INTERNSHIP REPORT

ON

WEB DEVELOPMENT

Submitted in Partial Fullfillment of the Requirements for

the award of the Degree of

B.SC COMPUTER SCIENCE

V.HEMA SHREE

REGISTER NUMBER: 222305893

INTERNSHIP ORGANIZATION

TECH POWER SOLUTIONS

Submitted To

DEPARTMENT OF COMPUTER SCIENCE

ANNA ADARSH COLLEGE FOR WOMEN

SIGNATURE OF THE

ORGANIZATION

WEEK	TOPICS COVERED	PAGE NO
WEEK 1 (HTML BASICS)	• Introduction to HTML and Structure • Headings and Paragraphs • Formatting Tags (Bold, Italic, underline) • Links and Images with Attributes • Lists : Ordered & Unorderd • Tables and Basic Forms	01
WEEK 2 (CSS BASICS)	• What is CSS ? • Inline, Internal, External CSS • Selectors : Tag, Class, ID • Colors and Fonts • Text Alignment and Styling • Box Model: Margin, Padding, Border	03
WEEK 3 (PRACTICE)	**Practice Tasks using HTML + CSS :** ✓ Basic Form layout ✓ Simple Personal Website ✓ Simple Website Design	05

WEEK 4
(JAVASCRIPT BASICS)
	• Introduction to JavaScript
	• JS Placement in HTML
	• Variables and Data Type
	• Operators and Conditional Statements
	• Basic Functions and Events
	• **Practice Tasks Using HTML + CSS + JS :**
	    ✓ Date and time
	    ✓ Using Operators
		22

1. Introduction to HTML and Structure:

HTML (Hyper Text Markup Language) is the foundational language used to create and structure content on the web. It consists of elements written with tags that define how content should be displayed. A typical HTML document includes a root element, a head section containing metadata and title, and a body section that holds the content visible to users.

The head section includes information like the page title and links to stylesheets, while the body contains text, images, links, tables, forms, and other elements that users interact with.

2. Headings and Paragraphs

HTML offers six levels of headings, from the most important (level 1) to the least important (level 6). These headings help organize content and improve accessibility and search engine optimization.

Paragraphs are used to group blocks of text. They provide spacing between sections and improve readability. Headings and paragraphs are essential for content structure and visual hierarchy on a web page.

3. Formatting Tags (Bold, Italic, Underline)

Formatting tags are used to style text and emphasize certain parts:

- **Bold** text is used to highlight important words or phrases.

- **Italic** text can indicate emphasis or titles of works.

- **Underlined** text is often used for emphasis, but it's less common today due to its association with links.

These tags help in drawing attention to specific parts of the content, making it more engaging and readable.

4. Links and Images with Attributes

Links (hyperlinks) connect one web page to another or to external resources. They can open in the same or a new browser tab and are essential for website navigation.

Images add visual content to a web page. They require a source path and alternative text, which is shown if the image can't load. Attributes like width and height control image dimensions. Proper use of links and images improves user experience and website design.

5. Lists: Ordered and Unordered

Lists are used to group related items:

- **Unordered lists** display items with bullets and are ideal for non-sequential data.

- **Ordered lists** use numbers or letters to indicate a specific sequence, useful for steps, rankings, or instructions.

Both types enhance content organization and make information easier to scan and understand.

6. Tables and Basic Forms

Tables are used to display data in rows and columns. They typically include headers for labeling each column and rows for data entries. Tables help present structured data like schedules, pricing, or comparisons clearly.

Forms are used to collect user input, such as names, emails, or passwords. Basic forms include fields for text input, password entry, and a submit button. Forms are the main method for user interaction and data submission on websites.

1. What is CSS?

CSS (Cascading Style Sheets) is a language used to style and format the layout of web pages. While HTML structures the content, CSS controls how that content looks—such as colors, fonts, spacing, and positioning.

CSS helps in creating visually attractive and responsive websites. It separates design from structure, making web development cleaner and more efficient.

2. Inline, Internal, External CSS

- **Inline CSS** is written directly inside an HTML element's tag. It applies styles to that specific element only. It's not recommended for large-scale styling due to poor maintainability.

- **Internal CSS** is placed within a <style> tag in the <head> section of an HTML document. It styles the entire page but is limited to that single HTML file.

- **External CSS** is stored in a separate .css file and linked to the HTML document. It's the most efficient way to manage styling across multiple web pages, allowing for consistent and scalable design.

3. Selectors: Tag, Class, ID

Selectors are patterns used to target HTML elements for styling.

