

# ActiveNET

101, Manjeera Plaza, Opp: Aditya  
Park Inn, Ameerpet, HYDERABAD –  
500038  
98 48 111 2 88

**By Suryanarayana**

## Web Standards Curriculum

The history of Web and Internet & the evolution of web standards
How does Internet works
Web Standard Model – HTML, CSS and JavaScript
Planning out a web site
What does a good web page need?
Color Theory
Building up a site wireframe
Color schemes and design mockups – KISS philosophy
Typography on the web

## HTML Curriculum

Markup textual content in HTML
HTML Lists
Images in HTML
HTML Links
HTML Tables
HTML Forms – Basic & Additions
HTML Structural Elements
Lesser – Knows Semantics Elements
Creating multiple pages with navigation menus
Validating your HTML
Accessibility basics

## CSS Curriculum

CSS Basics
Advanced CSS Selectors
Inheritance and Cascade
CSS text styling Part – 1
CSS Layout Model - Boxes, Borders, Margins, Padding
CSS Background Images
Styling Lists and Links
Styling tables

Styling Forms
Floats and Clearing
CSS static and relative Positioning
CSS absolute and relative Positioning
Debugging CSS
CSS Shorthand reference
Headers, Footers, Columns and Templates

<b>Java Script Curriculum</b>
A short History of JavaScript
Programming – the real basics
What can you do with JavaScript
Your first look at JavaScript
JavaScript best practices
The principles of unobtrusive JavaScript
JavaScript Functions
Objects in JavaScript
Traversing the DOM
Creating and Modifying HTML
Dynamic style – manipulating CSS with JavaScript
Handling events with JavaScript
JavaScript animation
SVG Basics

<b>HTML 5 Curriculum</b>
<ul style="list-style-type: none"> <li>• Introduction</li> </ul>
<ul style="list-style-type: none"> <li>• Laying out a Page with HTML5 Page Structure, New HTML5 Structural Tags, Page Simplification</li> </ul>
<ul style="list-style-type: none"> <li>• Sections and Articles The section Tag, The article Tag, Outlining, Accessibility</li> </ul>
<ul style="list-style-type: none"> <li>• HTML5 Audio and Video Supported Media Types, The audio Element (Audio Formats, Multiple Sources, Audio Tag Attributes, Getting and Creating Audio Files) The video Element (Video Tag Attributes, Creating and Converting Video Files)</li> </ul>
<ul style="list-style-type: none"> <li>• HTML5 Forms New Input Types (search, tel, url and email, date/time input types, number, range, min, max, and step attributes, color)</li> </ul>
<ul style="list-style-type: none"> <li>• HTML5 New Form Attributes Autocomplete, novalidate</li> </ul>
<ul style="list-style-type: none"> <li>• HTML5 New Form Field Attributes required, placeholder, autofocus, autocomplete, form, pattern</li> </ul>
<ul style="list-style-type: none"> <li>• New Form Elements Datalist, progress and meter</li> </ul>

<ul style="list-style-type: none"> <li>HTML5 Web Storage Overview of HTML5 Web Storage. Web Storage (Browser Support, Local Storage, Session Storage, Prefixing your Keys) Other Storage Methods (Web Database Storage, Indexed Database API)</li> </ul>
<ul style="list-style-type: none"> <li>HTML5 Canvas Getting Started with Canvas, Drawing Lines (Multiple Sub-Paths, The Path Drawing Process, The fill() Method) Color and Transparency, Rectangles, Circles and Arcs, Quadratic and Bezier Curves, Images (drawImage() – Basic, drawImage() – Sprites), Text, Integrated APIs, Offline Application API (Cache Manifest File, The HTML File, Managing Application Cache with JavaScript, A Sample Application), Drag and Drop API</li> </ul>

<b>CSS3 Curriculum</b>
<ul style="list-style-type: none"> <li><b>Initials (Fonts, Text Shadows, Font Face, Selectors (Positional selector, Target Selector, Selection Selector))</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Styling Form Inputs (Stateful Selectors)</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Layout (Layouts, BoxLayout)</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Columns (Demo Columns, Demo Box Layout, Background, Demo Multiple Image Background)</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Colors (Opacity, RGBA, HSLA), Alpha, Current Color, Gradients</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Borders (Resizable Borders, Rounded Corners, Shadowing, Complex Box Shadows)</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Transforms (Rotate, Skew, Scale, Translate)</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Animations (Animations, Key Frames, Key Frames &amp; Animations, Transitions)</b></li> </ul>
<ul style="list-style-type: none"> <li><b>Media Queries (Overview of RWD, Responding to View Dimensions, Responsive to Device Constraints, Applying Media Queries)</b></li> </ul>

