**HTML 5 (5.2)**

1. Introduction to HTML 5
2. HTML 5 Syntax
3. HTML 5 Document
4. HTML 5 Attributes
5. HTML 5 events
6. HTML 5 Webform 2.0
7. HTML 5 SVG
8. HTML 5 MathML
9. HTML 5 web storage
10. HTML 5 Canvas
11. HTML 5 Video and Audio
12. HTML 5 Drag and Drop
13. HTML 5 Geolcation API
14. **Introduction to HTML 5**

HTML 5 is major and latest version of HTML  
All latest browser supports

New features:

* New Semantic Elements: header, footer, section etc..
* Form elements: inside <input> attributes
* Persistent local storage : No third party plugin
* Websocket
* Server-Sent events (SSE)
* Canvas: This supports a two-dimensional drawing surface that you can program with JavaScript
* Audio- and Video
* Geolacation
* Microdata: Custom semantics
* Drag and Drop

1. **HTML 5 Syntax**

* Uppercase tag names.
* Quotes are optional for attributes.
* Attribute values are optional.
* Closing empty elements are optional.

<script type = "text/javascript" src = "example.js"></script>

<script src = "example.js"></script>

<link rel = "stylesheet" type = "text/css" href = "example.css">

<link rel = "stylesheet" href = "example.css">

1. HTML 5 Document



* Section
* Article
* Aside
* Header
* Footer
* Nav

<!DOCTYPE html>

<html>

<head>

<meta charset = "utf-8">

<title>...</title>

</head>

<body>

<header>...</header>

<nav>...</nav>

<article>

<section>

...

</section>

</article>

<aside>...</aside>

<footer>...</footer>

</body>

</html>

<!DOCTYPE html>

<html>

<head>

<meta charset = "utf-8">

<title>...</title>

</head>

<body>

<header role = "banner">

<h1>HTML5 Document Structure Example</h1>

<p>This page should be tried in safari, chrome or Mozila.</p>

</header>

<nav>

<ul>

<li><a href = "http://laratechnology.com/">Lara Java/j2ee</a></li>

<li><a href = "http://laratechnology.com/">Java/J2ee </a></li>

<li><a href = "http://laratechnology.com/">

CurrentBatch</a></li>

</ul>

</nav>

<article>

<section>

<p>Once article can have multiple sections</p>

</section>

</article>

<aside>

<p>This is aside part of the web page</p>

</aside>

<footer>

<p>Created by <a href = "http://laratechnology.com/">Lara Tech</a></p>

</footer>

</body>

</html>

1. **HTML 5 attributes**

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Options** | **Function** |
| accesskey | User Defined | Specifies a keyboard shortcut to access an element. |
| align | right, left, center | Horizontally aligns tags |
| background | URL | Places an background image behind an element |
| bgcolor | numeric, hexidecimal, RGB values | Places a background color behind an element |
| class | User Defined | Classifies an element for use with Cascading Style Sheets. |
| contenteditable | true, false | Specifies if the user can edit the element's content or not. |
| contextmenu | Menu id | Specifies the context menu for an element. |
| data-XXXX | User Defined | Custom attributes. Authors of a HTML document can define their own attributes. Must start with "data-". |
| draggable | true,false, auto | Specifies whether or not a user is allowed to drag an element. |
| height | Numeric Value | Specifies the height of tables, images, or table cells. |
| hidden | hidden | Specifies whether element should be visible or not. |
| id | User Defined | Names an element for use with Cascading Style Sheets. |
| item | List of elements | Used to group elements. |
| itemprop | List of items | Used to group items. |
| spellcheck | true, false | Specifies if the element must have it's spelling or grammar checked. |
| style | CSS Style sheet | Specifies an inline style for an element. |
| subject | User define id | Specifies the element's corresponding item. |
| tabindex | Tab number | Specifies the tab order of an element. |
| title | User Defined | "Pop-up" title for your elements. |
| valign | top, middle, bottom | Vertically aligns tags within an HTML element. |
| width | Numeric Value | Specifies the width of tables, images, or table cells. |