- **Tag Selector** targets all elements of a specific type (e.g., all paragraphs or all headings).

- **Class Selector** targets elements that share the same class name. It allows multiple elements to share the same styles.

- **ID Selector** targets a single unique element by its ID. It is more specific than class and is typically used when an element needs a distinct style.

Using selectors correctly ensures organized and flexible styling.

4. Colors and Fonts

CSS allows customization of text and background colors using names, hexadecimal codes, RGB, or HSL values. This adds personality and branding to web pages.

Fonts can be changed using properties like font-family, font-size, and font-weight. Web-safe fonts are recommended for compatibility, and custom fonts can be imported for advanced designs.

Controlling fonts and colors enhances readability, mood, and the overall user experience.

5. Text Alignment and Styling

Text alignment can be controlled using properties like text-align (left, right, center, justify). This helps in organizing content in a visually pleasing manner.

Other text styling options include:

- **Line height** for spacing between lines
- **Letter spacing** for character gaps
- **Text decoration** for underlines or line-through effects
- **Text transform** for uppercase, lowercase, or capitalizing text

These properties help improve legibility and design consistency.

6. Box Model: Margin, Padding, Border

The CSS box model is the foundation of layout design. Every element is treated as a rectangular box made up of the following parts:

- **Content**: The actual text or image inside the box.
- **Padding**: Space between the content and the border.
- **Border**: The line surrounding the padding (can be styled with color, thickness, and pattern).
- **Margin**: Space outside the border that separates the element from others

PRACTICE

USING

HTML

&

CSS

CODING FOR STUDENT FORM

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title> STUDENT FORM</title>
    <link rel="stylesheet" href="index.css">
</head>
<body>
 <center>
    <h1 style="height: 30px; color: black;">STUDENT APLLICATION FORM</h1>
    <div class="my">
      <form>
        <label style="font-size: 20px;">NAME : </label>
        <input type="text" placeholder="Enter your name" required>
        <br><br>
        <label style="font-size: 20px;">EMAIL: </label>
        <input type="text" placeholder="Enter your mail"required >
        <br><br>
        <label style="font-size: 20px;">MOBILE NUMBER: </label>
        <input type="text" placeholder="Enter your contact"required >
        <br><br>
        <label style="font-size: 20px;">GENDER: </label>
        <select>
          <option>MALE</option>
          <option>FEMALE</option>
        </select>
        <br><br>
        <label style="font-size: 20px;">COURSE: </label>
```

```html
    <select>

        <option>Bsc.computer science</option>

        <option>Bsc.pyscology</option>

        <option>b.com general</option>

        <option>b.com accountancy</option>

        </select><br><br>
    <div class="yo">

        <h3>

        <p> Lorem ipsum dolor sit amet consectetur adipisicing elit.<br>

            Aliquam, repudiandae incidunt? Deleniti repudiandae tempore <br>

            facilis! Voluptate, quae fugit? Sed assumenda vel, dignissimos<br>

            fuga nostrum neque ipsam rem facere tenetur dolorum.</p>

        </h3>

        <marquee>

        <img src="C:\Users\anand\OneDrive\Desktop\website\thankyou.jpg"
alt="jackpic"width="100px" height="100px">

            <h1>THANKYOU</h1>

        </marquee>

    </div>

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




    </form>

    </div>

 <br><br>

</center>

</body>

</html>
```

CODING FOR CSS

```css
input[type="text"]{
  font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;

  font-size: 20px;

  padding: 20 px;

  font-weight: bold;

  color: black;

  cursor: pointer;

}
  input[type="number"]{

  font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;

  font-size: 20px;

  padding: 20 px;

  font-weight: bold;

  color: black;

  cursor: pointer;

  }
 label[for="email"]{

   font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva,
Verdana, sans-serif;

   font-size:20px;

 }


 .yo{font-family:Cambria, Cochin, Georgia, Times, 'Times New Roman', serif;

  font-size: 20px;

  font-weight:bold;

  font-style: italic;

  color: rgb(196, 15, 84);

  background-color: aqua;

