

Lab 1: Hello World!

Aim: Create an HTML file and open it in a web browser

Code:

```
<!DOCTYPE html>
<html>
<head>
  <title>My First Webpage</title>
</head>
<body>
  <h1>Hello World!</h1>
  <p>Welcome to Web Design Lab</p>
</body>
</html>
```

Lab 2: Inspect, Ping

Common Terms

- **URL (Uniform Resource Locator):**
 - A URL is used to identify and locate documents on the internet.
 - Browsers fetch web pages from servers using URLs.
 - **Syntax:** scheme://prefix.domain:port/path/filename
- **Hosting:**
 - Hosting is the service that stores websites on a web server.
 - It makes websites accessible via the World Wide Web.
- **Bandwidth:**
 - Bandwidth refers to the maximum data transfer rate across a network.
 - It measures how much data can be sent and received per second.
- **Cache:**
 - Cache is a temporary storage area for frequently accessed web content.
 - It improves speed and efficiency by reusing stored data.
- **Cookies:**
 - Cookies are small pieces of data stored by websites on a browser.
 - They help websites retrieve user data when needed.
- **Responsive Design:**
 - Responsive design ensures a website adapts to all screen sizes.
 - It improves usability and looks consistent on any device.
- **Web Standards:**
 - Web standards are guidelines that ensure websites function consistently.
 - They maintain uniformity across different browsers and devices.

Practical Demonstration

- **Ping:**
 - Open Command Prompt.
 - Type ping <IP Address> and press Enter.
 - It will show minimum, maximum, and average ping times.
- **Page Source:**
 - Right-click on any webpage and select **View Page Source**.
 - Shortcut: Press Ctrl + U.
- **Inspect Element:**
 - Right-click on a webpage and select **Inspect**.
 - Shortcut: Press Ctrl + Shift + I or F12.
- **NS2 Simulator:**

- Create a TCL script for network simulation.
- Run it in NS2 and observe network behavior in the trace file.

Interactive Exercise

- **Tracing Web Requests:**
 - Open Developer Tools with Ctrl + Shift + I.
 - Go to the **Network** tab.
 - Reload the page (Ctrl + R) and monitor network requests.
- **Types of Website's Source:**
 - Explore different websites by viewing their source code.
- **MIME (Multipurpose Internet Mail Extensions):**
 - MIME types define file types and formats.
 - Common examples:
 - HTML → text/html
 - CSS → text/css
 - JavaScript → application/javascript
 - Images → image/jpeg
 - Video → video/mp4

Lab 3: Forms and Tables

Code:

Forms:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Forms</title>
</head>
<body>
  <h1>Registration Form</h1>
  <form>

    <label>Username: </label>
    <input type="text" placeholder="Username">
    <br/><br/>

    <label>Password:</label>
    <input type="password" placeholder="Password">
    <br/><br/>

    <label>Full Name: </label>
    <input type="text" placeholder="Enter Your Name"/>
    <br/><br/>

    <label>Phone number: </label>
    <input type="number" placeholder="Enter your number"/>
    <br/><br/>

    <label for="gender">Gender: </label>
    <br/>
    <label for="gender-male">Male</label>
    <input type="radio" value="male" name="gender" id="gender-male" checked>
    <label for="gender-female">Female</label>
    <input type="radio" value="female" name="gender" id="gender-female">
    <label for="gender-other">Other</label>
    <input type="radio" value="other" name="gender" id="gender-other">
    <br/><br/>

    <label for="education">Education: </label>
    <br/>
```

```

    <label for="education-primary">Primary School</label>
    <input type="checkbox" name="education" value="Primary School" id="education-
primary" checked>
    <label for="education-secondary">Secondary Education</label>
    <input type="checkbox" value="Secondary School" name="education" id="education-
secondary">
    <label for="education-undergrad">Under Graduate</label>
    <input type="checkbox" value="Under Graduate" name="education" id="education-
undergrad">
    <label for="education-grad">Graduate</label>
    <input type="checkbox" value="Graduate" name="education" id="education-grad">
    <br/><br/>

    <label for="site-rating">Site Rating:</label>
    <select>
        <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>
        <option value="6">6</option>
        <option value="7">7</option>
        <option value="8">8</option>
        <option value="9">9</option>
        <option value="10" selected>10</option>
    </select>
    <br/><br/>
    <input type="submit" value="Submit">
    <input type="reset" value="Reset">

</form>
<br/>
</body>
</html>

```

Table:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Enhanced Table</title>
</head>
<body>