* AccessKey

<!DOCTYPE html>

<html>

<body>

<a href="http://laratechnology.com/" accesskey="h">Lara tech</a><br>

<a href="http://laratechnology.com/" accesskey="c">Lara tech</a>

<p>The accesskey attribute specifies a shortcut key to activate/focus an element.</p>

<p><strong>Note:</strong> The shortcut is varying in different browsers:</p>

<ul>

<li>IE, Chrome, Safari, Opera 15+: [ALT] + <em>accesskey</em></li>

<li>Opera prior version 15: [SHIFT] [ESC] + <em>accesskey</em></li>

<li>Firefox: [ALT] [SHIFT] + <em>accesskey</em></li>

</ul>

</body>

</html>

Custom attributes:

<div class = "example" data-subject = "physics" data-level = "complex">

...

</div>

1. **HTML 5 Events**

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| offline | script | Triggers when the document goes offline |
| onabort | script | Triggers on an abort event |
| onafterprint | script | Triggers after the document is printed |
| onbeforeonload | script | Triggers before the document loads |
| onbeforeprint | script | Triggers before the document is printed |
| onblur | script | Triggers when the window loses focus |
| oncanplay | script | Triggers when media can start play, but might has to stop for buffering |
| oncanplaythrough | script | Triggers when media can be played to the end, without stopping for buffering |
| onchange | script | Triggers when an element changes |
| onclick | script | Triggers on a mouse click |
| oncontextmenu | script | Triggers when a context menu is triggered |
| ondblclick | script | Triggers on a mouse double-click |
| ondrag | script | Triggers when an element is dragged |
| ondragend | script | Triggers at the end of a drag operation |
| ondragenter | script | Triggers when an element has been dragged to a valid drop target |
| ondragleave | script | Triggers when an element leaves a valid drop target |
| ondragover | script | Triggers when an element is being dragged over a valid drop target |
| ondragstart | script | Triggers at the start of a drag operation |
| ondrop | script | Triggers when dragged element is being dropped |
| ondurationchange | script | Triggers when the length of the media is changed |
| onemptied | script | Triggers when a media resource element suddenly becomes empty. |
| onended | script | Triggers when media has reach the end |
| onerror | script | Triggers when an error occur |
| onfocus | script | Triggers when the window gets focus |
| onformchange | script | Triggers when a form changes |
| onforminput | script | Triggers when a form gets user input |
| onhaschange | script | Triggers when the document has change |
| oninput | script | Triggers when an element gets user input |
| oninvalid | script | Triggers when an element is invalid |
| onkeydown | script | Triggers when a key is pressed |
| onkeypress | script | Triggers when a key is pressed and released |
| onkeyup | script | Triggers when a key is released |
| onload | script | Triggers when the document loads |
| onloadeddata | script | Triggers when media data is loaded |
| onloadedmetadata | script | Triggers when the duration and other media data of a media element is loaded |
| onloadstart | script | Triggers when the browser starts to load the media data |
| onmessage | script | Triggers when the message is triggered |
| onmousedown | script | Triggers when a mouse button is pressed |
| onmousemove | script | Triggers when the mouse pointer moves |
| onmouseout | script | Triggers when the mouse pointer moves out of an element |
| onmouseover | script | Triggers when the mouse pointer moves over an element |
| onmouseup | script | Triggers when a mouse button is released |
| onmousewheel | script | Triggers when the mouse wheel is being rotated |
| onoffline | script | Triggers when the document goes offline |
| onoine | script | Triggers when the document comes online |
| ononline | script | Triggers when the document comes online |
| onpagehide | script | Triggers when the window is hidden |
| onpageshow | script | Triggers when the window becomes visible |
| onpause | script | Triggers when media data is paused |
| onplay | script | Triggers when media data is going to start playing |
| onplaying | script | Triggers when media data has start playing |
| onpopstate | script | Triggers when the window's history changes |
| onprogress | script | Triggers when the browser is fetching the media data |
| onratechange | script | Triggers when the media data's playing rate has changed |
| onreadystatechange | script | Triggers when the ready-state changes |
| onredo | script | Triggers when the document performs a redo |
| onresize | script | Triggers when the window is resized |
| onscroll | script | Triggers when an element's scrollbar is being scrolled |
| onseeked | script | Triggers when a media element's seeking attribute is no longer true, and the seeking has ended |
| onseeking | script | Triggers when a media element's seeking attribute is true, and the seeking has begun |
| onselect | script | Triggers when an element is selected |
| onstalled | script | Triggers when there is an error in fetching media data |
| onstorage | script | Triggers when a document loads |
| onsubmit | script | Triggers when a form is submitted |
| onsuspend | script | Triggers when the browser has been fetching media data, but stopped before the entire media file was fetched |
| ontimeupdate | script | Triggers when media changes its playing position |
| onundo | script | Triggers when a document performs an undo |
| onunload | script | Triggers when the user leaves the document |
| onvolumechange | script | Triggers when media changes the volume, also when volume is set to "mute" |
| onwaiting | script | Triggers when media has stopped playing, but is expected to resume |

