solid black;
padding: 8px;
text-align: left;
      }
th {
          background-color: #f2f2f2;
      }
   </style>
</head>
<body>
   <h1>HTML Table Example</h1>
   <table>
      <caption>Student Grades</caption>
      <thead>
         <tr>
```

```
            <th>Name</th>
            <th>Subject</th>
            <th>Grade</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td>John Doe</td>
            <td>Math</td>
            <td>A</td>
        </tr>
        <tr>
            <td>Jane Smith</td>
            <td>English</td>
            <td>B+</td>
        </tr>
    </tbody>
    <tfoot>
        <tr>
            <td colspan="3">End of Table</td>
        </tr>
    </tfoot>
  </table>
</body>
</html>
```

OUTPUT:

HTML Table Example

Student Grades

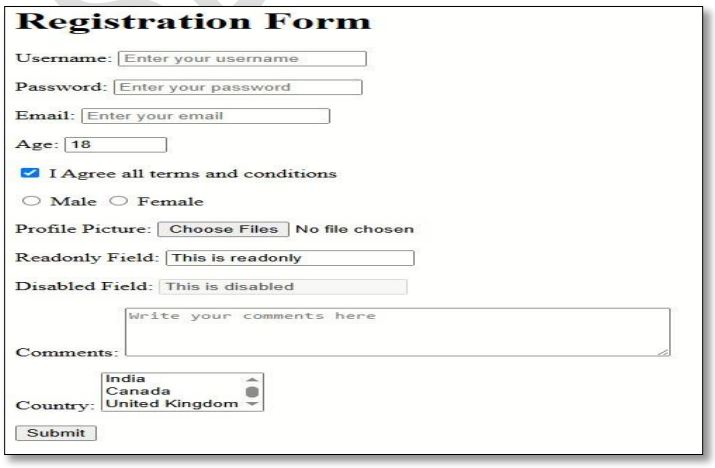
Name	Subject	Grade
John Doe	Math	A
Jane Smith	English	B+
End of Table		

b) Form Tag

```
<!DOCTYPE html>
<html>
<head>
<title> Registration Form </title>
</head>
<body>
<h1> Registration Form </h1>
<form action="/submit-form" method="POST"  enctype="multipart/form-data"
target="_blank" autocomplete="on">
<!-- Text Input -->
<label for="username">Username:</label> <input
type="text" id="username" name="username"
placeholder="Enter your username" maxlength="20"
minlength="3" required > <br/><br/>
<!-- Password Input -->
<label for="password">Password:</label>
<input type="password" id="password" name="password"  placeholder="Enter
your password" required>
<br/><br/>
<!-- Email Input -->
<label for="email">Email:</label>
<input type="email" id="email" name="email"  placeholder="Enter
your email" required>
<br/><br/>
<!-- Number Input -->
<label for="age">Age:</label>
<input type="number" id="age" name="age" min="1" max="100"  step="1"
value="18" maxvalue="100" required>
<br/><br/>
<!-- Checkbox -->
<label>
<input type="checkbox" name="terms" checked>
I Agree all terms and conditions
</label>
<br/><br/>

<!-- Radio Buttons -->
<label>
<input type="radio" name="gender" value="male" required> Male</label> <label>
<input type="radio" name="gender" value="female"> Female</label>
<br/><br/>

<!-- File Input -->
```

```
<label for="profile-picture">Profile Picture:</label> <input
type="file" id="profile-picture" name="profile-picture"
multiple> <br/><br/>
<!-- Readonly Input -->
<label for="readonly">Readonly Field:</label>
<input type="text" id="readonly" value="This is readonly" readonly>
<br/><br/>

