**HTML5 Introduction**

|  |  |
| --- | --- |
| HTML  Abbreviation | Hyper Text Markup Language |
| Purpose | Used to design web pages |
| Building Blocks | Predefined elements/tags |
| Tags Purpose | Display data |
| Types of Tags | (Singleton(closing part)) and Paired tags(opening and closing parts)) |
| HTML latest version | HTML5 |
| HTML5 Features | Uppercase tag names, optional (Quotes, Attribute and Closing empty elements) |

**Html Tags:** use keywords for formatting and display the content on a web browser.

|  |  |  |
| --- | --- | --- |
| **Tag Name** | **Description** | **Output** |
| Headings | Used to display heading with different fonts. <h1>heading 1</h1> <h2>heading 2</h2> <h3>heading 3</h3> <h4>heading 4</h4> <h5>heading 5</h5> <h6>heading 6</h6> | heading 1 heading 2 heading 3 heading 4 heading 5 heading 6 |
| Paragraph | Used to display Paragraph. <p>Type Text</p> | Type Text |
| Anchor | It is used for page redirection and downloads for a file. <a href=" http://www.wisdommaterials.com"> links to site wisdommaterials.com </a>  \_blank - Opens new window or  \_self - Opens document in same window/tab | Links to that site (wisdommaterials.com). target=\_parent - Opens document in parent frame \_top - Opens document in same window get="\_blank" |
| Image | Used Displays Image on webpage. <img src="a.jpg" alt="Describe image" width="100" height="150"/> | Image |
| Button | <button>Click me</button> | Click Me |
| Lists It is a collection of items. Types of Lists 1. Ordered. 2. Unordered. | Ordered <ol type="1" >   <li>Coffee</li>    <li>Tea</li>   <li>Milk</li> </ol> Type can be type="1"         type="A"      type="a"  type="I"         type="i" Unordered <ul style="list-style-type:square" >   <li>Coffee</li>   <li>Tea</li>  <li>Milk</li> </ul> | Ordered Example 1. Coffee 2. Tea 3. Milk   Unordered Example .Coffee . Tea .Milk Type: disc, bullet, circle, square, none, not be marked |
| Horizontal Rules | Give a horizontal line. | ------------------------ |
| Meta | <meta charset="UTF-8"> | Character Encoding |
| New Line | <br> | Singleton tag |
| Pre | <pre>1   2 3   4 </pre> | 1 2 3 4 |
| Quotations | <q>Quotations</q> | “Quotations” |
| Comment | <!-- Write your comments here --> |  |
| Div | It is used as a Positioning element. <div>Positions element</div> | Positions element |
| Link | <link rel = "stylesheet" type = "text/css" href = "stylefile.css"> | Used to link files. |
| <address> | Defines contact information of owner in italic. <address> Wisdom Materials,<br /> Hyderabad<br /> </address> | Wisdom Materials, Hyderabad |
| <bdo> | It is display the text from left to right. <bdo  dir="rtl">text is written from right to left</bdo> | Tfel ot thgir  morf nettirw si txet |
| <form> | It is used to send data to the server.  <form name=”f1”> </form> |  |

**Html Formatting Elements**

|  |  |  |
| --- | --- | --- |
| Bold text | <b>Bold text</b> | Html Formating Tags |
| Important text | <strong>Important text</strong> |
| Italic text | <i>Italic text</i> |
| Emphasized text | <em>Emphasized text</em> |
| Marked text | <mark>Marked text</mark> |
| Small text | <small>Small text</small> |
| Deleted text | <del>Deleted text</del> |
| Inserted text | <ins>Inserted text</ins> |
| Subscript text | <p>It is <sub>subscripted</sub> text.</p> |
| Superscript text | <p>It is <sup>superscripted</sup> text.</p> |

**HTML VS HTML 5**

|  |  |  |
| --- | --- | --- |
|  | **HTML** | **HTML 5** |
| Audio & Video | support using flash player | It supports <audio> and <video> tags. |
| Temporary Data Store | uses cookies | uses SQL databases and application cache |
| Javascript | supports | supports |
| Vector Graphics | possible with technologies such as VML, Silver-light, Flash, etc. | It is an integral a part of HTML5 like SVG and canvas. |
| Drag & Drop | Don’t support | support |
| Draw Shapes | Don’t support | circle, rectangle, triangle etc |
| Special Elements | nav, header were not present. | nav, header, footer etc are present. |