<b>jQuery Curriculum</b>
<b>jQuery Fundamentals</b>
<ul style="list-style-type: none"> <li>Traversing HTML with the jQuery() function</li> <li>Introducing the Sizzle CSS selector engine</li> <li>Matching nodes by element name, ID, class, position, content, behavior and context</li> <li>Chaining calls to the jQuery() function</li> </ul>
<b>Manipulating the Document Object Model (DOM) for Cross-Browser DHTML</b>
<b>Adding and replacing content with jQuery</b>
<ul style="list-style-type: none"> <li>Updating, adding and deleting element content</li> <li>Inserting nodes into the DOM and manipulating parents and siblings</li> </ul>
<b>Dynamically assigning CSS properties</b>
<ul style="list-style-type: none"> <li>Adding and removing CSS rules and classes</li> <li>Controlling element size and position</li> </ul>
<b>Creating accessible, unobtrusive JavaScript</b>
<ul style="list-style-type: none"> <li>Leveraging the .ready() method</li> <li>Assigning event handlers programmatically</li> </ul>
<b>Animating Web Pages with jQuery Effects</b>

Enhancing pages with animations
<ul style="list-style-type: none"> <li>• Controlling visibility with <code>.hide()</code>, <code>.show()</code> and <code>.toggle()</code></li> <li>• Combining animations and responding to callbacks</li> <li>• Leveraging object literals to control animations</li> <li>• Developing custom animations with <code>.animate()</code></li> </ul>
Building Responsive Pages with Ajax
Updating page components asynchronously
<ul style="list-style-type: none"> <li>• Downloading HTML with the <code>.load()</code> method</li> <li>• Calling web services with <code>.get()</code> and <code>.post()</code></li> <li>• Replacing callbacks with chained deferred objects</li> <li>• Combining Ajax calls with <code>.when()</code> and <code>.then()</code></li> </ul>
Converting serialized server data to HTML
<ul style="list-style-type: none"> <li>• Retrieving JSON with <code>.getJSON()</code></li> <li>• Returning and parsing XML with <code>.ajax()</code></li> </ul>
Designing User-Friendly Forms
Exploring jQuery tools for forms manipulation
<ul style="list-style-type: none"> <li>• Selecting and setting focus on the first element</li> <li>• Responding to focus and blur events</li> <li>• Providing real-time feedback via keyboard events</li> <li>• Integrating a validation plugin</li> </ul>
Extending jQuery with Plugins
Integrating popular plugins
<ul style="list-style-type: none"> <li>• Leveraging contributions from the jQuery community</li> <li>• Extracting embedded data from semantic HTML</li> <li>• Manipulating images with slideshows and carousels</li> </ul>
Developing jQuery plugins
<ul style="list-style-type: none"> <li>• Conforming to best practices and naming conventions</li> <li>• Handling multiple elements</li> <li>• Enabling method chaining</li> <li>• Aliasing <code>\$</code> to avoid namespace conflicts</li> <li>• Consuming object literals to override defaults</li> </ul>
Creating Sophisticated User Interfaces with jQuery UI
Integrating widgets for highly interactive web applications
<ul style="list-style-type: none"> <li>• Optimizing screen real estate with accordions and tabs</li> <li>• Adding resizable, floating windows with dialog</li> <li>• Providing input assistance with autocomplete</li> </ul>
Implementing effects

- Creating color animations with .animate()
- Leveraging complex animation easings and effects

#### Adding Web 2.0 functionality

- Resizing HTML elements
- Building drag-and-drop user interfaces

### Bootstrap Curriculum

- Introduction to Bootstrap
- Quick styles after Dark
- Our First Web Page
- What's is inside
- Scripts & Styles

#### Layout with Bootstrap

- Grid Layouts
- Simple Layout
- Fixed Grids
- Fluid Grids
- Responsive Design
- Responsive Utilities

#### Daily use of Bootstrap

- Typography
- Tables
- Forms
- Buttons
- Images & Icons

#### Bootstrap Components

- Drop Down Menus
- Buttons with Menus
- Tabs and Pills
- The Nav Bar
- Heros, Badges, Lables and Media