<!-- Disabled Input -->
<label for="disabled">Disabled Field:</label>
<input type="text" id="disabled" value="This is disabled" disabled>
<br/><br/>
<!-- Textarea -->
<label for="comments">Comments:</label>
<textarea id="comments" name="comments" rows="4" cols="50"  placeholder="Write
your comments here"></textarea>
<br/><br/>
<!-- Select Dropdown -->
<label for="country">Country:</label>
<select id="country" name="country" size="3" multiple>
<option value="India">India</option>
<option value="canada">Canada</option>
<option value="uk">United Kingdom</option>
<option value="australia">Australia</option>
</select>
<br/><br/>
<!-- Submit Button -->
<button type="submit">Submit</button>
</form>
</body>
</html>
```

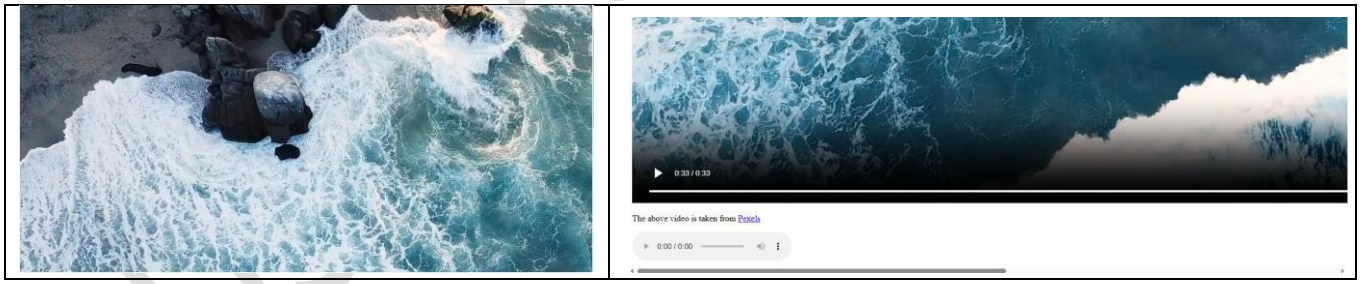
OUTPUT:



c) Embedded Multimedia Element:

```
<!DOCTYPE html>

<html>

<head>

<title> Video Element </title>

</head>

<body>

<video controls>

<source src="nature.mp4" type="video/mp4">  Nature </video>

<p>The above video is taken from <a href="https://www.pexels.com/video/drone-view-
ofbig-waves-rushing-to-the-shore-3571264/">

Pexels</a></p>

<audio controls>

<source src="9PBX76J-birds.mp3" type="audio/mp3">

</body>

</html>
```

OUTPUT:



Practical No. 3

Aim: Design a Webpage that makes use of CSS properties to change the background of a Page, to change fonts and Text Styles and for positioning an element

---

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Stylish Webpage</title>
<style>
          body {
         background-image: url('background.jpg');
         background-size: cover;
        background position: center;
        margin: 0;
        height: 100vh;
       display: flex;
       justify-content: center;
       align-items: center;
      font-family: Arial, sans-serif;
     }
     /* 2. Changing fonts and text styles */
   h1 {
        font-family: 'Arial', sans-serif;
       color: #ff6347;
       font-size: 3rem;
      text-align: center;
      text-transform: uppercase;
       text-shadow: 2px 4px rgba(0, 0, 0, 0.2);
     }
p {
       font-family: 'Verdana', sans-serif;
      font-size: 1.2rem;
      color: #333;
      text-align: center;
      line-height: 1.6;
      margin: 20px;
     }
     /* 3. Positioning an element */
     .box {
        position: absolute;
       top: 50%;
       left: 50%;
       transform: translate(-50%, -50%); /* Center the box both horizontally and vertically */
```

```
        background-color: rgba(255, 255, 255, 0.7); /* Semi-transparent white background */
        padding: 30px;
        border-radius: 10px; /* Rounded corners */
        box-shadow: 0px 8px 15px rgba(0, 0, 0, 0.1); /* Soft shadow */
        width: 80%; /* Width of the box */
        max-width: 600px; /* Maximum width of the box */
    }
</style>
</head>
<body>
<div class="box">
<h1>Welcome to My Stylish Webpage!</h1>
<p>This page demonstrates CSS for changing backgrounds, fonts, and positioning
elements.</p>
</div>
</body>
</html>
```

Practical No. 4

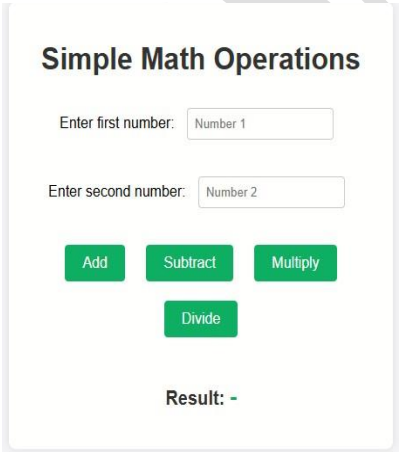
Aim Write JavaScript code for performing various mathematical operations