1. **HTML 5 webform 2.0**

<label for = "firstname">first name: </label>

<input type = "text" id = "firstname"><br />

<label for = "lastname">last name: </label>

<input type = "text" id = "lastname"><br />

<label for = "email">email: </label>

<input type = "text" id = "email"><br>

<input type = "radio" name = "sex" value = "male"> Male<br>

<input type = "radio" name = "sex" value = "female"> Female<br>

<input type = "submit" value = "send"> <input type = "reset">

* + Datetime

<!DOCTYPE HTML>

<html>

<body>

<form action = "" method = "get">

Date and Time : <input type = "datetime" name = "newinput" />

<input type = "submit" value = "submit" />

</form>

</body>

</html>

* + Datetime local

<!DOCTYPE HTML>

<html>

<body>

<form action = "" method = "get">

Local Date and Time : <input type = "datetime-local" name = "newinput" />

<input type = "submit" value = "submit" />

</form>

</body>

</html>

* + Date

<!DOCTYPE HTML>

<html>

<body>

<form action = "" method = "get">

Date : <input type = "date" name = "newinput" />

<input type = "submit" value = "submit" />

</form>

</body>

</html>

* + Month/week/time

<!DOCTYPE HTML>

<html>

<body>

<form action = "/cgi-bin/html5.cgi" method = "get">

Month : <input type = "month" name = "newinput" />

<input type = "submit" value = "submit" />

</form>

</body>

</html>

* + Number

<input type = "number" min = "0" max = "10" step "1"

value = "5" name = "newinput" />

<input type = "submit" value = "submit" />

* + Range

<!DOCTYPE HTML>

<html>

<body>

<form action = “" method = "get">

Select Range : <input type = "range" min = "0" max = "10" step "1"

value = "5" name = "newinput" />

<input type = "submit" value = "submit" />

</form>

</body>

</html>

* + Email
  + url
  + Placeholder

<input type = "email" name = "newinput"

placeholder = "email@example.com"/>

* + Autofocus

<input type = "text" name = "search" autofocus/>

* + Required

<input type = "text" name = "newinput" required/>

1. **HTML 5 SVG**

Scalable Vector Graphics , like pie chart

2-dimensional charts

* + SVG Circle

<!DOCTYPE html>

<html>

<head>

<style>

#svgelem {

position: relative;

left: 50%;

-webkit-transform: translateX(-20%);

-ms-transform: translateX(-20%);

transform: translateX(-20%);

}

</style>

<title>SVG</title>

<meta charset = "utf-8" />

</head>

<body>

<h2 align = "center">HTML5 SVG Circle</h2>

<svg id = "svgelem" height = "200" >

<circle id = "redcircle" cx = "50" cy = "50" r = "50" fill = "red" />

</svg>

</body>

</html>

* + Rectangle

<!DOCTYPE html>

<html>

<head>

<style>

#svgelem {

position: relative;

left: 50%;

-webkit-transform: translateX(-50%);

-ms-transform: translateX(-50%);

transform: translateX(-50%);