#### Bootstrap and JavaScript

- More Buttons
- Modals
- Tooltips & popovers
- Scrollspy
- Typeahead
- Carousel

### AngularJS Curriculum

- **Introduction to AngularJS**
  - **What does AngularJS do for me?**
  - **Who controls AngularJS?**
  - **How can I get AngularJS?**

<ul style="list-style-type: none"> <li>• <b>Our first AngularJS application</b> <ul style="list-style-type: none"> <li>○ <b>A basic application</b></li> <li>○ <b>Using angular-seed</b></li> <li>○ <b>The pieces of the puzzle</b> <ul style="list-style-type: none"> <li>▪ <b>Two-way data binding</b></li> <li>▪ <b>Directives</b></li> </ul> </li> <li>○ <b>How it fits together</b> <ul style="list-style-type: none"> <li>▪ <b>How much of the page is an Angular application?</b></li> </ul> </li> <li>○ <b>Model, View, Controller from the AngularJS Perspective</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Single Page Applications</b> <ul style="list-style-type: none"> <li>○ <b>What do we mean by Single Page Application?</b></li> <li>○ <b>Creating Angular Modules</b></li> <li>○ <b>Using Angular's Routing Service</b> <ul style="list-style-type: none"> <li>▪ <b>Routing Basics</b></li> <li>▪ <b>Accessing URL Data</b></li> <li>▪ <b>Using the \$location Service</b></li> </ul> </li> <li>○ <b>Creating a Skeleton Single Page Application</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Controllers</b> <ul style="list-style-type: none"> <li>○ <b>Where Controllers fit in, and what they do, from Angular's perspective</b></li> <li>○ <b>Managing Scope</b></li> <li>○ <b>Setting up Behavior</b></li> <li>○ <b>Building a basic controller</b></li> <li>○ <b>A more advanced controller</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Models</b> <ul style="list-style-type: none"> <li>○ <b>How to create a model</b></li> <li>○ <b>Explicit models</b></li> <li>○ <b>Implicit models</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Views</b> <ul style="list-style-type: none"> <li>○ <b>Angular's take on the View: a little bit different</b></li> <li>○ <b>Tying a View to a Controller</b></li> <li>○ <b>Tying a View to a model</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Expressions</b> <ul style="list-style-type: none"> <li>○ <b>Expressions are lightweight code snippets</b></li> <li>○ <b>Expression capabilities</b></li> <li>○ <b>Limitations</b></li> <li>○ <b>The border between expressions and \$eval</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Filters</b> <ul style="list-style-type: none"> <li>○ <b>Standard filters</b></li> <li>○ <b>Writing your own filter</b></li> <li>○ <b>Tying filters together</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Scopes</b> <ul style="list-style-type: none"> <li>○ <b>What are scopes?</b></li> <li>○ <b>What do scopes provide?</b></li> <li>○ <b>Scope lifecycle</b></li> <li>○ <b>Scopes as glue between controller and view</b></li> <li>○ <b>Scope hierarchies</b></li> <li>○ <b>Scope and events</b></li> </ul> </li> </ul>

<ul style="list-style-type: none"> <li>• <b>Angular Forms</b> <ul style="list-style-type: none"> <li>○ <b>Angular forms vs HTML forms</b></li> <li>○ <b>Angular form controls</b></li> <li>○ <b>Form events</b></li> <li>○ <b>The form controller</b></li> <li>○ <b>Form validation</b> <ul style="list-style-type: none"> <li>▪ <b>CSS classes for form data</b></li> </ul> </li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Ajax, Data, and Angular</b> <ul style="list-style-type: none"> <li>○ <b>High level interactions with servers</b></li> <li>○ <b>Low-level server interactions with \$http</b></li> <li>○ <b>The deferred/promises API</b></li> <li>○ <b>Making RESTful Service calls with \$resource</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Directives</b> <ul style="list-style-type: none"> <li>○ <b>Teaching HTML new tricks</b></li> <li>○ <b>Binding text and attributes</b></li> <li>○ <b>Directive processing lifecycle</b> <ul style="list-style-type: none"> <li>▪ <b>DOM Processing</b></li> <li>▪ <b>Compilation</b></li> <li>▪ <b>Linking</b></li> </ul> </li> <li>○ <b>A basic directive</b></li> <li>○ <b>Directives and scopes</b></li> <li>○ <b>Creating reusable directives</b></li> <li>○ <b>Turning directives into components</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Testing in Angular</b> <ul style="list-style-type: none"> <li>○ <b>Unit testing</b> <ul style="list-style-type: none"> <li>▪ <b>Working with Dependency Injection</b></li> <li>▪ <b>Other unit testing issues</b></li> </ul> </li> <li>○ <b>End-to-end testing</b> <ul style="list-style-type: none"> <li>▪ <b>Angular's E2E testing framework</b></li> <li>▪ <b>Commands and expectations</b></li> <li>▪ <b>Controlling what happens before and after the test</b></li> <li>▪ <b>Running a scenario</b></li> </ul> </li> </ul> </li> </ul>