---

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Simple Math Operations</title>
<style>
      body {
         font-family: Arial, sans-serif;
          background-color: #f4f4f9;
            text-align: center;
             padding: 20px;
      }
h1 {
         color: #333;
      }
 input {
 padding: 8px;
 margin: 10px;
border: 1px solid #ccc;
border-radius: 4px;
 width: 150px;
      }
      button {
padding: 10px 20px;
 margin: 10px;
border: none;
 border-radius: 4px;
background-color: #4CAF50;
 color: white;
cursor: pointer;
font-size: 16px;
      }
      button:hover {
         background-color: #45a049;
      }
h3 {
color: #333;
font-size: 20px;
      }
      #result {
```

```
font-size: 24px;
font-weight: bold;
color: #4CAF50;
      }

  .container {
max-width: 400px;
margin: 0 auto;
padding: 20px;
background-color: white;
border-radius: 8px;
boxshadow:0 0 10px rgba(0,0,0,0.1);
      }
   </style>
</head>
<body>
<div class="container">
<h1>Simple Math Operations</h1>

<label for="num1">Enter first number:</label>
<input type="number" id="num1" placeholder="Number 1">
<br><br>

<label for="num2">Enter second number:</label>
<input type="number" id="num2" placeholder="Number 2">
<br><br>

<button onclick="add()">Add</button>
<button onclick="subtract()">Subtract</button>
<button onclick="multiply()">Multiply</button>
<button onclick="divide()">Divide</button>
<br><br>

<h3>Result: <span id="result">-</span></h3>
</div>
<script>
      // Addition
function add() {
         var num1 = parseFloat(document.getElementById("num1").value);
         var num2 = parseFloat(document.getElementById("num2").value);
         var result = num1 + num2;
         document.getElementById("result").textContent = result;
      }
```

```
      // Subtraction
function subtract() {
        var num1 = parseFloat(document.getElementById("num1").value);
        var num2 = parseFloat(document.getElementById("num2").value);
        var result = num1 - num2;
        document.getElementById("result").textContent = result;
      }

      // Multiplication
function multiply() {
        var num1 = parseFloat(document.getElementById("num1").value);
        var num2 = parseFloat(document.getElementById("num2").value);
        var result = num1 * num2;
        document.getElementById("result").textContent = result;
      }
      // Division
function divide() {
        var num1 = parseFloat(document.getElementById("num1").value);
        var num2 = parseFloat(document.getElementById("num2").value);
        if (num2 !== 0) {
        var result = num1 / num2;
       document.getElementById("result").textContent = result;
        } else {
        document.getElementById("result").textContent = "Cannot divide by zero";
        }
   }
</script>
</body>
</html>
```



Practical No. 5

Aim: Write JavaScript code for

a. Demonstrating different objects such as String, Math, Date etc.

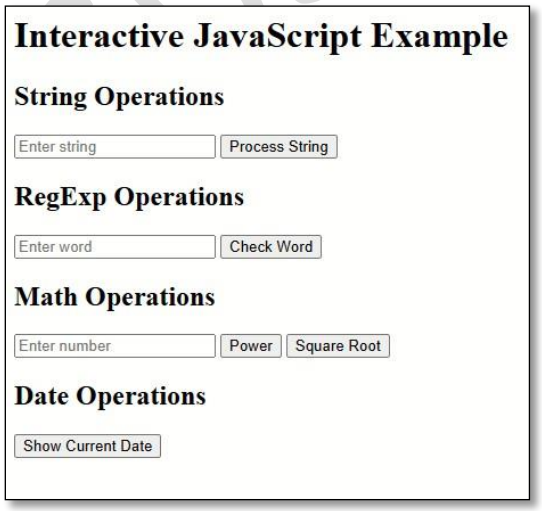
b. Demonstrating Browser objects.

---

a) Demonstrating different objects such as String. Math. Date etc.