 }
```

STUDENT APLLICATION FORM

NAME : Enter your name

EMAIL : Enter your mail

MOBILE NUMBER : Enter your contact

GENDER: MALE

COURSE: Bsc.computer science

Lorem ipsum dolor sit amet consectetur adipisicing elit. Aliquam, repudiandae incidunt? Deleniti repudiandae tempore facilis! Voluptate, quae fugit? Sed assumenda vel, dignissimos fuga nostrum neque ipsam rem facere tenetur dolorum.

SUBMIT

THANKYOU

CODING FOR HTML

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>MY WEBSITE</title>
  <link rel="stylesheet" href="hemu.css">
</head>
<body>
  <center>
  <nav>
   <h1> WELCOME TO MY WEBSITE</h1>

     <a href="https://annaadarsh.edu.in">HOME</a>


     <a href="https://annaadarsh.edu.in/history-of-college">ABOUTUS</a>


     <a href="https://annaadarsh.edu.in/wp-content/uploads/2024/03/College-governing-council-2023.pdf">COUNCIL</a>


     <a href="https://annaadarsh.edu.in/events-1">EVENTS</a>
  </nav>
  <main>
   <h2>GOOD MORNING SIR/MAM !</h2>
   <p>THIS IS OUR COLLEGE WEBSITE<br>
     Hello everyone this is our college website
     here you can know about our college <br>
     academics, events and others and also about
     our achievements.</p>
   </main>
<div class="box">
   <footer>
```

```html
    <p>@2025 MY COLLLEGE WEBSITE</p>

   </footer>

</div>

   </center>


</body>

</html>
```

CODING FOR CSS

```css
.body{
    font-family: Arial, Helvetica, sans-serif;
    background-color: brown;
    color: aqua;
    text-align: center;
    cursor: pointer;
}
nav{
    background-color: rgb(9, 9, 9);
    color: rgb(222, 218, 218);
    transition:background-color 0.5s ease;
    padding: 20px;
    box-shadow: 2px;
}
nav h1{
    margin:0;
    font-size: 2em;
}
nav ul{
    margin-top: 10px;
}
nav ul li a{
    color: bisque;
    font-size: 10em;
}
main{
    padding: 50px;
}
main p{
```

```css
        font-size: 1em;

        color: black;

    }

    main h2{

        font-size: 2em;

    }

    .box{

        background-color: black;

        color: beige;

        padding:15px;

        position:absolute;

        width:100%;

        transition:background-color 0.5s ease;

        cursor: pointer;

    }

    .box:hover{

        background-color: brown;

    }
```

WELCOME TO MY WEBSITE

GOOD MORNING SIR/MAM !

THIS IS OUR COLLEGE WEBSITE

Hello everyone this is our college website here you can know about our college academics,events and others and also about our achievements.

@2025 MY COLLIEGE WEBSITE

SIMPLE WEBSITE DESIGN

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

</head>

<body>
  <div class="cat"></div>
  <div class="anan">
   <nav>
    <div class="log">
     <a href="#">web<span>site.</span></a>
    </div>
    <ul>
     <li><a href="#">Home</a></li>
     <li><a href="#">About</a></li>
     <li><a href="#">Services</a></li>
     <li><a href="#">Contact</a></li>
    </ul>
    <div class="buttons">
     <a href="#" class="login">Log in</a>
     <a href="#" class="cls">Register</a>
    </div>
   </nav>
```

```html
    <div class="hem">

      <h2>Hello,<br>TECH POWER SOLUTIONS</h2>

      <p>We are here to create an website<br>You Can Find Me On Google.</p>

    </div>

    <div class="my">

      <a href="#" class="bby">CLICK ME</a>

    </div>

  </div>


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


</script>
</html>
```

CODING FOR CSS

```css
.cat{
 width: 100%;
 height: 100;
 background-image:url(cat.jpg);
 background-size:100%;
 background-position:center;
 background-repeat: no-repeat;
 color: antiquewhite;
 display: flex;
 align-items: center;
 justify-content: center;
 text-align: center;
}
*{
 margin: 0;
 padding: 0;
 box-sizing: border-box;
 font-family: 'Karla', sans-serif;
}
body{
 color: #fff;
}
.anan{
 width: 100%;
 height: 100vh;
 background-color: #ba0799;
 background-position: center;
 background-size: cover;
 padding-top: 35px;
```

```css
  padding-left: 8%;

  padding-right: 8%;

}

nav{

  padding: 10px 0;

  display: flex;

  align-items: center;

  justify-content: space-between;

}

.log a{

  font-size: 40px;

  text-decoration: none;

}

span{

  color: hwb(179 1% 84%);

}

nav ul li{

  display: inline-block;

  list-style: none;

  margin: 10px 15px;

}

nav ul li a{

  text-decoration: none;

  transition: 0.5s;

}

nav ul li a:hover{

  color: #060506;

