# **HTML**

Hypertext markup language (set of markup tags)

<doctype! html>

<html>

<*start tag*> content </end tag>

<head>

<title>

<body>

<h></h><p></p>

<br/>

Linkings: <a href=”fb.com”>click facebook</a>

Images: <img src=”Rajesh.jpg”

alt=”Rajesh2.jpg”

width=”104”

height=”120”>

Heading size: <h6>idu dan heading size 6 </h6> h1 h2 h3 h4 h5 h6 available

Paragraph: <p title=”idu title”> this is paragraph</p>

Horizontal line: <hr>

Line break (a enter to next line) :</br>

Comments: <!--this is comment-->

<b>**bold la varum**</b> <strong>**iduvum bold dan</**strong**>**

<i>*italic*</i> <cite> *italic<*/cite*>*

<u>underline</u>

<strike>~~strike~~</strike> <del> ~~strike the word~~</del>

<tt> small la teriyum </tt> <small> small la teriyum </small>

<sub> subscript</sub>

<sup>superscript</sup>

<q>”quotaion”</q>

<address> *google.com* </address>

<p align=”center”></p>

<hr width=”50px”/>

<hr width=”50%”/>

<img src=”image.jpg” width=”150px” height=”150%” border=”1px” alt= ” google.in”/>

<font color=”red”>this is red</font>

<a href=” jkjh.com ” target=”\_blank”open in new window></a>

<ol>

<li>hiii</li> //will give with number in front

<li>hello</li>

</ol>

<ul>

<li>hiii</li> //will give with dot in front of list

<li>hello</li>

</ul>

<table border=”2”>

<tr align=”center”>

<td colspan=”2”></td>

<td bgcolor=”red”></td>

<td></td>

</tr>

</table>

<html>

<body>

<div style="background-color:green; color:white; padding:20px;"></div>

This alone shown in <span style="color:red">red color</span>

</body>

</html>

**Blocks vs Inline:**

Block- they will appear on a new line, and any content that goes after it will also appear on a new line.

<div><p><address><blockquote><dd><dl><dt><fieldset><form><h1><hr><li><main><nav><noscript><ol><pre><tables><tfoot><ul>

html5

<article><aside><canvas><figcaption><footer><figure><header><hgroup><output><section><video>

Inline-surround only small parts of content’

FORM Element

<body>

<form action=”http:rajesh.com” method=”GET”> //show details in url

<input type=”text” name=”usr”> enter username:</input>

</form>

<form action=”http:rajesh.com” method=”POST”> //not show details

<input type=”password” name=”pwd”> enter password:</input>

<input type=”radio” name=”gender” value=”male”> male</input>

<input type=”checkbox” name=”gender” value=”female”> male</input>

<input type=”submit” value=”enter”></input>

</form>

</body>

**HTML 5**

doctype declaration <!doctype html>

The character encoding (charset) declaration <meta charset="UTF-8">

New in HTML5

Forms

- The Web Forms 2.0 specification allows for creation of more powerful forms

- Date pickers, color pickers, and numeric stepper controls have been added.

- Input field types now include email, search, and URL.

- PUT and DELETE form methods are now supported.

Integrated API (Application Programming Interfaces)

- Drag and Drop, Audio and Video, Offline Web Applications, History, Local Storage, Geolocation, Web Messaging.

HTML 5 provide 7Content Models:

Metadata: Content that sets up the presentation or behavior of the rest of the content. These elements are found in the head of the document.

Elements: <base>, <link>, <meta>, <noscript>, <script>, <style>, <title>

Embedded: Content that imports other resources into the document.

Elements: <audio>, <video>, <canvas>, <iframe>, <img>, <math>, <object>, <svg>

Interactive: Content specifically intended for user interaction.

Elements: <a>, <audio>, <video>, <button>, <details>, <embed>, <iframe>, <img>, <input>, <label>, <object>, <select>, <textarea>

Heading: Defines a section header.

