

**Name of student:** ARIF MUSTAFA PATEL (TE)

**Seat no:** T191074252

**Roll no:** 164

### Assignment-01

Sr.no	WebsiteURL	Purposeof website	Things liked	Things disliked	Overall evaluation
1.	Unipune.ac.in	Education alsiteforP uneuniversity	<ul style="list-style-type: none"><li>• Web pagealignment</li><li>• Colorcombination</li></ul>	<ul style="list-style-type: none"><li>• Fontsizeand style</li><li>• Hardtofindlinks</li><li>• Too muchcontentsand flashedtext</li></ul>	Good
2.	Amazon.in	E-commerce	<ul style="list-style-type: none"><li>• Attractiv ecolors</li><li>• Navigation panel</li></ul>	<ul style="list-style-type: none"><li>• UIcouldbebetter</li><li>• Blurimages</li></ul>	Better
3.	Adobe.com	Comput ersoftware	<ul style="list-style-type: none"><li>• Great use ofvariousimag es</li><li>• Simpleand productive design</li></ul>	<ul style="list-style-type: none"><li>• Colorusage</li></ul>	Better
4.	LinkedIn.in	Employment serviceprovider	<ul style="list-style-type: none"><li>• Easyto use</li><li>• Colorcoordinationandfo nts</li></ul>	<ul style="list-style-type: none"><li>• Not thatoptimize dasformobile</li></ul>	Better
5.	Cricbuzz.com	Crick etsco res	<ul style="list-style-type: none"><li>• Easy to useandman age</li><li>• Design</li></ul>	<ul style="list-style-type: none"><li>• Not that greatuseoffont andimages</li></ul>	Good

## Assignment-02

### HTML CODE:

```
<!DOCTYPE html>

<html>
<head>
    <title> Landing page design using html and css in hindi</title>
</head>
<body>
    <header>
        <nav>
            <div class="logo"><span>Pi</span>zzahat</div>
            <div class="menu">
                <a href="#">Home</a>
                <a href="#">Foods</a>
                <a href="#">Offer's</a>
                <a href="#">Your Order</a>
                <a href="#">Delivery</a>
                <a href="#">Cart</a>
            </div>
        </nav>
        <div class="h-text">
            <p>Free Home Delivery</p>
            <h1>Quality F<span>oo</span>ds </h1>
            <h2>Pizza of the Week</h2>
            <a href="#" id="order">Order now!!</a>
            <a href="#" id="dis">25% Off Hurry!!</a>
        </div>
    </header>
</body>
</html>
```

## CSS Code:

```
<style type="text/css">

* {
    padding: 0;
    margin: 0;
    box-sizing: border-box;
    font-family: sans-serif;
}

header {
    width: 100%;
    height: 100vh;
    background: url(images/pizza2.png);
    background-size: 50% 90%;
    background-repeat: no-repeat;
    background-position: bottom right;
    position: relative;
}

nav {
    width: 100%;
    height: 70px;
    padding: 0px 70px;
    background: white;
    color: black;
    display: flex;
    justify-content: space-between;
    align-items: center;
    box-shadow: 2px 0px 5px grey;
}
```

```
.logo {  
    font-weight: bold;  
    font-size: 2em;  
    letter-spacing: 2px;  
}  
  
Scanned with CamScanner
```

```
.logo span {  
    color: brown;  
}
```

```
.menu a {  
    text-decoration: none;  
    color: black;  
    padding: 10px 20px;  
}
```

```
.menu a:last-child {  
    background: brown;  
    color: white;  
    border-radius: 8px;  
}
```

```
.h-text {  
    position: absolute;  
    top: 30%;  
    left: 5%;  
    padding: 15px;  
}
```

```
.h-text p {  
    color: #262626;
```

```
letter-spacing: 3px;  
font-weight: bold;  
}
```

```
.h-text h1 {  
    font-size: 4em;  
    line-height: 1.5em;  
}
```

```
.h-text h1 span {  
    color: brown;  
}
```

```
.h-text h2 {  
    font-size: 4em;  
    margin-bottom: 0.5em;  
}
```

```
#order {  
    text-decoration: none;  
    color: white;  
    background: brown;  
    border-radius: 8px;  
    padding: 10px 20px;  
}
```

```
#dis {  
    text-decoration: none;  
    color: white;  
    background: #262626;  
    padding: 10px 20px;  
    border-radius: 8px;  
}
```

```
</style>
```

## OUTPUT

← → C ⓘ 127.0.0.1:5500/pizz.html

Pizzahat

Home Foods Offer's Your Order Delivery Cart

Free Home Delivery

# Quality Foods

## Pizza of the Week

Order now!! 25% Off Hurry!!

Scanned with CamScanner

```
<!--      Assignment - 3
Design the XML document to store the information of the employees of any business
organization and demonstrate the use of:
a) DTD
b) XML Schema
And display the content in (e.g., tabular format) by using CSS/XSL.
-->
```

employee.dtd

```
<!DOCTYPE Person[
<!ELEMENT Person (Employee+)
<!ELEMENT Employee (EmpId,EmpName,Department)
<!ELEMENT EmpId (#PCDATA)
<!ELEMENT EmpName (#PCDATA)
<!ELEMENT Department (#PCDATA)
]>
```

employee.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="employeeSchema.xsl" ?>
<Person>
    <Employee>
        <EmpId>119</EmpId>
        <EmpName>Harsh</EmpName>
        <Department>Backend</Department>
    </Employee>

    <Employee>
        <EmpId>145</EmpId>
        <EmpName>Mayuri</EmpName>
        <Department>Front End</Department>
    </Employee>

    <Employee>
        <EmpId>156</EmpId>
        <EmpName>Vaibhavi</EmpName>
        <Department>Front End</Department>
    </Employee>

    <Employee>
        <EmpId>152</EmpId>
        <EmpName>Yash</EmpName>
        <Department>Cyber Security</Department>
    </Employee>