}

.login{

  text-decoration: none;

  margin-right: 15px;
```

```css
    font-size: 18px;
}
.cls{
  background: #000;
  border-radius: 6px;
  padding: 9px 25px;
  text-decoration: none;
  transition: 0.5s;
  font-size: 18px;
}
.hem{
  margin-top: 10%;
  max-width: 600px;
}
.hem h2{
  font-size: 60px
}
.hem p{
  margin-top: 10px;
  line-height: 25px;
}
a{
  color: #fff;
}
.my{
  margin-top: 30px;
}
.bby{
  color: #000;
  text-decoration: none;
  background: #fff;
```

```css
    padding: 9px 25px;

    font-weight: bold;

    border-radius: 6px;

    transition: 0.5s;

}
.my.bby:hover{

    background: transparent;

    border: 1px solid #d615bc;

    color: #fff;

}
```

website.

Home About Services Contact

Log in Register

Hello,
TECH POWER
SOLUTIONS

We are here to create an website
You Can Find Me On Google.

CLICK ME

1. Introduction to JavaScript

JavaScript is a powerful, high-level scripting language used to create dynamic and interactive elements on web pages. It works in the browser, allowing developers to manipulate HTML content, respond to user actions, validate forms, create animations, and much more.

Unlike HTML and CSS, which control structure and style, JavaScript adds behavior and logic to web applications. It is an essential part of modern web development alongside HTML and CSS.

2. JavaScript Placement in HTML

JavaScript can be included in an HTML document in three main ways:

- **Inline**: JavaScript is written directly inside an HTML element's event attribute (e.g., onclick). It's generally used for very small scripts.

- **Internal**: Code is written inside a <script> tag within the HTML file, usually in the <head> or just before the closing <body> tag.

- **External**: JavaScript is placed in a separate .js file and linked to the HTML. This is best practice for larger projects as it keeps code organized and reusable.

Placing JavaScript just before the closing body tag is common to ensure that HTML elements are loaded before the script runs.

3. Variables and Data Types

Variables are used to store data that can be used and modified later. In JavaScript, variables can be declared using var, let, or const. Modern best practice is to use let for variables that change, and const for those that don't.

Common **data types** in JavaScript include:

- **String** – for text

- **Number** – for numeric values

- **Boolean** – true or false

- **Array** – list of values

- **Object** – collection of key-value pairs

- **Null** and **Undefined** – for empty or uninitialized variables

Understanding variables and data types is crucial for handling logic and data in scripts.

4. Operators and Conditional Statements

JavaScript uses **operators** to perform operations on values:

- **Arithmetic operators** (+, -, *, /)

- **Comparison operators** (==, ===, !=, >, <)

- **Logical operators** (&&, ||, !)

Conditional statements allow the program to make decisions based on conditions. The most common structure is the if, else if, and else statement.

These tools let JavaScript perform logic-based tasks and respond to different scenarios in a program.

5. Basic Functions and Events

Functions are reusable blocks of code that perform specific tasks. They can accept inputs (called parameters) and return outputs. Using functions promotes modular and clean code.

Events are user actions (like clicks, key presses, or page loads) that JavaScript can respond to. You can assign functions to events, so something happens when the user interacts with the page.

Functions and events are core to making websites interactive and responsive to user actions.

6. Simple DOM Methods

JavaScript interacts with web pages through the **DOM (Document Object Model)**, which represents the page structure as objects.

Two basic and frequently used DOM methods are:

- **getElementById**: Used to select a specific HTML element by its ID for manipulation.

- **innerText**: Used to read or change the visible text content of an element.

PRACTICE USING HTML, CSS & JAVASCRIPT

CODING FOR HTML

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>My Clock</title>
</head>
<body onload="showTime()">
  <div id="MyclockDisplay" > time </div>
    <link rel="stylesheet" href="date.css">
  <script >
   function showTime(){
    var date = new Date();
    var h = date.getHours(); // 0 - 23
    var m = date.getMinutes(); // 0 - 59
    var s = date.getSeconds(); // 0 - 59
    var session = "PM";

    if(h == 0){
       h = 12;
    }

    if(h > 12){
       h = h - 12;
       session = "PM";
    }

    h = (h < 10) ? "0" + h : h;
    m = (m < 10) ? "0" + m : m;
    s = (s < 10) ? "0" + s : s;
```
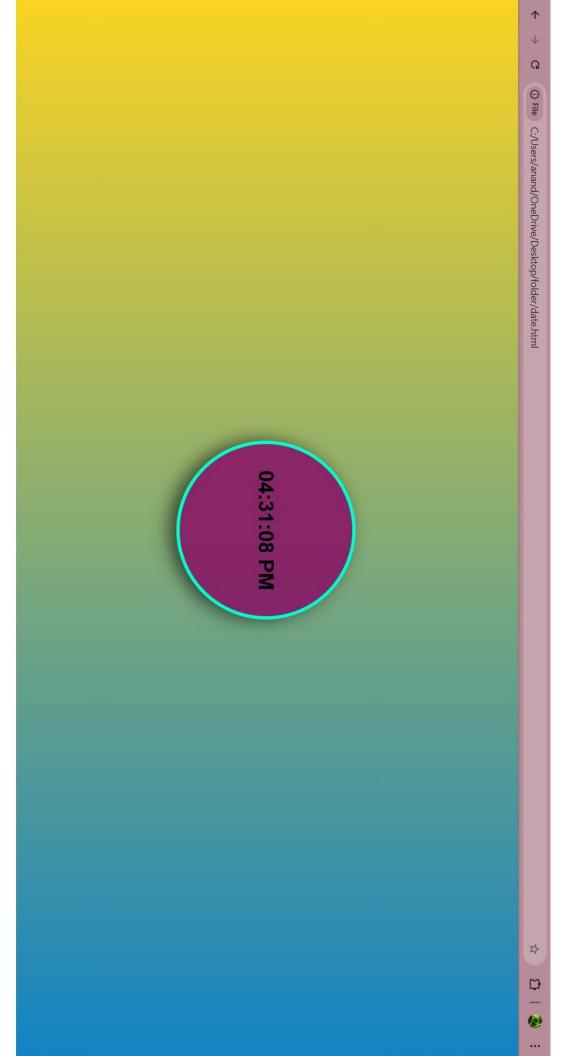
```
    var time = h + ":" + m + ":" + s + " " + session;

    document.getElementById("MyclockDisplay").innerText  = time;

    setTimeout(showTime, 1000);


}
showTime();
 </script>
</body>
</html>
```