Elements: <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <hgroup>

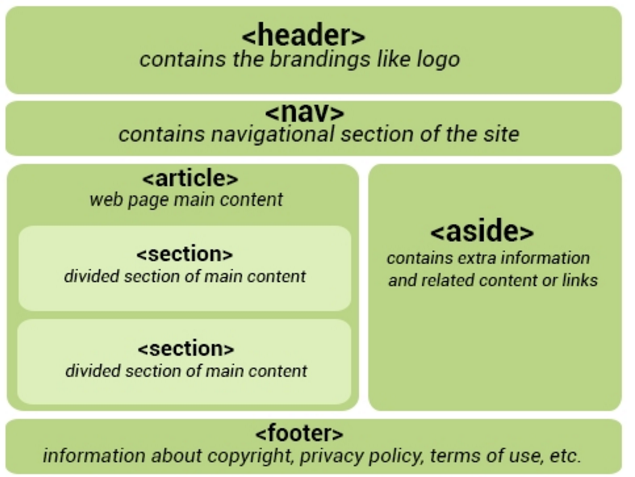
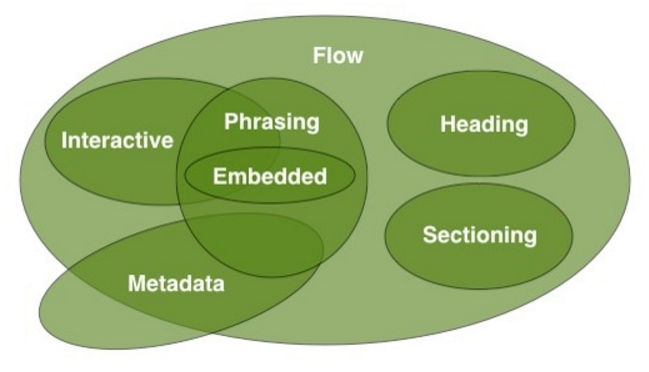
Phrasing: This model has a number of inline level elements in common with HTML4.

Elements: <img>, <span>, <strong>, <label>, <br />, <small>, <sub>, and more. Flow content:

Flow content: Contains the majority of HTML5 elements that would be included in the normal flow of the document.

Sectioning content: Defines the scope of headings, content, navigation, and footers.

Elements: <article>, <aside>, <nav>, <section>



Header is comes inside body tag and it s totally different from <head></head>

<nav>

<ul>

<li> <a href=”home.com” >home</a> </li>

<li> <a href=”tools.com” >tools</a> </li>

</ul>

</nav>

<article>

<h1>newsblogs entries</h1>

<p>Contents of the article element, widgets, gadjet, comment, magazines, forum post </p>

</article>

<article>

<h1>Welcome</h1>

<section>

<h1>Heading</h1>

<p>content or image</p>

</section>

</article>

<article>

<aside>Seperated from article but Indirectly related to the article </aside>

</article>

AUDIO

<audio src="audio.mp3" controls>

Audio element not supported by your browser

</audio>

<audio controls>

<source src="audio.mp3" type="audio/mpeg">

<source src="audio.ogg" type="audio/ogg">

</audio>

<audio controls autoplay> automatically play without need of visitor permission

<audio controls autoplay loop> make a loop

VIDEO

<video controls>

<source src="video.mp4" type="video/mp4"> Video is not supported by your browser

</video>

<video controls autoplay loop>

<source src="video.ogg" type="video/ogg"> Video is not supported by your browser

</video>

PROGRESS BAR

Status loading: <progress min="0" max="100" value="35">

</progress>

HTML5 webstorage:

Before that storage done by javascript cookies

Two types: 1.session storage(destroy when browser close) 2.local storage(local storage)

Storing a Value:

localStorage.setItem("key1", "value1");

sessionStorage.setItem("key1", "value1");

Getting a Value:

//this will print the value

alert(localStorage.getItem("key1"));

alert(sessionStorage.getItem("key1"));

Removing a Value:

localStorage.removeItem("key1");

sessionStorage.removeItem("key1");

Removing All Values:

localStorage.clear();

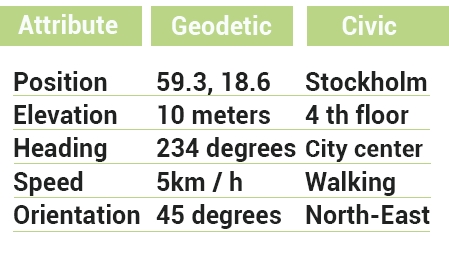
sessionStorage.clear();