    <Employee>
        <EmpId>181</EmpId>
```

```
<EmpName>Pratik</EmpName>
<Department>Full Stack</Department>
</Employee>
</Person>

employeeSchema.xsl

<?xml version="1.0" encoding="ISO-8859-1"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
    <xsl:template match="/">
        <html>
            <body>
                <center>
                    <h2>Employee Information</h2>
                    <table border="1">
                        <tr bgcolor="gray">
                            <th>EmpId</th>
                            <th>EmpName</th>
                            <th>Department</th>
                        </tr>
                        <xsl:for-each select="Person/Employee">
                            <tr>
                                <td bgcolor="white"><xsl:value-of select="EmpId"/></td>
                                <td bgcolor="white"><b><xsl:value-of
select="translate(EmpName, 'abcdefghijklmnopqrstuvwxyz', 'ABCDEFGHIJKLMNOPQRSTUVWXYZ')"/>
                                    </b></td>
                                <td bgcolor="cyan"><xsl:value-of
select="Department"/></td>
                            </tr>
                        </xsl:for-each>
                    </table>
                </center>
            </body>
        </html>
    </xsl:template>
</xsl:stylesheet>
```

EmpID	EmpName	Department
119	HARSII	Backend
145	MAYURI	Front End
151	PRATIK	Marketing
126	VAIBHAVI	Front End
122	YASH	Cyber Security

## Assignment-04

### HTML CODE:

```
<!DOCTYPE html>

<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<link rel="stylesheet" href="style.css">
<title>Calculator</title>
</head>

<body>
<div class="container">
<h1>Calculator by Arif</h1>
<div class="calculator">
<input type="text" name="screen" id="screen">
<table>
<tr>
<td><button>(</button></td>
<td><button>)</button></td>
<td><button>C</button></td>
<td><button>%</button></td>
</tr>
<tr>
<td><button>7</button></td>
<td><button>8</button></td>
```

```
<td><button>9</button></td>
<td><button>X</button></td>
</tr>

<tr>
    <td><button>4</button></td>
    <td><button>5</button></td>
    <td><button>6</button></td>
    <td><button>-</button></td>
</tr>

<tr>
    <td><button>1</button></td>
    <td><button>2</button></td>
    <td><button>3</button></td>
    <td><button>+</button></td>
</tr>

<tr>
    <td><button>0</button></td>
    <td><button>. </button></td>
    <td><button>/</button></td>
    <td><button>=</button></td>
</tr>
</table>
</div>
</div>

</body>
<script src="index.js"></script>
</html>
```

## CSS Code:

```
.container{  
    text-align: center;  
    margin-top: 23px  
}  
  
table{  
    margin: auto;  
}  
  
input{  
    border-radius: 21px;  
    border: 5px solid #244624;  
    font-size: 34px;  
    height: 65px;  
    width: 456px;  
}  
  
button{  
    border-radius: 20px;  
    font-size: 40px;  
    background: #1ecef8;  
    width: 102px;  
    height: 90px;  
    margin: 6px;  
}  
  
.calculator{  
    border: 4px solid #121312;  
    background-color: #28ffoc;  
    padding: 23px;  
    border-radius: 53px;
```

