Week1

1a) Create a Web page using html which contains a Heading, Image and 2 hyperlinks. Each hyperlink opens a new page in the same web browser. New page contains “Go Back” link that takes you to main page.

Source Code:

Main.html

```
<!DOCTYPE htmnl>
<html>
   <head>
      <meta charset='utf-8'>
      <meta http-equvi='X-UA-Compatible' content="IE=edge">
      <title> G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND
SCIENCES</title>
      <meta name="viewport" content="width=device-width, initial-scale=1">
      <style>
         #header{
            background-color:aquamarine;
            margin:10px;
            padding:5px;
            height:100px;
         }
         #menu{
            background-color:rgb(175, 82, 167);
            margin:10px;
            padding:5px;
            height:100px;
         }
         #content{
            height:300px;
            }
            img{
               float:left;
            }
            h1{
               text-align :center;
            }
            #menu a{
               padding:8px;
               display: inline-flex;
               float:inline-start;
            }
            #footer{
               background-color:rgb(160, 72, 72);
               height:25px;
            }
         </style>
      </head>
      <body>
         <div id="header">
            <img src="C:\Users\Admin\Desktop\gnits.jpeg" alt="image not found!"
height="90" width="100"/>
            <h1> G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCES
</h1>
         </div>
         <DIV ID="menu">
            <a href="it.html"> IT</a>
            <a href="cst.html"> CST</a>
         </DIV>
```

```
        <div id="content">


        </div>
        <div id="footer">
            <p style="text-align:center";>
                contact @ 8639460069
            </p>
        </div>
    </body>
</html>
```

lt.html

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset='utf-8'>
        <meta http-equvi='X-UA-Compatible' content="IE=edge">
        <title> G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND
SCIENCES</title>
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <style>
            #header{
                background-color:aquamarine;
                margin:10px;
                padding:5px;
                height:100px;
            }
            #menu{
                background-color:rgb(175, 82, 167);
            margin:10px;
            padding:5px;
            height:100px;
        }
        #content{
            height:300px;
        }
        img{
            float:left;
        }
        h1{
            text-align :center;
        }
        #menu a{
            padding:8px;
            display: inline-flex;
            float:inline-end;
        }
        #footer{
            background-color:rgb(160, 72, 72);
            height:25px;
        }
    </style>
</head>
<body>
    <div id="header">
        <img src="C:\Users\Admin\Desktop\gnits.jpeg" alt="image not found!"
height="90" width="100"/>
```

```
    <h1> G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCES
</h1>
  </div>
  <DIV ID="menu">
    <a href="main.html"> go Back</a>
  </DIV>
  <div id="content">
   <h2><b>DEPARTMENT VISION </b> </h2>
   <p>To build a prime transformative learning community that
      responds swiftly to the challenges of Information Technology. </p>
    <h2><b>DEPARTMENT MIsSION </b> </h2>
    <p>
    To foster an intellectual environment that delivers virtuous
    Information Technocrats with commitment to industry and society
    by strengthening the logical, analytical and applicative skills to
    excel academically and professionally. To inculcate good
    communication skills in students and introduce them to various
    codes of professional practices for carrying out effective team
    collaborations and project management in the field of IT.</p>
    <h2><b>PROGRAM EDUCATIONALOBJECTIVES:  </b> </h2>
    <p>
    PEO-1: Providing students with a compelling foundation in Engineering and
Basic
    Sciences that will further help them conduct investigations of complex
    problems. <br>
    PEO-2: Applying scientific and engineering methodologies using modern
tools
    and techniques in the analysis, design, implementation and evaluation of
    information in the field of IT. <br>
```

```
            PEO-3: Promoting lifelong learning and help students in aiming for higher
            education and become successful Engineers in the society. <br>
            PEO-4: Inculcating strong communication skills, ethics and various codes of
            professional practices useful in performing effective project management
            & team collaborations and enable them to sustain and excel in various
            environments. <br>
            </p>
        </div>
        <div id="footer">
            <p style="text-align:center";>
                contact @ 8639460069
            </p>
        </div>
    </body>
</html>
```

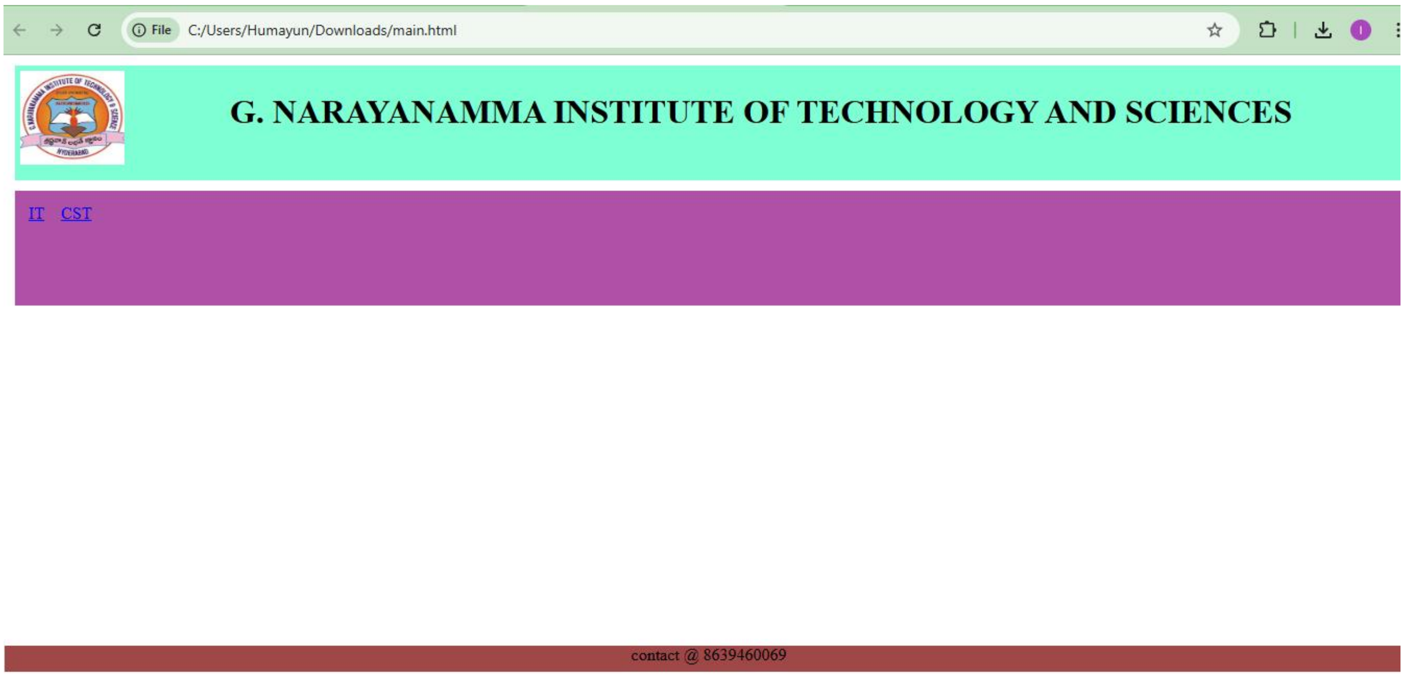
cst.html

```
<!DOCTYPE htmnl>
<html>
    <head>
        <meta charset='utf-8'>
        <meta http-equvi='X-UA-Compatible' content="IE=edge">
        <title> G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND
SCIENCES</title>
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <style>
            #header{
                background-color:aquamarine;
                margin:10px;
        padding:5px;
        height:100px;
    }
    #menu{
        background-color:rgb(175, 82, 167);
        margin:10px;
        padding:5px;
        height:100px;
    }
    #content{
        height:300px;
    }
    img{
        float:left;
    }
    h1{
        text-align :center;
    }
    #menu a{
        padding:8px;
        display: inline-flex;
        float:inline-end;
    }
    #footer{
        background-color:rgb(160, 72, 72);
        height:25px;
    }
</style>
```

```
   </head>
   <body>
       <div id="header">
           <img src="C:\Users\Admin\Desktop\gnits.jpeg" alt="image not found!"
height="90" width="100"/>
           <h1> G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCES
</h1>
       </div>
       <DIV ID="menu">
           <a href="main.html"> go Back</a>
       </DIV>
       <div id="content">
           <h2><b>DEPARTMENT VISION </b> </h2>
          <p>To build a prime transformative learning community that
             responds swiftly to the challenges of Information Technology. </p>
           <h2><b>DEPARTMENT MIsSION </b> </h2>
           <p>
           To foster an intellectual environment that delivers virtuous
           Information Technocrats with commitment to industry and society
           by strengthening the logical, analytical and applicative skills to
           excel academically and professionally. To inculcate good
           communication skills in students and introduce them to various
           codes of professional practices for carrying out effective team
           collaborations and project management in the field of IT.</p>
           <h2><b>PROGRAM EDUCATIONALOBJECTIVES:  </b> </h2>
           <p>
           PEO-1: Providing students with a compelling foundation in Engineering and
Basic
           Sciences that will further help them conduct investigations of complex
```

