Experiment No. 02

Title: Design Website forms using HTML5.

Batch:SY-IT(B3) **Roll No.:16010423076** **Experiment No.: 2**

Aim: To design website forms to accept data from the user through the HTML 5.0 form elements.

Resources needed:HTML 5.0 editor

Theory:

Basics of HTML Forms:

HTML forms contain **form elements**. Form elements are different types of input elements, checkboxes, radio buttons, submit buttons, and more.

For Example:
<input type="text"> defines a one-line input field for **text input.**
<input type="radio"> defines a **radio button**.

The other input elements are:

- Checkboxes
- Button
- Textarea
- Select

The different attributes of forms are:

The Action Attribute: The **action attribute** defines the action to be performed when the form is submitted.The common way to submit a form to a server, is by using a submit button.Normally, the form is submitted to a web page on a web server.
For example:
<form**action="action_page.php**">

The Method Attribute:Themethod attribute **specifies the HTTP method (**GET **or** POST**) to be used when submitting the forms:**
For example:
<formaction="action_page.php"**method="get"**> or <form action="action_page.php" **method="post"**>

A history of HTML5 forms:

The forms section of HTML5 was originally a specification titled Web Forms 2.0 that added new types of controls for forms. Started by Opera and edited by then-Opera employee Ian Hickson, it was submitted to the W3C in early 2005. The work was initially carried out under the W3C. It was then combined with the Web Applications 1.0 specification to create the basis of the breakaway Web Hypertext Application Technology Working Group (WHATWG) HTML5 specification.

Using HTML5 design principles

One of the best things about HTML5 forms is that you can use almost all of these new input types and attributes right now. They don’t even need any shivs, hacks, or workarounds. That isn’t to say they’re all “supported” right now, but they do cool things in modern browsers that do support them-and degrade gracefully in browsers that don’t understand them. This is thanks to HTML5’s design principles. In this instance we’re specifically referring to the principle of graceful degradation. In essence, this means that there’s no excuse for not using these features right now. In fact, it means you’re ahead of the curve.

HTML5 form attributes

There are 14 new attributes provided by HTML5

placeholder
autocomplete
pattern
multiple
formnovalidate
formaction
formmethod

autofocus
required
list
novalidate
form
formenctype
formtarget

1. placeholder

First up is the placeholder attribute, which allows us to set placeholder text as we would currently do in HTML4 with the value attribute. It should only be used for short descriptions. For anything longer, use the title attribute. The difference from HTML4 is that the text is only displayed when the field is empty and hasn’t received focus. Once the field receives focus (e.g., you click or tab to the field), and you begin to type, the text simply disappears. It’s very similar to the search box you see in Safari (see Figure 1).

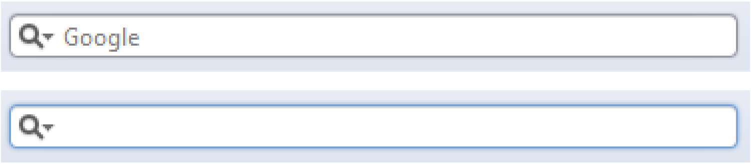


Figure 1. Browser search box in Safari without and with focus

Let’s have a look at how to implement the placeholder attribute.

```
<input type="text"name="user-name"id="user-name"placeholder="at least 3 characters">
```

Figure 2 shows the placeholder attribute working in Chrome.

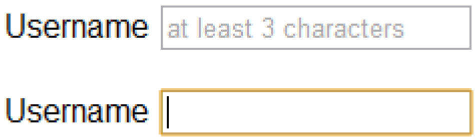


Figure 2. Placeholder attribute support in Chrome, unfocused and focused

2. **autofocus**

autofocus does exactly what it says on the tin. Adding it to an input automatically focuses that field when the page is rendered. It is a Boolean attribute (except if you are writing XHTML5; see the note) and is implemented as follows:
<input type="text"name="first-name"id="first-name" autofocus>

3. **autocomplete**

The autocomplete attribute helps users complete forms based on earlier input. The default state is set to on. This means that generally we won't have to use it. However, if you want to insist that a form field be entered each time a form is completed (as opposed to the browser autofilling the field), you would implement it like so:

<input type="text"name="tracking-code"id="tracking-code"autocomplete="off">

The autocomplete state on a field overrides any autocomplete state set on the containing form element.