<b>node.js Curriculum</b>
<ul style="list-style-type: none"> <li>• <b>Introduction</b></li> </ul>
<ul style="list-style-type: none"> <li>• <b>Foundation</b> <ul style="list-style-type: none"> <li>○ <b>The Node.js framework</b></li> <li>○ <b>Installing Node.js</b></li> <li>○ <b>Using Node.js to execute scripts</b></li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• <b>Node Projects</b> <ul style="list-style-type: none"> <li>○ <b>The Node Package Manager</b></li> <li>○ <b>Creating a project</b></li> <li>○ <b>The package.json configuration file</b></li> <li>○ <b>Global vs. local package installation</b></li> <li>○ <b>Automating tasks with Gulp.</b></li> </ul> </li> </ul>





<ul style="list-style-type: none"> <li>• HTTP <ul style="list-style-type: none"> <li>○ The HTTP protocol</li> <li>○ Building an HTTP server</li> <li>○ Rendering a response</li> <li>○ Processing query strings</li> <li>○ Using Representational State Transfer</li> <li>○ Configuring TLS</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• File System <ul style="list-style-type: none"> <li>○ Synchronous vs. asynchronous I/O</li> <li>○ Path and directory operations</li> <li>○ <code>__dirname</code> and <code>__filename</code></li> <li>○ Asynchronous file reads and writes</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• Buffers, Streams, and Events <ul style="list-style-type: none"> <li>○ Using buffers for binary data</li> <li>○ Flowing vs. non-flowing streams</li> <li>○ Streaming I/O from files and other sources</li> <li>○ Processing streams asynchronously</li> <li>○ Configuring event handlers</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• Modules and Unit Testing <ul style="list-style-type: none"> <li>○ Modularization</li> <li>○ The CommonJS and RequireJS specifications</li> <li>○ Defining modules with exports</li> <li>○ Modules are singletons</li> <li>○ Creating a package</li> <li>○ Module scope and construction</li> <li>○ Unit testing frameworks</li> <li>○ What to test and how to test it</li> <li>○ Building unit tests with Mocha</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• Express <ul style="list-style-type: none"> <li>○ The model-view-controller pattern</li> <li>○ Defining Jade and Handlebars templates</li> <li>○ Building a front-end controller</li> <li>○ Defining routes</li> <li>○ Creating actions</li> <li>○ Configuring Express to use Handlebars</li> <li>○ Using REST</li> <li>○ Reading POST data</li> <li>○ Building Handlebars helpers</li> <li>○ Adding middleware</li> </ul> </li> </ul>
<ul style="list-style-type: none"> <li>• Data Sources <ul style="list-style-type: none"> <li>○ How Node.js connects to databases</li> <li>○ RDBMS databases and NoSQL databases</li> <li>○ Connecting to RDBMS and NoSQL databases</li> <li>○ Performing CRUD operations</li> <li>○ Building client requests to web services</li> </ul> </li> </ul>













Tag	Description	Version
<!--...-->	Specifies a comment	
<!DOCTYPE>	Specifies the document type	
<a>	Specifies an anchor	
<abbr>	Specifies an abbreviation	
<acronym>	Specifies an acronym	
<address>	Specifies an address element	
<applet>	Deprecated. Specifies an applet	
<area>	Specifies an area inside an image map	
<article>	Specifies an article	5
<aside>	Specifies some content loosely related to the page content. If it is removed, the remaining content still makes sense	5
<audio>	Specifies a sound content	5
<b>	Specifies bold text	
<base>	Specifies a base URL for all the links in a page	
<basefont>	Deprecated. Specifies a base font	
<bdo>	Specifies the direction of text display	
<bdi>	Represents text that must be isolated from its surrounding for bidirectional text formatting. It allows embedding a span of text with a different, or unknown, directionality	5
<bgsound>	Specifies background music	
<big>	Specifies big text	