**GeoLocation API:**

navigator.geolocation.getCurrentPosition(); showLocation(mandatory)

ErrorHandler(optional)

Options(optional)



**Html5 element can be draggable:**

<img draggable=”true”/>

**Scalabble vector graphics SVG:**used to draw style

drawing circle

<svg width=”1000” height=”1000”>

<circle cx=”80”, cy=”80” r=”50” fill=”green”/>

</svg>

Shapes: circle, rec,line, polyline,ellipse, polygon

|  |  |
| --- | --- |
| cx  cy  r  fill  stoke | pushes the center of the circle further to the right in screen  pushes the center of the circle further down from top of the screen  defines the radius  determines the color of our circle  adds an outline to the circle |

**SVG animations:**

<svg width="1000" height="250">

<rect width="150" height="150" fill="orange">

<animate attributeName="x" from="0" to="300"

dur="3s" fill="freeze" repeatCount="2"/>

</rect>

</svg>

|  |  |
| --- | --- |
| attributeName:  from:  to:  dur:  fill:  repeatCount: | Specifies which attribute will be affected by the animation  Specifies the starting value of the attribute  Specifies the ending value of the attribute  Specifies how long the animation runs (duration)  Specifies the attribute value not return back to intial value after animation  Specifies the repeat count of the animation |

Path:

<svg width="500" height="500">

<path d="M 0 0 L200 200 L200 0 Z" style="stroke:#000; fill:none;" />

</svg>

|  |  |
| --- | --- |
| M  L  H  V  C  S  Q  T  A  Z | moveto  lineto  horizontal lineto  vertical lineto  curveto  smooth curveto  quadratic Bézier curve  smooth quadratic curveto  elliptical Arc  closepath |

**Canvas:** is a container for graphics

<canvas id="canvas1" width="200" height="100">

</canvas>

fillreact(x axis, y axis, width, height); //drawing shape

translate(100, 150); //move

rotate( (Math.PI / 180) \* 25; //rotate

scale(x,y) //size

example:

var canvas = document.getElementById('canvas1');

ctx =canvas.getContext('2d');

ctx.font="bold 22px Tahoma";

ctx.textAlign="start";

ctx.fillText("start", 10, 30);

ctx.translate(100, 150);

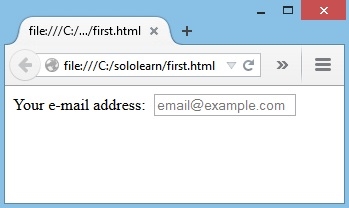
ctx.fillText("after translate", 0, 0);

ctx.rotate(1);

ctx.fillText("after rotate", 0, 0);

ctx.scale(1.5, 4);

ctx.fillText("after scale", 0,20);

****

**Forms:**

<form>

<label for="email">Your e-mail address: </label>

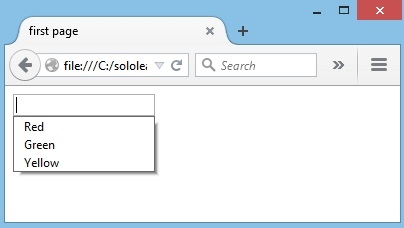
<input type="text" **required //***must needed to fill*

name="email" placeholder="email@example.com" //*display some stuff*

id="mysearch" name="searchitem" type="search" list=”colours” //a search bar

/>

</form>

<datalist id="colors">

<option value="Red">

<option value="Green">

<option value="Yellow">

</datalist>

<input type="text" name="email" autofocus/> // *makes the desired input focus when the form loads(normal uh varum)*

HTML5 added several new input types:

- color

- date

- datetime

- datetime-local

- email

- month

- number

- range

- search

- tel

- time

- url

- week

New input attributes in HTML5:

- autofocus

- form

- formaction

- formenctype

- formmethod

- formnovalidate

- formtarget

- height and width

- list

- min and max

- multiple

- pattern (regexp)

- placeholder

- required

- step