}

</style>

<title>SVG</title>

<meta charset = "utf-8" />

</head>

<body>

<h2 align = "center">HTML5 SVG Rectangle</h2>

<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">

<rect id = "redrect" width = "300" height = "100" fill = "red" />

</svg>

</body>

</html>

* + SVG Line

<!DOCTYPE html>

<html>

<head>

<style>

#svgelem {

position: relative;

left: 50%;

-webkit-transform: translateX(-50%);

-ms-transform: translateX(-50%);

transform: translateX(-50%);

}

</style>

<title>SVG</title>

<meta charset = "utf-8" />

</head>

<body>

<h2 align = "center">HTML5 SVG Line</h2>

<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">

<line x1 = "100" y1 = "0" x2 = "0" y2 = "100"

style = "stroke:red;stroke-width:2"/>

</svg>

</body>

</html>

* + Ellipse

<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">

<ellipse cx = "100" cy = "50" rx = "100" ry = "50" fill = "red" />

</svg>

* + Polygon

<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">

<polygon points = "20,10 300,20, 170,50" fill = "red" />

</svg>

* + Polyline

<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">

<polyline points = "0,0 0,20 20,20 20,40 40,40 40,60" fill = "red" />

</svg>

* + Star

<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">

<polygon points = "100,10 40,180 190,60 10,60 160,180" fill = "red"/>

</svg>

1. **HTML 5 Math ML**

<https://www.w3.org/TR/MathML2/chapter3.html#presm.mi>

* + x 2 + 4 ⁢ x + 4 = 0

<!doctype html>

<html>

<head>

<meta charset = "UTF-8">

<title>MathML Examples</title>

</head>

<body>

<math xmlns = "http://www.w3.org/1998/Math/MathML">

<mrow>

<mrow>

<msup>

<mi>x</mi>

<mn>2</mn>

</msup>

<mo>+</mo>

<mrow>

<mn>4</mn>

<mo>⁢</mo>

<mi>x</mi>

</mrow>

<mo>+</mo>

<mn>4</mn>

</mrow>

<mo>=</mo>

<mn>0</mn>

</mrow>

</math>

</body>

</html>

<!doctype html>

<html>

<head>

<meta charset = "UTF-8">

<title>MathML Examples</title>

</head>

<body>

<math xmlns = "http://www.w3.org/1998/Math/MathML">

<mrow>

<mi>A</mi>

<mo>=</mo>

<mfenced open = "[" close="]">

<mtable>

<mtr>

<mtd><mi>x</mi></mtd>

<mtd><mi>y</mi></mtd>

</mtr>

<mtr>

<mtd><mi>z</mi></mtd>

<mtd><mi>w</mi></mtd>

</mtr>

</mtable>

</mfenced>

</mrow>

</math>

</body>

</html>

1. **HTML 5 webstorage**
   * Session storage : available until the browser is alive

<!DOCTYPE HTML>

<html>

<body>

<script type = "text/javascript">

if( sessionStorage.hits ) {

sessionStorage.hits = Number(sessionStorage.hits) +1;

} else {

sessionStorage.hits = 1;

}

document.write("Total Hits :" + sessionStorage.hits );

</script>

<p>Refresh the page to increase number of hits.</p>

<p>Close the window and open it again and check the result.</p>

</body>

</html>

* + Local storage : No expiration, within same window storage available

<!DOCTYPE HTML>

<html>

<body>

<script type = "text/javascript">

if( localStorage.hits ) {

localStorage.hits = Number(localStorage.hits) +1;

} else {

localStorage.hits = 1;

}

document.write("Total Hits :" + localStorage.hits );

</script>

<p>Refresh the page to increase number of hits.</p>

<p>Close the window and open it again and check the result.</p>

</body>

</html>

* + Clear storage

localStorage.clear();

1. **HTML 5 Canvas**

<html>

<head>

<style>

#mycanvas{border:1px solid red;}

</style>

</head>

<body>

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

</body>

</html>

<!DOCTYPE HTML>

<html>

<head>

<style>

#test {

width: 100px;

height:100px;

margin: 0px auto;

}

</style>

<script type = "text/javascript">

function drawShape() {

// Get the canvas element using the DOM

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

// Make sure we don't execute when canvas isn't supported

if (canvas.getContext) {

// use getContext to use the canvas for drawing

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

// Draw shapes

ctx.fillRect(25,25,100,100);

ctx.clearRect(45,45,60,60);

ctx.strokeRect(50,50,50,50);

} else {

alert('You need Safari or Firefox 1.5+ to see this demo.');

}

}

</script>

</head>

<body id = "test" onload = "drawShape();">

<canvas id = "mycanvas"></canvas>

</body>

</html>

1. **HTML 5 Video and Audio**

<https://www.sample-videos.com/index.php#sample-mp4-video>

.ogg, mpeg4, mp4

* + <audio> and <video> tag

<!DOCTYPE HTML>

<html>

<body>

<audio controls autoplay>

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

Your browser does not support the <audio> element.

</audio>

</body>

</html>

* + Video

<html>

<body>

<video src = "SampleVide.mp4" width = "300" height = "200" controls>

Your browser does not support the <video> element.

</video>

</body>

</html>

<html>

<body>

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

<source src = " foo.ogg" type ="video/ogg" />

<source src = " foo.mp4" type = "video/mp4" />

Your browser does not support the <video> element.

</video>

</body>

</html>

1. **Drag and Drop**

<!DOCTYPE HTML>

<html>

<head>

<style>

#div1 {

width: 350px;

height: 70px;

padding: 10px;

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>

</head>

<body>

<p>Drag the image into the rectangle:</p>

<div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)"></div>

<br>

<img id="drag1" src="img\_logo.gif" draggable="true" ondragstart="drag(event)" width="336" height="69">

</body>

</html>

1. **Geolocation API**

Var geolocation = navigator.geolocation;

HTML5 Geolocation API lets you share your location with your favorite web sites. A JavaScript can capture your latitude and longitude and can be sent to backend web server and do fancy location-aware things like finding local businesses or showing your location on a map.

Today most of the browsers and mobile devices support Geolocation API.

Geolocation is most accurate for devices with GPS, like iPhone.

<!DOCTYPE html>

<html>

<body>

<p id="demo">Click the button to get your position.</p>

<button onclick="getLocation()">Try It</button>

<div id="mapholder"></div>

<script>

var x = document.getElementById("demo");

function getLocation() {

if (navigator.geolocation) {

navigator.geolocation.getCurrentPosition(showPosition, showError);

} else {

x.innerHTML = "Geolocation is not supported by this browser.";

}

}

function showPosition(position) {

var latlon = position.coords.latitude + "," + position.coords.longitude;

var img\_url = "https://maps.googleapis.com/maps/api/staticmap?center="

+latlon+"&zoom=14&size=400x300&key=AIzaSyBu-916DdpKAjTmJNIgngS6HL\_kDIKU0aU";

document.getElementById("mapholder").innerHTML = "<img src='"+img\_url+"'>";

}

function showError(error) {

switch(error.code) {

case error.PERMISSION\_DENIED:

x.innerHTML = "User denied the request for Geolocation."

break;

case error.POSITION\_UNAVAILABLE:

x.innerHTML = "Location information is unavailable."

break;

case error.TIMEOUT:

x.innerHTML = "The request to get user location timed out."

break;

case error.UNKNOWN\_ERROR:

x.innerHTML = "An unknown error occurred."

break;

}

}

</script>

</body>

</html>

|  |  |
| --- | --- |
| Sr.No. | Method & Description |
| 1 | [getCurrentPosition()](https://www.tutorialspoint.com/html5/geolocation_getcurrentposition.htm)  This method retrieves the current geographic location of the user. |
| 2 | [watchPosition()](https://www.tutorialspoint.com/html5/geolocation_watchposition.htm)  This method retrieves periodic updates about the current geographic location of the device. |
| 3 | [clearWatch()](https://www.tutorialspoint.com/html5/geolocation_clearwatch.htm)  This method cancels an ongoing watchPosition call. |