```
            problems. <br>
            PEO-2: Applying scientific and engineering methodologies using modern
tools
            and techniques in the analysis, design, implementation and evaluation of
            information in the field of IT. <br>
            PEO-3: Promoting lifelong learning and help students in aiming for higher
            education and become successful Engineers in the society. <br>
            PEO-4: Inculcating strong communication skills, ethics and various codes of
            professional practices useful in performing effective project management
            & team collaborations and enable them to sustain and excel in various
            environments. <br>
            </p>
        </div>
        <div id="footer">
            <p style="text-align:center";>
                contact @ 8639460069
            </p>
        </div>
    </body>
</html>
```

Output:

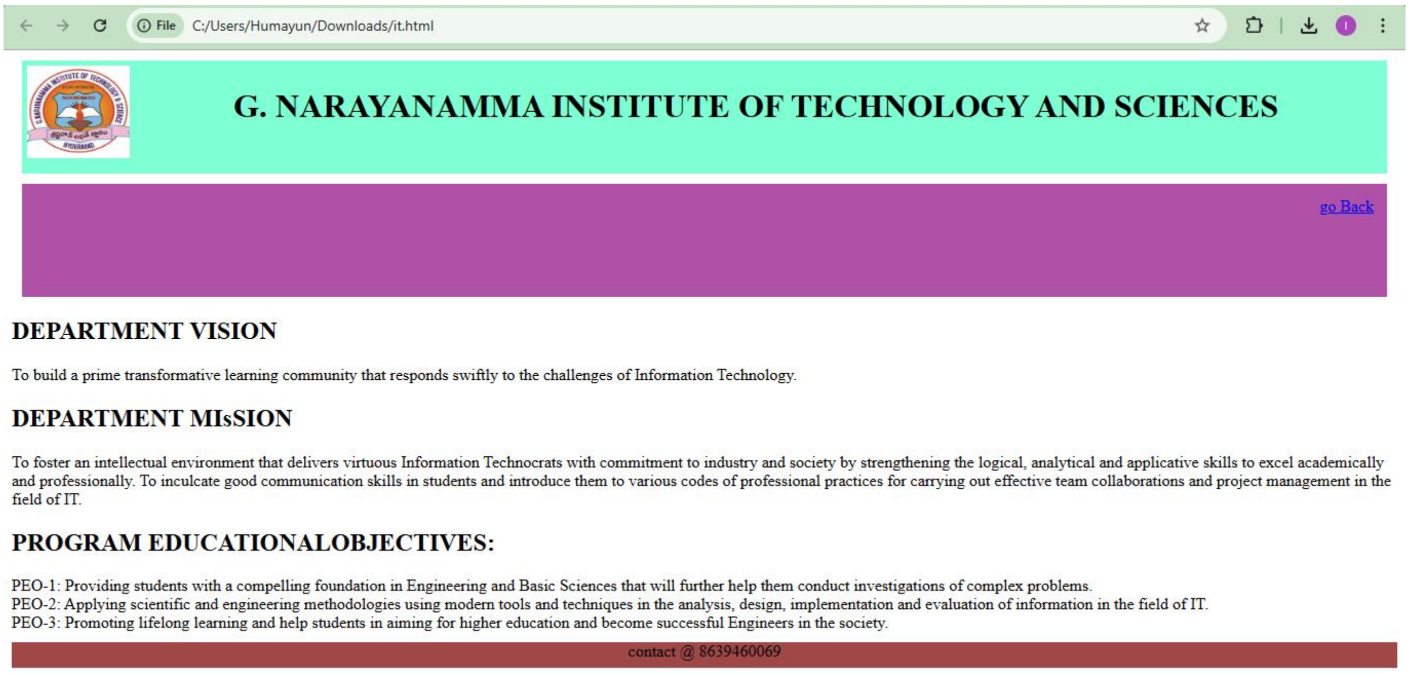


C:/Users/Humayun/Downloads/main.html

G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCES

IT CST

contact @ 8639460069



C:/Users/Humayun/Downloads/it.html

G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCES

go Back

DEPARTMENT VISION

To build a prime transformative learning community that responds swiftly to the challenges of Information Technology.

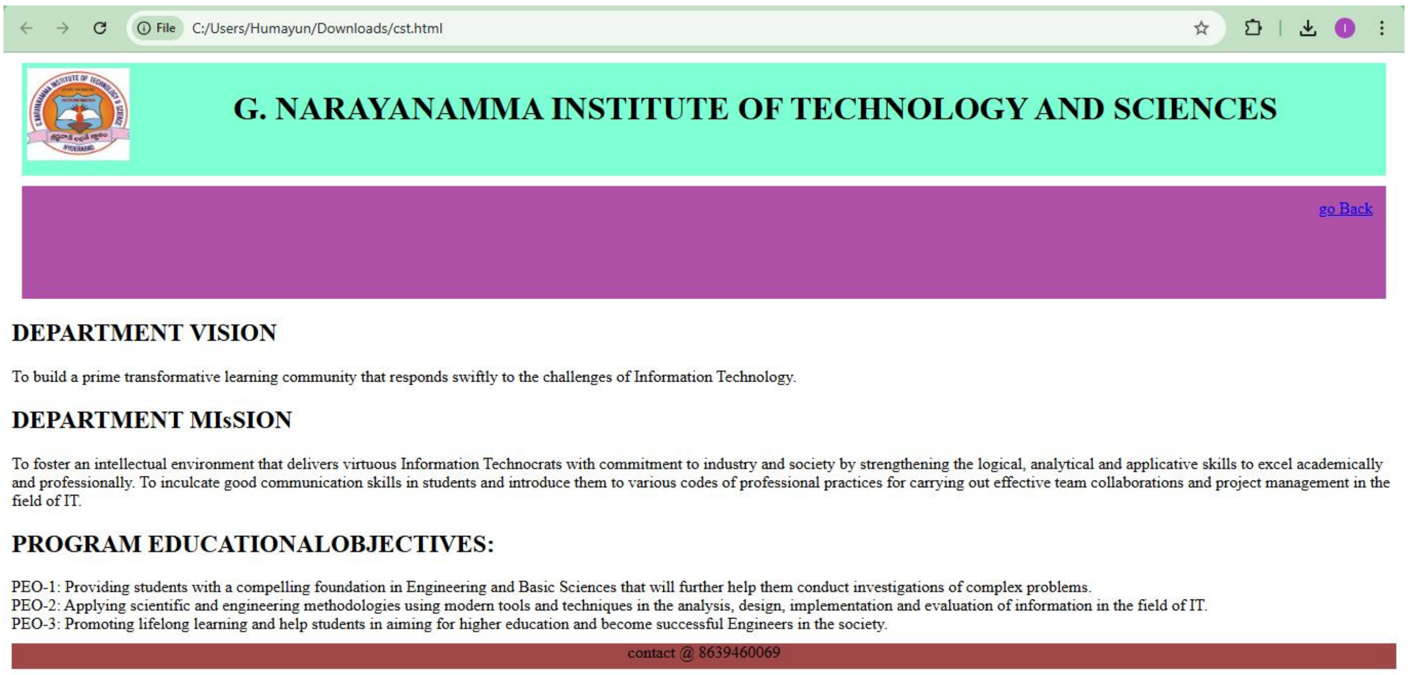
DEPARTMENT MIsSION

To foster an intellectual environment that delivers virtuous Information Technocrats with commitment to industry and society by strengthening the logical, analytical and applicative skills to excel academically and professionally. To inculcate good communication skills in students and introduce them to various codes of professional practices for carrying out effective team collaborations and project management in the field of IT.

PROGRAM EDUCATIONALOBJECTIVES:

PEO-1: Providing students with a compelling foundation in Engineering and Basic Sciences that will further help them conduct investigations of complex problems.
PEO-2: Applying scientific and engineering methodologies using modern tools and techniques in the analysis, design, implementation and evaluation of information in the field of IT.
PEO-3: Promoting lifelong learning and help students in aiming for higher education and become successful Engineers in the society.

contact @ 8639460069



C:/Users/Humayun/Downloads/cst.html

G. NARAYANAMMA INSTITUTE OF TECHNOLOGY AND SCIENCES

go Back

DEPARTMENT VISION

To build a prime transformative learning community that responds swiftly to the challenges of Information Technology.

DEPARTMENT MIsSION

To foster an intellectual environment that delivers virtuous Information Technocrats with commitment to industry and society by strengthening the logical, analytical and applicative skills to excel academically and professionally. To inculcate good communication skills in students and introduce them to various codes of professional practices for carrying out effective team collaborations and project management in the field of IT.

PROGRAM EDUCATIONALOBJECTIVES:

PEO-1: Providing students with a compelling foundation in Engineering and Basic Sciences that will further help them conduct investigations of complex problems.
PEO-2: Applying scientific and engineering methodologies using modern tools and techniques in the analysis, design, implementation and evaluation of information in the field of IT.
PEO-3: Promoting lifelong learning and help students in aiming for higher education and become successful Engineers in the society.

contact @ 8639460069

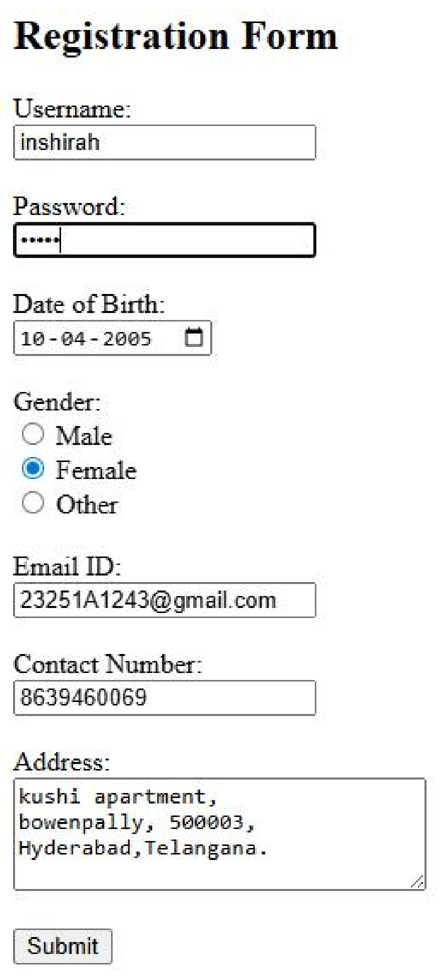
1b. Write a HTML program to create a Registration Form, which contains User Name, Password, Date of Birth, Gender, Mail-id, Contact number, Address and Submit button.

Source Code:

```
<!DOCTYPE html>

<html>

<head>

   <title>Registration Form</title>

</head>

<body>

   <h2>Registration Form</h2>

   <form>

      <label>Username:</label><br>

      <input type="text" name="username"><br><br>

      <label>Password:</label><br>

      <input type="password" name="password"><br><br>

      <label>Date of Birth:</label><br>

      <input type="date" name="dob"><br><br>

      <label>Gender:</label><br>

      <input type="radio" name="gender" value="male"> Male<br>

      <input type="radio" name="gender" value="female"> Female<br>

      <input type="radio" name="gender" value="other"> Other<br><br>

      <label>Email ID:</label><br>

      <input type="email" name="email"><br><br>


      <label>Contact Number:</label><br>

      <input type="tel" name="contact"><br><br>

      <label>Address:</label><br>
```

```
        <textarea name="address" rows="4" cols="30"></textarea><br><br>
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

Output:



Registration Form

Username:
inshirah

Password:
•••••

Date of Birth:
10-04-2005

Gender:
○ Male
● Female
○ Other

Email ID:
23251A1243@gmail.com

Contact Number:
8639460069

Address:
```
kushi apartment,
bowenpally, 500003,
Hyderabad,Telangana.
```

Week 2:

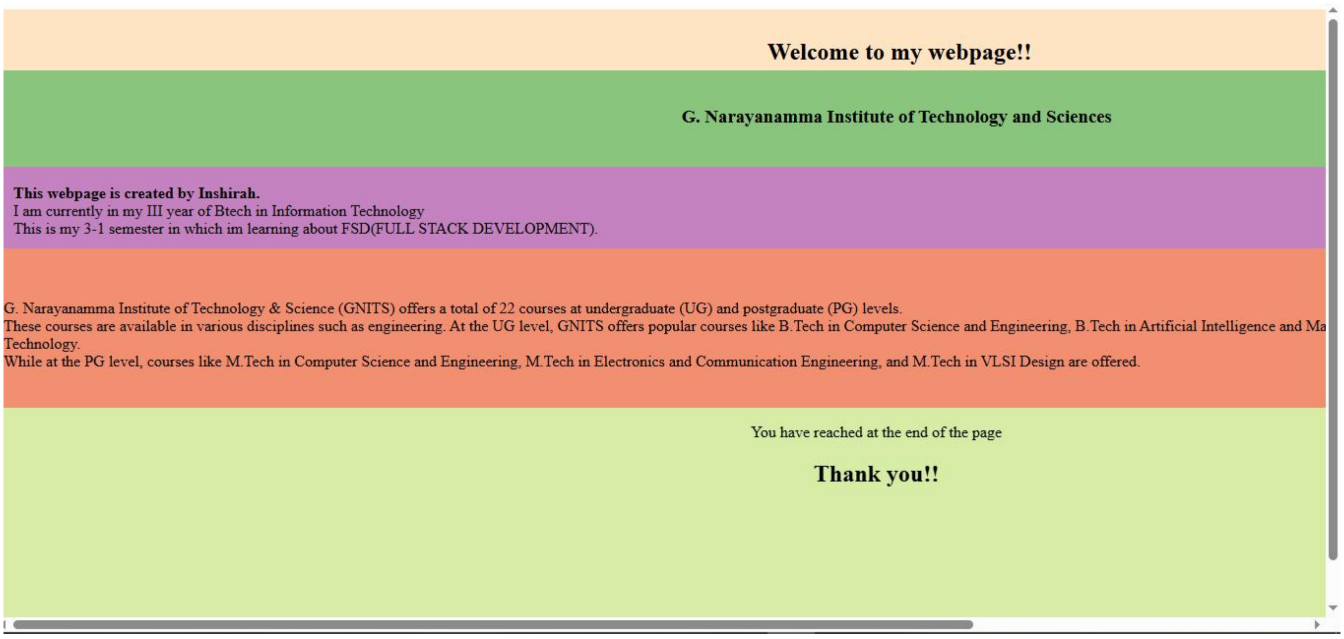
2a) Create a web page to demonstrate Position Property in CSS.

Source Code:

```
<!doctype html>
<head>
   <title> Week2a</title>
   <style>
      #w{
         background-color: bisque;
         text-align:center;
         width: 1735px;
         height: 164px;
         position: fixed;
         padding: 10px;
      }
      #header{
          background-color: rgb(137, 197, 125);
         text-align:center;
         width: 1780px;
         height: 170px;
         position: sticky;
         top:70px;
         padding: 17px;
      }
       #body{
          background-color: rgb(196, 131, 192);
         width: 1720px;
```

```
            height: 235px;
            position: static;
            padding: 10px;
        }
         #body2{
             background-color: rgb(243, 143, 113);
            width: 1777px;
            height: 722px;
            position: relative;
        }
         #footer{
             background-color: rgba(168, 109, 207, 0.685);
            text-align:center;
            width: 1773px;
            height: 659px;
            position: absolute;
        }
    </style>
</head>
<body>
    <div id="w">
        <h1> Welcome to my webpage!!</h1>
    </div>
    <div id="header">
        <h2> G. Narayanamma Institute of Technology and Sciences </h2>
    </div>
    <div id="body">
        <br>
        <br>
        <br>
        <p> <b>This webpage is created by Inshirah.</b><br>
        I am currently in my III year of Btech in Information Technology<br>
        This is my 3-1 semester in which im learning about FSD(FULL STACK
    DEVELOPMENT).
        </p>
      </div>
      <DIV id="body2">
         <br>
        <br>
        <br>
        <br>
        <p> G. Narayanamma Institute of Technology & Science (GNITS) offers a total
    of 22 courses at undergraduate (UG) and postgraduate (PG) levels. <br>
            These courses are available in various disciplines such as engineering. At
    the UG level, GNITS offers popular courses like B.Tech in Computer Science and
    Engineering, B.Tech in Artificial Intelligence and Machine Learning, B.Tech in Data
    Science, and B.Tech in Information Technology. <br>
            While at the PG level, courses like M.Tech in Computer Science and
    Engineering, M.Tech in Electronics and Communication Engineering, and M.Tech in
    VLSI Design are offered.</p>
      </DIV>
      <div id="footer">
        <p> You have reached at the end of the page</p>
        <h2> Thank you!!</h2>
      <div>
    </body>
    </html>
```

Output:



Welcome to my webpage!!