<b>&lt;blink&gt;</b>	Specifies a text which blinks	
<b>&lt;blockquote&gt;</b>	Specifies a long quotation	
<b>&lt;body&gt;</b>	Specifies the body element	
<b>&lt;br&gt;</b>	Inserts a single line break	
<b>&lt;button&gt;</b>	Specifies a push button	
<b>&lt;canvas&gt;</b>	For making graphics with a script	
<b>&lt;caption&gt;</b>	Specifies a table caption	
<b>&lt;center&gt;</b>	Deprecated. Specifies centered text	
<b>&lt;cite&gt;</b>	Specifies a citation	
<b>&lt;code&gt;</b>	Specifies computer code text	
<b>&lt;col&gt;</b>	Specifies attributes for table columns	
<b>&lt;colgroup&gt;</b>	Specifies groups of table columns	
<b>&lt;comment&gt;</b>	Puts a comment in the document	
<b>&lt;datalist&gt;</b>	A list of options for input values	
<b>&lt;dd&gt;</b>	Specifies a definition description	
<b>&lt;del&gt;</b>	Specifies deleted text	
<b>&lt;dfn&gt;</b>	Specifies a definition term	
<b>&lt;dialog&gt;</b>	Specifies a dialog box or window	
<b>&lt;dir&gt;</b>	Deprecated. Specifies a directory list	
<b>&lt;div&gt;</b>	Specifies a section in a document	
<b>&lt;dl&gt;</b>	Specifies a definition list	
<b>&lt;dt&gt;</b>	Specifies a definition term	
<b>&lt;em&gt;</b>	Specifies emphasized text	
<b>&lt;embed&gt;</b>	Specifies a container for an external (non-HTML) application	
<b>&lt;fieldset&gt;</b>	Specifies a fieldset	

<b>&lt;figcaption&gt;</b>	Specifies a caption for a <figure> element	
<b>&lt;figure&gt;</b>	Specifies self-contained content	
<b>&lt;font&gt;</b>	Deprecated. Specifies text font, size, and color	
<b>&lt;footer&gt;</b>	Specifies a footer for a document or section	
<b>&lt;form&gt;</b>	Specifies a form	
<b>&lt;frame&gt;</b>	Specifies a sub window (a frame)	
<b>&lt;frameset&gt;</b>	Specifies a set of frames	
<b>&lt;h1&gt; to &lt;h6&gt;</b>	Specifies header 1 to header 6	
<b>&lt;head&gt;</b>	Specifies information about the document	
<b>&lt;header&gt;</b>	Specifies a header for a document or section	
<b>&lt;hr&gt;</b>	Specifies a horizontal rule	
<b>&lt;html&gt;</b>	Specifies an html document	
<b>&lt;i&gt;</b>	Specifies italic text	
<b>&lt;iframe&gt;</b>	Specifies an inline sub window (frame)	
<b>&lt;ilayer&gt;</b>	Specifies an inline layer	
<b>&lt;img&gt;</b>	Specifies an image	
<b>&lt;input&gt;</b>	Specifies an input field	
<b>&lt;ins&gt;</b>	Specifies inserted text	
<b>&lt;isindex&gt;</b>	Deprecated. Specifies a single-line input field	
<b>&lt;kbd&gt;</b>	Specifies keyboard text	
<b>&lt;keygen&gt;</b>	Generate key information in a form	
<b>&lt;label&gt;</b>	Specifies a label for a form control	
<b>&lt;layer&gt;</b>	Specifies a layer	
<b>&lt;legend&gt;</b>	Specifies a title in a fieldset	
<b>&lt;li&gt;</b>	Specifies a list item	