```
display: inline-block;  
}  
  
h1{  
font-size: 28px;  
font-family: 'Courier New', Courier, monospace;  
}
```

---

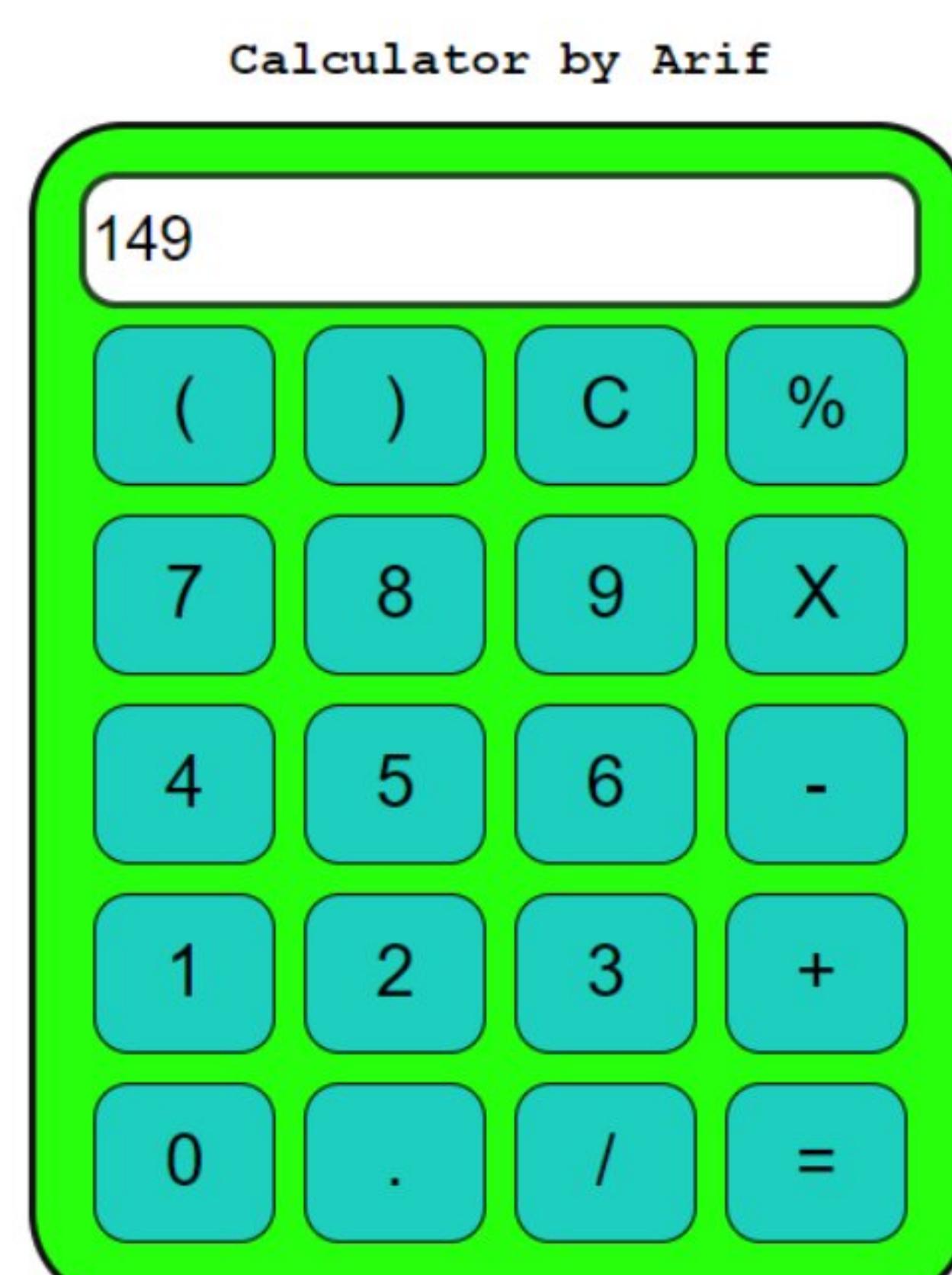
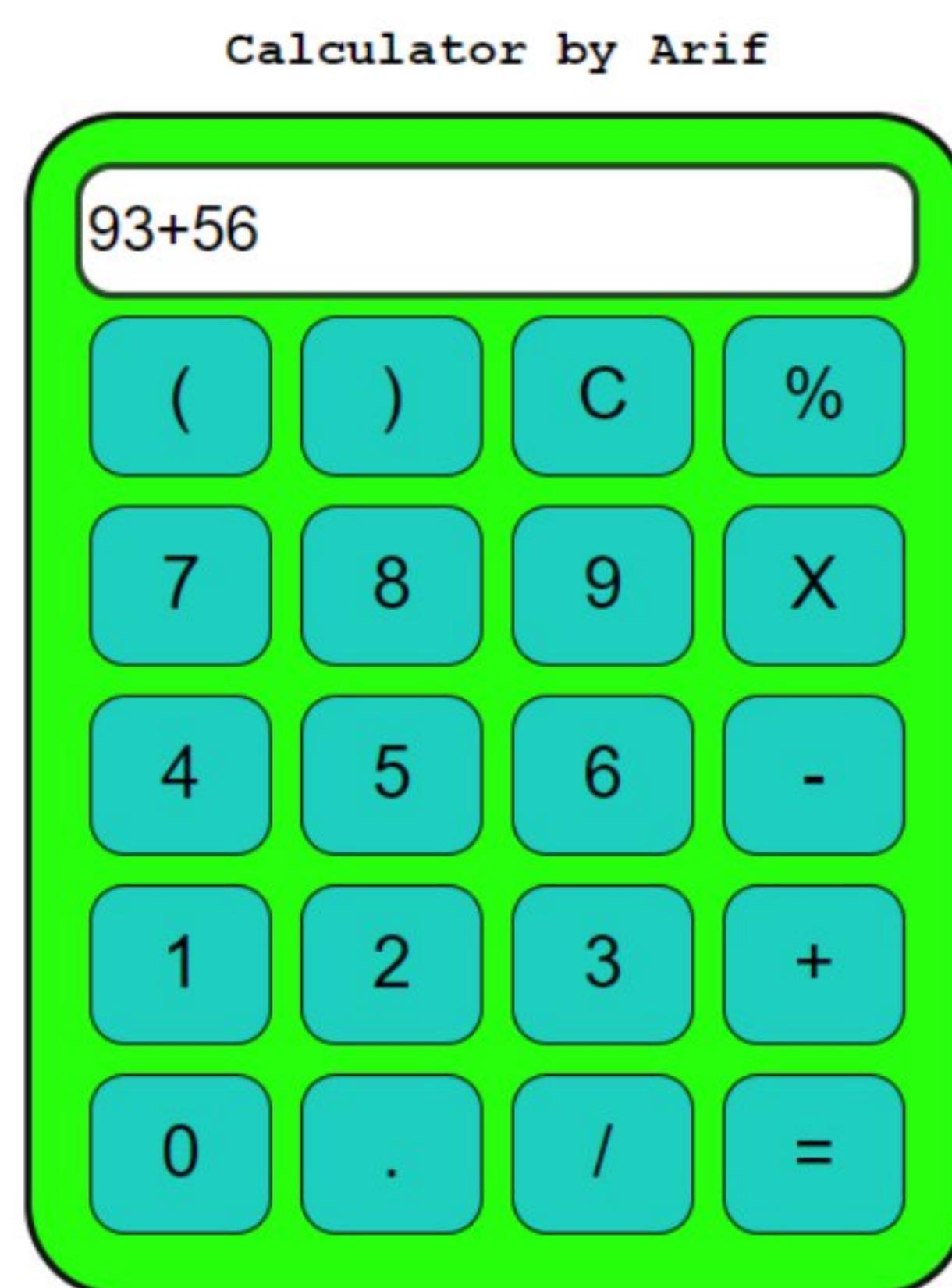
### Javascript Code:

```
let screen = document.getElementById('screen');  
buttons = document.querySelectorAll('button');  
let screenValue = " ";  
for (item of buttons) {  
    item.addEventListener('click', (e) => {  
        buttonText = e.target.innerText;  
        console.log('Button text is ', buttonText);  
        if (buttonText == 'X') {  
            buttonText = '*';  
            screenValue += buttonText;  
            screen.value = screenValue;  
        }  
        else if (buttonText == 'C') {  
            screenValue = "";  
            screen.value = screenValue;  
        }  
        else if (buttonText == '=') {  
            screen.value = eval(screenValue);  
        }  
        else {  
            screenValue += buttonText;  
        }  
    })  
}
```

```
screen.value = screenValue;  
}  
})  
}
```

---

## **OUTPUT**



```
<!-- Assignment 5  
Implement the sample program demonstrating the use of Servlet.  
e.g., Create a database table ebookshop (book_id, book_title, book_author,  
book_price,  
quantity) using database like Oracle/MySQL etc. and display (use SQL select query)  
the table  
content using servlet.  
-->
```

ebookdb.sql