4. **required**

The required attribute doesn't need much introduction; likeautofocus, it does exactly what you'd expect. By adding it to a form field, the browser requires the user to enter data into that field before submitting the form. required is a Boolean attribute, like autofocus. Let's see it in action.

<input type="text"id="given-name"name="given-name" required>

New Input Types in HTML5

- color
- date
- datetime
- datetime-local
- email
- month
- number
- range
- search
- tel
- time
- url
- week

The new Elements added by HTML5

list and the datalist element
The list attribute enables the user to associate a list of options with a particular field. The value of the list attribute must be the same as the ID of a datalist element that resides in the

same document. The following example shows how list and datalist are combined (see Figure)

```
<label>Your favorite fruit:
<datalist id="fruits">
  <option value="Blackberry">Blackberry</option>
  <option value="Blackcurrant">Blackcurrant</option>
  <option value="Blueberry">Blueberry</option>
  <!-- … -->
</datalist>
If other, please specify:
  <input type="text" name="fruit" list="fruits">
</label>
```

By adding a select element inside the datalist you can provide superior graceful degradation than by simply using an option element.

```
<label>Your favorite fruit:
<datalist id="fruits">
  <select name="fruits">
    <option value="Blackberry">Blackberry</option>
    <option value="Blackcurrant">Blackcurrant</option>
    <option value="Blueberry">Blueberry</option>
    <!-- … -->
  </select>
If other, please specify:
</datalist>
  <input type="text" name="fruit" list="fruits">
</label>
```

Browser support for list and datalist is currently limited to Opera 9.5+ (see Figure 5), Chrome 20+, Internet Explorer 10 and Firefox 4+.

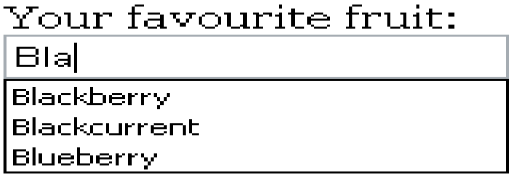


Figure 3 :Thedatalist element rendered in Opera

<u>Attributes of the Form tag</u>:

- Formaction
- Formenctype
- Formmethod
- Formtarget
- Novalidate
- formnovalidate

The novalidate and formnovalidate attributes indicate that the form shouldn’t be validated when submitted. They are both Boolean attributes. formnovalidate can be applied to submit or image input types. The novalidate attribute can be set only on the form element.

The following example shows how to use formnovalidate:

```
<form action="process.php">
  <label for="email">Email:</label>
  <input type="text" name="email"value="gordo@example.com">
  <input type="submit" formnovalidate value="Submit">
</form>
```

And this example shows how to use novalidate:

```
<form action="process.php" novalidate>
  <label for="email">Email:</label>
  <input type="text" name="email"value="gordo@example.com">
  <input type="submit" value="Submit">
</form>
```

Activity:

Design a form (eg. Registration form/feedback form/admission form etc) with HTML 5.0 new form features.