```

```

<table border="5">
  <caption><h1>Student Information</h1></caption>
  <thead>
    <tr>
      <th colspan="6">Contents</th>
    </tr>
    <tr>
      <th>S No.</th>
      <th>Name</th>
      <th>Age</th>
      <th>Class</th>
      <th>Roll No.</th>
      <th>Department</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>1</td>
      <td>Hitesh</td>
      <td>17</td>
      <td>2203</td>
      <td>12411078</td>
      <td>CSE</td>
    </tr>
    <tr>
      <td>2</td>
      <td>Sushil</td>
      <td>23</td>
      <td>2203</td>
      <td>12411096</td>
      <td>CSE</td>
    </tr>
    <tr>
      <td>3</td>
      <td>Anil</td>
      <td>19</td>
      <td>2205</td>
      <td>12411067</td>
      <td>ECE</td>
    </tr>
    <tr>
      <td>4</td>
      <td>Piyush</td>
      <td>19</td>
      <td>2204</td>
      <td>12411103</td>
      <td></td>
    </tr>
  </tbody>
</table>

```

```
        <td>CSE</td>
    </tr>
    <tr>
        <td>5</td>
        <td>Garv</td>
        <td>20</td>
        <td>2202</td>
        <td>12411029</td>
        <td>ECE</td>
    </tr>
</tbody>
<tfoot>
    <tr>
        <td colspan="5">Total Students</td>
        <td>5</td>
    </tr>
</tfoot>
</table>

</body>
</html>
```

Lab 4: Inline, Embedded

Code:

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Hitesh's Portfolio</title>
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css">
  <style>
    body {
      font-family: 'Roboto', sans-serif;
      margin: 0;
      padding: 0;
      background: skyblue;
    }
    h1 {
      color: #2c3e50;
      text-align: center;
      padding: 20px;
      font-weight: 700;
    }
  </style>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <div class="container">
    <h1 style="background: #3498db; color: white; margin: 0;">Hitesh's Portfolio</h1>
    <section id="about">
      <h2>About Me</h2>
      <p class="content-box">
        Hi, I'm Hitesh! I'm currently in my 2nd semester at IIIT Sonapat.
        My native home is in Karnal, Haryana. I'm passionate about web development
        and C++ programming, and I'm excited to build creative and functional projects.
      </p>
    </section>

    <section id="contact">
      <h2>Contact Us</h2>
      <div class="content-box contact-info">
```



```

        <p><i class="fas fa-envelope"></i> Email: hiteshshonak1@gmail.com</p>
        <p><i class="fab fa-linkedin"></i> LinkedIn: <span class="social-link">[Your ID
Here]</span></p>
        <p><i class="fab fa-twitter"></i> X: <span class="social-link">[Your ID
Here]</span></p>
        <p><i class="fab fa-instagram"></i> Instagram: <span class="social-link">[Your ID
Here]</span></p>
    </div>
</section>
</div>
</body>
</html>

```

CSS:

```

.container {
    width: 900px;
    margin: 0 auto;
    padding: 20px;
}
h2 {
    font-family: 'Montserrat', sans-serif;
    color: #2980b9;
    font-style: normal;
    border-bottom: 2px solid #3498db;
    padding-bottom: 5px;
    margin-bottom: 20px;
}
.content-box {
    background-color: rgba(255, 255, 255, 0.9);
    padding: 20px;
    margin: 20px 0;
    border-radius: 8px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
    border-left: 5px solid #3498db;
    transition: transform 0.3s ease;
}
#about .content-box:hover {
    transform: scale(1.05);
}
#about {
    position: relative;
    height: 130px;
}
.contact-info p {
    margin: 15px 0;
    font-weight: 500;
}

```

```
.contact-info i {
    color: #2c3e50;
    margin-right: 10px;
    width: 20px;
}
.social-link {
    color: #7f8c8d;
    font-style: italic;
}
.social-link:hover {
    color: #3498db;
    text-decoration: underline;
}
/* Media Query */
@media screen and (max-width: 600px) {
    body{
        background-color: lightgoldenrodyellow;
    }
    .container {
        padding: 10px;
        background-color: lightgoldenrodyellow;
    }
    h1 {
        font-size: 1.5em;
        padding: 15px;
    }
    .content-box {
        margin: 10px 0;
        padding: 15px;
    }
    h2 {
        font-size: 1.2em;
    }
}
```

Lab 5: Sun

Code:

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Simple Sun with Radial Gradient</title>
  <link rel="stylesheet" href="sunstyles.css">
</head>
<body>
  <div class="sun"></div>
</body>
</html>
```