G. Narayanamma Institute of Technology and Sciences

This webpage is created by Inshirah.
I am currently in my III year of Btech in Information Technology
This is my 3-1 semester in which im learning about FSD(FULL STACK DEVELOPMENT).

G. Narayanamma Institute of Technology & Science (GNITS) offers a total of 22 courses at undergraduate (UG) and postgraduate (PG) levels.
These courses are available in various disciplines such as engineering. At the UG level, GNITS offers popular courses like B.Tech in Computer Science and Engineering, B.Tech in Artificial Intelligence and Ma
Technology.
While at the PG level, courses like M.Tech in Computer Science and Engineering, M.Tech in Electronics and Communication Engineering, and M.Tech in VLSI Design are offered.

You have reached at the end of the page

Thank you!!

2b) Create a Newspaper style Design to print minimum 2 articles using HTML and CSS.

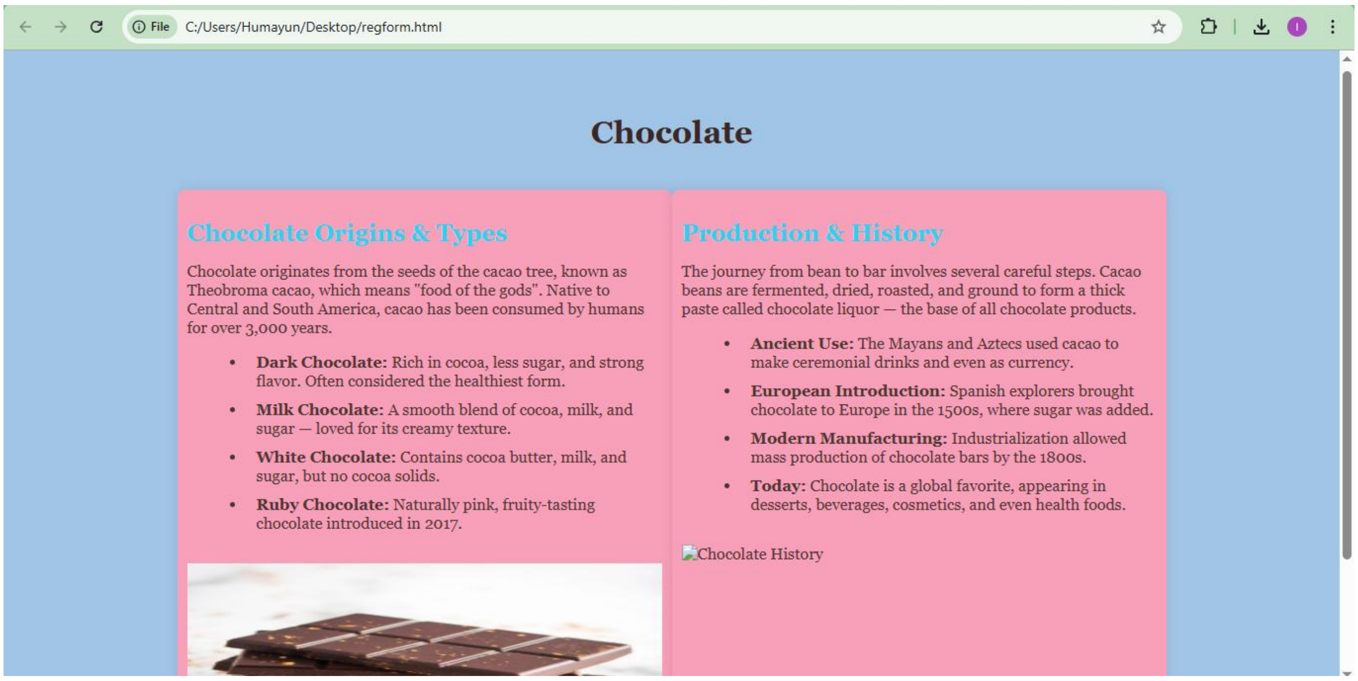
Source Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Chocolate Newsletter</title>
  <style>     body {       background-color: #fdf6f0;       font-family: 'Georgia', serif;
color: #4e342e;       padding: 40px;
    }
    .container {       max-width: 1000px;       margin: auto;       display: flex;
    }
    .column {
      flex: 1;
      background-color: #fff8e1;       padding: 10px;       border-radius: 8px;       box-
shadow: 0 0 12px rgba(0, 0, 0, 0.1);
    }     h1 {       text-align: center;      margin-bottom: 40px; font-size: 2em;       color:
#3e2723;
    }     h2 {       color: #5d4037;       margin-bottom: 15px;
    }     ul {
      margin-left: 20px;
    }     li {
      margin-bottom: 10px;       padding-left: 10px;
    }
    img {       width: 100%;       height: 200px;       margin-top: 15px;
    }     p {
      margin-bottom: 15px;
    }
  </style>
```

```
</head>
<body>
  <h1>Chocolate Lovers</h1>
  <div class="container">
   <div class="column"> <h2>Chocolate Origins & Types</h2>
      <p>Chocolate originates from the seeds of the cacao tree, known as Theobroma
cacao, which means "food of the gods". Native to Central and South America, cacao
has been consumed by humans for over 3,000 years.</p>
      <ul>
        <li><strong>Dark Chocolate:</strong> Rich in cocoa, less sugar, and strong
flavor. Often considered the healthiest form.</li>
        <li><strong>Milk Chocolate:</strong> A smooth blend of cocoa, milk, and sugar
— loved for its creamy texture.</li>
        <li><strong>White Chocolate:</strong> Contains cocoa butter, milk, and sugar,
but no cocoa solids.</li>
        <li><strong>Ruby Chocolate:</strong> Naturally pink, fruity-tasting chocolate
introduced in
2017.</li>
      </ul>
      <img
src="https://i.pinimg.com/originals/b6/03/b6/b603b668f5603d55ad32a12b9ac38d12.j
pg" alt="Chocolate Types" height="50px" width="50px">
    </div>
    <div class="column">
      <h2>Production & History</h2>
      <p>The journey from bean to bar involves several careful steps. Cacao beans
are fermented, dried, roasted, and ground to form a thick paste called chocolate
liquor — the base of all chocolate products.</p>
      <ul>
        <li><strong>Ancient Use:</strong> The Mayans and Aztecs used cacao to
make ceremonial drinks and even as currency.</li>
        <li><strong>European Introduction:</strong> Spanish explorers brought
chocolate to Europe in the 1500s, where sugar was added.</li>
```

```
        <li><strong>Modern Manufacturing:</strong> Industrialization allowed mass production of chocolate bars by the 1800s.</li>

        <li><strong>Today:</strong> Chocolate is a global favorite, appearing in desserts, beverages, cosmetics, and even health foods.</li>

      </ul>

<img src="http://1.bp.blogspot.com/i4LOH6cEbfs/T42vByl5VBI/AAAAAAAAAaM/1PSBfjtmdbo/s1600/box-of-chocolate11297440000LZg.jpg" alt="Chocolate History" height="5px" width="50px">

    </div>

  </div>

</body>

</html>
```

Output:



Chocolate

Chocolate Origins & Types

Chocolate originates from the seeds of the cacao tree, known as Theobroma cacao, which means "food of the gods". Native to Central and South America, cacao has been consumed by humans for over 3,000 years.

- **Dark Chocolate:** Rich in cocoa, less sugar, and strong flavor. Often considered the healthiest form.
- **Milk Chocolate:** A smooth blend of cocoa, milk, and sugar — loved for its creamy texture.
- **White Chocolate:** Contains cocoa butter, milk, and sugar, but no cocoa solids.
- **Ruby Chocolate:** Naturally pink, fruity-tasting chocolate introduced in 2017.

Production & History

The journey from bean to bar involves several careful steps. Cacao beans are fermented, dried, roasted, and ground to form a thick paste called chocolate liquor — the base of all chocolate products.

- **Ancient Use:** The Mayans and Aztecs used cacao to make ceremonial drinks and even as currency.
- **European Introduction:** Spanish explorers brought chocolate to Europe in the 1500s, where sugar was added.
- **Modern Manufacturing:** Industrialization allowed mass production of chocolate bars by the 1800s.
- **Today:** Chocolate is a global favorite, appearing in desserts, beverages, cosmetics, and even health foods.

Chocolate History

Week 3:

3 a) Create a web page to demonstrate Position Property in CSS.

Source Code:

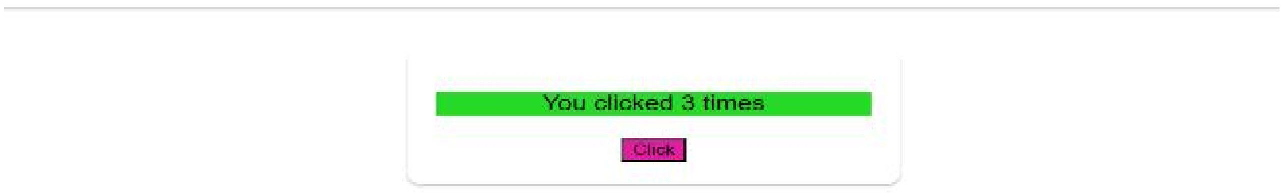
```
<!DOCTYPE html>
<html>
<head>
   <meta charset="UTF-8">
   <title>Program</title>
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <style>
      #box {
         width: 300px;
         margin: 40px auto;
         padding: 20px;
         border-radius: 10px;
         text-align: center;
         font-family: Arial, sans-serif;
         box-shadow: 0 2px 2px rgba(0,0,0,0.2);
      }

      #message {
         font-size: 18px;
         margin-bottom: 20px;
         transition: color 0.3s ease;
      }

      #button:hover {
         background-color: hsl(320, 80%, 50%);
```

```
        }
    </style>
</head>
<body>
    <div id="box">
        <p id="message">Message</p>
        <button id="button">Click</button>
    </div>
    <script src="script.js"></script>
</body>
</html>
```

Script.js

```
document.addEventListener("DOMContentLoaded", () => {
    const msg = document.getElementById("message");
    const btn = document.getElementById("button");

    let cb = 0, hue = 0;

    btn.addEventListener("click", () => {
        cb++;
        if (cb == 1) {
            msg.textContent = "Clicked once";
        } else if (cb == 2) {
            msg.textContent = "You clicked twice & dynamic";
        } else {
            msg.textContent = "You clicked " + cb + " times";
        }


        hue = (hue + 40) % 360;
        msg.style.backgroundColor = `hsl(${hue}, 70%, 50%)`;
    });
});
```

Output:



```
You clicked 3 times
```

3 b)Write a javascript program to validate registration page using regular expression

Source Code:

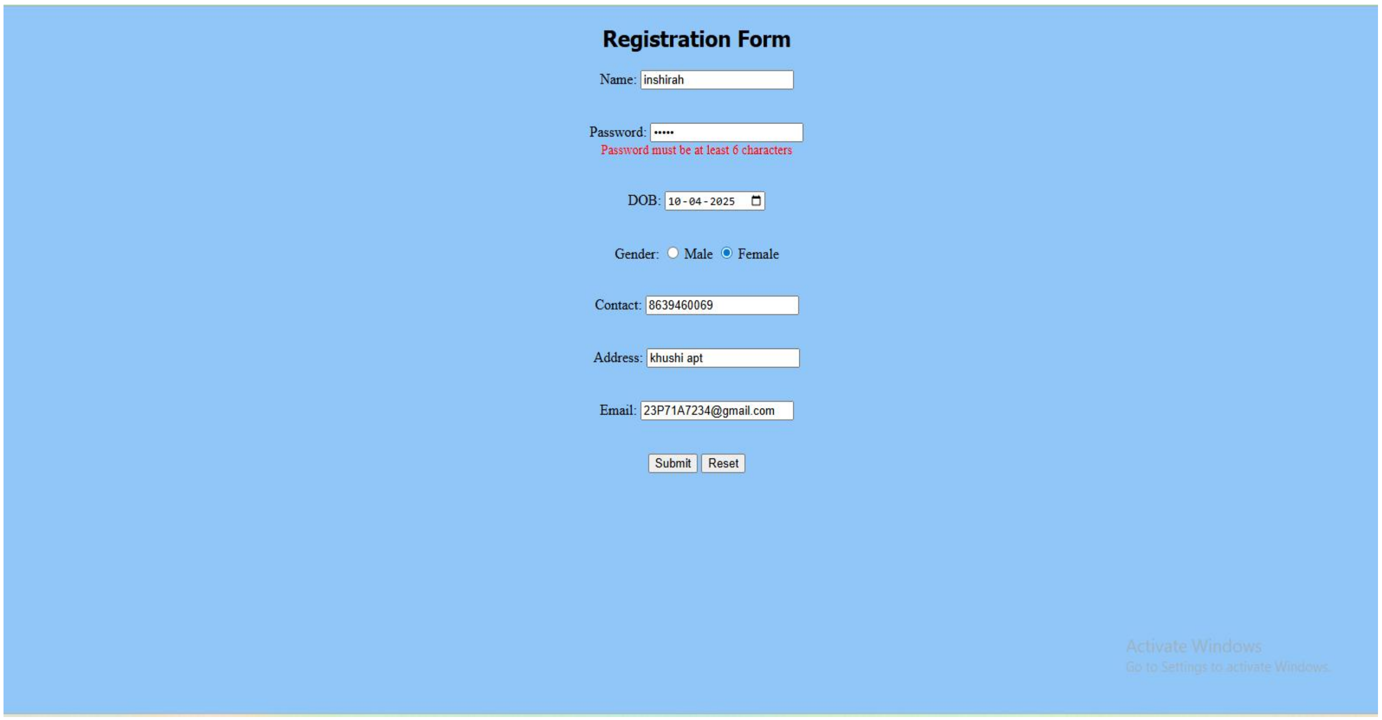
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <title>Registration Form</title>
  <style>
    h2 {
      text-align: center;
      font-family: Tahoma, sans-serif;
    }
    body {
      background-color: rgb(144, 199, 248);
    }
    form {
      text-align: center;
    }
    .error {
      color: red;
      font-size: 0.9em;
    }
  </style>
</head>
<body>
  <h2>Registration Form</h2>
  <form id="regForm" onsubmit="return validateForm()">
```

```
<label for="name">Name: </label>
<input type="text" id="name" name="name" />
<div id="nameError" class="error"></div>
<br /><br />

<label for="pwd">Password: </label>
<input type="password" id="pwd" name="pwd" />
<div id="pwdError" class="error"></div>
<br /><br />

<label for="birthday">DOB: </label>
<input type="date" id="birthday" name="birthday" />
<div id="dobError" class="error"></div>
<br /><br />

<label>Gender: </label>
<input type="radio" id="male" name="gender" value="MALE" />
<label for="male">Male</label>
<input type="radio" id="female" name="gender" value="FEMALE" />
<label for="female">Female</label>
<div id="genderError" class="error"></div>
<br /><br />

<label for="contact">Contact: </label>
<input type="text" id="contact" name="contact" />
<div id="contactError" class="error"></div>
<br /><br />
```

```
  <label for="address">Address: </label>
  <input type="text" id="address" name="address" />
  <div id="addressError" class="error"></div>
  <br /><br />

  <label for="email">Email: </label>
  <input type="email" id="email" name="email" />
  <div id="emailError" class="error"></div>
  <br /><br />

  <input type="submit" value="Submit" />
  <input type="reset" value="Reset" />
</form>

<script>
  function validateForm() {
    document.querySelectorAll('.error').forEach(e => e.textContent = '');

    let valid = true;

    const name = document.getElementById('name').value.trim();
    if (name === '') {
      document.getElementById('nameError').textContent = 'Name is required';
      valid = false;
    }

    const pwd = document.getElementById('pwd').value;
    if (pwd === '') {
```

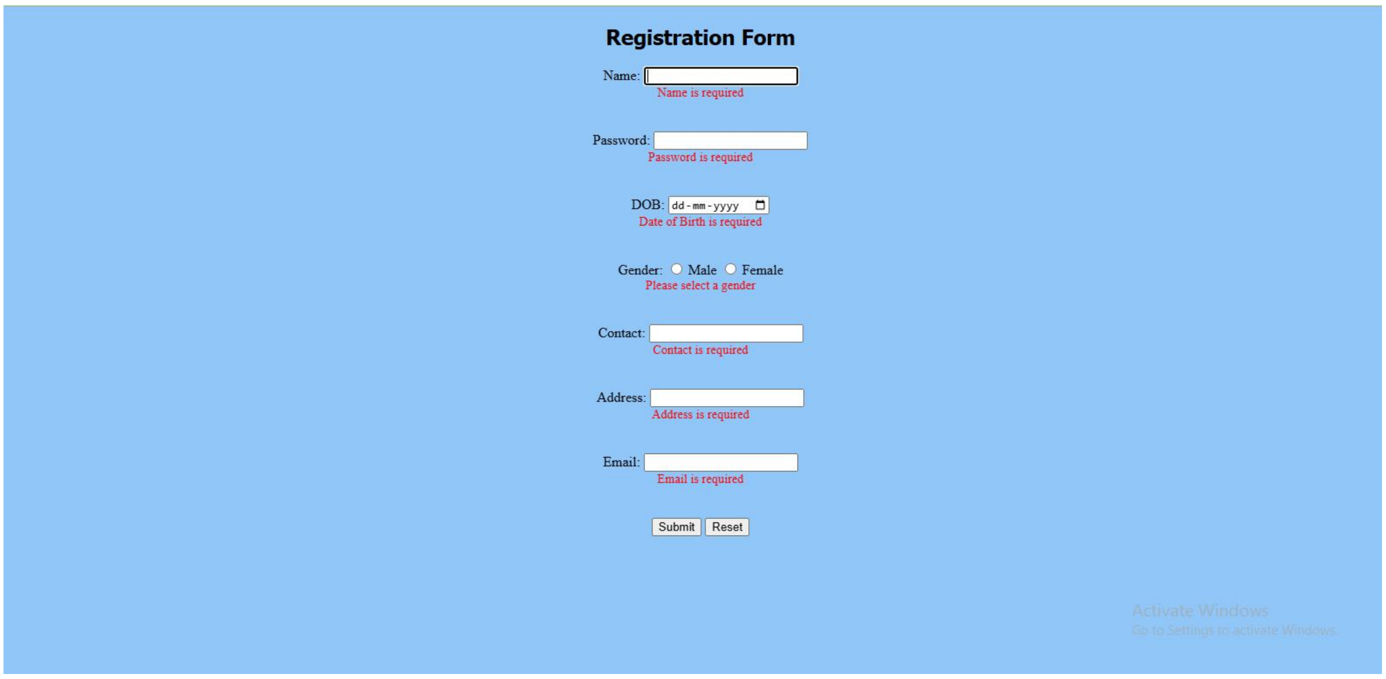
```
    document.getElementById('pwdError').textContent = 'Password is required';
    valid = false;
  } else if (pwd.length < 6) {
    document.getElementById('pwdError').textContent = 'Password must be at least
6 characters';
    valid = false;
  }



  const dob = document.getElementById('birthday').value;
  if (dob === '') {
    document.getElementById('dobError').textContent = 'Date of Birth is required';
    valid = false;
  }

  const genderMale = document.getElementById('male').checked;
  const genderFemale = document.getElementById('female').checked;
  if (!genderMale && !genderFemale) {
    document.getElementById('genderError').textContent = 'Please select a gender';
    valid = false;
  }

  const contact = document.getElementById('contact').value.trim();
  const contactRegex = /^\d{10}$/;
  if (contact === '') {
    document.getElementById('contactError').textContent = 'Contact is required';
    valid = false;
  } else if (!contactRegex.test(contact)) {
        document.getElementById('contactError').textContent = 'Contact must be 10 
digits';
        valid = false;
      }

      const address = document.getElementById('address').value.trim();
      if (address === '') {
        document.getElementById('addressError').textContent = 'Address is required';
        valid = false;
      }

      const email = document.getElementById('email').value.trim();
      if (email === '') {
        document.getElementById('emailError').textContent = 'Email is required';
        valid = false;
      } else {

        const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
        if (!emailRegex.test(email)) {
          document.getElementById('emailError').textContent = 'Invalid email format';
          valid = false;
        }
      }

      return valid;
    }
  </script>
</body>
</html>
```

Output:



Registration Form

Name:	inshirah
Password:	•••••
	Password must be at least 6 characters
DOB:	10-04-2025
Gender:	Male [✓ Female]
Contact:	8639460069
Address:	khushi apt
Email:	23P71A7234@gmail.com



Registration Form

Name:	
	Name is required
Password:	
	Password is required
DOB:	dd-mm-yyyy
	Date of Birth is required
Gender:	Male Female
	Please select a gender
Contact:	
	Contact is required
Address:	
	Address is required
Email:	
	Email is required

Week 4

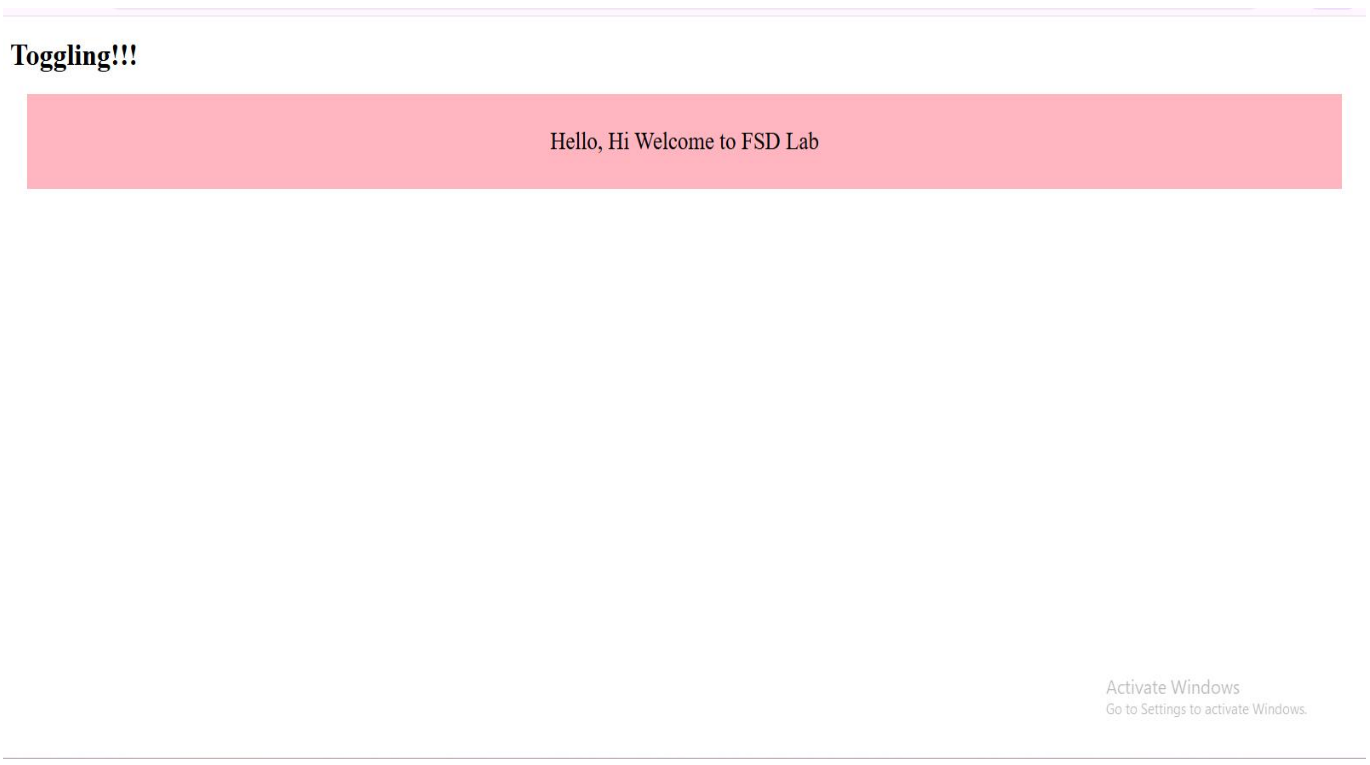
4a) Write a code to hide and show an element in a periodic interval without any action from the user using JQuery.