**Html5 Document Tags**

|  |  |
| --- | --- |
| **Tag Names** | **Details** |
| section | It represents a generic document / application section and uses h1-h6 tags to indicate the document structure. |
| article | It is a tag represents an independent piece of content of a document (blog entry /newspaper article). |
| aside | It represents a piece of content related to the rest of the page. |
| header | It represents header section. |
| footer | It contain information about the author, copyright information, etc. |
| nav | It represents navigation section of a document. |
| dialog | It can be used to mark up a conversation. |
| figure | It is used with a caption together with some embedded content ( graphic / video). |

**HTML5 Template Preparation**

|  |  |
| --- | --- |
| Program | <header role = "banner" style="color:white; width:100%; height:25%; background-color:orange;" align="center">  <h1>Header Part1</h1>  <p>Header Part2</p>  <a href = "https://www.abcd.com/" style=" color: white;">Home</a> &nbsp;&nbsp;&nbsp;  <a href = "https://www.abcd.com/" style=" color: white;">About</a>&nbsp;&nbsp;&nbsp;  <a href = "https://www.abcd.com/" style=" color: white;">Deptartments</a>&nbsp;&nbsp;&nbsp;  <a href = "https://www.abcd.com/" style=" color: white;">Contact</a>  </aside>  </header>  <nav style="color:white; width:100%; height:50%; background-color:blue;" >  <section>  <center>  <br><br><br><br>  Body1  <br><br><br><br>  </center>  </section>  </nav>    <footer style="color:white; width:100%; height:25%; background-color:red;"align="center"> <br />  <center>Footer Part</center>  </footer> |
| Output |  |

**HTML Video Tag:** Used to show a video on a web page.

**Examples**

|  |  |
| --- | --- |
| **Program** | **Output** |
| <video width="320" height="240" controls autoplay muted >  <source src="movie.mp4" type="video/mp4">  </video> | vid.png |

**HTML Audio Tag:**  Used play an audio file on a web page.

|  |  |
| --- | --- |
| **Example** | **Output** |
| <audio src=“flute.mp3” controls> |  |

**HTML Audio / Video Tag Attributes**

|  |  |
| --- | --- |
| **Attributes** | **Purpose** |
| controls | adds Audio / video controls, like play, pause, and volume |
| source | Allows you to specify alternative video files which the browser may choose from. |
| autoplay | To start a Audio / video automatically |
| muted | To mute Audio / video |

**HTML Canvas**

|  |  |
| --- | --- |
| <canvas>  Tag | It is a container for graphics |
| Purpose | Draw graphics, on the fly, via JavaScript. |
| Canvas methods | drawing paths, boxes, circles, text, and adding images |
| Canvas attributes | id attribute, width, height attribute and border |

**Examples**

|  |  |
| --- | --- |
| **Programs** | **Outputs** |
| <canvas id="myCanvas" width="200" height="100" style="border:1px solid #d3d3d3;"> </canvas> | canvas op.png |
| <script>  var c = document.getElementById("myCanvas");  var ctx = c.getContext("2d");  ctx.moveTo(0,0);ctx.lineTo(200,100);ctx.stroke();  </script> | canvas op2.png |
| ctx.beginPath();  ctx.arc(95,50,40,0,2\*Math.PI); |  |
| ctx.font = "30px Arial";  ctx.fillText("Hello World",10,50); |  |

**SVG (Scalable Vector Graphics)**

|  |  |
| --- | --- |
| SVG | It is a container for SVG graphics for the Web Recommended by W3C |
| Methods | drawing paths, boxes, circles, text, and graphic images |
| Examples | <svg width="100" height="100">  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4" fill="yellow" />  </svg><br>  <svg width="100" height="100">  <rect width="400" height="100" style="fill:rgb(0,0,255);stroke-width:10;stroke:rgb(0,0,0)" />  </svg><br>  <svg width="300" height="200">  <polygon points="100,10 40,198 190,78 10,78 160,198"  style="fill:lime;stroke:purple;stroke-width:5;fill-rule:evenodd;" />  </svg><br>  <svg height="130" width="500">  <defs>  <linearGradient id="grad1" x1="0%" y1="0%" x2="100%" y2="0%">  <stop offset="0%" style="stop-color:rgb(255,255,0);stop-opacity:1" />  <stop offset="100%" style="stop-color:rgb(255,0,0);stop-opacity:1" />  </linearGradient>  </defs>  <ellipse cx="100" cy="70" rx="85" ry="55" fill="url(#grad1)" />  <text fill="#ffffff" font-size="45" x="50" y="86">SVG</text>  </svg> |
| Output |  |

**Differences between Canvas and SVG**

|  |  |  |
| --- | --- | --- |
| Property | Canvas | SVG |
| Resolution | dependent | independent |
| Event Handlers | Don’t support | support |
| Text Rendering Capabilities | Poor | Best suited for applications with large rendering areas (Google Maps) Slow rendering if complex (anything that uses the DOM a lot will be slow) |
| Suitable For Games | yes Well suited | not |

**Web Storage**

Used by web applications can store data locally within the user's browser. Before HTML5, application data had to be stored in cookies, included in every server request. Web storage is more secure, and large amounts of data can be stored locally, without affecting website performance. Unlike cookies, the storage limit is far larger (at least 5MB) and information is never transferred to the server. Web storage is per origin (per domain and protocol). All pages, from one origin, can store and access the same data.

HTML web storage provides two objects for storing data on the client:

|  |  |
| --- | --- |
| **Object Name** | **Details= Stores Data** |
| window.localStorage | with no expiration date |
| window.sessionStorage | for one session (data is lost when the browser tab is closed) |

**The localStorage Object**

It stores the data with no expiration date. The data will not be deleted when the browser is closed, and will be available forever or till the user removes it. Example: Execute the following program in any IDE to obsrve how can key and values can be stored in the local storage.

|  |  |
| --- | --- |
| Program | <div id="result"></div>  <script>  if (typeof(Storage) !== "undefined") {  localStorage.setItem("FirstName", "CMRIT");  document.getElementById("result").innerHTML = localStorage.getItem("lastname");  } else {  document.getElementById("result").innerHTML = "Sorry, your browser does not support Web Storage...";  }  </script> |
| Output | Key = FirstName, value = wisdom |

**The sessionStorage Object**

It is equal to the localStorage object, except that it stores the data for only one session. The data is deleted when the user closes the specific browser tab.

Example: Implement a counter of user clicks on a button, in the current session

|  |  |
| --- | --- |
| Program | <script>  function clickCounter() {  if (typeof(Storage) !== "undefined")  {  if (sessionStorage.clickcount)  sessionStorage.clickcount = Number(sessionStorage.clickcount)+1;  else  sessionStorage.clickcount = 1;  document.getElementById("result").innerHTML = "You have clicked the button " + sessionStorage.clickcount + " time(s) in this session.";  }  else  document.getElementById("result").innerHTML = "web storage browser dont support ";  }  </script>  <p><button onclick="clickCounter()" type="button">Click me!</button></p>  <div id="result"></div>  <p>Click button to increase counter value & Close browser resets counter.</p> |
| Output | Counter increments on button click |

Support

|  |  |
| --- | --- |
|  |  |

**Delete Web Storage**

1. Storing sensitive data on local machine could be dangerous and could raise a security issue.

2. After session gets terminated browsers delete the Session Storage Data.

3. Execute methods in console & observe the changes under application tab local storage.

|  |  |
| --- | --- |
| Methods to clear local storage | Details |
| localStorage.clear(); | where 'key' is the key of the value you want to remove |
| localStorage.remove('FirstName'); | If you want to clear all setting |

**Drag and Drop**

Drag and drop "grab" an object (Image) and drag it to a different location in the web page.

|  |  |
| --- | --- |
| Program | <style>#div1 {width: 150px; height: 150px;border: 1px solid #aaaaaa;} </style>  <script>  function allowDrop(ev) { ev.preventDefault(); }  function drag(ev) { ev.dataTransfer.setData("text", ev.target.id);}  function drop(ev) {  ev.preventDefault();  var data = ev.dataTransfer.getData("text");  ev.target.appendChild(document.getElementById(data));  }  </script>  <p>Drag the image into the rectangle:</p>  <div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)"></div><br>  <img id="drag1" src="rose.jpg" draggable="true" ondragstart="drag(event)" width="100" height="100"> |
| Output |  |

**GeoLocation**

Its API is used to locate a user's position on his approval using getCurrentPosition() method.

|  |  |
| --- | --- |
| Program | <button onclick="getLocation()">Click to get Latitude and Longitude</button>  <script>  function getLocation() {  if (navigator.geolocation)  navigator.geolocation.getCurrentPosition(showPosition);  else  document.write("Browser Dont support Geolocation.");  }  function showPosition(position) { document.write("Latitude:"+position.coords.latitude + "Latitude:"+position.coords.longitude);  }</script> |
| Output | Allow location, Latitude:17.6041294 Latitude:78.4838642 |

**Cascading Style Sheets (CSS)**

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes. CSS saves a lot of work. It can control the layout of multiple web pages all at once.

CSS Syntax



**Types of CSS:**

|  |  |  |
| --- | --- | --- |
| **Type** | **Use** | **Attribute / Tag** |
| Inline | Style attribute inside the HTML elements | style attribute |
| Internal | Use <style>  inside the HTML document | <style> |
| External | Link tag within the HTML document | <link> |

**Inline CSS**

|  |  |
| --- | --- |
| Definition | It use a style attribute to individual HTML element the HTML document. |
| Example | <h1 style="color: blue;">A Blue Heading</h1>  <p style="color: green;">A green paragraph</p> |
| Output | Color applied to H1 is blue and p is green |

**Internal CSS**

|  |  |
| --- | --- |
| Definition | It defines a style for a single HTML page using style tag. |
| Example Program | <style>  h1 {color: blue;} p {color: red;}  </style>  <h1>This is a heading</h1>  <p>This is a paragraph.</p> |
| Output | h1, p are appeared in blue and red colors. |

**External CSS**

|  |  |
| --- | --- |
| Definition | It is used to define the style for many HTML pages |
| Example | ECSS.html  <link rel="stylesheet" href="styles.css">  <h1>This is a heading</h1>  <p>This is a paragraph.</p>  styles.css:  body {  background-color: powderblue;}  h1 {  color: blue;}  p {  color: red;} |
| Output | H1, p are assigned the colors blue and red. |

**CSS Layout - The position Property :** positioning method positions the element.

|  |  |
| --- | --- |
| Position Values | Position Type |
| Static  Relative  Fixed  Absolute |  |

**Bootstrap**

|  |  |
| --- | --- |
| Bootstrap | Front-end framework to create Responsive web designs which automatically adjust themselves to look good on all devices (Phones, desktops & Tabs). |
| Developed By | Mark Otto & Jacob Thornton as open source product, 8/2011at GitHub. |
| Features | Free, faster and easier web development |
| typography Includes | Forms, buttons, tables, navigation, modals, image carousels, optional JavaScript plugins e.t.c. |

**Bootstrap Advantages:**

|  |  |
| --- | --- |
| Advantages | Details |
| Easy to use | Everyone can use Bootstrap with HTML and CSS knowledge. |
| Responsive features | Bootstrap's responsive CSS adjusts to phones, tablets, and desktops |
| Browser compatibility | all browsers (Chrome, Firefox, Internet Explorer, Edge, Safari, and Opera) |

**Bootstrap Template Preparation**

|  |  |
| --- | --- |
| Resources | https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css  src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js |
| Header | <!DOCTYPE html>  <html lang="en">  <head>  <title>Bootstrap 5 Example</title>  <meta charset="utf-8">  <meta name="viewport" content="width=device-width, initial-scale=1">  <link href="bootstrap.min.css" rel="stylesheet">  <script src=" bootstrap.bundle.min.js"></script>  </head>  <body> </body>  </html> |
| header | <div class="container-fluid p-5 bg-primary text-white text-center">  <h1>My First Bootstrap Page</h1>  <p>Resize this responsive page to see the effect!</p>  </div> |
| Body | <div class="container mt-5">  <div class="row">  <div class="col-sm-4">  <h3>Column 1</h3>  <p>Column 1 text</p>  <p>Column 1 text</p>  </div>  <div class="col-sm-4">  <h3>Column 2</h3>  <p>Column 2 text</p>  <p>Column 2 text</p>  </div> <br><br><br><br>  </div>  </div> |
| Footer | <style>  .footer { position: fixed; left: 0; bottom: 0; width: 100%; background-color: red; color: white; text-align: center; }  </style>  <div class="footer">  <p>Footer</p>  </div> |
| Menu | <div class="container mt-3">  <div class="btn-group">  <button type="button" class="btn btn-primary">Home</button>  <div class="btn-group">  <button type="button" class="btn btn-primary dropdown-toggle" data-bs-toggle="dropdown">About</button>  <ul class="dropdown-menu">  <li><a class="dropdown-item" href="#">Chairman</a></li>  <li><a class="dropdown-item" href="#">Principle</a></li>  <li><a class="dropdown-item" href="#">College</a></li>  </ul>  </div>  <button type="button" class="btn btn-primary">Depts</button>  <button type="button" class="btn btn-primary">Contact</button>  </div>  </div> |