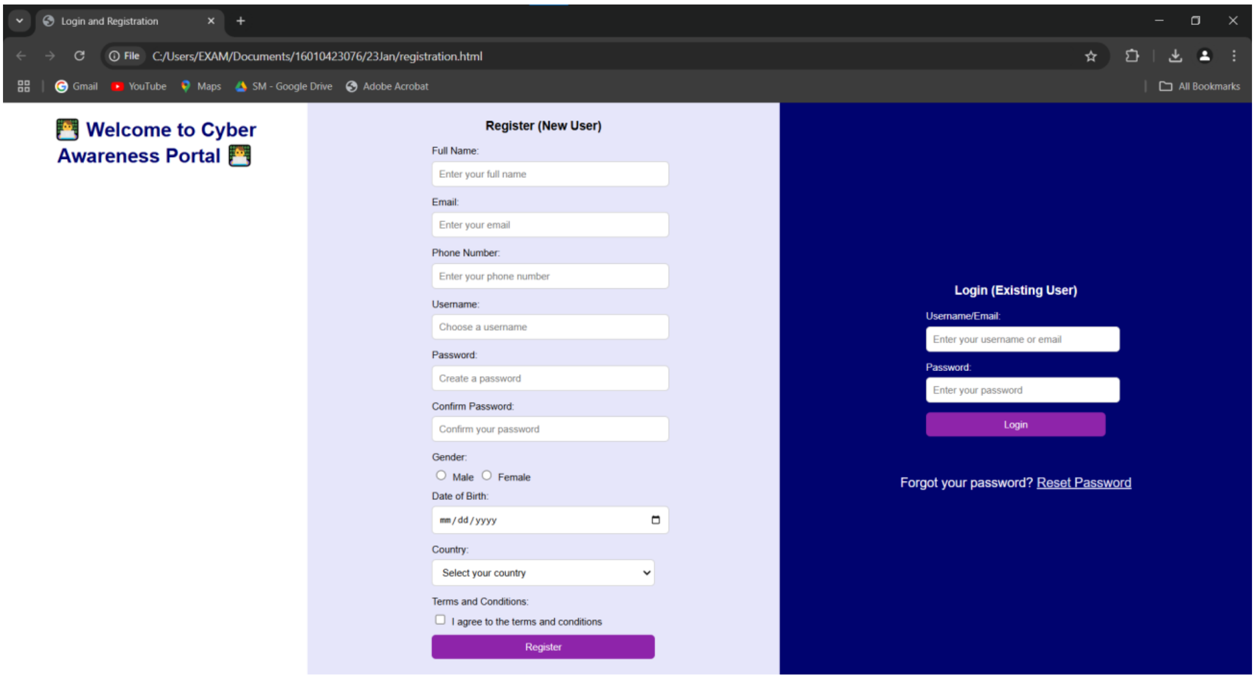
Results: (Program printout with output / Document printout as per the format)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Login and Registration</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
      display: flex;
      height: 100vh;
    }
    .container {
      display: flex;
      width: 100%;
    }
```

```
.left, .right {
  flex: 1;
  padding: 20px;
}
.left {
  background-color: #E6E6FA;
  display: flex;
  flex-direction: column;
  justify-content: center;
}
.right {
  background-color: #00036f;
  color: white;
  display: flex;
  flex-direction: column;
  justify-content: center;
}
h2 {
  text-align: center;
  margin-bottom: 15px;
  font-size: 15px;
}
form {
  max-width: 400px;
  margin: 0 auto;
}
label {
  display: block;
  margin-bottom: 6px;
```

```
      font-size: 12px;
    }
    input, select, button {
      width: 100%;
      padding: 8px;
      margin-bottom: 12px;
      border-radius: 5px;
      border: 1px solid #ccc;
      font-size: 12px;
    }
    input[type="radio"], input[type="checkbox"] {
      width: auto;
    }
    button {
      background-color: #8E24AA;
      color: white;
      border: none;
      cursor: pointer;
    }
    button:hover {
      background-color: #6d1b7b;
    }
    footer {
      text-align: center;
      margin-top: 20px;
    }
  </style>
</head>
<body>
```

```
  <header>

    <center>

      <h1 style="color: #00036f; font-size: 24px;">👨‍💻 Welcome to Cyber Awareness Portal
👨‍💻</h1>

    </center>

  </header>

  <div class="container">



    <div class="left">

      <h2>Register (New User)</h2>

      <form action="Updated.html" method="get">

        <label for="name">Full Name:</label>

        <input type="text" id="name" name="name" placeholder="Enter your full name"
required>



        <label for="email">Email:</label>

        <input type="email" id="email" name="email" placeholder="Enter your email"
required>



        <label for="phone">Phone Number:</label>

        <input type="tel" id="phone" name="phone" placeholder="Enter your phone number"
required>



        <label for="username">Username:</label>

        <input type="text" id="username" name="username" placeholder="Choose a username"
required>



        <label for="password">Password:</label>

        <input type="password" id="password" name="password" placeholder="Create a
password" required>



        <label for="confirm-password">Confirm Password:</label>
        <input type="password" id="confirm-password" name="confirm-password"
placeholder="Confirm your password" required>


        <label for="gender">Gender:</label>

        <input type="radio" id="male" name="gender" value="male" required>

        <label for="male" style="display: inline;">Male</label>

        <input type="radio" id="female" name="gender" value="female">

        <label for="female" style="display: inline;">Female</label>


        <label for="dob">Date of Birth:</label>

        <input type="date" id="dob" name="dob" required>


        <label for="country">Country:</label>

        <select id="country" name="country" required>

          <option value="">Select your country</option>

          <option value="India">India</option>

          <option value="USA">USA</option>

          <option value="UK">UK</option>

          <option value="Canada">Canada</option>

        </select>


        <label for="agree">Terms and Conditions:</label>

        <input type="checkbox" id="agree" name="agree" required>

        <label for="agree" style="display: inline;">I agree to the terms and conditions</label>


        <button type="submit">Register</button>

      </form>

    </div>
    <div class="right">
      <h2>Login (Existing User)</h2>
      <form action="Updated.html" method="get">
        <label for="userId">Username/Email:</label>
        <input type="text" id="userId" name="userId" placeholder="Enter your username or email" required>

        <label for="password">Password:</label>
        <input type="password" id="password" name="password" placeholder="Enter your password" required>

        <button type="submit">Login</button>
      </form>
      <footer>
        <p>Forgot your password? <a href="#" style="color: #E6E6FA;">Reset Password</a></p>
      </footer>
    </div>
  </div>
</body>
</html>
```



Welcome to Cyber Awareness Portal

Register (New User)

Full Name: Enter your full name

Email: Enter your email

Phone Number: Enter your phone number

Username: Choose a username

Password: Create a password

Confirm Password: Confirm your password

Gender: ○ Male ○ Female

Date of Birth: mm/dd/yyyy

Country: Select your country

Terms and Conditions: □ I agree to the terms and conditions

Register

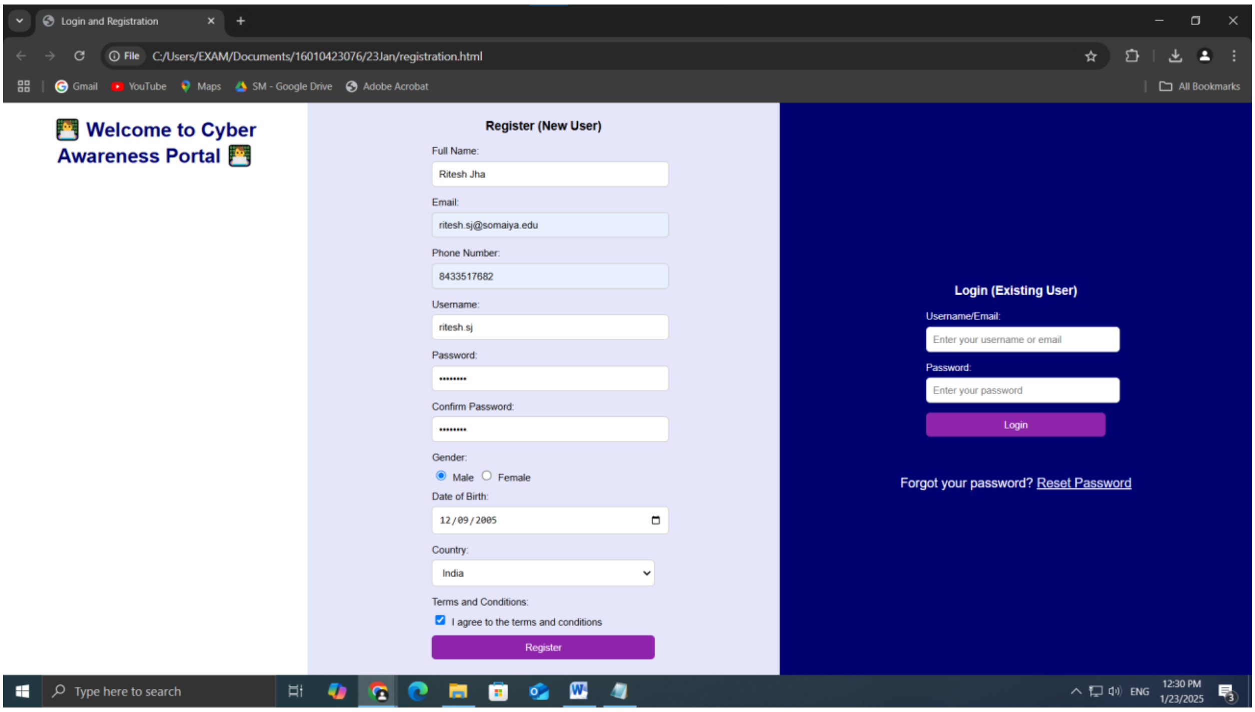
Login (Existing User)

Username/Email: Enter your username or email

Password: Enter your password

Login

Forgot your password? Reset Password



Welcome to Cyber Awareness Portal

Register (New User)

Full Name: Ritesh Jha

Email: ritesh.sj@somaiya.edu

Phone Number: 8433517682

Username: ritesh.sj

Password: ••••••••

Confirm Password: ••••••••

Gender: ● Male ○ Female

Date of Birth: 12/09/2005

Country: India

Terms and Conditions: ☑ I agree to the terms and conditions

Register

Login (Existing User)

Username/Email: Enter your username or email

Password: Enter your password

Login

Forgot your password? Reset Password

Questions:

1. **What is the use of multiple in list and datalist element?**
   The multiple attribute allows users to select more than one option from a dropdown list or a datalist. For example, in a datalist, users can input multiple values separated by commas if multiple is enabled. This is helpful when gathering varied user inputs in a single field, like selecting multiple skills or categories.

2. **What is the importance of pattern attribute?**
   The pattern attribute ensures that user input follows a specific format. It uses a regular expression to validate the input before the form is submitted. For instance, you can use it to enforce rules like entering a 10-digit phone number or a properly formatted email address. This improves data accuracy and reduces errors in user submissions.

3. **What are the three types of button that can be used in form?**
   The three types of buttons in a form are:
   - Submit Button (type="submit"): Used to send the form data to the server for processing.
   - Reset Button (type="reset"): Clears all the input fields in the form, resetting them to their default values.
   - Button (type="button"): A generic button that doesn’t perform any default action on its own. It’s often used with JavaScript to trigger custom functionalities.

Outcomes:

CO2: Create Web pages using HTML 5 and CSS

Conclusion:
(Conclusion to be based on the outcomes achieved)

From this experiment, I learned that attributes like multiple, pattern, and button types significantly enhance form usability and functionality. The multiple attribute allows users to select multiple values, while the pattern attribute enforces input validation. Additionally, understanding the different button types helps in designing forms that are user-friendly and efficient.

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of faculty in-charge with date

References:

Books/ Journals/ Websites:

- "HTML5: Black Book", Dreamtech Publication.
- "Web Technologies: Black Book", Dreamtech Publication.
- http://www.w3schools.com