Source Code:

```
<!DOCTYPE html>

<html lang="en">

<head>

   <meta charset="UTF-8">

   <meta name="viewport" content="width=device-width, initial-scale=1.0">

   <title>Document</title>

   <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

   <style>

      .id1{

         text-align:center;

         background-color: lightpink;

         padding:10px;

         width:1340;

         margin: 20px;

         font-size:25px;

      }


   </style>

   <script>

      $(document).ready(function(){

         setInterval(function(){

            $(".id1").fadeToggle(2000);

         },3000);

      });
```

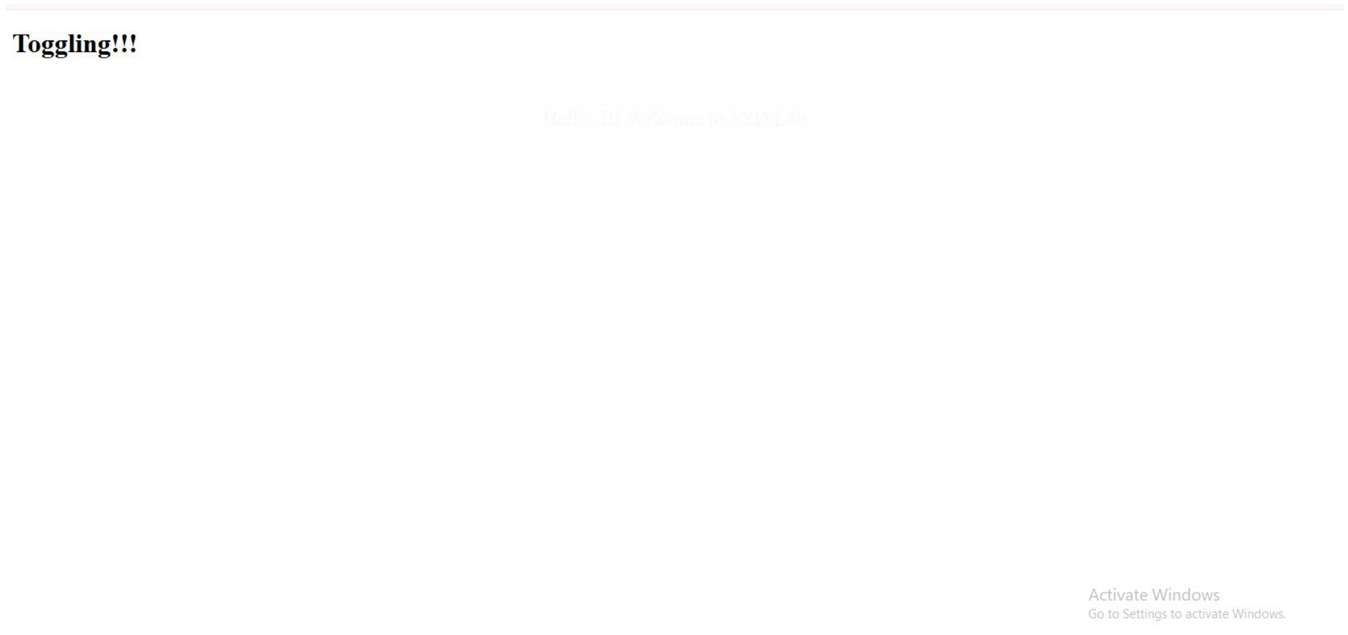
```
    </script>
</head>
<body>
    <h1> Toggling!!!</h1>
    <div class="id1">
        <p id="para"> Hello, Hi Welcome to FSD Lab</p>
    </div>
</body>
</html>
```

Output:



Toggling!!!

Hello, Hi Welcome to FSD Lab



Toggling!!!

Hello, Hi Welcome to FSD Lab

4b) Write a program to create and Build a star rating system using JQuery.

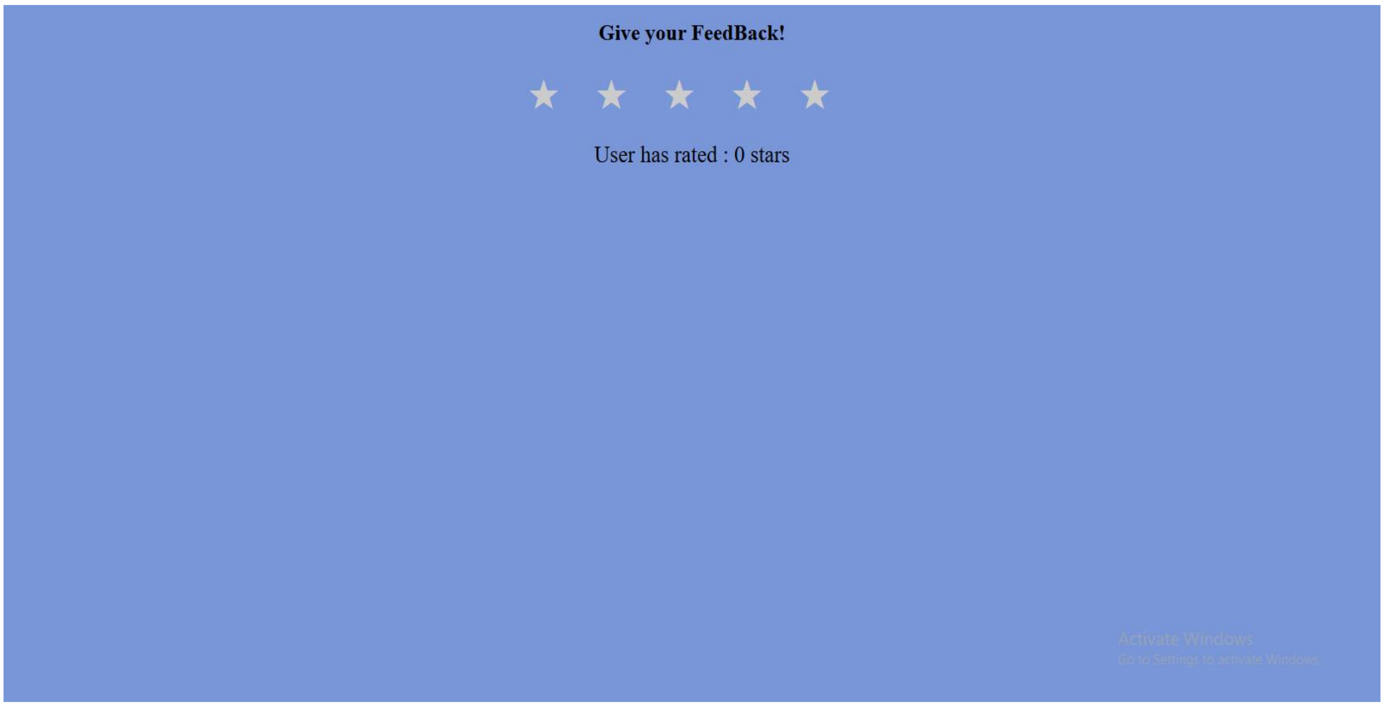
Source Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<title>FeedBack Rating System </title>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>
<style>
   #b{
      background-color: rgb(121, 150, 216);
      text-align: center;
   }
  .rating {
   display: inline-block;
   font-size: 45px;
   cursor: pointer;
   color: #ccc;
  }
  .rating .star {
   margin-right:30px;
   display: inline-block;
   transition: color 0.2s;
  }
  .rating .star.filled {
   color: rgb(244, 219, 76);
  }
```

```
</style>
</head>
<body id="b">
   <h2> Give your FeedBack! </h2>
<div class="rating">
  <span class="star" data-value="1">&#9733;</span>
  <span class="star" data-value="2">&#9733;</span>
  <span class="star" data-value="3">&#9733;</span>
  <span class="star" data-value="4">&#9733;</span>
  <span class="star" data-value="5">&#9733;</span>
</div>
<p style="font-size: 25px;">User has rated :  <span id="rating-value">0</span>
stars</p>
<script>
   $(function() {
   let selectedRating = 0;
   $('.rating .star').on('mouseenter', function() {
      let rating = $(this).data('value');
      highlightStars(rating);
   });

   $('.rating .star').on('mouseleave', function() {
      highlightStars(selectedRating);
   });

   $('.rating .star').on('click', function() {
      selectedRating = $(this).data('value');
      $('#rating-value').text(selectedRating);
      highlightStars(selectedRating);
```

```
    });
function highlightStars(rating) {
        $('.rating .star').each(function() {
        let starValue = $(this).data('value');
        if (starValue <= rating) {
            $(this).addClass('filled');
        } else {
            $(this).removeClass('filled');
        }
        });
    }
    });
</script>
</body>
</html>
```

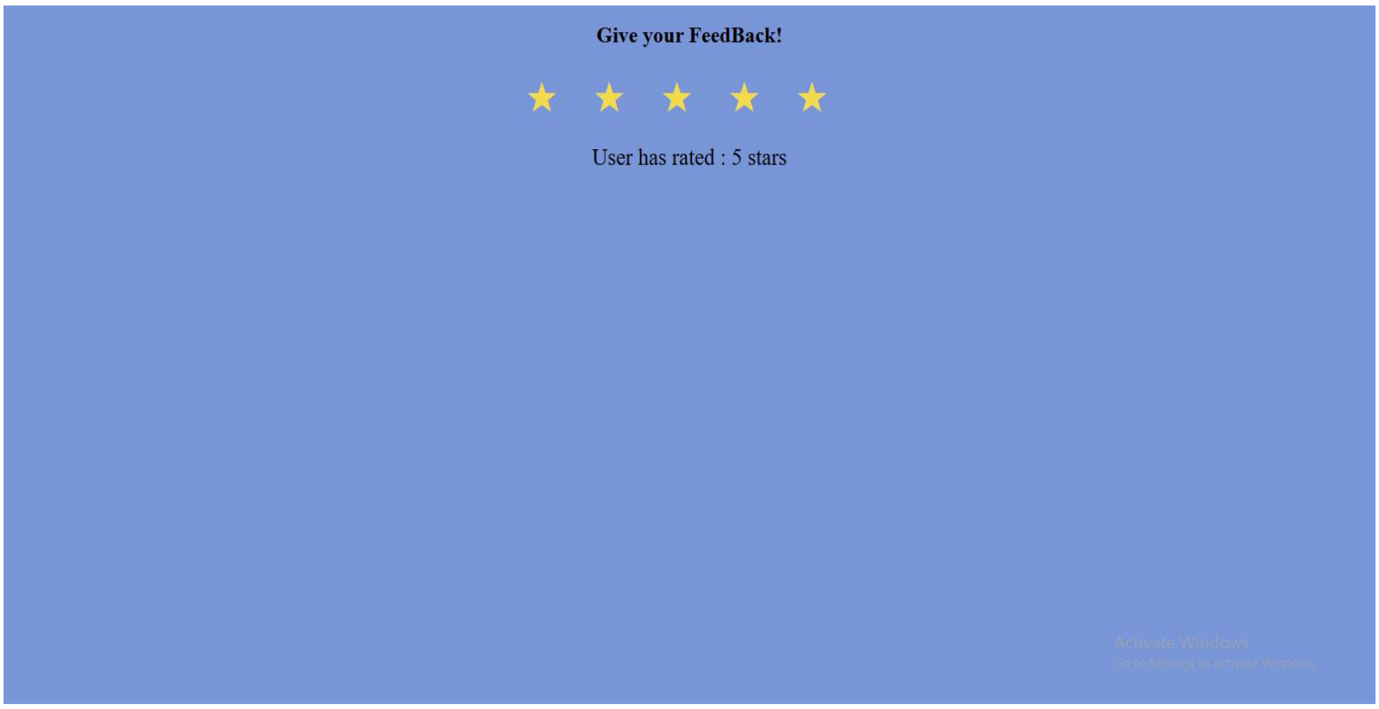
Output:



Give your FeedBack!

★ ★ ★ ★ ★

User has rated : 0 stars



Give your FeedBack!

★ ★ ★ ★ ★

User has rated : 5 stars

Week 5:

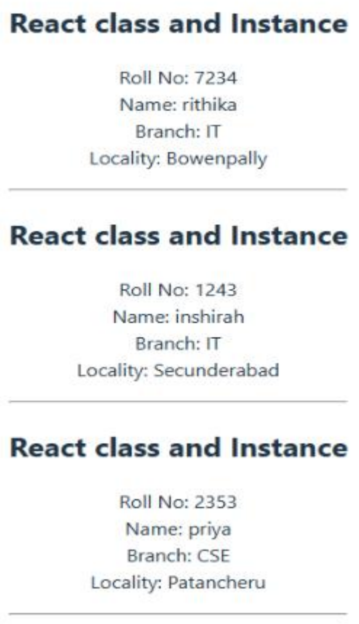
a) Write a program to demonstrate ReactJS Class and Instance.

Source Code:

App.jsx

```
import React,{component} from 'react'
import './App.css'

class Student extends React.Component{
  constructor(props){
    super(props);
    this.state={enrolled:true};
  }
  render(){
    return(
      <div>
        <h2>React class and Instance</h2>
        <p>
          Roll No: {this.props.rollno}<br />
          Name: {this.props.name}<br />
          Branch: {this.props.branch}<br />
          Locality: {this.props.locality}<br />
        </p>
      </div>
    );
  }
}

class App extends React.Component{
  render(){
    return(
    <div>
      <Student rollno="7234" name="rithika" branch="IT" locality="Bowenpally" />
      <hr/>
      <Student rollno="1243" name="inshirah" branch="IT" locality="Secunderabad" />
      <hr/>
      <Student rollno="2353" name="priya" branch="CSE" locality="Patancheru" />
      <hr/>
    </div>


    );
  }
}
export default App;
```

Output:



```
React class and Instance

Roll No: 7234
Name: rithika
Branch: IT
Locality: Bowenpally

React class and Instance

Roll No: 1243
Name: inshirah
Branch: IT
Locality: Secunderabad

React class and Instance

Roll No: 2353
Name: priya
Branch: CSE
Locality: Patancheru
```

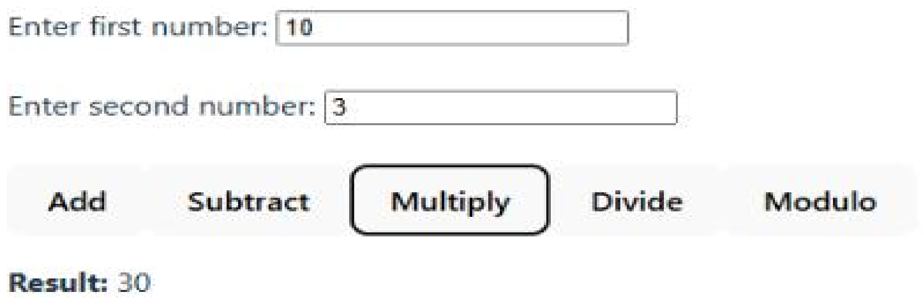
5b) Write a program to create a basic calculator to perform arithmetic operations using ReactJS.

Source Code:

App.jsx