CSS

```
body {
  margin: 0;
  height: 100vh;
  background: skyblue;
  display: flex;
  justify-content: center;
  align-items: center;
}

.sun {
  width: 200px;
  height: 200px;
  border-radius: 50%;
  background: radial-gradient(circle, yellow 0%, orange 70%, red 100%);
}
```

Lab 6: Solar System

Code:

HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>Animated Solar System</title>
</head>
<body>
  <div class="solar-system">
    <div class="sun"></div>
    <div class="orbit-container orbit1"><div class="planet"></div></div>
    <div class="orbit-container orbit2"><div class="planet"></div></div>
    <div class="orbit-container orbit3"><div class="planet"></div></div>
    <div class="orbit-container orbit4"><div class="planet"></div></div>
    <div class="orbit-container orbit5"><div class="planet"></div></div>
    <div class="orbit-container orbit6"><div class="planet"></div></div>
    <div class="orbit-container orbit7"><div class="planet"></div></div>
    <div class="orbit-container orbit8"><div class="planet"></div></div>
  </div>
</body>
</html>
```

CSS

```
<style>
  body {
    background: #000;
    margin: 0;
    height: 100vh;
    display: flex;
    justify-content: center;
    align-items: center;
    overflow: hidden;
  }

  .solar-system {
    position: relative;
    width: 800px;
    height: 800px;
  }
```

```

.sun {
  position: absolute;
  width: 70px;
  height: 70px;
  background: #fc3;
  border-radius: 50%;
  box-shadow: 0 0 50px #f90;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  z-index: 10;
}

.orbit-container {
  position: absolute;
  top: 50%;
  left: 50%;
  border: 1px solid rgba(255, 255, 255, 0.3); /* Visible orbit line */
  border-radius: 50%;
  transform: translate(-50%, -50%);
  animation: rotate 30s linear infinite;
}

.planet {
  position: absolute;
  border-radius: 50%;
}

/* Mercury */
.orbit1 {
  width: 120px;
  height: 120px;
  animation-duration: 4s;
}
.orbit1 .planet {
  width: 8px;
  height: 8px;
  background: #aaa;
  top: 50%;
  left: 100%;
  transform: translate(-50%, -50%) rotate(10deg);
}

/* Venus */
.orbit2 {
  width: 180px;
  height: 180px;

```

```

    animation-duration: 8s;
}
.orbit2 .planet {
    width: 14px;
    height: 14px;
    background: #e8c;
    top: 50%;
    left: 100%;
    transform: translate(-50%, -50%) rotate(80deg);
}

/* Earth */
.orbit3 {
    width: 240px;
    height: 240px;
    animation-duration: 12s;
}
.orbit3 .planet {
    width: 16px;
    height: 16px;
    background: #49f;
    top: 50%;
    left: 100%;
    transform: translate(-50%, -50%) rotate(160deg);
}

/* Mars */
.orbit4 {
    width: 300px;
    height: 300px;
    animation-duration: 20s;
}
.orbit4 .planet {
    width: 12px;
    height: 12px;
    background: #f74;
    top: 50%;
    left: 100%;
    transform: translate(-50%, -50%) rotate(230deg);
}

/* Jupiter */
.orbit5 {
    width: 360px;
    height: 360px;
    animation-duration: 35s;
}

```

```

.orbit5 .planet {
  width: 30px;
  height: 30px;
  background: #eb7;
  top: 50%;
  left: 100%;
  transform: translate(-50%, -50%) rotate(45deg);
}

/* Saturn */
.orbit6 {
  width: 420px;
  height: 420px;
  animation-duration: 50s;
}
.orbit6 .planet {
  width: 25px;
  height: 25px;
  background: #dc8;
  top: 50%;
  left: 100%;
  transform: translate(-50%, -50%) rotate(110deg);
}
.orbit6 .planet::after {
  content: "";
  position: absolute;
  width: 40px;
  height: 8px;
  background: rgba(220, 200, 150, 0.5);
  border-radius: 50%;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%) rotate(20deg);
}

/* Uranus */
.orbit7 {
  width: 480px;
  height: 480px;
  animation-duration: 70s;
}
.orbit7 .planet {
  width: 18px;
  height: 18px;
  background: #8df;
  top: 50%;
  left: 100%;

```