```
<!DOCTYPE html>
<head>
<title>Interactive JavaScript Example</title>
</head>
<body>
  <h1>Interactive JavaScript Example</h1>
<h2>String Operations</h2>
 <input type="text" id="inputString" placeholder="Enter string">
 <button onclick="processString()">Process String</button>
  <p id="stringExample"></p>
 <h2>RegExp Operations</h2>
 <input type="text" id="inputRegExp" placeholder="Enter word">
 <button onclick="processRegExp()">Check Word</button>
 <p id="regexpExample"></p>

 <h2>Math Operations</h2>
 <input type="text" id="inputNumber" placeholder="Enter number">
  <button onclick="performMathOperation('power')">Power</button>
  <button onclick="performMathOperation('sqrt')">Square Root</button>
  <p id="mathExample"></p>

 <h2>Date Operations</h2>
 <button onclick="showCurrentDate()">Show Current Date</button>
 <p id="dateExample"></p>
    <script>
      // String Operation: Convert input string to uppercase
function processString()
{
 var inputStr = document.getElementById("inputString").value;
document.getElementById("stringExample").innerText = inputStr.toUpperCase();
}
      // RegExp Operation: Check if input string matches a word (e.g., "JavaScript")
function processRegExp() {
          var inputWord = document.getElementById("inputRegExp").value;
           var result = "JavaScript".includes(inputWord);
          document.getElementById("regexpExample").innerText =  "Does the string contain '"
+ inputWord + "'? " + result;
```

```
      }
       // Math Operations: Power and Square Root
function performMathOperation(operation) {
         var number = document.getElementById('inputNumber').value;

 // Check if the input is a valid number
 if (isNaN(number) || number === "")
{
 alert("Please enter a valid number!");
 return;
}
number = parseFloat(number); // Convert to number
if (operation === 'power')
{
 document.getElementById('mathExample').innerText =
"Power of " + number + ": " + Math.pow(number, 2);
 }
 else if (operation === 'sqrt')
{
 document.getElementById('mathExample').innerText =
"Square Root of " + number + ": " + Math.sqrt(number);
}
}
// Date Operation: Show current date and time
function showCurrentDate()
{
 document.getElementById("dateExample").innerText =
"Current Date and Time: " + new Date();
 }
   </script>
</body>
</html>
```



b) Demonstrating browser objects.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Browser Objects in JavaScript</title>
</head>
<body>
<h1>JavaScript Browser Objects</h1>
<button onclick="showWindowInfo()">Show Window Info</button>
<button onclick="showNavigatorInfo()">Show Navigator Info</button>
<button onclick="showScreenInfo()">Show Screen Info</button>
<button onclick="showLocationInfo()">Show Location Info</button>
<button onclick="goBack()">Go Back</button>

<div id="output" style="margin-top:20px; white-space:pre;"></div>
<script>
function showWindowInfo() {
let info = `Inner Width: ${window.innerWidth}\n` +
           `Inner Height: ${window.innerHeight}\n` +
           `Outer Width: ${window.outerWidth}\n` +
            `Outer Height: ${window.outerHeight}`;
 displayOutput(info);
}
     function showNavigatorInfo() {
 let info =  `User Agent: ${navigator.userAgent}\n`+
               `Platform: ${navigator.platform}\n` +
               `Language: ${navigator.language}\n` +
                `Online: ${navigator.onLine }`;
displayOutput(info);
}
function showScreenInfo() {
let info = `Screen Width: ${screen.width}\n` +
               `Screen Height: ${screen.height}\n` +
               `Available Width: ${screen.availWidth}\n` +
               `Available Height: ${screen.availHeight}\n` +
               `Color Depth: ${screen.colorDepth}\n` +
               `Pixel Depth: ${screen.pixelDepth}`;
                displayOutput(info);
     }
     function showLocationInfo() {
 let info = `URL: ${location.href}\n` +
            `Protocol: ${location.protocol}\n` +
            `Host: ${location.host}\n` +
            Pathname: ${location.pathname}\n` +
```

```
                `Search: ${ location.search }`;
displayOutput(info);
      }

      function goBack() {
if (history.length > 1) {
history.back();
         } else {
            alert("No previous page in history!");
         }
      }
      function displayOutput(text) {
         document.getElementById('output').innerText = text;
      }
   </script>
</body>
</html>
```