```
import React,{useState} from 'react'
function App() {
  const[inp1,setInp1] = useState();
  const[inp2,setInp2] = useState();
  const[res,setRes] = useState();

  const add = () => {
    const ans=Number(inp1)+Number(inp2);
    setRes(ans) ;
    alert("Result is: "+ans);
  };
  const sub = () => {
    setRes(Number(inp1)-Number(inp2));
    //to display result in alert, we follow the syntax given below
    const ans=Number(inp1)-Number(inp2);
    setRes(ans) ;
    alert("Result is: "+ans);
  };
  const mul = () => {
    setRes(Number(inp1)*Number(inp2));
  };
  const div = () => {
    setRes(Number(inp1)/Number(inp2));
  };
  const mod = () => {
    setRes(Number(inp1)%Number(inp2));
  };
  return (
    <>
      <div id="input">
        <label htmlFor="inp1">Enter first number: </label>
        <input type="text" id="inp1" value={inp1} onChange={(e) =>
setInp1(e.target.value)}/>
        <br/>
        <br/>
        <label htmlFor="inp2" >Enter second number: </label>
        <input type="text" id="inp2" value={inp2} onChange={(e) =>
setInp2(e.target.value)}/>
        <br/>
         <br />
      </div>

      <div id="opt">
        <button onClick={add}>Add</button>
        <button onClick={sub}>Subtract</button>
        <button onClick={mul}>Multiply</button>
        <button onClick={div}>Divide</button>
        <button onClick={mod}>Modulo</button>
        <br />
      </div>

      <div id="output">
        <p><strong>Result:</strong> {res}</p>
      </div>
      </>


  );
}

export default App;
```

Output:



Enter first number: 10

Enter second number: 3

Result: 30

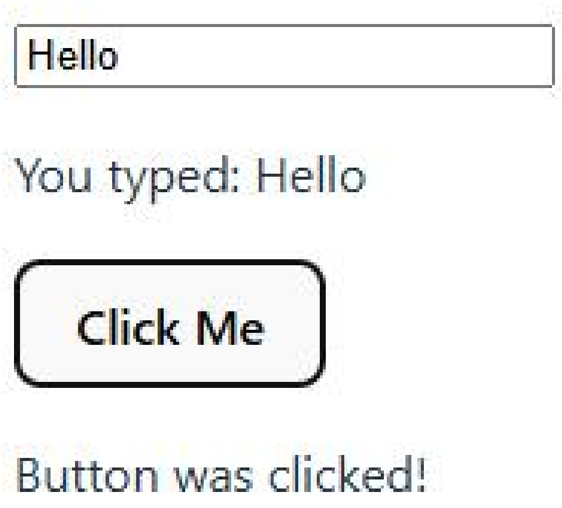
Week 6:

a) Demonstrate simple event handling examples using ReactJS.

Source Code:

```
import React,{ useState } from 'react'
function App() {
  const [input, setInput] = useState();
  const [msg, setMsg] = useState();
  const click = () => {
    setMsg('Button was clicked!');
  };

  const inputChange = (e) => {
    setInput(e.target.value);
  };

  return (
    <div style={{ padding: '20px' }}>

      <input
        type="text"
        value={input}
        onChange={inputChange}
      />
      <p>You typed: {input}</p>

      <button onClick={click}>Click Me</button>
      <p>{msg}</p>
    </div>
  );
}
export default App;
```

Output:



```
Hello
You typed: Hello
Click Me
Button was clicked!
```

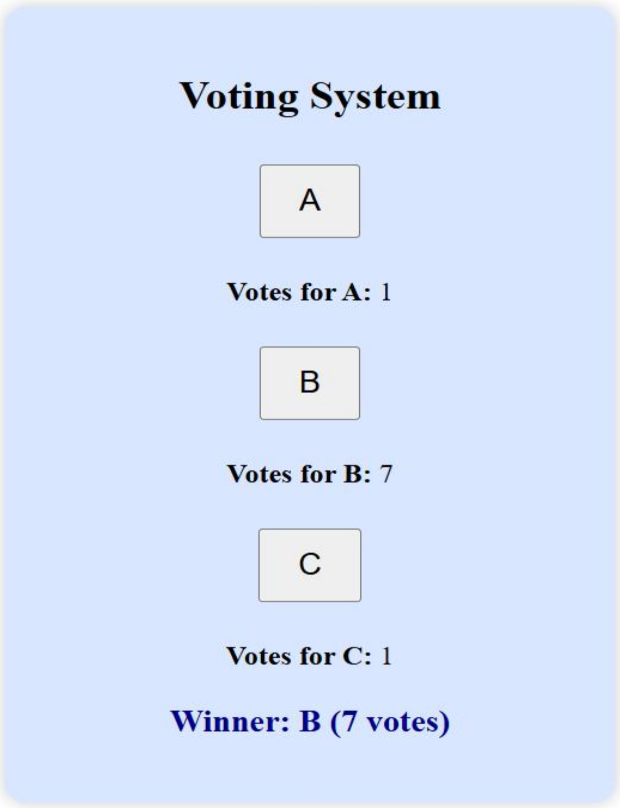
b) Write a program to create a simple voting application system using ReactJS.

Source Code:

App.jsx

```
import { useState } from 'react'
import reactLogo from './assets/react.svg'
import viteLogo from '/vite.svg'
import './App.css'
function App() {
  const [c1, setC1] = useState(0)
  const [c2, setC2] = useState(0)
  const [c3, setC3] = useState(0)

  return (
    <>
      <div id="inp">
        <button id="b1" onClick={() => setC1((add1) => c1 + 1)}>
          A
        </button>
          <p><strong>votes for A is </strong>{c1} </p>
        <button id="b2" onClick={() => setC2((add1) => c2 + 1)}>
          B
        </button>
          <p><strong>votes for B is </strong>{c2} </p>
        <button id="b3" onClick={() => setC3((add1) => c3 + 1)}>
         C
        </button>
          <p><strong>votes for C is </strong>{c3} </p>
      </div>
    </>
  )
}
export default App
```

Output:



Voting System

Votes for A: 1

Votes for B: 7

Votes for C: 1

Winner: B (7 votes)

Week 7:

a) Create a webpage to display “Hello World!” using SERVLET.

Source Code:

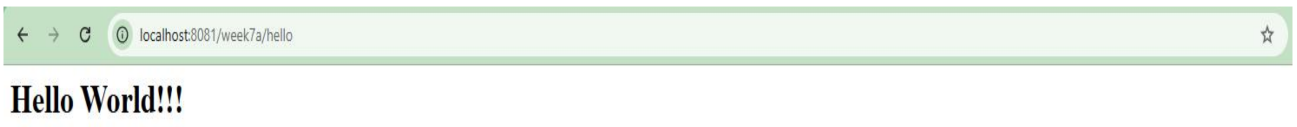
HelloWorld.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class HelloWorld extends HttpServlet{
   //private String msg;
   /*public void init() throws ServletException{
      msg="Hello World!";
   } */
   public void doGet(HttpServletRequest req,HttpServletResponse res) throws
ServletException, IOException{
      res.setContentType("text/html");
      PrintWriter out=res.getWriter();
      out.println("<html><body>");
      out.println("<h1> Hello World!!!</h1>");
      out.println("</body></html>");
   }
   public void destroy(){
   }
}
```

web.xml

```
<web-app>
    <servlet>
        <servlet-name>HelloWorld</servlet-name>
        <servlet-class>HelloWorld</servlet-class>
    </servlet>
    <servlet-mapping>
            <servlet-name>HelloWorld</servlet-name>
            <url-pattern>/hello</url-pattern>
   </servlet-mapping>
</web-app>
```

Output:



Hello World!!!

7 b) Implement a web application using SERVLET, which takes a name as input and on submitting it, shows a hello page. It shows start time at the right top comer of the page and provides a logout button. On clicking logout button, it should show a logout page with Thank You message with the duration of usage (hint: Use session to store name and time).

Source Code:

LoginServlet.java

```
import java.io.IOException;
import java.io.PrintWriter;
import java.util.Date;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
public class LoginServlet extends HttpServlet {
   @Override
   protected void doGet(HttpServletRequest request, HttpServletResponse response)
         throws ServletException, IOException {
      // Allow GET requests (browser URL or redirect) to work
      doPost(request, response);
   }

   @Override
   protected void doPost(HttpServletRequest request, HttpServletResponse
response)
         throws ServletException, IOException {

      response.setContentType("text/html");
      PrintWriter out = response.getWriter();
```

```
        String username = request.getParameter("username");
        if (username == null || username.trim().isEmpty()) {
            out.println("<html><body>");
            out.println("<h3>Please enter a valid username.</h3>");
            out.println("<a href='login.html'>Go back</a>");
            out.println("</body></html>");
            return;
        }
        // Create session and set attributes
        HttpSession session = request.getSession();
        session.setAttribute("username", username);
        session.setAttribute("startTime", new Date());
        out.println("<html><body>");
        out.println("<h2>Welcome, " + username + "!</h2>");
        out.println("<p>You can now <a href='logout'>Logout (GET)</a></p>");
        out.println("<form action='logout' method='post'>");
        out.println("<input type='submit' value='Logout (POST)'>");
        out.println("</form>");
        out.println("</body></html>");
    }
}
```

LogOutServlet.java

```
import java.io.IOException;
import java.io.PrintWriter;
import java.util.Date;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
public class LogOutServlet extends HttpServlet {
   @Override
   protected void doGet(HttpServletRequest request, HttpServletResponse response)
         throws ServletException, IOException {
      // Allow GET to work (so logout link works)
      doPost(request, response);
   }
   @Override
   protected void doPost(HttpServletRequest request, HttpServletResponse
response)
         throws ServletException, IOException {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();
      HttpSession session = request.getSession(false); // don't create new session
      if (session != null) {
         String name = (String) session.getAttribute("username");
         Date startTime = (Date) session.getAttribute("startTime");
         Date endTime = new Date();
         long duration = (endTime.getTime() - startTime.getTime()) / 1000; // seconds
         out.println("<html><body>");
         out.println("<h2>Thank you, " + name + "!</h2>");
         out.println("<p>You used this application for " + duration + " seconds.</p>");
         out.println("</body></html>");
         session.invalidate(); // destroy session
      } else {
         out.println("<html><body>");
```

```
            out.println("<h3>No active session found!</h3>");
            out.println("</body></html>");
        }
    }
}
```

web.xml

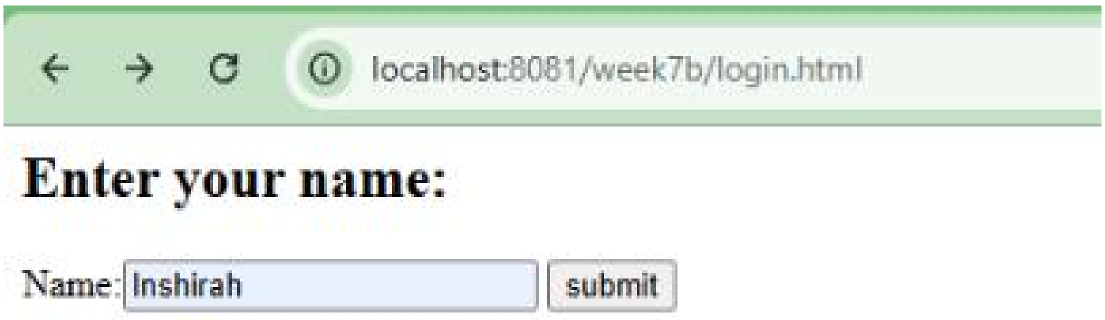
```
<web-app>
    <servlet>
        <servlet-name>LoginServlet</servlet-name>
        <servlet-class>LoginServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>LoginServlet</servlet-name>
        <url-pattern>/login</url-pattern>
     </servlet-mapping>
     <servlet>
          <servlet-name>LogOutServlet</servlet-name>
          <servlet-class>LogOutServlet</servlet-class>
    </servlet>
    <servlet-mapping>
         <servlet-name>LogOutServlet</servlet-name>
         <url-pattern>/logout</url-pattern>
    </servlet-mapping>
</web-app>
```

login.html

```
<html>
```

```
<head>
<title>web page</title>
</head>
<body>
<h2>Enter your name:</h2>
<form action="login" method="post">
Name:<input type="text" name="username" required/>
<input type="submit" value="submit"/>
</form>
</body>
</html>
```