```
    transform: translate(-50%, -50%) rotate(200deg);
  }

/* Neptune */
.orbit8 {
  width: 540px;
  height: 540px;
  animation-duration: 100s;
}
.orbit8 .planet {
  width: 17px;
  height: 17px;
  background: #59f;
  top: 50%;
  left: 100%;
  transform: translate(-50%, -50%) rotate(300deg);
}

@keyframes rotate {
  to {
    transform: translate(-50%, -50%) rotate(360deg);
  }
}
</style>
```


Lab 7: Theme Quotes

Code:

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <title>Dashboard</title>
  <link rel="stylesheet" href="style.css"/>
</head>
<body>
  <div class="container">
    <h1 id="greeting"></h1>
    <p id="quote"></p>

    <label for="theme">Theme:</label>
    <select id="theme">
      <option value="light">Light</option>
      <option value="dark">Dark</option>
    </select>

    <label for="font-size">Font Size:</label>
    <select id="font-size">
      <option value="16px">Small</option>
      <option value="20px">Medium</option>
      <option value="24px">Large</option>
    </select>

    <h2>To-Do List</h2>
    <input type="text" id="task-input" placeholder="Add a task" />
    <button onclick="addTask()">Add</button>
    <ul id="task-list"></ul>
  </div>

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

CSS

```
body {
```

```
margin: 0;
padding: 20px;
font-family: Arial, sans-serif;
transition: background 0.3s, color 0.3s;
}
```

```
.container {
  max-width: 500px;
  margin: auto;
}
```

```
.dark {
  background: #111;
  color: #eee;
}
```

```
.light {
  background: #f9f9f9;
  color: #111;
}
```

```
#task-list {
  list-style-type: none;
  padding: 0;
}
```

```
#task-list li {
  margin: 5px 0;
  padding: 5px;
  background: #2ecec6;
  display: flex;
  justify-content: space-between;
}
```

JavaScript

```
// Greeting based on time
function getTimeGreeting() {
  const hour = new Date().getHours();
  if (hour < 12) return "Good Morning";
  else if (hour < 18) return "Good Afternoon";
  else return "Good Evening";
}
```

```
// Ask for name once per session
let name = sessionStorage.getItem("name");
```

```

if (!name) {
  name = prompt("What is your name?");
  if (name) {
    sessionStorage.setItem("name", name);
  }
}

// Show greeting
const greeting = document.getElementById("greeting");
const timeGreeting = getTimeGreeting();
greeting.textContent = `${timeGreeting}, ${name || "Guest"}!`;

// Quotes
const quotes = [
  "Believe in yourself!",
  "Stay positive and strong.",
  "Every day is a new chance.",
  "You are capable of amazing things.",
  "Don't give up!"
];
const quote = document.getElementById("quote");
quote.textContent = quotes[Math.floor(Math.random() * quotes.length)];

// Theme
const themeSelect = document.getElementById("theme");
const savedTheme = localStorage.getItem("theme") || "light";
document.body.classList.add(savedTheme);
themeSelect.value = savedTheme;

themeSelect.addEventListener("change", () => {
  document.body.classList.remove("light", "dark");
  document.body.classList.add(themeSelect.value);
  localStorage.setItem("theme", themeSelect.value);
});

// Font size
const fontSizeSelect = document.getElementById("font-size");
const savedFont = localStorage.getItem("fontSize") || "16px";
document.body.style.fontSize = savedFont;
fontSizeSelect.value = savedFont;

fontSizeSelect.addEventListener("change", () => {
  const size = fontSizeSelect.value;
  document.body.style.fontSize = size;
  localStorage.setItem("fontSize", size);
});

```

```

// To-do list
const taskInput = document.getElementById("task-input");
const taskList = document.getElementById("task-list");
let tasks = JSON.parse(localStorage.getItem("tasks")) || [];

function renderTasks() {
  taskList.innerHTML = "";
  tasks.forEach((task, index) => {
    const li = document.createElement("li");
    li.textContent = task;
    const btn = document.createElement("button");
    btn.textContent = "✕";
    btn.onclick = () => {
      tasks.splice(index, 1);
      localStorage.setItem("tasks", JSON.stringify(tasks));
      renderTasks();
    };
    li.appendChild(btn);
    taskList.appendChild(li);
  });
}

function addTask() {
  if (taskInput.value.trim()) {
    tasks.push(taskInput.value.trim());
    localStorage.setItem("tasks", JSON.stringify(tasks));
    taskInput.value = "";
    renderTasks();
  }
}

