

CUI
Comsats University Islamabad (Vehari Campus)



Academic Year 22-27
Department: CS

Topic of Assignment:

"Assignment # 02"

Student Name:

M. Taimoor
(FA22-BSE-072)

Subject:

Web Technology

Batch: "19"

Teacher Name:

Ma'am Yasmeen Jana

Q no.1: Explain why semantic HTML is important for web accessibility and SEO.

Assistive Technologies (AT) and Web Accessibility:

1. **Screen Readers:** Semantic HTML helps screen readers understand the page structure, allowing them to provide accurate and contextual information to users with visual impairments.
2. **Keyboard Navigation:** By using semantic HTML, developers can ensure that keyboard navigation is logical and consistent, making it easier for users with mobility or dexterity impairments to navigate the page.
3. **Other Assistive Technologies:** Semantic HTML also benefits other assistive technologies, such as braille displays, switch access, and eye-tracking devices.

Search Engine Optimization (SEO):

1. **Improved Crawling:** Search engines like Google can better understand the structure and content of web pages when semantic HTML is used, leading to improved crawling and indexing.
2. **Relevant Search Results:** By providing meaningful structure to web pages, semantic HTML helps search engines deliver more relevant search results, increasing the chances of your website being listed higher in search engine results pages (SERPs).
3. **Enhanced User Experience:** Semantic HTML also contributes to a better user experience, as it provides a clear and consistent structure, making it easier for users to find what they're looking for.

Benefits of Semantic HTML:

1. **Improved Accessibility:** Semantic HTML ensures that web pages are accessible to users with disabilities, promoting inclusivity and equal access to information.
2. **Better SEO:** By providing meaningful structure to web pages, semantic HTML improves search engine crawling, indexing, and ranking.
3. **Future-Proofing:** Semantic HTML prepares your website for future technologies, such as voice assistants, augmented reality, and other emerging technologies.
4. **Easier Maintenance:** Semantic HTML makes it easier to maintain and update web pages, as the structure and content are clearly defined.

Best Practices for Semantic HTML:

1. **Use Header Tags (H1-H6):** Organize content with header tags, indicating the importance and structure of the content.
2. **Use Meaningful Element Names:** Choose element names that accurately describe the content, such as `<nav>`, `<footer>`, and `<article>`.
3. **Use ARIA Attributes:** Provide additional meaning to elements using ARIA (Accessible Rich Internet Applications) attributes, such as `role` and `aria-label`.
4. **Validate Your HTML:** Regularly validate your HTML to ensure it's error-free and semantic.

Create a Student Information Table with columns for Name, RollNumber, Class, Marks, and Grade. Use proper table attributes

HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>Student Information Table</title>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  <h1>Student Information Table</h1>
  <table id="student-table">
    <thead>
      <tr>
        <th>Name</th>
        <th>Roll Number</th>
        <th>Semester</th>
        <th>Marks</th>
        <th>Grade</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>taimoor</td>
        <td>072</td>
        <td>6 semester</td>
        <td>85</td>
        <td>A</td>
      </tr>
      <tr>
```

```

        <td>Asad</td>
        <td>037</td>
        <td>6 semester</td>
        <td>90</td>
        <td>A+</td>
    </tr>
    <tr>
        <td>Anas</td>
        <td>039</td>
        <td>6 semester</td>
        <td>78</td>
        <td>B</td>
    </tr>
</tbody>
</table>
</body>
</html>

```

CSS:

```

body {
    font-family: Arial, sans-serif;
}

#student-table {
    border-collapse: collapse;
    width: 80%;
    margin: 20px auto;
}

```

```

#student-table th, #student-table td {

```

```
border: 1px solid #ddd;
padding: 10px;
text-align: left;
}
```

```
#student-table th {
    background-color: #f0f0f0;
}
```

```
#student-table tr:nth-child(even) {
    background-color: #f9f9f9;
}
```



Student Information Table

Name	Roll Number	Semester	Marks	Grade
taimoor	072	6 semester	85	A
Asad	037	6 semester	90	A+
Anas	039	6 semester	78	B

Design a Feedback Form with fields for Name, Email, Feedback, Rating (1-5), and a Submit button. Include basic form validation.

HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" type="text/css" href="feedback.css">
</head>
<body>
  <h1>Feedback Form</h1>
  <form id="feedback-form">
    <label for="name">Name:</label>
    <input type="text" id="name" name="name" required>

    <label for="email">Email:</label>
    <input type="email" id="email" name="email" required>

    <label for="feedback">Feedback:</label>
    <textarea id="feedback" name="feedback" required></textarea>

    <label for="rating">Rating (1-5):</label>
    <select id="rating" name="rating" required>
      <option value="">Select a rating</option>
      <option value="1">1</option>
      <option value="2">2</option>
```

```
        <option value="3">3</option>
        <option value="4">4</option>
        <option value="5">5</option>
    </select>

    <button type="submit">Submit</button>
</form>

<div id="error-message"></div>

<script src="script.js"></script>
</body>

</html>
```

CSS:

```
/* Global Styles */
```

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

```
body {
    font-family: Arial, sans-serif;
    font-size: 16px;
    line-height: 1.5;
    color: #333;
    background: linear-gradient(to bottom, #99ccff, #66ccff);
    background-attachment: fixed;
    animation: background-animation 10s infinite;
}
```



```
@keyframes background-animation {  
  0% {  
    background: linear-gradient(to bottom, #99ccff, #66ccff);  
  }  
  50% {  
    background: linear-gradient(to bottom, #99ccff, #66ccff);  
  }  
  100% {  
    background: linear-gradient(to bottom, #99ccff, #66ccff);  
  }  
}
```

```
.container {  
  max-width: 1200px;  
  margin: 40px auto;  
  padding: 20px;  
  background: linear-gradient(to bottom, #99ccff, #66ccff);  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  display: flex;  
  flex-wrap: wrap;  
  justify-content: space-between;  
  animation: container-animation 5s infinite;  
}
```

```
@keyframes container-animation {  
  0% {  
    transform: scale(1);  
  }  
  50% {  
    transform: scale(1.1);  
  }  
}
```

```
100% {  
    transform: scale(1);  
}  
}
```

```
.profile-section {  
    flex-basis: 40%;  
    margin: 20px;  
    padding: 20px;  
    background: linear-gradient(to bottom, #ebf0f5, #eaecec);  
    border: 1px solid #ddd;  
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
    animation: profile-section-animation 5s infinite;  
}
```

```
@keyframes profile-section-animation {  
    0% {  
        transform: translateX(0);  
    }  
    50% {  
        transform: translateX(10px);  
    }  
    100% {  
        transform: translateX(0);  
    }  
}
```

```
.feedback-form {  
    flex-basis: 50%;  
    margin: 20px;  
    padding: 20px;  
    background: linear-gradient(to bottom, #99ccff, #66ccff);  
}
```

```
border: 1px solid #ddd;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
animation: feedback-form-animation 5s infinite;
}
```

```
@keyframes feedback-form-animation {
  0% {
    transform: translateY(0);
  }
  50% {
    transform: translateY(10px);
  }
  100% {
    transform: translateY(0);
  }
}
```

```
form {
  background: linear-gradient(to bottom, #99ccff, #66ccff);
  padding: 20px;
  border: 1px solid #ddd;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
```

```
label {
  display: block;
  margin-bottom: 10px;
}
```

```
input, textarea {
  width: 100%;
  height: 40px;
```

```
margin-bottom: 20px;
padding: 10px;
border: 1px solid #ccc;
border-radius: 5px;
background: linear-gradient(to bottom, #f9f9f9, #e9e9e9);
}
```

```
input:hover, textarea:hover {
    background: linear-gradient(to bottom, #e9e9e9, #f9f9f9);
}
```

```
button {
    width: 100%;
    height: 40px;
    margin-bottom: 20px;
    padding: 10px;
    border: none;
    border-radius: 5px;
    background: linear-gradient(to bottom, #4CAF50, #3e8e41);
    color: #fff;
    cursor: pointer;
}
```

```
button:hover {
    background: linear-gradient(to bottom, #3e8e41, #2f6d31);
}
```

Java Script:

```
const form = document.getElementById('feedback-form');
const errorMessage = document.getElementById('error-message');

form.addEventListener('submit', (e) => {
```

```
let isValid = true;
```

```
// Check if name is empty
if (form.name.value.trim() === '') {
    errorMessage.textContent = 'Please enter your name.';
    isValid = false;
}
```

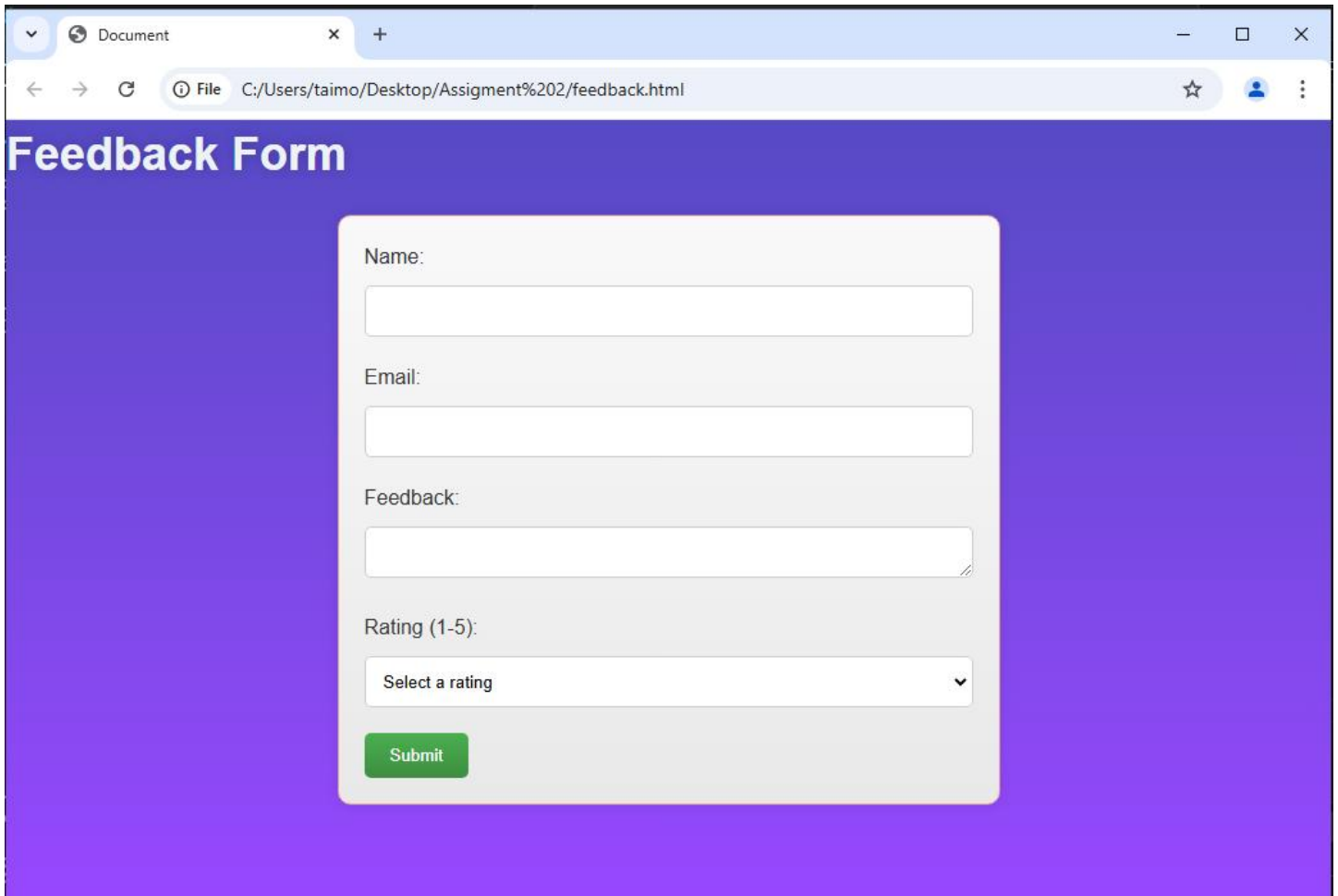
```
// Check if email is valid
const emailRegex = /^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;
if (!emailRegex.test(form.email.value)) {
    errorMessage.textContent = 'Please enter a valid email address.';
    isValid = false;
}
```

```
// Check if feedback is empty
if (form.feedback.value.trim() === '') {
    errorMessage.textContent = 'Please enter your feedback.';
    isValid = false;
}
```

```
// Check if rating is selected
if (form.rating.value === '') {
    errorMessage.textContent = 'Please select a rating.';
    isValid = false;
}
```

```
if (!isValid) {
    e.preventDefault();
}
});
```

Output:



The screenshot shows a web browser window with a single tab titled 'Document'. The address bar shows the file path 'C:/Users/taimo/Desktop/Assignment%202/feedback.html'. The page has a solid purple background. At the top left, the text 'Feedback Form' is displayed in a white, bold, sans-serif font. Centered on the page is a light gray rounded rectangle containing the feedback form. The form consists of four labeled input fields: 'Name:', 'Email:', 'Feedback:', and 'Rating (1-5):'. The 'Name', 'Email', and 'Feedback' fields are white text boxes. The 'Rating (1-5):' field is a dropdown menu with the text 'Select a rating' and a downward arrow. Below these fields is a green rectangular button with the word 'Submit' in white text.

- 1. Create a CSS file named style.css and link it to your HTML files. Apply consistent color schemes and fonts.**
- 2. Implement a Flexbox Layout to display the profile section and feedback form side by side on larger screens and stacked on smaller screens. Use media queries for responsiveness.**

HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" type="text/css" href="flexbox.css">
</head>
<body>
  <div class="container">
    <div class="profile-section">
      <h2>Profile Section</h2>
      <p>This is the profile section.</p>
    </div>
    <div class="feedback-form">
      <h2>Feedback Form</h2>
      <form>
        <label for="name">Name:</label>
        <input type="text" id="name" name="name"><br><br>
        <label for="email">Email:</label>
        <input type="email" id="email" name="email"><br><br>
        <label for="feedback">Feedback:</label>
        <textarea id="feedback" name="feedback"></textarea><br><br>
        <input type="submit" value="Submit">
      </form>
    </div>
  </div>
</body>

</html>
```

CSS:

```
/* Global Styles */
```

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

```
body {  
    font-family: Arial, sans-serif;  
    font-size: 16px;  
    line-height: 1.5;  
    color: #333;  
    background: linear-gradient(to bottom, #99ccff, #66ccff);  
    background-attachment: fixed;  
    animation: background-animation 10s infinite;  
}
```

```
@keyframes background-animation {  
    0% {  
        background: linear-gradient(to bottom, #99ccff, #66ccff);  
    }  
    50% {  
        background: linear-gradient(to bottom, #99ccff, #66ccff);  
    }  
    100% {  
        background: linear-gradient(to bottom, #99ccff, #66ccff);  
    }  
}
```



```
.container {
  max-width: 1200px;
  margin: 40px auto;
  padding: 20px;
  background: linear-gradient(to bottom, #99ccff, #66ccff);
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  animation: container-animation 5s infinite;
}
```

```
@keyframes container-animation {
  0% {
    transform: scale(1);
  }
  50% {
    transform: scale(1.1);
  }
  100% {
    transform: scale(1);
  }
}
```

```
.profile-section {
  flex-basis: 40%;
  margin: 20px;
  padding: 20px;
  background: linear-gradient(to bottom, #ebf0f5, #eaecec);
  border: 1px solid #ddd;
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
```

```
    animation: profile-section-animation 5s infinite;
}
```

```
@keyframes profile-section-animation {
    0% {
        transform: translateX(0);
    }
    50% {
        transform: translateX(10px);
    }
    100% {
        transform: translateX(0);
    }
}
```

```
.feedback-form {
    flex-basis: 50%;
    margin: 20px;
    padding: 20px;
    background: linear-gradient(to bottom, #99ccff, #66ccff);
    border: 1px solid #ddd;
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    animation: feedback-form-animation 5s infinite;
}
```

```
@keyframes feedback-form-animation {
    0% {
        transform: translateY(0);
    }
    50% {
        transform: translateY(10px);
    }
}
```

```
100% {  
    transform: translateY(0);  
}  
}
```

```
form {  
    background: linear-gradient(to bottom, #99ccff, #66ccff);  
    padding: 20px;  
    border: 1px solid #ddd;  
    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
}
```

```
label {  
    display: block;  
    margin-bottom: 10px;  
}
```

```
input, textarea {  
    width: 100%;  
    height: 40px;  
    margin-bottom: 20px;  
    padding: 10px;  
    border: 1px solid #ccc;  
    border-radius: 5px;  
    background: linear-gradient(to bottom, #f9f9f9, #e9e9e9);  
}
```

```
input:hover, textarea:hover {  
    background: linear-gradient(to bottom, #e9e9e9, #f9f9f9);  
}
```

```
button {
```

```
width: 100%;  
height: 40px;  
margin-bottom: 20px;  
padding: 10px;  
border: none;  
border-radius: 5px;  
background: linear-gradient(to bottom, #4CAF50, #3e8e41);  
color: #fff;  
cursor: pointer;  
}
```

```
button:hover {  
    background: linear-gradient(to bottom, #3e8e41, #2f6d31);  
}
```

OUTPUT:

Profile Section
This is the profile section.

Feedback Form

Name:

Email:

Feedback:

Use the Grid Layout to create a Photo Gallery with at least 6 images. The gallery should be responsive, adjusting the number of columns based on screen size. Create a Navigation Bar using Bootstrap with links to Home, About, Gallery, and Contact pages. Use utility classes for styling.

HTML:

```
<!DOCTYPE html>
<html>
<head>
  <title>Photo Gallery</title>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css">
  <link rel="stylesheet" href="bootstrap.css">
</head>
<body>
  <nav class="navbar navbar-expand-lg navbar-light bg-light">
    <a class="navbar-brand" href="#">Photo Gallery</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav mr-auto">
        <li class="nav-item active">
          <a class="nav-link" href="#">Home <span class="sr-
only">(current)</span></a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="#">About</a>
```

```

        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Gallery</a>
        </li>
        <li class="nav-item">
            <a class="nav-link" href="#">Contact</a>
        </li>
    </ul>
</div>
</nav>
<div class="container">
    <div class="row">
        <div class="col-lg-4 col-md-6 col-sm-12">
            
        </div>
        <div class="col-lg-4 col-md-6 col-sm-12">
            
        </div>
        <div class="col-lg-4 col-md-6 col-sm-12">
            
        </div>
        <div class="col-lg-4 col-md-6 col-sm-12">
            
        </div>
        <div class="col-lg-4 col-md-6 col-sm-12">
            
        </div>
        <div class="col-lg-4 col-md-6 col-sm-12">
            
        </div>
    </div>
</div>

```

```
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"></script>
<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
</body>
</html>
```

CSS:

```
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
  background-color: #f9f9f9;
}
```

```
.navbar {
  background-color: #333;
  color: #fff;
  padding: 1rem;
  text-align: center;
}
```

```
.navbar-brand {
  color: #fff;
  font-size: 1.5rem;
  font-weight: bold;
  text-decoration: none;
}
```

```
.navbar-nav {
  list-style: none;
```

```
margin: 0;
padding: 0;
display: flex;
justify-content: space-between;
}
```

```
.nav-link {
  color: #fff;
  text-decoration: none;
  transition: color 0.2s ease;
}
```

```
.nav-link:hover {
  color: #ccc;
}
```

```
.container {
  max-width: 1200px;
  margin: 40px auto;
  padding: 20px;
}
```

```
.row {
  display: flex;
  flex-wrap: wrap;
  justify-content: center;
}
```

```
.col-lg-4 {
  margin: 20px;
  padding: 10px;
  border: 1px solid #ddd;
}
```



```
border-radius: 10px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}
```

```
.img-fluid {
width: 100%;
height: 200px;
object-fit: cover;
border-radius: 10px 10px 0 0;
transition: transform 0.2s ease;
}
```

```
.img-fluid:hover {
transform: scale(1.1);
}
```

```
.image-overlay {
position: absolute;
top: 0;
left: 0;
width: 100%;
height: 100%;
background-color: rgba(0, 0, 0, 0.5);
color: #fff;
padding: 10px;
border-radius: 10px;
display: none;
}
```

```
.col-lg-4:hover .image-overlay {
display: block;
}
```

```
.image-overlay h3 {  
    font-size: 18px;  
    font-weight: bold;  
    margin-bottom: 10px;  
}
```

```
.image-overlay p {  
    font-size: 14px;  
    margin-bottom: 20px;  
}
```

```
/* Add gap between images */  
.row {  
    gap: 20px;  
}
```

```
/* Add gap between rows */  
.row + .row {  
    margin-top: 20px;  
}
```

OUTPUT:

