



# Course Title: HTML, CSS, Javascript and Bootstrap

Faculty: Ajay Pooreti



L&T Technology Services



LTTS

GLOBAL  
ENGINEERING  
ACADEMY



# Agenda

A background image showing a person's hands interacting with a tablet. The tablet screen displays various data visualizations, including a bar chart, a line graph, and a pie chart. The person is wearing a light blue shirt. In the background, there is a desk with a pair of glasses and some papers.

Introduction about HTML

# Version

Version	Reviewed by	Approved by	Remarks
1.0	Ajay Pooreti		

Embed Syllabus of the Couse on this slide



# Learning Outcome of the Course

---

# Pre-Requisites for the Course

---



# Introduction of HTML



# Introduction of HTML

---

- HTML stands for Hyper Text Markup Language, which is the most widely used language on Web to develop web pages.
- HTML was created by Berners-Lee in late 1991.
- "HTML 2.0" was the first standard HTML specification which was published in 1995.
- HTML 4.01 was a major version of HTML and it was published in late 1999. Though HTML 4.01 version is widely used .
- Currently we are having HTML-5 version which is an extension to HTML 4.01, and this version was published in 2012.

# Introduction of HTML

---

- HTML is a markup language and makes use of various tags to format the content.
- These tags are enclosed within angle braces <Tag Name>.
- HTML Supports 2 types tags. They are
  - >Paired Tags
  - >Un-Paired Tags OR Single Tag
- Except few tags, most of the tags have their corresponding closing tags.
- For example, <html> has its closing tag </html> and <body> tag has its closing tag </body> tag etc.
- HTML is not a case sensitive language



# Basic Elements Or Tags in HTML

---

- Above example of HTML document uses the following tags –

1        `<!DOCTYPE...>`

This tag defines the document type and HTML version.

2        `<html>`

This tag encloses the complete HTML document and mainly comprises of document header which is represented by `<head>...</head>` and document body which is represented by `<body>...</body>` tags.

3        `<head>`

This tag represents the document's header which can keep other HTML tags like `<title>`, `<link>` etc.

4.       `<title>`

The `<title>` tag is used inside the `<head>` tag to mention the document title.

# Basic Elements Or Tags in HTML

---

5      <body>

This tag represents the document's body which keeps other HTML tags like <h1>, <div> etc..

Example program on Basic Elements:

```
<!DOCTYPE>
```

```
<html>
```

```
<head>
```

```
<title>First Program in HTML</title>
```

```
</head>
```

```
<body>
```

```
<h2>Hello World</h2>
```

```
</body>
```

```
</html>
```

# Global Attributes

---

- id
- class
- style
- title
- dir
- data\*
- hidden
- lang



# Header Tags



# Header Tags in HTML

Any document starts with a heading. You can use different sizes for your headings.

HTML also has six levels of headings, which use the elements `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, and `<h6>`. While displaying any heading, browser adds one line before and one line after that heading.

```
<!DOCTYPE html>
```

```
<html> <head>
```

```
  <title>Heading Example</title>
```

```
</head>
```

```
<body>
```

```
  <h1>This is heading 1</h1>
```

```
  <h2>This is heading 2</h2>
```

```
  <h3>This is heading 3</h3>
```

```
  <h4>This is heading 4</h4>
```

```
  <h5>This is heading 5</h5>
```

```
  <h6>This is heading 6</h6>
```

```
</body>
```

```
</html>
```

# Paragraph Tag In HTML



# Paragraph Tag in HTML

The <p> tag offers a way to structure your text into different paragraphs.

Each paragraph of text should go in between an opening <p> and a closing </p> tag as shown below in the example –

Example

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Paragraph Example</title>
```

```
  </head>
```

```
  <body>
```

```
    <p>Here is a first paragraph of text.</p>
```

```
    <p>Here is a second paragraph of text.</p>
```

```
    <p>Here is a third paragraph of text.</p>
```

```
  </body>
```

```
</html>
```



# Image Tag in HTML





# Image Tag in HTML

Images are very important to beautify as well as to depict many complex concepts in simple way on your web page. This tutorial will take you through simple steps to use images in your web pages.

## Insert Image:

You can insert any image in your web page by using `<img>` tag.

Following is the simple syntax to use this tag.

```
<img src = "Image URL" ... attributes-list/>
```

The `<img>` tag is an empty tag, which means that, it can contain only list of attributes and it has no closing tag.

# Image Tag Example Program

---

```
<!DOCTYPE html>

<html>
  <head>
    <title>Set Image Alignment</title>
  </head>
  <body>
    <p>Setting image Alignment</p>
    <img src = "C:\Users\AJAYREDDY\Desktop\ganesh.jpg" border = "3"
      align="right" />
  </body>
</html>
```



# Link Tag in HTML



# Link Tag In HTML

// In html to link from one web page into another web page is known as hyperlink.  
using anchor tag to link between one page to another page.it follows the given syntax:  
`<a href="target.html"> content</a>`

example:

```
-----  
<!DOCTYPE html>  
<html>  
  <head>  
    <title>link tag in html</title>  
  </head>  
  <body>  
    <a href="one.html"> click here to get one.html file</a>  
  </body>  
</html>
```

# Table Tag in HTML



# Table Tag in HTML

The HTML tables allow web authors to arrange data like text, images, links, other tables, etc. into rows and columns of cells.

The HTML tables are created using the <table> tag in which the <tr> tag is used to create table rows and <td> tag is used to create data cells. The elements under <td> are regular and left aligned by default

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>    <title>HTML Tables</title>  </head>
```

```
<body>
```

```
  <table border = "1" bordercolor="red">
```

```
    <tr>
```

```
      <td>Row 1, Column 1</td>
```

```
      <td>Row 1, Column 2</td>
```

```
    </tr>
```

```
  </table>    </body>  </html>
```

# Table Tag in HTML

---

## Table Heading:

Table heading can be defined using `<th>` tag. This tag will be put to replace `<td>` tag, which is used to represent actual data cell.

Normally you will put your top row as table heading as shown below, otherwise you can use `<th>` element in any row. Headings, which are defined in `<th>` tag are centered and bold by default.

## Cellpadding and Cellspacing Attributes:

There are two attributes called cellpadding and cellspacing which you will use to adjust the white space in your table cells.

The cellspacing attribute defines space between table cells, while cellpadding represents the distance between cell borders and the content within a cell

# Table Tag in HTML

```
<!DOCTYPE html>
```

```
<html>
```

```
<head><title>HTML Table Cellpadding</title> </head>
```

```
<body>
```

```
<table border = "1" cellpadding = "5" cellspacing = "5">
```

```
<tr>
```

```
<th>Name</th>
```

```
<th>Salary</th>
```

```
</tr> <tr>
```

```
<td>Suresh</td>
```

```
<td>1000</td>
```

```
</tr> </table></body> </html>
```



# List in HTML



# List Elements in HTML

---

HTML offers web authors three ways for specifying lists of information.

All lists must contain one or more list elements. Lists may contain –

1. Order List (OR) Numbering List
2. Un-Order List (OR) Bullet list
3. Definition List

`<ul>` – An unordered list. This will list items using plain bullets.

`<ol>` – An ordered list. This will use different schemes of numbers to list your items.

`<dl>` – A definition list. This arranges your items in the same way as they are arranged in a dictionary.

# List Elements in HTML-Order List

Order List :

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>HTML Ordered List</title>
```

```
  </head>
```

```
  <body>
```

```
    <ol>
```

```
      <li>Beetroot</li>
```

```
      <li>Ginger</li>
```

```
      <li>Potato</li>
```

```
      <li>Radish</li>
```

```
    </ol>
```

```
  </body>
```

```
</html>
```

# List Elements in HTML-Un-Order List

Un-Order List :

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>HTML Ordered List</title>
```

```
  </head>
```

```
  <body>
```

```
    <ul>
```

```
      <li>Beetroot</li>
```

```
      <li>Ginger</li>
```

```
      <li>Potato</li>
```

```
      <li>Radish</li>
```

```
    </ul>
```

```
  </body>
```

```
</html>
```

# List Elements in HTML-Definition List

Definition List:

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML Definition List</title>
  </head>
  <body>
    <dl>
      <dt><b>HTML</b></dt>
      <dd>This stands for Hyper Text Markup Language</dd>
      <dt><b>HTTP</b></dt>
      <dd>This stands for Hyper Text Transfer Protocol</dd>
    </dl>
  </body>
</html>
```

# HTML Frames



# HTML Frames

// HTML frames are used to divide your browser window into multiple sections where each section can load a separate HTML document. A collection of frames in the browser window is known as a frameset.

The window is divided into frames in a similar way the tables are organized into rows and columns.

## Creating Frames

=====

To use frames on a page we use `<frameset>` tag instead of `<body>` tag.

The `<frameset>` tag defines, how to divide the window into frames.

The `rows` attribute of `<frameset>` tag defines horizontal frames and `cols` attribute defines vertical frames. Each frame is indicated by `<frame>` tag and it defines which HTML document shall open into the frame.

# Example program on Frameset tag

## frameset tag with cols attribute

```
<!DOCTYPE html>
<html>
<head>
<title>frameset </title>
</head>
<frameset cols="50%,50%">
<frame name=f1 src="one.html">
<frame name=f2 src="two.html">
</frameset>
</html>
```



# Frameset Program with rows Attribute

## frameset with rows attribute

```
<!DOCTYPE html>
<html>
<head>
<title>frameset </title>
</head>
<frameset rows="50%,50%">
<frame name=f1 src="one.html">
<frame name=f2 src="two.html">
</frameset>
</html>
```

# Frameset with rows and columns

---

frameset with rows and cols attribute

```
<html>
```

```
<head>
```

```
<title>frameset </title>
```

```
</head>
```

```
<frameset rows="50%,50%">
```

```
<frame name=f1 src="one.html">
```

```
<frameset cols="50%,50%">
```

```
<frame name=f2 src="two.html">
```

```
<frame name=f3 src="three.html">
```

```
</frameset>
```

```
</html>
```

# HTML Forms



# Html Form Elements

---

There are different types of form controls that you can use to collect data using HTML form –

- Text Input Controls
- Text Area box
- Checkboxes Controls
- Radio Box Controls
- Select Box Controls
- Clickable Buttons
- Submit and Reset Button

# Input Types

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`

# Input Types

```
<input  
type="password">
```

```
<input  
type="radio">
```

```
<input  
type="range">
```

```
<input  
type="reset">
```

```
<input  
type="search">
```

```
<input  
type="submit">
```

```
<input type="tel">
```

```
<input  
type="text">
```

```
<input  
type="time">
```

```
<input type="url">
```

```
<input  
type="week">
```

# Html Form Elements- TextBox , CheckBox

Text Input controls (Textbox, Password)

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head> <title>Password Input Control</title>  </head>
```

```
  <body>
```

```
    <form >
```

```
      User ID : <input type = "text" name = "user_id" /><br>
```

```
      Password: <input type = "password" name = "password" />
```

```
    </form>
```

```
  </body>
```

```
</html>
```

# Html Form Elements- TextArea

Multiple-Line Text Input Controls:

-----

This is used when the user is required to give details that may be longer than a single sentence.

Multi-line input controls are created using HTML `<TextArea>` tag.

`<form>`

Description : `<br>`

`<textarea rows = "5" cols = "50" name = "description">`

Enter description here...

`</textarea>`

`</form>`



# Html Form Elements- Check Box

Checkbox Control:

-----

Checkboxes are used when more than one option is required to be selected.

They are also created using HTML `<input>` tag but type attribute is set to checkbox..

`<form>`

`<input type = "checkbox" name = "cricket" value = "on">` Cricket

`<input type = "checkbox" name = "chess" value = "on">` Chess

`</form>`

# Html Form Elements- Radio Button

## Radio Button Control

---

Radio buttons are used when out of many options, just one option is required to be selected.

They are also created using HTML `<input>` tag but type attribute is set to radio

```
<form>
```

```
  <input type = "radio" name = "subject" value = "maths"> Maths
```

```
  <input type = "radio" name = "subject" value = "physics"> Physics
```

```
</form>
```

# Html Form Elements- Selection Box

## Select Box Control

---

A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options.

```
<form>
```

```
  <select name = "dropdown">
```

```
    <option value = "one" selected>one</option>
```

```
    <option value = "two">two</option>
```

```
    <option value=" three">three</option>
```

```
  </select>
```

```
</form>
```

# Html Form Elements- Button Controls

## Button Controls

---

There are various ways in HTML to create clickable buttons.

You can also create a clickable button using `<input>` tag by setting its type attribute to button

`<form>`

`<input type = "submit" name = "submit" value = "Submit" />`

`<input type = "reset" name = "reset" value = "Reset" />`

`<input type = "button" name = "ok" value = "OK" />`

`</form>`

# HTML5



# HTML 5 Introduction

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0,

and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

HTML5 features:

-----

New Semantic Elements – These are like `<header>`, `<footer>`, and `<section>`.

Forms 2.0 – Improvements to HTML web forms where new attributes have been introduced for `<input>` tag.

# HTML 5 Features

---

Persistent Local Storage – To achieve without resorting to third-party plugins.

WebSocket – A next-generation bidirectional communication technology for web applications.

Server-Sent Events – HTML5 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE).

Canvas – This supports a two-dimensional drawing surface that you can program with JavaScript.

Audio & Video – You can embed audio or video on your webpages without resorting to third-party plugins.

Geolocation – Now visitors can choose to share their physical location with your web application.

# HTML5- SVG





## <SVG> tag

---

SVG stands for Scalable Vector Graphics and it is a language for describing 2D-graphics and graphical applications in XML and the XML is then rendered by an SVG viewer.

SVG is mostly useful for vector type diagrams like Pie charts,

Two-dimensional graphs in an X,Y coordinate system etc.

SVG became a W3C Recommendation 14. January 2003 and you can check latest version of SVG specification at [SVG Specification](#).

### Viewing SVG Files

Most of the web browsers can display SVG just like they can display PNG, GIF, and JPG.

Internet Explorer users may have to install the Adobe SVG

Viewer to be able to view SVG in the browser.

# <SVG> Tag in HTML

HTML5 allows embedding SVG directly using <svg>...</svg>

tag which has following simple syntax –

```
<svg>-----</svg>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>  <title>SVG Tag in HTML5</title> </head>
```

```
  <body>
```

```
    <svg width=100 height=100 style="border:1px solid green">
```

```
      </svg>
```

```
    </body>
```

```
</html>
```

# <SVG> Tag in HTML: Circle

You can use the style attribute which allows you to set additional style information like stroke and fill colors, width of the stroke, etc.

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
  </head>
```

```
  <body>
```

```
    <svg width="200" height="200" style="border:1px solid green;">
```

```
      <circle cx="100" cy="100" r="50" fill="yellow" stroke="red" stroke-width="2px"/>
```

```
    </svg>
```

```
  </body>
```

```
</html>
```

# <SVG> Tag in HTML: Rectangle

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title> HTML5 Rectangle </title>
```

```
  </head>
```

```
  <body>
```

```
    <svg width="200" height="200" style="border:1px solid green;">
```

```
      <rect x=10 y=20 width=100 height=50 fill="yellow" stroke="red" stroke-width="2px"/>
```

```
    </svg>
```

```
  </body>
```

```
</html>
```

## <SVG> Tag in HTML: Line

Following is the HTML5 version of an SVG example which would draw a line using <line> tag –You can use the style attribute which allows you to set additional style information like stroke and fill colors, width of the stroke, etc.

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head></head>
```

```
  <body>
```

```
    <svg width="200" height="200" style="border:1px solid green;">
```

```
      <line x1=20 y1=20 x2=100 y2=100 stroke="red" stroke-width="2px"/>
```

```
    </svg>
```

```
  </body>
```

```
</html>
```

# <SVG> Tag in HTML: Ellipse

Following is the HTML5 version of an SVG example which would draw an ellipse using <ellipse> tag –

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head><title>SVG</title>
```

```
  <meta charset = "utf-8" />
```

```
</head>
```

```
<body>
```

```
  <h2 align = "center">HTML5 SVG Ellipse</h2>
```

```
  <svg width="300" height = "100" style="border: 1px solid;">
```

```
    <ellipse x="10px" y="10px" cx = "100" cy = "50" rx = "100" ry = "50" fill = "red" />
```

```
  </svg> </body> </html>
```

# HTML5-Canvas



## <Canvas> Tag in HTML:

- >The <canvas> element is supported in all recent browsers such as IE 9+, Chrome and Firefox.
- >By default, the canvas element has 300px of width and 150px height without any border and content. The canvas is a 2-dimensional rectangular area.
- >The coordinates of the left-corner of the canvas are (0,0) which is known as origin and co-ordinates in the bottom-right corner are.

HTML5 element <canvas> gives you an easy and powerful way to draw graphics using JavaScript. It can be used to draw graphs, make photo compositions or do simple (and not so simple) animations.

Here is a simple <canvas> element which has only two specific attributes width and height plus all the core HTML5 attributes like id, name and class, etc.

```
<canvas id = "mycanvas" width = "100" height = "100"> </canvas>
```



# <Canvas> Tag in HTML:

## Example Program on <canvas>

```
<!DOCTYPE HTML>
```

```
<html>
```

```
  <head>
```

```
    <style>
```

```
      #mycanvas{border:1px solid red;}
```

```
    </style>
```

```
  </head>
```

```
  <body>
```

```
    <canvas id = "mycanvas"  width = "100" height = "100"></canvas>
```

```
  </body>
```

```
</html>
```

# <Canvas> Tag in HTML: Line

```
<!DOCTYPE html>

<html><head><title>html simple line </title>

<style type="text/css">
canvas{border:1px solid;}
</style>

<script type="text/javascript">
window.onload=function(){
var canvas=document.getElementById("mycanvas");
var context=canvas.getContext("2d");
//here 'W','S','T', and must be in capital letter
context.lineWidth=10; context.strokeStyle="orange";
context.moveTo(50,150);
context.lineTo(250,50); context.stroke();
};</script></head><body><canvas id="mycanvas" width="300" height="200"></canvas></body></html>
```

# <Canvas> Tag in HTML: Line

```
<!DOCTYPE html>

<html><head><title>html simple line </title>

<style type="text/css">
canvas{border:1px solid;}
</style>

<script type="text/JavaScript">
window.onload=function(){
var canvas=document.getElementById("mycanvas");
var context=canvas.getContext("2d");
//here 'W','S','T', and must be in capital letter
context.lineWidth=10; context.strokeStyle="orange";
context.moveTo(50,150);
context.lineTo(250,50); context.rect(x,y,width, height);//context.rect(75,50,200,75);
context.stroke();
};</script></head><body><canvas id="mycanvas" width="300" height="200"></canvas></body></html>
```



# HTML5-Audio & Video



# Tag in HTML: Video

We can use <source> tag to specify media along with media type and many other attributes. A video element allows multiple source elements and browser will use the first recognized format –

```
<!DOCTYPE HTML>
```

```
<html>
```

```
  <body>
```

```
    <video width = "300" height = "200" controls autoplay >
```

```
      <source src = "D:\fullstack\video.mp4" type = "video/mp4" />
```

Your browser does not support the <video> element.

```
    </video>
```

```
  </body>
```

```
</html>
```

# <Canvas> Tag in HTML: Video

We can use <source> tag to specify media along with media type and many other attributes. An audio element allows multiple source elements and browser will use the first recognized format –

```
<!DOCTYPE HTML>
```

```
<html>
```

```
  <body>
```

```
    <audio controls autoplay>
```

```
      <source src = "D:\fullstack\song.mp3" type ="audio/mp3" />
```

```
      Your browser does not support the <audio> element.
```

```
    </audio>
```

```
  </body>
```

```
</html>
```



# Storage in HTML 5



HTML 5 provides mechanisms by which browsers can store key/value pairs.



There are two types

`sessionStorage`

`localStorage`



# localStorage

---

- // Data persists even when the browser is closed and reopened.
- // Stores data with no expiration date, and gets cleared only through JavaScript, or clearing the Browser cache / Locally Stored Data.
- // Storage limit is the maximum amongst the three.

# Example

```
<script>
if( localStorage.hits )
{
localStorage.hits = Number(localStorage.hits)
+1;
}
else { localStorage.hits = 1; }
document.write("Total Hits : " +
localStorage.hits );

</script>
```

# sessionStorage



sessionStorage is a separate storage area for each given origin that's available for the duration of the page session.



Stores data only for a session, meaning that the data is stored until the browser (or tab) is closed.



Data is never transferred to the server.



Storage limit is larger than a cookie

# Example

```
<script>
if( sessionStorage.hits )
{
sessionStorage.hits =
Number(sessionStorage.hits) +1;
}
else { sessionStorage.hits = 1; }
document.write("Total Hits : " +
sessionStorage.hits );
</script>
```



Thank You !



L&T Technology Services