```
CREATE DATABASE IF NOT EXISTS ebookshop;
```

```
USE ebookshop;
```

```
DROP TABLE IF EXISTS books;
```

```
CREATE TABLE books (
```

```
    id INT,  
    title VARCHAR(50),  
    author VARCHAR(50),  
    price FLOAT,  
    qty INT,  
    PRIMARY KEY (id)
```

```
);
```

```
INSERT INTO books VALUES (1001, 'Introduction to Python', 'Reema Thareja', '499',  
16);
```

```
INSERT INTO books VALUES (1002, 'OAuth in Action', 'Justin Richer', '399', 15);
```

```
INSERT INTO books VALUES (1003, 'Kafka on the Shore', 'Haruki Murakami', '150',  
21);
```

```
INSERT INTO books VALUES (1004, 'Two Scoops of Django', 'Daniel and Audrey', '250',  
19);
```

```
INSERT INTO books VALUES (1005, 'Game of Thrones', 'George R.R. Martin', '999',  
25);
```

```
SELECT * FROM books;
```

web.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<!--
```

```
Licensed to the Apache Software Foundation (ASF) under one or more contributor  
license agreements. See the NOTICE file distributed with this work for additional  
information regarding copyright ownership. The ASF licenses this file to You under  
the Apache License, Version 2.0 (the "License"); you may not use this file except  
in compliance with the License. You may obtain a copy of the License at
```

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-->

```
<web-app xmlns="http://java.sun.com/xml/ns/javaee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
  http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" version="2.5">

  <servlet>
    <servlet-name>display</servlet-name>
    <servlet-class>display</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>display</servlet-name>
    <url-pattern>/display</url-pattern>
  </servlet-mapping>
</web-app>
```

Servle\WEB-INF\classes\display.java

```
import java.io.*;
import javax.servlet.*;
import java.servlet.http.*;
import java.sql.*;

public class display extends HttpServlet {
    public void doGet(HttpServletRequest req, HttpServletResponse res) throws
IOException, ServletException {
        PrintWriter out = res.getWriter();
        res.setContentType("text/html");
        try {
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/ebookshop","","");
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery("SELECT * FROM books");
            out.println("<table border=1 width=50% height=50%>");

            out.println("<tr><th>BookID</th><th>BookName</th><th>Author</th><th>Price</th><th>Q
uantity</th><tr>");
            while (rs.next()) {
                String n = rs.getString("id");
                String nm = rs.getString("title");
                String an = rs.getString("author");
                Float pri = rs.getFloat("price");
                out.println("<tr><td>" + n + "</td><td>" + nm + "</td><td>" + an + "</td><td>" + pri + "</td><td>" + rs.getInt("quantity") + "</td></tr>");
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
        int quan = rs.getInt("quantity");
        out.println("<tr><td>" + n + "</td><td>" + nm + "</td><td>" + an +
"</td><td>" + pri + "</td><td>" + quan + "</td></tr>");
    }
    out.println("</table>");
    out.println("</html></body>");
    con.close();
}
catch (Exception e){
    out.println("error");
}
}
}
```

```
Command Prompt  
C:\Users\User\Visual Studio Code\Web Technology - Assignment\Assignment-5\Servlet\WEB-INF\classes>javac -cp servlet-api.jar display.java  
C:\Users\User\Visual Studio Code\Web Technology - Assignment\Assignment-5\Servlet\WEB-INF\classes>
```



BookID	BookName	Author	Price	Quantity
1001	Introduction to Python	Reema Thareja	499.0	16
1002	OAuth in Action	Justin Richer	399.0	15
1003	Kafka on the Shore	Haruki Murakami	150.0	21
1004	Two Scoops of Django	Daniel and Audrey	250.0	19
1005	Game of Thrones	George R.R. Martin	999.0	25

## Assignment-06

### Jsp Files

#### LoginPage.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>login page</title>
</head>
<body>
<form method="post" action="Logincheck">
    <table>
        <tr>
            <td>User Name</td>
            <td><input type="text"
name="uname"></td>
        </tr>
        <tr>
            <td>password</td>
            <td><input type="password"
name="password"></td>
        </tr>
        <tr>
            <td></td>
            <td><input type="submit"
name="Login"></td>
        </tr>
    </table>
</form>
</body>
</html>
```

## Success.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>
Welcome you are Logged in </h1>
</body>
</html>
```

---

## error.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
please Check your
UserName OR password
</body>
</html>
```

## Servlet Files

### LoginCheck.java

```
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import java.io.IOException;
/***
 * Servlet implementation class Logincheck
 */
public class Logincheck extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public Logincheck() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
     * response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.getWriter().append("Served at:
").append(request.getContextPath());
    }
}
```

```
/**  
 * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse  
response)  
*/  
  
protected void doPost(HttpServletRequest request, HttpServletResponse response)  
throws ServletException, IOException {  
  
    String uname=request.getParameter("uname");  
  
    String password=request.getParameter("password");  
  
    if(uname.equals("Arif") && password.equals("9503643167"))  
    {  
        response.sendRedirect("success.jsp");  
    }  
  
    else  
    {  
        response.sendRedirect("error.jsp");  
    }  
}
```

## OUTPUT

A screenshot of a web browser window. The address bar shows the URL `localhost:8080/LoginPage/LoginPage.jsp`. The page contains a form with two text input fields: "User Name" containing "Arif" and "password" containing ".....". Below the inputs is a "Submit" button.

A screenshot of a web browser window. The address bar shows the URL `localhost:8080/LoginPage/error.jsp`. The page displays the text "please Check your UserName OR password" in red.

A screenshot of a web browser window. The address bar shows the URL `localhost:8080/LoginPage/success.jsp`. The page displays the text "Welcome you are Logged in" in large, bold, black font.