CODING FOR CSS

```css
body {
  background: linear-gradient(to right, #f9d423,rgb(18, 131, 197));
  height: 100vh;
  display: flex;
  justify-content: center;
  align-items: center;
  margin: 0;
}

#MyclockDisplay {
  font-size: 30px;
  font-weight: bold;
  background: rgba(137, 14, 102, 0.85);
  color: #010407;
  width: 250px;
  height: 250px;
  border-radius: 50%; /* makes the box a perfect circle */
  display: flex;
  justify-content: center;
  align-items: center;
  text-align: center;
  box-shadow: 0 10px 25px rgba(0, 0, 0, 0.6);
  border: 5px solid #00ffcc;
  font-family: 'Orbitron', sans-serif;
}
```

04:31:08 PM

```html
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Operator Demo</title>
  <link rel="stylesheet" href="operator.css" />
  <script src="operator.js"></script>
</head>
<body>

  <div class="container">
    <h2>JavaScript Operators Demo</h2>
    <p>Value: <span id="value">0</span></p>
    <button onclick="increment()">Increment (+)</button>
    <button onclick="decrement()">Decrement (-)</button>
    <button onclick="addFive()">Add 5</button>
    <button onclick="subtractFive()">Subtract 5</button>
    <button onclick="multiplyTwo()">Multiply 2</button>
    <button onclick="reset()">Reset</button>
  </div>
</body>
</html>
```

CODING FOR CSS

```css
body {
    font-family: Arial, sans-serif;
    text-align: center;
    padding: 50px;
    background-color: #f0f8ff;
  }
  .container {
    display: inline-block;
    padding: 20px;
    border: 2px solid #007acc;
    border-radius: 10px;
    background: #fff;
  }
  button {
    margin: 10px;
    padding: 10px 20px;
    font-size: 16px;
    cursor: pointer;
    background-color: #007acc;
    color: white;
    border: none;
    border-radius: 5px;
  }
  p {
    font-size: 18px;
  }
```

CODING FOR JAVASCRIPT

```javascript
let value = 0;

function updateDisplay() {
  document.getElementById('value').textContent = value;
  }

function increment() {
  value++;
  updateDisplay();
}

function decrement() {
  value--;
  updateDisplay();
}

function addFive() {
  value += 5;
  updateDisplay();
}

function subtractFive() {
  value -= 5;
  updateDisplay();
}
function multiplyTwo() {
  value*=2;
  updateDisplay();
}
```

```
function reset() {
  value = 0;
  updateDisplay();
}
```

JavaScript Operators Demo

Value: 4

Increment (+)   Decrement (-)   Add 5   Subtract 5   Multiply 2   Reset