renderTasks();

```

Lab 8: Rectangle Sorter

Code:

HTML

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Rectangle Area Sorter</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <h1>Rectangle Area Sorter</h1>

  <div class="input-section">
    <label for="length">Length:</label>
    <input type="number" id="length" min="1" required>

    <label for="breadth">Breadth:</label>
    <input type="number" id="breadth" min="1" required>

    <button onclick="addRectangle()">Add Rectangle</button>
  </div>

  <table>
    <thead>
      <tr>
        <th>Length</th>
        <th>Breadth</th>
        <th>Area</th>
      </tr>
    </thead>
    <tbody id="rectangleTableBody">
      <!-- Rows will be added dynamically -->
    </tbody>
  </table>

  <h2>Visual Prototypes (Scaled)</h2>
  <div class="prototypes" id="prototypeContainer">
    <!-- Visual rectangles will appear here -->
  </div>

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

CSS

```
body {  
  font-family: Arial, sans-serif;  
  background-color: #f0f4f8;  
  color: #333;  
  padding: 30px;  
}
```

```
h1, h2 {  
  text-align: center;  
  color: #2c3e50;  
}
```

```
.input-section {  
  text-align: center;  
  margin-bottom: 30px;  
}
```

```
.input-section label {  
  font-weight: bold;  
  margin: 0 10px;  
}
```

```
.input-section input {  
  padding: 8px;  
  width: 80px;  
  margin-right: 10px;  
  border-radius: 5px;  
  border: 1px solid #ccc;  
}
```

```
.input-section button {  
  padding: 8px 16px;  
  background-color: #3498db;  
  color: white;  
  border: none;  
  border-radius: 5px;  
  font-weight: bold;  
  cursor: pointer;  
}
```

```
.input-section button:hover {  
  background-color: #2980b9;  
}
```

```
table {  
  width: 60%;  
  margin: 0 auto 30px auto;  
  border-collapse: collapse;  
  box-shadow: 0 4px 8px rgba(0,0,0,0.1);  
  background-color: white;  
  border-radius: 10px;  
  overflow: hidden;  
}
```

```
th, td {  
  border: 1px solid #ddd;  
  padding: 10px;  
  text-align: center;  
}
```

```
th {  
  background-color: #3498db;  
  color: white;  
}
```

```
.prototypes {  
  display: flex;  
  align-items: flex-start;  
  justify-content: center;  
  gap: 15px;  
  padding-top: 20px;  
  border-top: 2px solid #ccc;  
  margin-top: 30px;  
  position: relative;  
  background-color: transparent;  
  height: auto;  
}
```

```
.rectangle-div {  
  background: linear-gradient(145deg, #74b9ff, #0984e3);  
  border: 2px solid #2c3e50;  
  transition: transform 0.3s ease, margin-top 0.3s ease;  
}
```

```
.rectangle-div:hover {  
  transform: scale(1.1);  
}
```

Javascript

```
let rectangles = [];
```

```
function addRectangle() {  
  const length = parseInt(document.getElementById('length').value);  
  const breadth = parseInt(document.getElementById('breadth').value);  
  
  if (isNaN(length) || isNaN(breadth) || length <= 0 || breadth <= 0) {  
    alert("Please enter valid positive numbers for length and breadth.");  
    return;  
  }  
}
```

```
const area = length * breadth;  
rectangles.push({ length, breadth, area });
```

```
// Sort rectangles by area in ascending order  
rectangles.sort((a, b) => a.area - b.area);
```

```
updateTable();  
updatePrototypes();  
}
```

```
function updateTable() {  
  const tableBody = document.getElementById("rectangleTableBody");  
  tableBody.innerHTML = "";  
  
  rectangles.forEach(rect => {  
    const row = document.createElement("tr");  
    row.innerHTML = `  
      <td>${rect.length}</td>  
      <td>${rect.breadth}</td>  
      <td>${rect.area}</td>  
    `;  
    tableBody.appendChild(row);  
  });  
}
```

```
function updatePrototypes() {  
  const container = document.getElementById("prototypeContainer");  
  container.innerHTML = "";  
  
  const scale = 4; // 1 unit = 4 pixels  
  const maxHeight = Math.max(...rectangles.map(r => r.breadth * scale));  
  rectangles.forEach(rect => {  
    const heightPx = rect.breadth * scale;  
    const offset = maxHeight - heightPx;
```



```
const div = document.createElement("div");
div.className = "rectangle-div";
div.style.width = `${rect.length * scale}px`;
div.style.height = `${heightPx}px`;
div.style.marginTop = `${offset}px`;
container.appendChild(div);
});
}
```