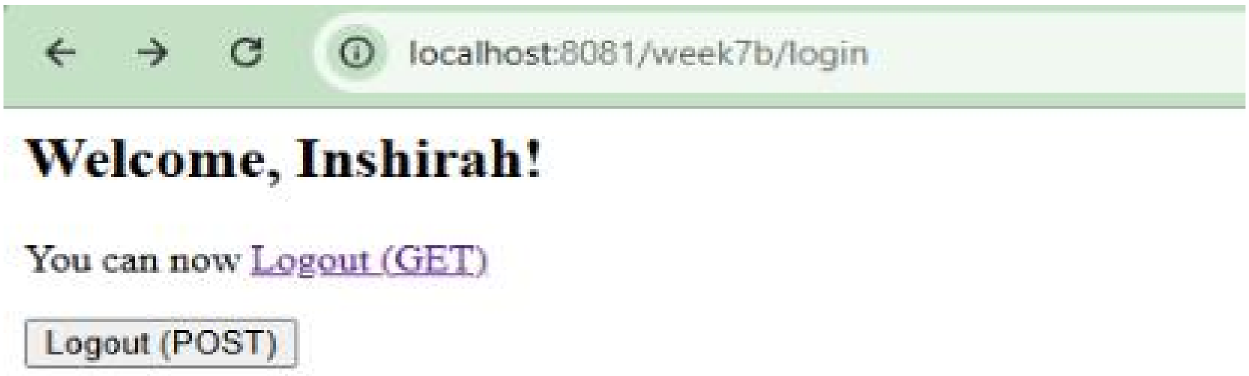
Output:



localhost:8081/week7b/login.html

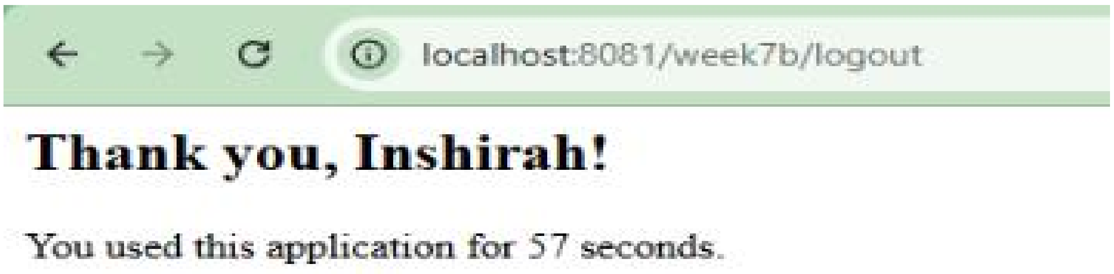
Enter your name:

Name: Inshirah


localhost:8081/week7b/login

Welcome, Inshirah!

You can now Logout (GET)


localhost:8081/week7b/logout

Thank you, Inshirah!

You used this application for 57 seconds.

Week 8:

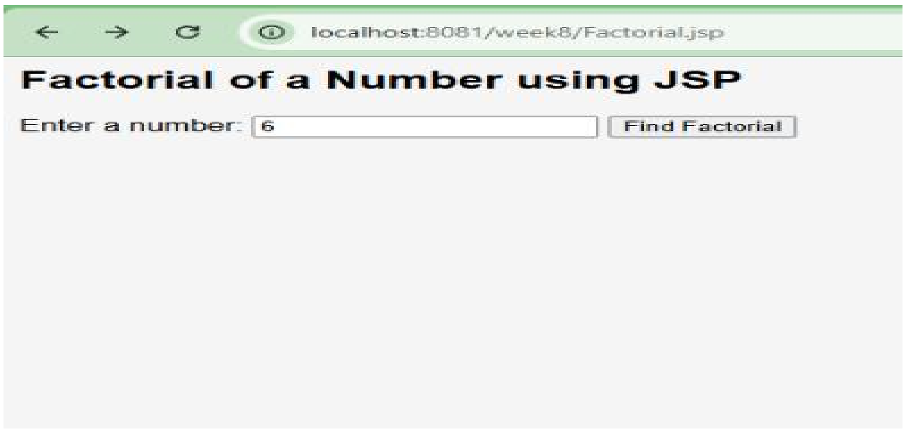
a) Write a JSP program to find a factorial of the given number.

Source Code:

factorial.jsp

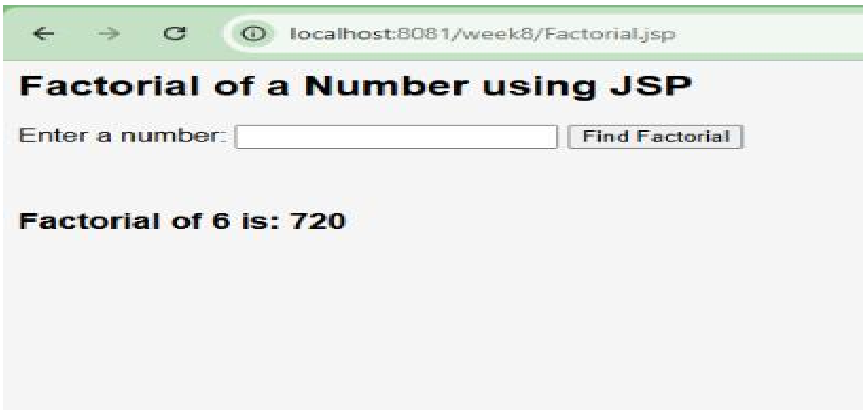
```
<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
<html>
<head>
   <title>Factorial Calculator</title>
</head>
<body style="font-family: Arial; background-color:#f5f5f5;">
   <h2>Factorial of a Number using JSP</h2>
   <form method="post" action="">
      Enter a number:
      <input type="text" name="num" required>
      <input type="submit" value="Find Factorial">
   </form>
   <br>
   <%
      // Get the number entered by the user
      String numStr = request.getParameter("num");
      if (numStr != null && !numStr.isEmpty()) {
         try {
            int num = Integer.parseInt(numStr);
            long fact = 1;
            for (int i = 1; i <= num; i++) {
               fact *= i;
            }
            out.println("<h3>Factorial of " + num + " is: " + fact + "</h3>");
        } catch (NumberFormatException e) {
            out.println("<p style='color:red;'>Please enter a valid integer!</p>");
        }
    }
%>
</body>
</html>
```

Output:



Factorial of a Number using JSP

Enter a number: 6 Find Factorial



Factorial of a Number using JSP

Enter a number: Find Factorial

Factorial of 6 is: 720

Week 9:

a) Demonstrate a simple example of Spring web MVC framework.

Source Code:

Step 1 – Create a New Maven Web Project

1. Open Eclipse → **File** → **New** → **Maven Project**
2. Check **Create a simple project (skip archetype selection)** → Next
3. Fill in:
    - **Group Id:** com.example.MySampleApp
    - **Artifact Id:** MySampleApp
4. Click **Finish**

Step 2 – Add Spring Dependencies

Open pom.xml and replace everything inside with this (from your PDF):

```
<project>
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.example.MySampleApp</groupId>
  <artifactId>MySampleApp</artifactId>
  <version>1.0</version>
  <packaging>war</packaging>
<properties>
    <java.version>1.8</java.version>
    <spring.version>3.1.0.RELEASE</spring.version>
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-context</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-orm</artifactId>
      <version>${spring.version}</version>
    </dependency>
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>servlet-api</artifactId>
      <version>2.4</version>
      <scope>provided</scope>
    </dependency>
    <dependency>
      <groupId>javax.servlet.jsp</groupId>
      <artifactId>jsp-api</artifactId>
      <version>2.1</version>
      <scope>provided</scope>
    </dependency>
  </dependencies>
</project>
```

Step 3 – Add web.xml

Create file:
src/main/webapp/WEB-INF/web.xml

```
<web-app version="3.0"
        xmlns="http://java.sun.com/xml/ns/j2ee"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
                  http://java.sun.com/xml/ns/j2ee/web-app_3_0.xsd">

  <display-name>MySampleApp</display-name>

  <servlet>
    <servlet-name>SpringDispatcher</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-
class>
    <load-on-startup>1</load-on-startup>
  </servlet>

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

Step 4 – Spring Configuration File

Create file:
src/main/webapp/WEB-INF/SpringDispatcher-servlet.xml

```
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:mvc="http://www.springframework.org/schema/mvc"
xmlns:context="http://www.springframework.org/schema/context"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
      xsi:schemaLocation="
        http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans.xsd
        http://www.springframework.org/schema/mvc
        http://www.springframework.org/schema/mvc/spring-mvc.xsd
        http://www.springframework.org/schema/context
        http://www.springframework.org/schema/context/spring-context.xsd">

  <context:component-scan base-package="com.example.MySampleApp.controller"
/>
  <mvc:annotation-driven />

  <bean id="viewResolver"
        class="org.springframework.web.servlet.view.InternalResourceViewResolver">
    <property name="prefix" value="/WEB-INF/views/" />
    <property name="suffix" value=".jsp" />
  </bean>
</beans>
```

Step 5 – Create Controller Classes

Path: src/main/java/com/example/MySampleApp/controller

HomeController.java

```
package com.example.MySampleApp.controller;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
@Controller
public class HomeController {
    @RequestMapping("/")
    public String homePage() {
        return "home";
    }
}
```

LoginController.java

```
package com.example.MySampleApp.controller;
import javax.servlet.http.HttpServletRequest;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.RequestMapping;
@Controller
public class LoginController {
    @RequestMapping("/login")
    public String showLoginPage() {
        return "login";
    }
    @RequestMapping("/user")
    public String authenticate(HttpServletRequest request, Model model) {
        String username = request.getParameter("uname");
        String password = request.getParameter("pwd");
        if ("admin".equals(password)) {
            model.addAttribute("message", "Welcome " + username + "!!");
            return "welcome";
        } else {
            model.addAttribute("message", "Invalid User!!");
            return "errorPage";
        }
    }
}
```

Step 6 – Create JSP Views

Create folder:
src/main/webapp/WEB-INF/views

Add files:

home.jsp

```
<h1>Hello World!</h1>
<p>This is the home page!</p>
<a href="login">Go to Login Page</a>
```

login.jsp

```
<h2>Login Form</h2>
<form action="user">
  Username: <input type="text" name="uname"/><br><br>
  Password: <input type="password" name="pwd"/><br><br>
  <button type="submit">Login</button>
</form>
```

welcome.jsp

```
<h1 style="color:green">${message}</h1>
```

errorPage.jsp

```
<h1 style="color:red">${message}</h1>
<a href="login">Try Again</a>
```

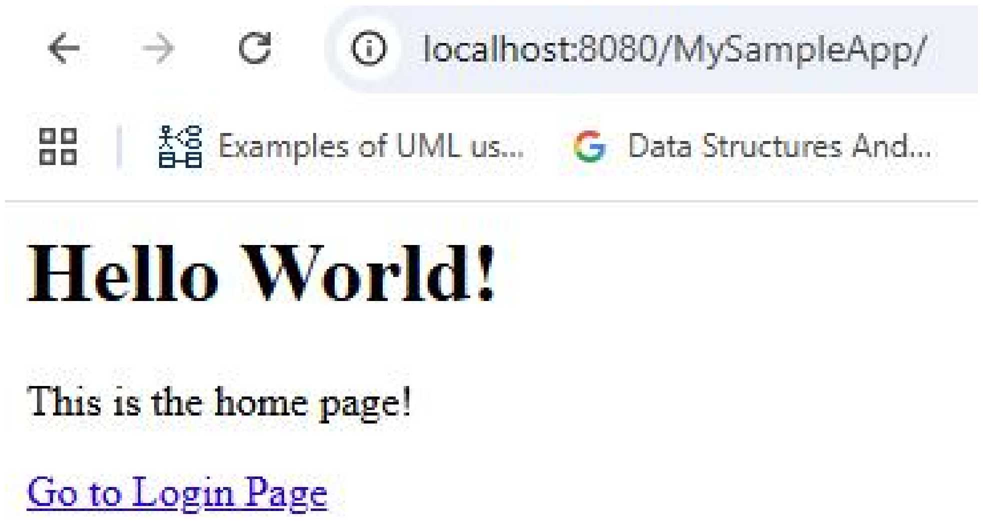
Step 7 – Run the App

1. Right-click project → **Run As** → **Run on Server**

2. Select **Apache Tomcat 9/10**
3. Open browser →
   - http://localhost:8080/MySampleApp/ → Home page
   - Click *Go to Login Page* → fill in details
   - Password admin → Welcome message
   - Any other password → Error page

Done! You've run your **first Spring MVC application**.

Output:



Hello World!

This is the home page!

Go to Login Page

Week 11 and 12:

CASE STUDY-1: Create a Chat module/Interface using HTML CSS and JavaScript. The chat interface primarily consists of two segments: the message header and the chat box.

Message-Header- The message header resides at the top of the chat box. It includes the user's name, avatar or profile image, and the user's last seen. Last seen is the last time the user was active.

The Chat-Box- The chat box consists of the message page and the message bottom sections.

● Message page-The message page consists of incoming and outgoing messages, as well as the avatars of the senders. It also displays the time at which each message is sent.

● The Message-Bottom-This section contains an input field where the user can type in the messages and a send button to send them.

Source Code:

script.js

```
const chatBox = document.getElementById('chatBox');

const input = document.getElementById('messageInput');

const lastSeen = document.getElementById('lastSeen');

const currentUser = {

  name: 'You',

  avatar: "av6.jpg",

};

const otherUser = {

  name: 'Rithika',

  avatar: "girl.jpg",

};

function formatTime(date = new Date()) {

  return date.toLocaleTimeString([], { hour: '2-digit', minute: '2-digit' });

}
```

```
function updateLastSeen() {
  const now = new Date();
  lastSeen.textContent = `${otherUser.name} last seen: ${formatTime(now)}`;
}
function addMessage(text, sender) {
  const messageElem = document.createElement('div');
  messageElem.classList.add('message');
  if (sender === currentUser.name) {
    messageElem.classList.add('outgoing');
  } else {
    messageElem.classList.add('incoming');
  }
  const avatarURL = sender === currentUser.name ? currentUser.avatar :
otherUser.avatar;
  messageElem.innerHTML = `
    <img src="${avatarURL}" alt="${sender} Avatar" class="avatar" />
    <div class="message-content">
      ${escapeHTML(text)}
      <div class="message-time">${formatTime()}</div>
    </div>
  `;
  chatBox.appendChild(messageElem);
  chatBox.scrollTop = chatBox.scrollHeight;
  if (sender === otherUser.name) {
    updateLastSeen();
  }
}
function escapeHTML(str) {
  const div = document.createElement('div');
  div.textContent = str;
  return div.innerHTML;
}
 async function getReply(userMsg) {
   const msg = userMsg.toLowerCase();
   if (msg.includes('hello') || msg.includes('hi')|| msg.includes('hey')) {
     return 'Hey! Nice to hear from you.';
    }
    if (msg.includes('how are you')) {
      return "I'm good, thanks! How about you?";
    }
    if (msg.includes('help')) {
      return 'Of course! What do you need help with?';
    }

    if (msg.includes("temperature") || msg.includes("temp")) {
      const temp = await getCurrentTemperature();
      if (!temp) {
        return "Sorry, I couldn't get the temperature right now.";
      }
      return `Current temperature is ${temp}°C.`;
    }
    // Current date and time
    if (msg.includes('date')) {
      const now = new Date();
      const cal = now.toLocaleDateString(undefined, { weekday: 'long', year: 'numeric',
month: 'long', day: 'numeric' });
      return `Today's date is ${cal}.`;
    }
    if(msg.includes('time')){
      const now = new Date();
      const timeStr = now.toLocaleTimeString([], { hour: '2-digit', minute: '2-digit' });
      return `Time is ${timeStr}.`;
    }
      if (msg.includes('thank')) {
    return "You're welcome!";
  }
  if (msg.includes('bye')) {
    return 'Talk soon! Take care.';
  }
  if (msg.includes('what are you doing?')) {
    return 'Nothing, I just finished my work.\n What about you??';
  }
  if (msg.includes('samhitha')) {
    return 'ohh! i knoww herr, she is your friend. She is one of the most important
persons in your lifeee. But she doesnt agree for pani puri, whenever u ask her. haha';
  }

  return "👍";
}

function sendMessage() {
  const msg = input.value.trim();
  if (!msg) return;
  addMessage(msg, currentUser.name);
  input.value = '';
  setTimeout(async () => {
  const reply = await getReply(msg);
  addMessage(reply, otherUser.name);
}, 1200);
}
```

index.html

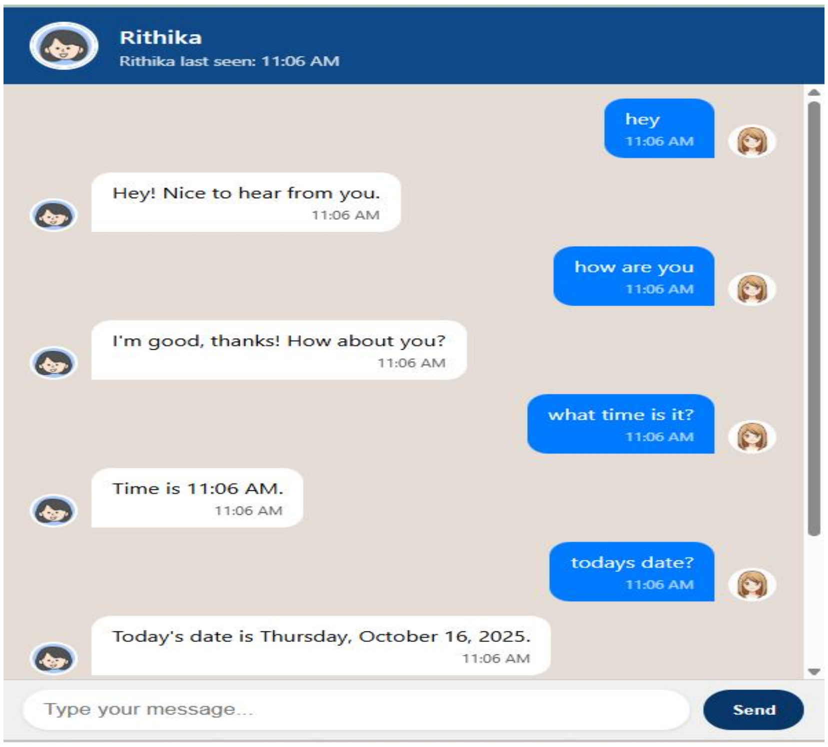
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<title>Full Screen Chat Interface</title>
<link rel="stylesheet" href="styles.css" />
</head>
<body>
  <div class="chat-container">
    <!-- Message Header -->
    <header class="message-header">
      <img src="girl.jpg" alt="User Avatar" class="avatar" />
      <div class="user-info">
        <h3>Rithika</h3>
        <p id="lastSeen">Last seen: just now</p>
      </div>
    </header>
    <!-- Chat Box -->
    <main class="chat-box" id="chatBox"></main>
    <!-- Message Bottom -->
    <footer class="message-bottom">
      <input
        type="text"
        id="messageInput"
        placeholder="Type your message..."
        autocomplete="off"
        onkeypress="if(event.key === 'Enter'){ sendMessage(); }"
      />
      <button onclick="sendMessage()">Send</button>
    </footer>
  </div>
<script src="script.js"></script>
</body>
</html>
```

styles.css

```
/* Reset */
* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}

body, html {
  height: 100%;
  font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
  background: #f0f2f5;
```

```
}

.chat-container {
  display: flex;
  flex-direction: column;
  height: 100vh; /* full screen height */
  max-width: 600px;
  margin: 0 auto;
  border: 1px solid #ccc;
  background: #fff;
}

/* Message Header */
.message-header {
  display: flex;
  align-items: center;
  padding: 15px 20px;
  background-color: #104a88;
  color: white;
  box-shadow: 0 2px 4px rgba(0,0,0,0.1);
}

.message-header .avatar {
  width: 50px;
  height: 50px;
  border-radius: 50%;
  margin-right: 15px;
  border: 2px solid white;
  object-fit: cover;
}

.user-info h3 {
  font-weight: 600;
  font-size: 1.2rem;
}

.user-info p {
  font-size: 0.85rem;
  opacity: 0.85;
  margin-top: 3px;
}

/* Chat Box */
.chat-box {
  flex-grow: 1;
  padding: 15px 20px;
  overflow-y: auto;
  background: #e5ddd5;
  display: flex;
  flex-direction: column;
}

/* Messages */
.message {
  display: flex;
  margin-bottom: 15px;
  max-width: 80%;
}

.message .avatar {
  width: 35px;
  height: 35px;
  border-radius: 50%;
  object-fit: cover;
  margin-top: auto;
}

/* Incoming messages (left) */
.message.incoming {
  flex-direction: row;
}

.message.incoming .message-content {
  background-color: white;
  color: black;
  border-radius: 15px 15px 15px 0;
  padding: 10px 15px;
  margin-left: 10px;
  position: relative;
  word-wrap: break-word;
}

/* Outgoing messages (right) */
.message.outgoing {
  flex-direction: row-reverse;
  margin-left: auto;
}

.message.outgoing .message-content {
  background-color: #007bff;
  color: white;
  border-radius: 15px 15px 0 15px;
  padding: 10px 15px;
  margin-right: 10px;
  position: relative;
  word-wrap: break-word;
}

/* Timestamp */
.message-time {
  font-size: 0.75rem;
  opacity: 0.7;
  margin-top: 5px;
  text-align: right;
  user-select: none;
}

/* Message Bottom */
.message-bottom {
  display: flex;
  padding: 10px 15px;
  background: #f0f0f0;
  border-top: 1px solid #ccc;
}

.message-bottom input[type="text"] {
  flex-grow: 1;
  border: none;
  border-radius: 20px;
  padding: 12px 15px;
  font-size: 1rem;
  outline: none;
  background: white;
  box-shadow: 0 1px 3px rgba(0,0,0,0.1);
}

.message-bottom button {
  background-color: #093769;
  border: none;
  color: white;
  font-weight: bold;
  padding: 12px 20px;
  margin-left: 10px;
  border-radius: 20px;
  cursor: pointer;
  transition: background-color 0.25s ease;
}
.message-bottom button:hover {
  background-color: #093769;
}
```

Output:



Rithika
Rithika last seen: 11:06 AM

hey
11:06 AM

Hey! Nice to hear from you.
11:06 AM

how are you
11:06 AM

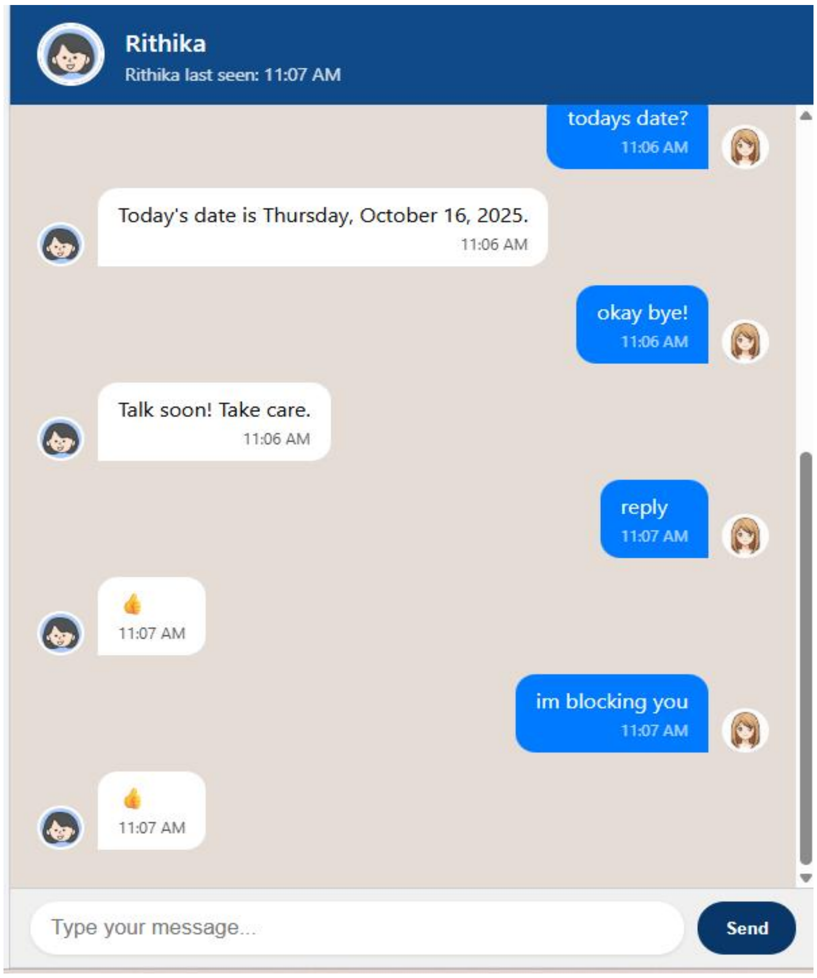
I'm good, thanks! How about you?
11:06 AM

what time is it?
11:06 AM

Time is 11:06 AM.
11:06 AM

todays date?
11:06 AM

Today's date is Thursday, October 16, 2025.
11:06 AM



Rithika
Rithika last seen: 11:07 AM

todays date?
11:06 AM

Today's date is Thursday, October 16, 2025.
11:06 AM

okay bye!
11:06 AM

Talk soon! Take care.
11:06 AM

reply
11:07 AM

👍
11:07 AM

im blocking you
11:07 AM

👍
11:07 AM