```
<!-- Assignment - 7
Build a dynamic web application using PHP and MySQL.
a. Create database tables in MySQL and create connection with PHP.
b. Create the add, update, delete and retrieve functions in the PHP web app
interacting
with MySQL database
-->

<?php

// Make a MySQL Connection
mysql_connect("localhost", "root", "mypassword") or die(mysql_error());
mysql_select_db("test") or die(mysql_error());
echo "Connected to database!";
mysql_query("CREATE TABLE mytable(id INT NOT NULL AUTO_INCREMENT, PRIMARY
KEY(id), name VARCHAR(30),
phoneINT,emailIdVARCHAR(30))")
or die(mysql_error());
print "<br/>";
echo "Table Created!";

// Insert a row of information into the table
mysql_query("INSERT INTO mytable(name, phone,emailId) VALUES('Priyanka',
'11111','abc123@gmail.com' )") or die(mysql_error());
mysql_query("INSERT INTO mytable(name, phone,emailId) VALUES('Kumar',
'22222','pqr11@yahoo.com' )") or die(mysql_error());
mysql_query("INSERT INTO mytable(name, phone,emailId) VALUES('Archana',
'33333','xyz@rediffmail.com' )") or die(mysql_error());
print "<br/>";
echo "Data Inserted!";
$result =mysql_query("SELECT * FROM mytable") or die(mysql_error());
print "<br/>";
print "<b>User Database</b>";
echo "<table border='1'>";
echo "<tr><th>ID</th><th>Name</th><th>Phone</th><th>Email-ID</th>
</tr>";
while($row = mysql_fetch_array( $result ))
{
    // Print out the contents of each row into a table
    echo "<tr><td>";
    echo $row['id'];
    echo "</td><td>";
    echo $row['name'];
    echo "</td><td>";
    echo $row['phone'];
    echo "</td><td>";
    echo $row['emailId'];
    echo "</td></tr>";
}
echo "</table>";
```

```
$result = mysql_query("UPDATE mytable SET phone='55555' WHERE phone='22222'") or  
die(mysql_error());  
print "<br/>";  
echo "Data Updated!";  
$result =mysql_query("SELECT * FROM mytable") or die(mysql_error());  
print "<br/>";  
print "<b>User Database</b>";  
echo "<table border='1'>";  
echo "<tr><th>ID</th><th>Name</th><th>Phone</th><th>Email-ID</th>  
</tr>";  
while($row = mysql_fetch_array( $result ))  
{  
    // Print out the contents of each row into a table  
    echo "<tr><td>";  
    echo $row['id'];  
    echo "</td><td>";  
    echo $row['name'];  
    echo "</td><td>";  
    echo $row['phone'];  
    echo "</td><td>";  
    echo $row['emailId'];  
    echo "</td></tr>";  
}  
echo "</table>";  
$result = mysql_query("DELETE FROM mytable WHERE phone='33333'") or  
die(mysql_error());  
print "<br/>";  
echo "Data Deleted!";  
$result =mysql_query("SELECT * FROM mytable") or die(mysql_error());  
print "<br/>";  
print "<b>User Database</b>";  
echo "<table border='1'>";  
echo "<tr><th>ID</th><th>Name</th><th>Phone</th><th>Email-ID</th>  
</tr>";  
while($row = mysql_fetch_array( $result ))  
{  
    // Print out the contents of each row into a table  
    echo "<tr><td>";  
    echo $row['id'];  
    echo "</td><td>";  
    echo $row['name'];  
    echo "</td><td>";  
    echo $row['phone'];  
    echo "</td><td>";  
    echo $row['emailId'];  
    echo "</td></tr>";  
}  
echo "</table>";  
?>
```

[+/-] http://localhost/p...amples/DBDemo.php

Back Forward

http://localhost/php-examples/DBDemo.php

☰ - Google

Connected to database!

Table Created!

Data Inserted!

User Database

ID	Name	Phone	Email-ID
1	Priyanka	11111	abc123@gmail.com
2	Kumar	22222	pqr11@yahoo.com
3	Archana	33333	xyz@rediffmail.com

Data Updated!

User Database

ID	Name	Phone	Email-ID
1	Priyanka	11111	abc123@gmail.com
2	Kumar	55555	pqr11@yahoo.com
3	Archana	33333	xyz@rediffmail.com

Data Deleted!

User Database

ID	Name	Phone	Email-ID
1	Priyanka	11111	abc123@gmail.com
2	Kumar	55555	pqr11@yahoo.com

```
<!-- Assignment 8
Design a login page with entries for name, mobile number email id and login button.
Use struts
and perform following validations
a. Validation for correct names
b. Validation for mobile numbers
c. Validation for email id
d. Validation if no entered any value
e. Re-display for wrongly entered values with message
f. Congratulations and welcome page upon successful entries
-->
```

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://xmlns.jcp.org/xml/ns/javaee"
  xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
  http://xmlns.jcp.org/xml/ns/javaee/webapp_3_1.xsd"
  id="WebApp_ID" version="3.1">
  <display-name>Struts2</display-name>
  <welcome-file-list>
    <welcome-file>LoginPg.jsp</welcome-file>
    <welcome-file>Welcome.jsp</welcome-file>
  </welcome-file-list>
  <filter>
    <filter-name>struts2</filter-name>
    <filter-class>
      org.apache.struts2.dispatcher.FilterDispatcher
    </filter-class>
    </filter>
    <filter-mapping>
      <filter-name>struts2</filter-name>
      <url-pattern>/*</url-pattern>
    </filter-mapping>
  </web-app>
```

struct.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts PUBLIC
"-//Apache Software Foundation//DTD Struts Configuration 2.0//EN"
"http://struts.apache.org/dtds/struts-2.0.dtd">
<struts>
  <constant name="struts.custom.i18n.resources" value="LoginProg" />
  <package name="default" extends="struts-default" namespace="/">
    <action name = "Login"
      class = "myPkg.LoginProg"
      method = "execute">
```

```
<result name = "success">/Welcome.jsp</result>
<result name = "input">/LoginPg.jsp</result>
</action>
</package>
</struts>
```

#### welcome.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@taglib uri="/struts-tags" prefix="s"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
        <title>login Page Demo</title>
    </head>
    <body>
        <h1>Congratulations!!! </h1>
        <h2>Welcome, User <s:property value="userName"/>!!!</h2>
    </body>
</html>
```

#### loginProh.java

```
package myPkg;
import com.opensymphony.xwork2.ActionSupport;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
public class LoginProg extends ActionSupport{
    private String userName;
    private String mobileNum;
    private String email;
    public String execute() {
        return SUCCESS;
    }
    public String getUserName() {
        return userName;
    }
    public void setUserName(String userName) {
        this.userName = userName;
    }
    public String getMobileNum() {
        return mobileNum;
    }
    public void setMobileNum(String mobileNum) {
        this.mobileNum = mobileNum;
    }
    public String getEmail() {
```

```
return email;
}
public void setEmail(String email) {
this.email = email;
}
public void validate() {
    //validating user name
    if (getUserName().length() == 0) {
        addFieldError("userName", "User Name is required");
    }
    else if (!getUserName().equals("Admin")) {
        addFieldError("userName", getText(userName)+" is not authentic user name");
    }
    //validating mobile number
    if (getMobileNum().length() == 0) {
        addFieldError("mobileNum", "Mobile Number is required");
    }
    else if(getMobileNum().length()!= 10)
        addFieldError("mobileNum", getText(mobileNum)+" is invalid Mobile Number");
    else if (!getMobileNum().equals("1234567890")) {
        addFieldError("mobileNum", getText(mobileNum)+" is not authentic mobile
number");
    }
    //validating email-id
    if (getEmail().length() == 0) {
        addFieldError("email", "Email-ID is required");
    }
    else if (!getEmail().equals("abc.xyz@gmail.com")) {
        addFieldError("email", getText(email)+" is not authentic email-id");
    }
    else {
        String expression="^[\w\w-]([\w\w])+[\w]+@[[\w\w-]+\w]+[A-Z]{2,4}$";
        CharSequence inputStr = email;
        Pattern pattern = Pattern.compile(expression,Pattern.CASE_INSENSITIVE);
        Matcher matcher = pattern.matcher(inputStr);
        if(!matcher.matches())
            addFieldError("email","Invalid email address");
    }
}
}
```

loginPg.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@taglib uri="/struts-tags" prefix="s" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
    <head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Login Page Demo</title>
<s:head />
</head>
<body>
    <s:form action="Login" validate="true">
        <s:textfield name="userNmae" label="User Name" />
        <s:textfield name="mobileNum" label="Mobile Number" />
        <s:textfield name="email" label="Email ID" />
        <s:submit value="Login" />
    </s:form>
</body>
</html>
```

Login Page Demo

User Name is required

User Name:

Mobile Number is required

Mobile Number:

Email-ID is required

Email ID:

Login Page Demo

abc is not authentic user name

User Name:

Mobile Number:

aa#gmail.com is not authentic email-id

Email ID:

Login Page Demo

abc is not authentic user name

User Name:

123 is invalid Mobile Number

Mobile Number:

Email-ID is required

Email ID:

Login Page Demo

localhost:8082/test/

User Name: Admin

Mobile Number: 1234567890

Email ID: abc.xyz@gmail.com

Login

login Page Demo

localhost:8082/test/Login

Congratulations!!!

Welcome, User Admin!!!

```
<!-- Assignment 9
Design an application using Angular JS.
e.g., Design registration (first name, last name, username, password) and login
page using
Angular JS.
-->
```

app.html

```
<!DOCTYPE html>
<html ng-app="myApp">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="stylesheet"
    href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
    <script
    src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>
    <script
    src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
    <script
    src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.4.7/angular.min.js"><scri
pt>
    <script
    src="https://cdnjs.cloudflare.com/ajax/libs/angular.js/1.4.7/angular-route.min.js">
</script>
</head>
<body ng-app="myApp">
    <div class = "row">
        <div class = "col-sm-12 well">
            <center><h1>OUR WEB SITE</h1></center>
        </div>
    </div>
    <div class="row">
        <div class ="col-sm-4 well" style = "background-color:pink;height:500px">
            <ul>
                <li><a href="#">Home</a></li>
                <li><a href="#registration">Registration</a></li>
                <li><a href="#login">Login</a></li>
            </ul>
        </div>
        <div class ="col-sm-8 well" style="background-color:aqua;height:500px">
            <div ng-view></div>
        </div>
    </div>
    <script src="app.js"></script>
</body>
</html>
```

app.js

```
var app = angular.module('myApp', ['ngRoute']);

app.config(function($routeProvider) {
    $routeProvider
        .when('/', {
            templateUrl : 'home.html',
            controller : 'HomeController'
        })
        .when('/registration', {
            templateUrl : 'registration.html',
            controller : 'RegistrationController'
        })
        .when('/login', {
            templateUrl : 'login.html',
            controller : 'LoginController'
        })
        .otherwise({redirectTo: '/'});
});

app.controller('HomeController', function($scope) {
    $scope.message = 'Welcome to a home page';
});

app.controller('RegistrationController', function($scope) {
    $scope.message = 'Please fill up the Registration form(New User)';
});

app.controller('LoginController', function($scope) {
    $scope.message = 'Please fill up the Login form(Existing User)';
});

home.html

<div>
    <center><h3>{{message}}</h3></center> <!-- Message for home page -->
</div>

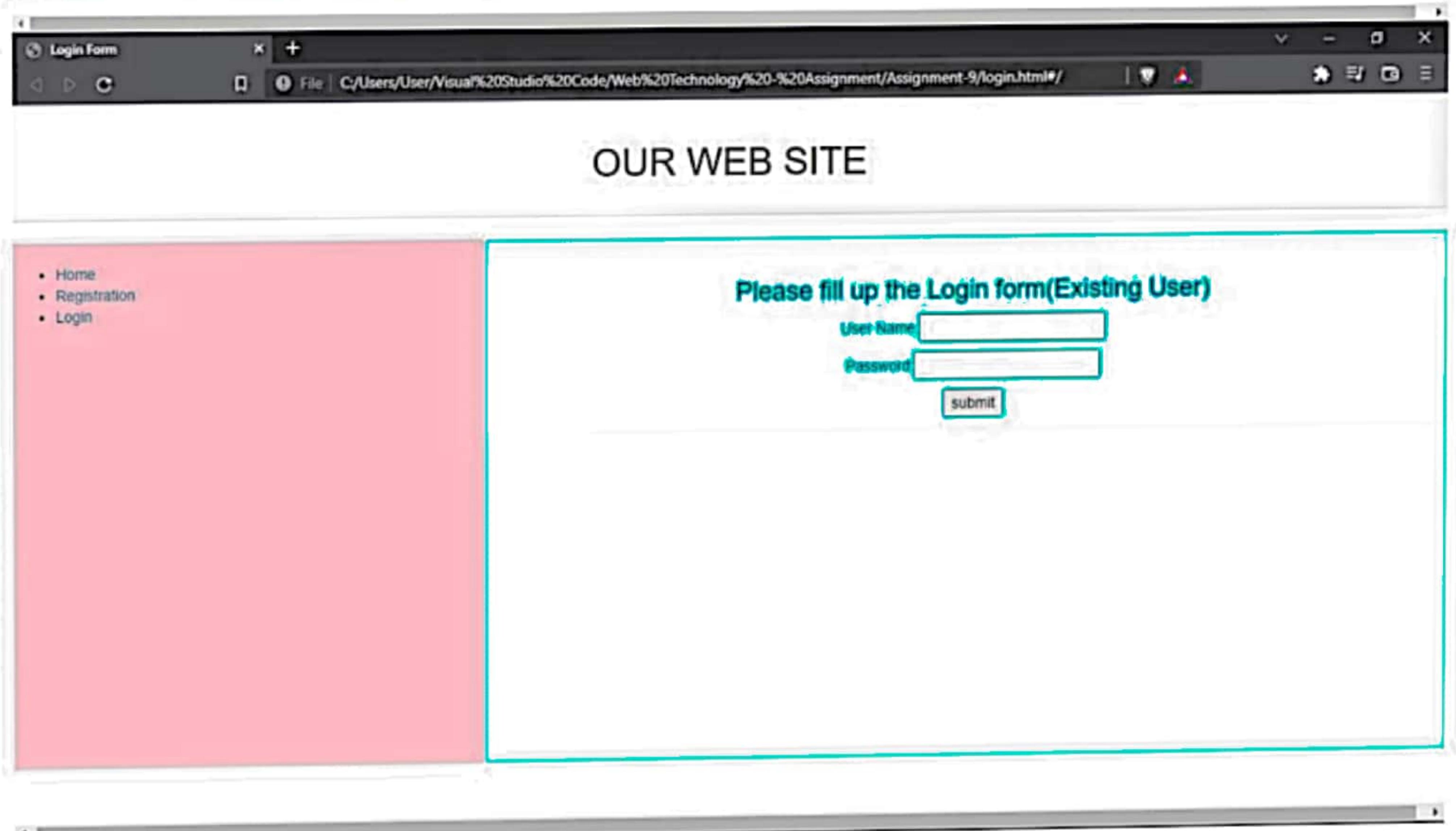
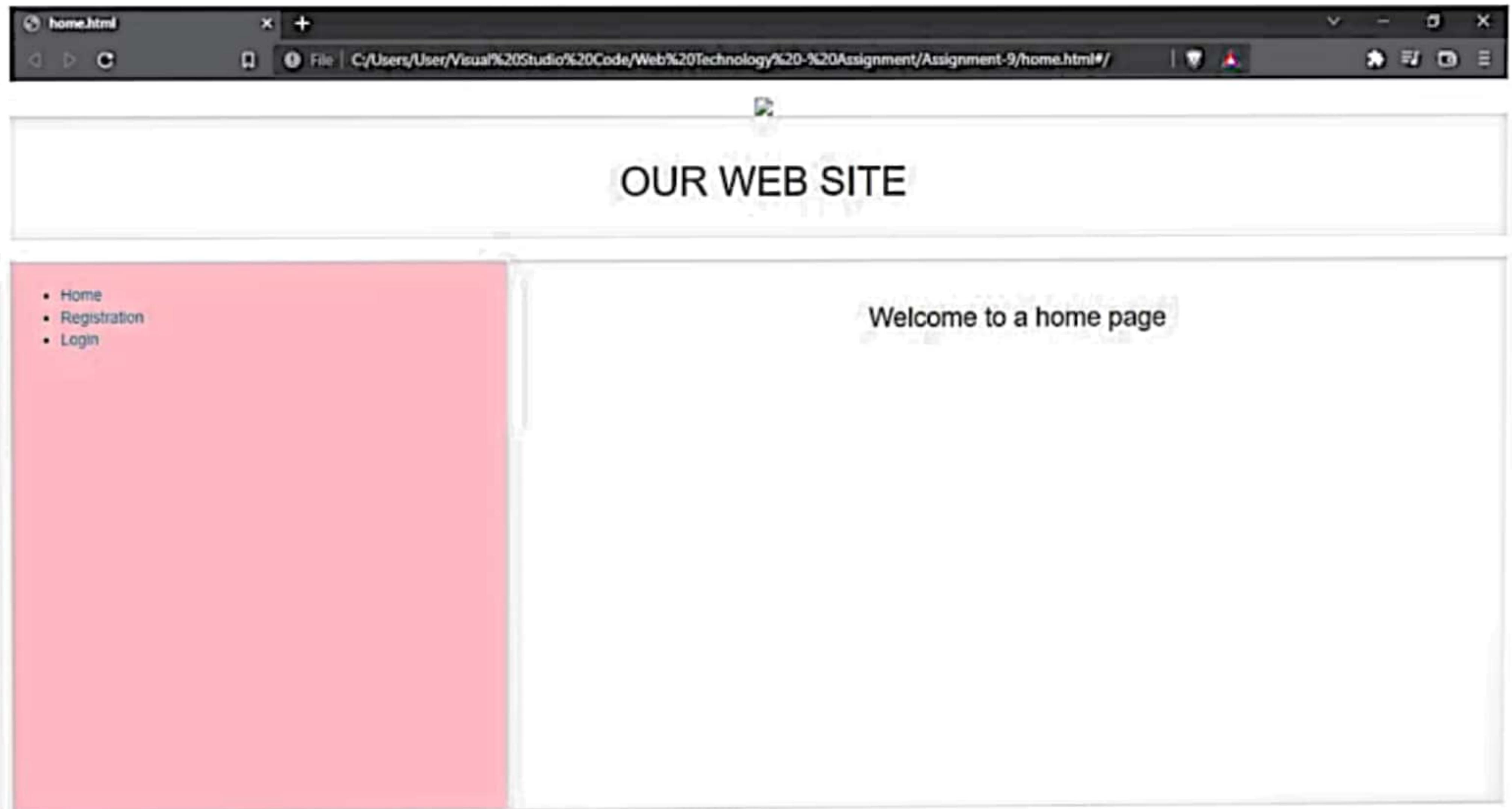
login.html

<html>
    <head>
        <title>
            Login Form
        </title>
    </head>
    <body>
        <h3>{{message}}</h3>
        <form>
            <p>User Name:<input type="text" name="username"></p>
```

```
<p>Password:<input type="password" name="password"></p>
<p><input type="submit" value="submit"></p>
</form>
</body>
</html>
```

registration.html

```
<html>
  <head>
    <title>
      Registration Form
    </title>
  </head>
  <body>
    <h3>{{message}}</h3>
    <form>
      <p>First Name:<input type="text" name="fname"></p>
      <p>Last Name:<input type="text" name="lname"></p>
      <p>User Name:<input type="text" name="username"></p>
      <p>Password:<input type="password" name="password"></p>
      <p><input type="submit" value="submit"></p>
    </form>
  </body>
</html>
```



Registration Form

File | C:/Users/User/Visual%20Studio%20Code/Web%20Technology%20-%20Assignment/Assignment-9/registration.html

# OUR WEB SITE

• Home  
• Registration  
• Login

Please fill up the Registration form(New User)

First Name

Last Name

User Name

Password

## Assignment 10

Code:

```
MyInterface.java
package myPkg;
import javax.ejb.Remote;
@Remote
public interface MyInterface {
    public int add(int a,int b);
}
```

```
MyClass.java
package myPkg;
import javax.ejb.Stateless;
@Stateless
public class MyClass implements MyInterface{
@Override
public int add(int a,int b) {
return a+b;
}
}
```

```
Glassfish-ejb-jar.xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE glassfish-ejb-jar PUBLIC "-//GlassFish.org//DTD GlassFish Application
Server 3.1
EJB 3.1//EN" "http://glassfish.org/dtds/glassfish-ejb-jar_3_1-1.dtd">
<glassfish-ejb-jar>
<enterprise-beans>
<ejb>
<ejb-name>MyClass</ejb-name>
<jndi-name>aaa</jndi-name>
</ejb>
</enterprise-beans>
</glassfish-ejb-jar>
```

```
DriverClass.java
package myPkg;
import java.util.Properties;
import javax.naming.InitialContext;
public class DriverClass {
public static void main(String arg[])throws Exception
{
Properties p=new Properties();
InitialContext c=new InitialContext();
Object obj=c.lookup("aaa");
MyInterface inter=(MyInterface)obj;
System.out.println("The addition of two numbers is = "+inter.add(10,20));
}
```

```
}
```

```
index.html
<html>
<head>
<title>Web Application For Bank Transactions</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>
<form action="Main">
Enter Amount: <input type="text" name="amt_txt">
<br/><br/>
Select Operation:
<br/><br/>
<select name="operation">
<option value="deposit">Deposit</option>
<option value="withdraw">Withdraw</option>
<option value="balance">Balance</option>
</select>
<br/><br/>
<input type="Submit" value="Submit">
</form>
</body>
</html>
```

```
MyBean.java
package myPkg;
import javax.ejb.Stateful;
@Stateful
public class MyBean implements MyBeanLocal {
int balance=0;
@Override
public void deposit(int amount) {
balance=balance+amount;
}
@Override
public boolean withdraw(int amount) {
if(amount<=balance)
{
balance=balance-amount;
return true;
}
return false;
}
@Override
public int getBalance() {
return balance;
}
}
```

```
Main.java
import java.io.IOException;
import java.io.PrintWriter;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.naming.Context;
import javax.naming.InitialContext;
import javax.naming.NamingException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import myPkg.MyBeanLocal;
public class Main extends HttpServlet {
MyBeanLocal myBean = lookupMyBeanLocal();
protected void processRequest(HttpServletRequest request, HttpServletResponse
response)
throws ServletException, IOException {
response.setContentType("text/html;charset=UTF-8");
try (PrintWriter out = response.getWriter()) {
String choice=request.getParameter("operation");
int amount=Integer.parseInt(request.getParameter("amt_txt"));
if(choice.equals("deposit"))
{
myBean.deposit(amount);
out.println("The amount is deposited successfully!!!");
}
if(choice.equals("withdraw"))
{
f(myBean.withdraw(amount))
{
out.println("The amount is wihsdrawn successfully!!!");
}
Else
out.println("Sorry, Balance insufficient!!!");
}
if(choice.equals("balance"))
{
out.println("The total balance is = "+myBean.getBalance());
}
}
}
...
...
private MyBeanLocal lookupMyBeanLocal() {
try {
Context c = new InitialContext();
return (MyBeanLocal) c.lookup("java:global/MyWebApp/MyWebAppejb/
MyBean!myPkg.MyBeanLocal");
```

```
} catch (NamingException ne) {  
    Logger.getLogger(getClass().getName()).log(Level.SEVERE, "exception caught", ne);  
    throw new RuntimeException(ne);  
}  
}  
}
```