Practical No. 6

Aim: Create an XML file with Internal / External DTD and display it using CSS & XSL

File Name: Productdata.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="t1.xsl"?>
<productdata>
<product id="P001">
<productname>Woody</productname>
<description>This is a toy</description>
<price>240</price>
<quantity>12</quantity>
</product>

<product id="P002">
<productname>Buzz</productname>
<description>This is a toy</description>
<price>280</price>
<quantity>14</quantity>
</product>
</productdata>
```

File Name: t1.xsl

```
<?xml version="1.0" encoding="UTF-8"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
<xsl:template match="/">
<html>
 <head>
 <title>Product List</title>
 <style>
        table { border-collapse: collapse; width: 50%; }
        th, td { border: 1px solid black; padding: 8px; text-align: left; }
        th { background-color: #f2f2f2; }
</style>
</head>
<body>
 <h2>Product Data</h2>
<table>
<tr>
 <th>Product ID</th>
 <th>Name</th>
 <th>Description</th>
```

```
 <th>Price</th>
 <th>Quantity</th>
 </tr>
<xsl:for-each select="productdata/product">
 <tr>
 <td><xsl:value-of select="@id"/></td>
<td><xsl:value-of select="productname"/></td>
<td><xsl:value-of select="description"/></td>
<td><xsl:value-of select="price"/></td>
 <td><xsl:value-of select="quantity"/></td>
 </tr>
</xsl:for-each>
</table> </body>
</html>
</xsl:template>
</xsl:stylesheet>
```

Output:

Open command Prompt type : cd desktop

c:/user/student/desktop : python –m http.server 8000

In any browser type in address bar: http://localhost:8000/productdata.xml

Product Data

Product ID	Name	Description	Price	Quantity
P001	Woody	This is a toy	240	12
P002	Buzz	This is a toy	280	14

Practical No. 7

Aim: Write PHP Scripts for Retrieving Data from HTML form

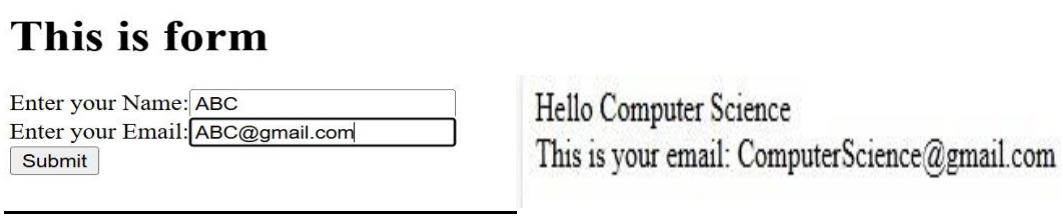
File name: Retrieving Data.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Document</title>
</head>
<body>
<h1>This is form</h1>
<form action="wel.php" method="post">
Enter your Name:<input type="text" name="t1"><br>
Enter your Email:<input type="text" name="email"><br>
<input type="submit">
</form>
</body>
</html>
```

File name: wel.php

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Document</title>
</head>
<body>
Hello  <?php
echo $_POST['t1'];
?>
<br>
This is your email:
<?php
echo $_POST['email'];
 ?>
</body>
</html>
```

Output:



Practical No. 8

Aim: Write PHP Scripts for Working with Databases (Storing Records / Retrieving Records and displaying them).

Create database in MySQL Console in Wampserver with query

1. ***Create database test_db;***
2. ***Create table users(id int Auto_increment primary key , name varchar(255) not null, email varchar(255) unique not null);***

```
mysql> create database test_db;
Query OK, 1 row affected (0.00 sec)

mysql> use test_db;
Database changed
mysql> create table users(id int Auto_increment primary key , name varchar(255) not null, email varchar(255) unique not null);
Query OK, 0 rows affected (0.00 sec)
```

File Name: Display.php