<b>&lt;link&gt;</b>	Specifies a resource reference	
<b>&lt;main&gt;</b>	Specifies the main or important content in the document. There is only one element in the document	
<b>&lt;map&gt;</b>	Specifies an image map	
<b>&lt;mark&gt;</b>	Specifies a text highlighted for reference purposes, that is for its relevance in another context	
<b>&lt;marquee&gt;</b>	Creates a scrolling-text marquee	
<b>&lt;menu&gt;</b>	Deprecated. Specifies a menu list	
<b>&lt;menuitem&gt;</b>	Specifies a command/menu item that the user can invoke from a popup menu	
<b>&lt;meta&gt;</b>	Specifies meta data of an html document which is not displayed on the page	
<b>&lt;meter&gt;</b>	Specifies a scalar measurement within a known range (a gauge)	
<b>&lt;multicol&gt;</b>	Specifies a multicolumn text flow	
<b>&lt;nav&gt;</b>	Specifies a section that contains only navigation links	
<b>&lt;noabr&gt;</b>	No breaks allowed in the enclosed text	
<b>&lt;noembed&gt;</b>	Specifies content to be presented by browsers that do not support the <embed> tag	
<b>&lt;noframes&gt;</b>	Specifies a noframe section	
<b>&lt;noscript&gt;</b>	Specifies a noscript section	
<b>&lt;object&gt;</b>	Specifies an embedded object	
<b>&lt;ol&gt;</b>	Specifies an ordered list	
<b>&lt;optgroup&gt;</b>	Specifies an option group	
<b>&lt;option&gt;</b>	Specifies an option in a drop-down list	
<b>&lt;output&gt;</b>	Specifies the result of a calculation	
<b>&lt;p&gt;</b>	Specifies a paragraph	
<b>&lt;param&gt;</b>	Specifies a parameter for an object	
<b>&lt;plaintext&gt;</b>	Deprecated. Render the remainder of the document as	

	preformatted plain text	
<b>&lt;pre&gt;</b>	Specifies preformatted text	
<b>&lt;progress&gt;</b>	Specifies a completion progress of a task	
<b>&lt;q&gt;</b>	Specifies a short quotation	
<b>&lt;rp&gt;</b>	Specifies to show browsers that do not support the ruby element	
<b>&lt;rt&gt;</b>	Specifies an text ruby annotation	
<b>&lt;ruby&gt;</b>	Specifies an ruby annotation	
<b>&lt;s&gt;</b>	Deprecated. Specifies strikethrough text	
<b>&lt;samp&gt;</b>	Specifies sample computer code	
<b>&lt;script&gt;</b>	Specifies a script	
<b>&lt;section&gt;</b>	Specifies a section in a document	
<b>&lt;select&gt;</b>	Specifies a selectable list	
<b>&lt;spacer&gt;</b>	Specifies a white space	
<b>&lt;small&gt;</b>	Specifies small text	
<b>&lt;source&gt;</b>	Specifies a media resources for media elements, defined inside video or audio elements	
<b>&lt;span&gt;</b>	Specifies a section in a document	
<b>&lt;strike&gt;</b>	Deprecated. Specifies strikethrough text	
<b>&lt;strong&gt;</b>	Specifies strong text	
<b>&lt;style&gt;</b>	Specifies a style definition	
<b>&lt;sub&gt;</b>	Specifies subscripted text	
<b>&lt;summary&gt;</b>	Specifies a summary, caption, or legend for a given <details>	
<b>&lt;sup&gt;</b>	Specifies superscripted text	
<b>&lt;table&gt;</b>	Specifies a table	
<b>&lt;tbody&gt;</b>	Specifies a table body	

<b>&lt;td&gt;</b>	Specifies a table cell	
<b>&lt;textarea&gt;</b>	Specifies a text area	
<b>&lt;tfoot&gt;</b>	Specifies a table footer	
<b>&lt;th&gt;</b>	Specifies a table heading	
<b>&lt;thead&gt;</b>	Specifies a table header	
<b>&lt;time&gt;</b>	Specifies a date and time <details>	
<b>&lt;title&gt;</b>	Specifies the document title	
<b>&lt;tr&gt;</b>	Specifies a table row	
<b>&lt;track&gt;</b>	Specifies a text tracks used in mediaplayers	
<b>&lt;tt&gt;</b>	Specifies teletype text	
<b>&lt;u&gt;</b>	Deprecated. Specifies underlined text	
<b>&lt;ul&gt;</b>	Specifies an unordered list	
<b>&lt;var&gt;</b>	Specifies a variable	
<b>&lt;video&gt;</b>	Specifies a text tracks used in mediaplayers	
<b>&lt;wbr&gt;</b>	Indicates a potential word break point within a <nobr> section	
<b>&lt;xmp&gt;</b>	Deprecated. Specifies preformatted text	