

**MiniProject-8**

DEPARTMENT OF COMPUTER APPLICATION

BCA-2024

SEMESTER-3

FRONT END WEB DEVELOPMENT

Topic: Student Registration Using Event Handlers

Mentor Name : Veera Raghav K

Done by: Muhammed Faiz C K

Reg No: 2411021240042  
Mentor Signature:\_\_\_\_\_\_\_

**1. Introduction about Using Technologies**

This project demonstrates how to create a Student Registration Form using HTML, CSS, and JavaScript.   
The form collects details such as registration ID, name, course, college, email, and address.   
HTML is used for structuring the form, CSS for styling the layout and design, and JavaScript for interactivity,   
including displaying user input dynamically. This project highlights form handling and user-friendly design principles.

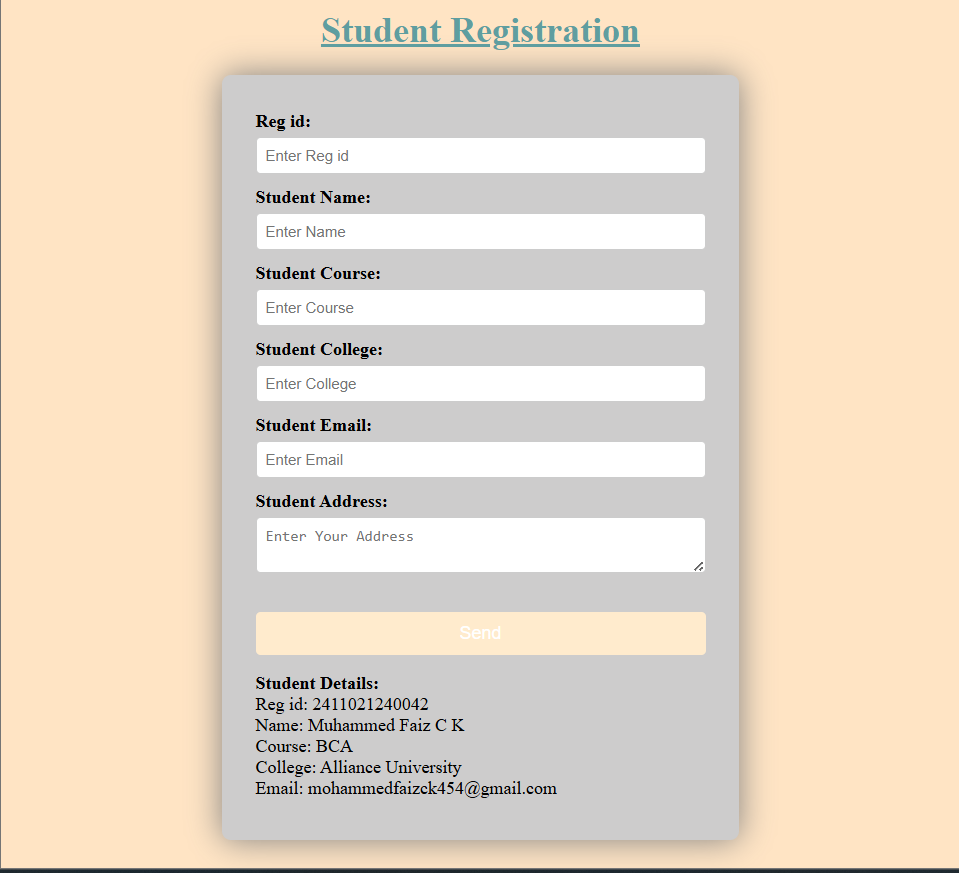
**2. Input Code & Explanation**

<html>  
 <head>  
 <title>Student Registration Form</title>  
 <style>  
 h1{  
 text-align: center;  
 text-decoration: underline;  
 color:cadetblue ;  
 }  
 .container{  
 background: rgb(205, 204, 204);  
 max-width: 400px;  
 margin: 0 auto;  
 padding: 20px 30px;  
 border-radius: 8px;  
 box-shadow: 0 0 29px grey;  
 }  
 label {  
 display: block;  
 margin-top: 12px;  
 margin-bottom: 4px;  
 font-weight: bold;  
 }  
 #User,#Address{  
 width: 100%;  
 }  
 button{  
 margin-top: 16px;  
 width: 100%;  
 background-color:blanchedalmond;  
 color: white;  
 border: none;  
 padding: 10px;  
 font-size: 16px;  
 border-radius: 4px;  
 }  
 button:hover {  
 background-color:burlywood;  
 }  
 body{  
 background-color:bisque;  
 }  
 input, textarea {  
 width: 100%;  
 padding: 8px;  
 border: 1px solid #ccc;  
 border-radius: 4px;  
 box-sizing: border-box;  
 }  
 </style>  
 </head>  
 <body>  
 <h1>Student Registration</h1>  
  
 <div class="container">  
 <form id="studform">  
 <label>Reg id:</label>  
 <input id="RegId" placeholder="Enter Reg id" type="number" required><br>  
  
 <label>Student Name:</label>  
 <input id="StudName" placeholder="Enter Name" type="text" required><br>  
  
 <label>Student Course:</label>  
 <input id="StudCourse" placeholder="Enter Course" type="text" required><br>  
   
 <label>Student College:</label>  
 <input id="StudCollege" placeholder="Enter College" type="text" required><br>  
   
 <label>Student Email:</label>  
 <input id="StudEmail" placeholder="Enter Email" type="email" required><br>  
  
 <label>Student Address:</label>   
 <textarea id="StudAddress" placeholder="Enter Your Address" required></textarea>  
 <br><br>  
 <button type="submit">Send</button>  
 </form>  
   
 <p id="para"></p>  
 </div>  
  
 <script>  
 const studform = document.getElementById("studform");  
 const para = document.getElementById("para");  
 function Display(event){  
 event.preventDefault();  
 const RegIdValue = document.getElementById("RegId").value;  
 const nameValue = document.getElementById("StudName").value;  
 const courseValue = document.getElementById("StudCourse").value;  
 const collegeValue = document.getElementById("StudCollege").value;  
 const emailValue = document.getElementById("StudEmail").value;  
 const addressValue = document.getElementById("StudAddress").value;  
  
 const full = "<b>Student Details:</b><br>"+  
 "Reg id: " + RegIdValue + "<br>"+  
 "Name: " + nameValue + "<br>"+  
 "Course: " + courseValue + "<br>"+  
 "College: " + collegeValue + "<br>"+  
 "Email: " + emailValue + "<br>"+  
 "Address: " + addressValue + "<br>";  
 para.innerHTML = full;  
 studform.reset();  
 }  
 studform.addEventListener('submit', Display);  
 </script>  
 </body>  
</html>

Explanation:  
- The <head> section includes the title and internal CSS for styling.  
- The <style> section defines layout, colors, and responsive form styling.  
- The <form> tag collects user inputs such as ID, name, course, college, email, and address.  
- The <script> section captures form data, prevents page reload, and displays entered information dynamically.

**3. Output & Explanation of Important HTML Tags**

Output :



<h1> : Represents the main heading of the page.  
<form> : Used to create an input form to collect student details.  
<label> : Provides text descriptions for input fields.  
<input> : Used to take different types of data like text, number, and email.  
<textarea> : Accepts multiple lines of text input for address entry.  
<button> : Submits the form when clicked.  
<script> : Contains JavaScript logic for event handling and displaying the result.

**4. Conclusion**

This project demonstrates how to build a user-friendly student registration form using HTML, CSS, and JavaScript.   
It provides an interactive way to collect and display user data on the same page without reloading.   
The project helps understand form validation, styling techniques, and DOM manipulation.