```
<?php
include 'db_connect.php';
$sql = "SELECT id, name, email FROM users";
$result = $conn->query($sql);
?>
<!DOCTYPE html>
<html>
<head>
  <title>Display Records</title>
</head>
<body>
  <h2>Users List</h2>
  <table border="1">
    <tr>
      <th>ID</th>
      <th>Name</th>
      <th>Email</th>
    </tr>
    <?php
if ($result->num_rows > 0)
{
while ($row = $result->fetch_assoc())
{
 echo "<tr><td>" . $row['id'] . "</td><td>" . $row['name'] . "</td><td>" . $row['email'] .
"</td></tr>";
}
```

```
} else
{
 echo "<tr><td colspan='3'>No records found</td></tr>";
       }
       $conn->close();
       ?>
   </table>
</body>
</html>
```

File Name: insert.php

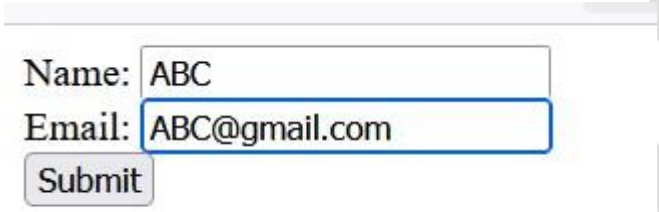
```
<?php
include 'db_connect.php';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
 $name = $_POST['name'];
 $email = $_POST['email'];
 $sql = "INSERT INTO users (name, email) VALUES ('$name', '$email')";

  if ($conn->query($sql) === TRUE) {
  echo "New record created successfully!";
   } else {
    echo "Error: " . $sql . "<br>" . $conn->error;
    }
}
$conn->close();
?>
<!DOCTYPE html>
<html>
<head>
<title>Insert Record</title>
</head>
<body>
 <form method="POST" action="">
 <label>Name:</label>
  <input type="text" name="name" required><br>
 <label>Email:</label>
  <input type="email" name="email" required><br>
  <input type="submit" value="Submit">
 </form>
</body>
</html>
```

File Name: db_connect.php

```
<?php
$host = "localhost";
// Change to your database host
$user = "root";
 // Change to your database username
$pass = "";
// Change to your database password
$dbname = "test_db";
 // Change to your database name $conn = new
mysqli($host, $user, $pass, $dbname); if
($conn>connect_error) {
  die("Connection failed: " . $conn->connect_error);
}
?>
```

Output:

Insert values:



Display values:

```
mysql> select*from users;
+----+--------+-----------------+
| id | name   | email           |
+----+--------+-----------------+
|  1 | ABC    | ABC@gmail.com   |
|  2 | fsedfv | dsrf@gmail.com  |
+----+--------+-----------------+
2 rows in set (0.00 sec)
```

Users List

ID	Name	Email
1	ABC	ABC@gmail.com
2	fsedfv	dsrf@gmail.com

Practical No. 9

Aim: Dynamically fetch and display server data using AJAX in a WAMP environment without page reloads.

File Name: index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>AJAX with WAMP</title>
</head>
<body>
<h2>AJAX Request Example</h2>
<button onclick="loadData()">Get Data</button>
<div id="result"></div>
<script>
function loadData()
{
var xhr = new XMLHttpRequest();
xhr.open("GET", "server.php", true);
xhr.onload = function ()
{
if (xhr.status == 200) {
document.getElementById("result").innerHTML = xhr.responseText;
}
else
{
document.getElementById("result").innerHTML = "Error loading data";
} };
xhr.send();
}
</script>
</body>
</html>
```

File Name: server.php

```
<?php
   header("Content-Type: text/plain");
echo "Hello from PHP Server!"; ?>
```

Output:

Practical No. 10

Aim: Load and display server data dynamically on hover using AJAX and jQuery in a WAMP environment.

File Name: Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>AJAX Hover</title>
<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
 <body>
<div id="box" style="width:200px; height:100px; background:lightblue; text-align:center;
padding:20px; cursor:pointer;">
   Hover Here
</div>
<div id="result"></div>
<script>
$("#box").hover(
   () => $("#result").load("data.php"),  // On hover, load data
   () => $("#result").empty()          // On mouseout, clear data
);
</script>
</body>
</html>
```

File Name: data.php

```
<?php
echo "Hello! AJAX Mouseover Triggered.";
?>
```

Output: