HTML

Prof. Jayakumar Sadhasivam,
Vellore Institute of Technology

Table of Contents

A Rough History of Web Standards

91-92	93-94	95-96	97-98	99-00	01-02	03-04	05-06	07-08	09-10	11-12	13-14
HTML 1	HTML 2		HTML 4	XHTML 1					**HTML 5**		
		CSS 1	CSS 2		T-less D	Web 2.0			**CSS3**		
		JS	ECMA, DOM	DOM 2			Ajax		**DOM, APIs**		

2004	WHATWG started
2008	W3C Working Draft
2012 (2010)	W3C Candidate Rec
2022	W3C Rec

1996 – CSS 1	W3C Rec
1998 – CSS 2	W3C Rec
1999 – CSS 3	Proposed
2005 – CSS 2.1	W3C Candidate Rec
2001 – CSS 3	W3C Working Draft

HTML(Hyper Text Markup Language)

- HTML is a language for describing web pages.
- HTML is not a programming language, it is a markup language
- A markup language is a set of markup tags
- HTML uses markup tags to describe web pages
- **.html** – file extension.
- HTML is case insensitive. for example,

  type TITLE or Title or title or even tItLE if you like.

HTML Documents = Web Pages

- HTML documents describe web pages
- HTML documents contain HTML tags and plain text
- HTML documents are also called web pages
- The purpose of a web browser (like Internet Explorer or Firefox) is to read HTML documents and display them as web pages.
- The browser does not display the HTML tags, but uses the tags to interpret the content of the page.

What is HTML5?

- HTML5 will be the new standard for HTML, XHTML, and the HTML DOM.
- The previous version of HTML came in 1999. The web has changed a lot since then.
- HTML5 is still a work in progress. However, most modern browsers have some HTML5 support.

How Did HTML5 Get Started?

- HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).
- WHATWG was working with web forms and applications, and W3C was working with XHTML 2.0. In 2006, they decided to cooperate and create a new version of HTML.

How Did HTML5 Get Started?

- Some rules for HTML5 were established:
  - ✓ New features should be based on HTML, CSS, DOM, and JavaScript
  - ✓ Reduce the need for external plugins (like Flash)
  - ✓ Better error handling
  - ✓ More markup to replace scripting
  - ✓ HTML5 should be device independent
  - ✓ The development process should be visible to the public

New Features

- Some of the most interesting new features in HTML5 :
  - ✓ The canvas element for drawing
  - ✓ The video and audio elements for media playback
  - ✓ Better support for local offline storage
  - ✓ New content specific elements, like article, footer, header, nav, section
  - ✓ New form controls, like calendar, date, time, email, url, search

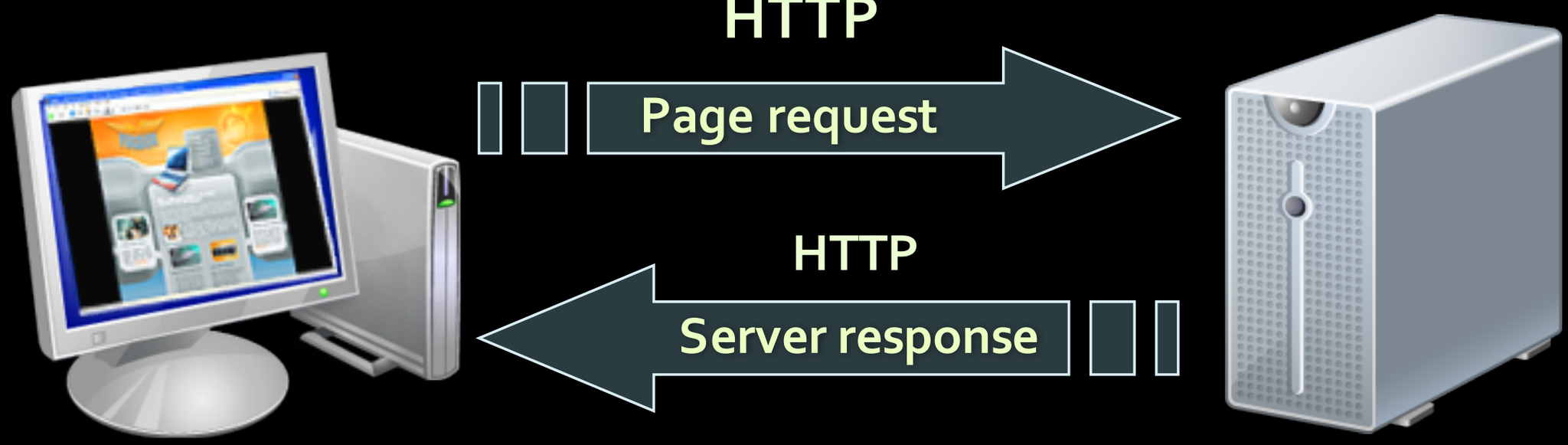
Browser Support

- HTML5 is not yet an official standard, and no browsers have full HTML5 support.

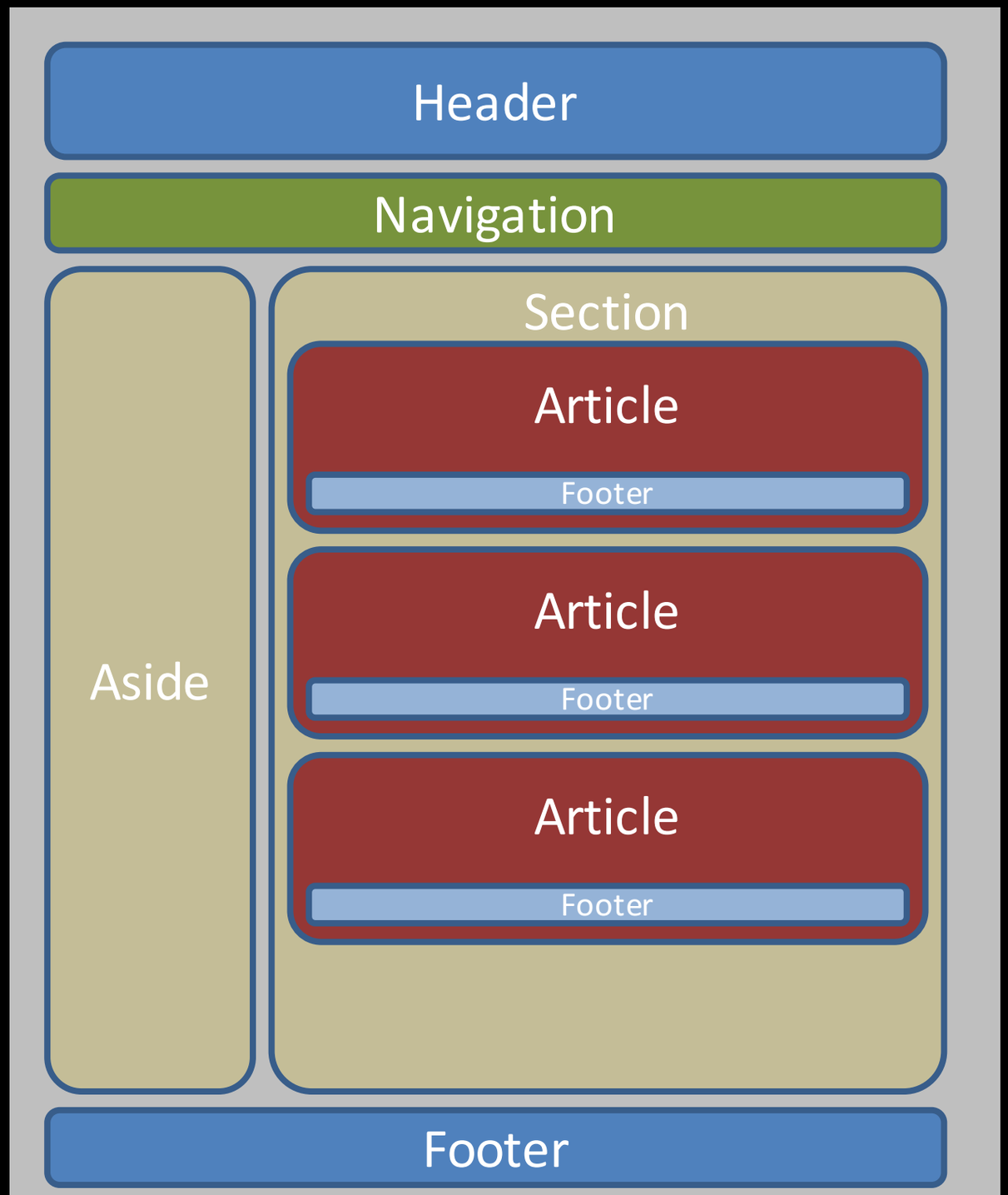
- But all major browsers (Safari, Chrome, Firefox, Opera, Internet Explorer) continue to add new HTML5 features to their latest versions.

How Web Works?

- WWW use classical client / server architecture
  - HTTP is text-based request-response protocol



HTML5


Header
Navigation
Aside
Section
Article
Footer
Article
Footer
Article
Footer
Footer

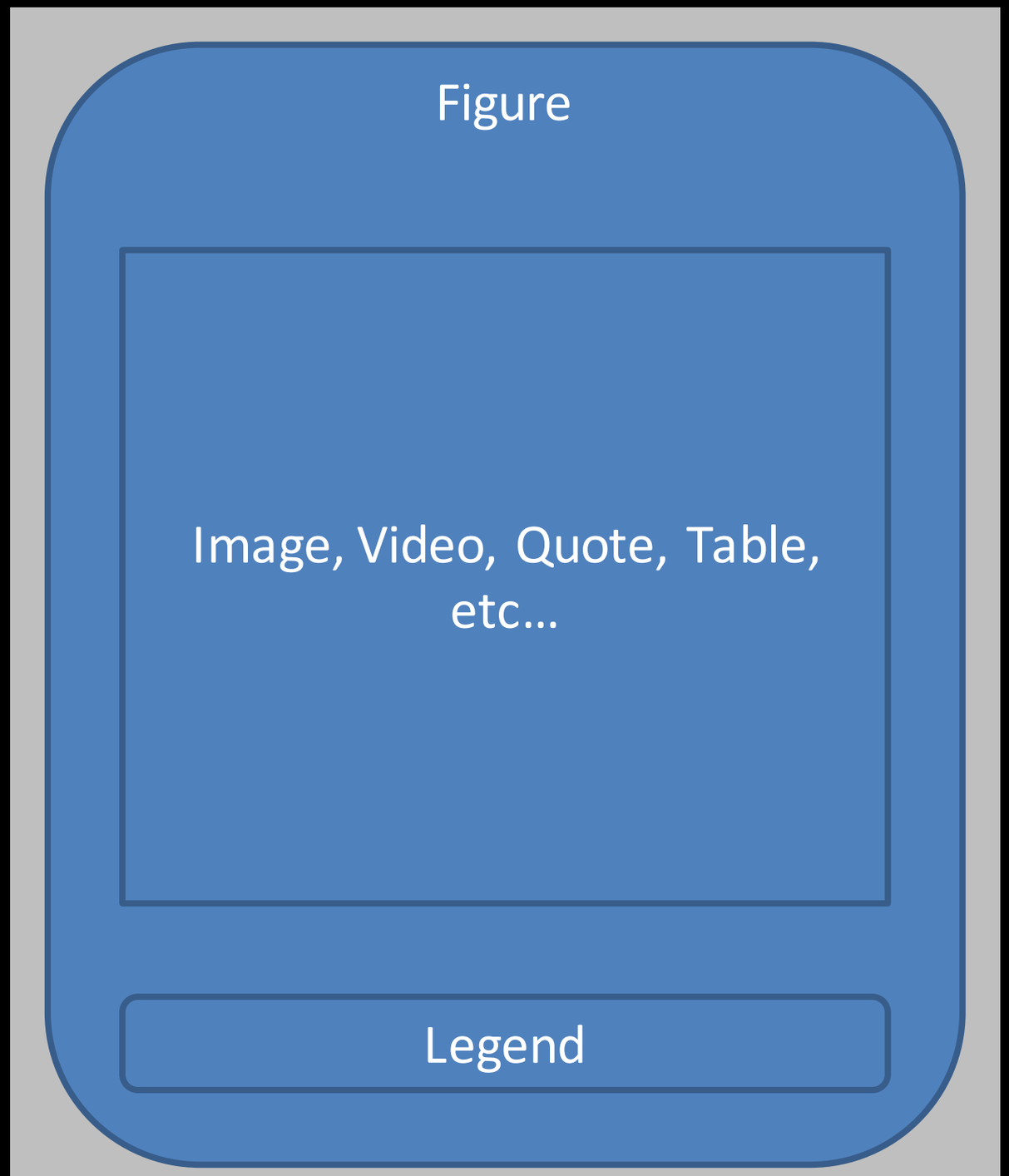
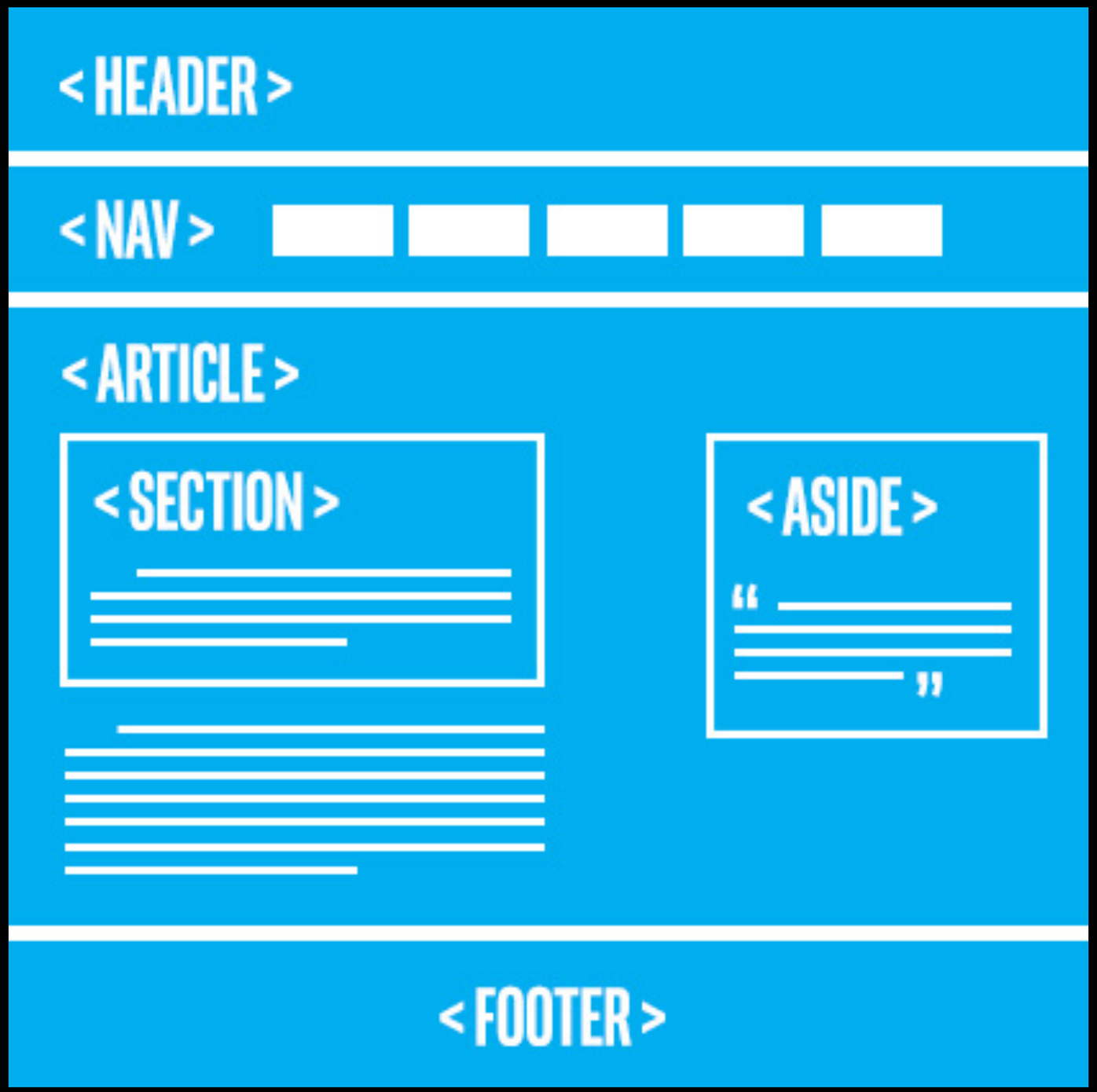
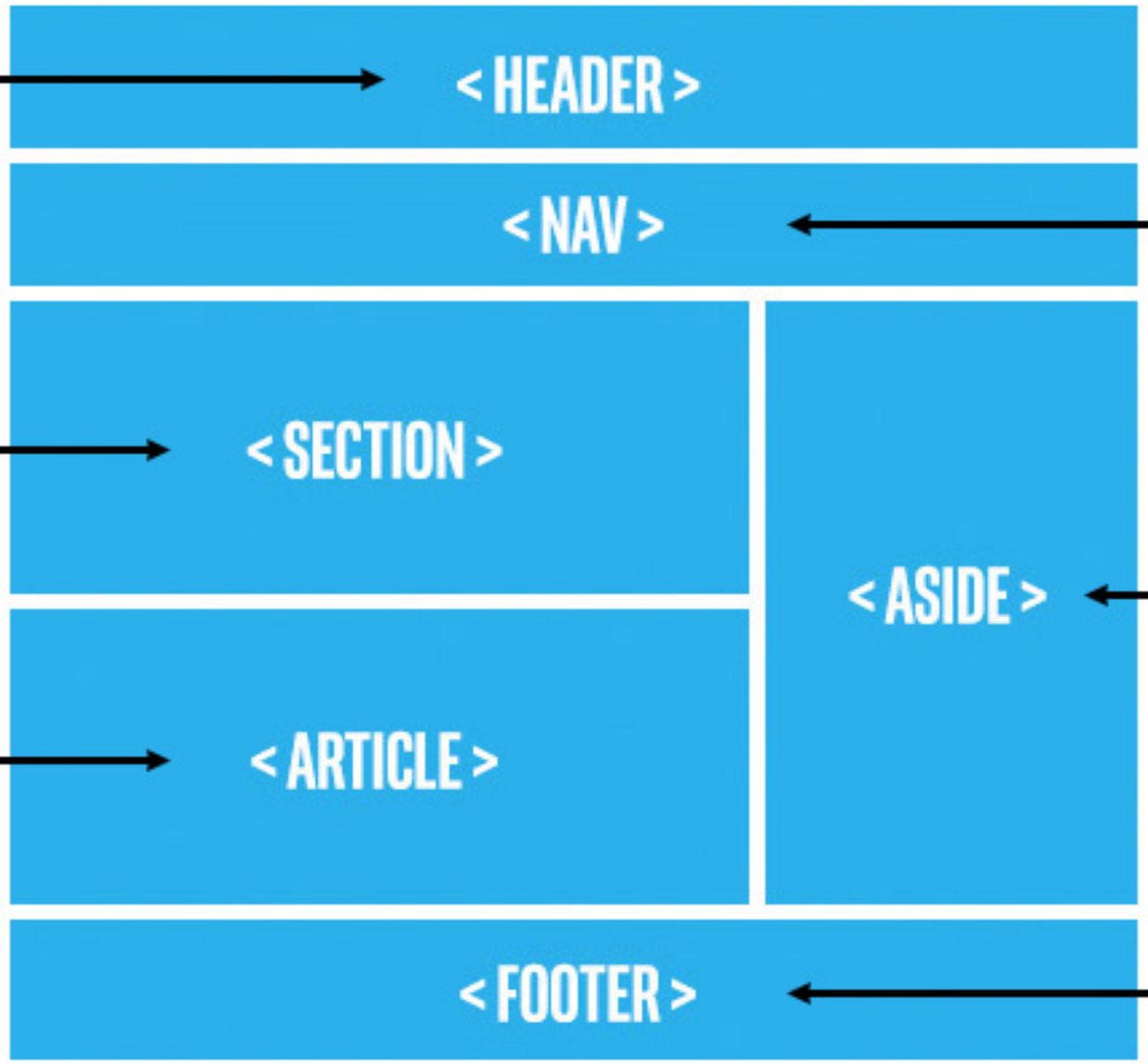


Figure
Image, Video, Quote, Table, etc…
Legend

HTML - typical Web page structure



The **header** can be a starting point for the whole document or individual sections. Typically contains the introduction.



The **nav** refers to navigation. It could contain a set of navigation links, such as a table of contents of a book.

The **section** refers to sections in a document. For example, a document about plants could contain several sections under headings such as perennials, annuals, soil type etc.

The **aside** refers to content that is apart from the main content. For example, in an article about a young architect from the Umbria region of Italy. The aside might be a small sidebar with information about Umbria, things like geographical details and population.

The **article** refers to independent content. If an article is extracted out of the document, it should make sense all by itself. Articles, blog posts, frequently asked questions are all examples of independent content.

The **footer** contains typical footer information such as authoring, copyrights and contact information.

HTML Tags

- HTML markup tags are usually called HTML tags
- HTML tags are keywords surrounded by angle brackets like <html>
- HTML tags normally come in pairs like and
- The first tag in a pair is the start tag, the second tag is the end tag
- Start and end tags are also called opening tags and closing tags

HTML Structure

- HTML is comprised of "elements" and "tags"
  - Begins with <html> and ends with </html>
- Elements (tags) are nested one inside another:

```
<html> <head></head> <body></body> </html>
```

- Tags have attributes:

```
<img src="logo.jpg" alt="logo" />
```

- HTML describes structure using two main sections: <head> and <body>

HTML Code Formatting

- The HTML source code should be formatted to increase readability and facilitate debugging.
  - ✓ Every block element should start on a new line.
  - ✓ Every nested (block) element should be indented.
  - ✓ Browsers ignore multiple whitespaces in the page source, so formatting is harmless.
- For performance reasons, formatting can be sacrificed

The <head> Section

- Contains information that doesn't show directly on the viewable page
- Starts after the <!DOCTYPE> declaration
- Begins with <head> and ends with </head>
- Contains mandatory single <title> tag
- Can contain some other tags, e.g.
  - ✓ <meta>
  - ✓ <script>
  - ✓ <style>
  - ✓ <!-- comments -->

<head> Section: <meta>

- Meta tags additionally describe the content contained within the page

```
<meta charset="utf-8" />
<meta name="description" content="HTML5" />
<meta keywords content="html,HTML5, web design, styles">
<meta name="author" content="Jayakumar Sadhasivam" />
<meta http-equiv="refresh" content="5 url=http://
jayakumars.in">
```

<head> Section: <script>

- The `<script>` element is used to embed scripts into an HTML document
  - Script are executed in the client's Web browser
  - Scripts can live in the <head> and in the <body> sections
- Supported client-side scripting languages:
  - JavaScript (it is not Java!)
  - VBScript
  - JScript

Comments: <!-- --> Tag

- **Comments can exist anywhere between the `<html></html>` tags**
- **Comments start with `<!--` and end with `-->`**

```
<!-- Telerik Logo (a JPG file) -->
<img src="logo.jpg" alt=“Telerik Logo">
<!-- Hyperlink to the web site -->
<a href="http://telerik.com/">Telerik</a>
<!-- Show the news table -->
<table class="newstable">
...
```

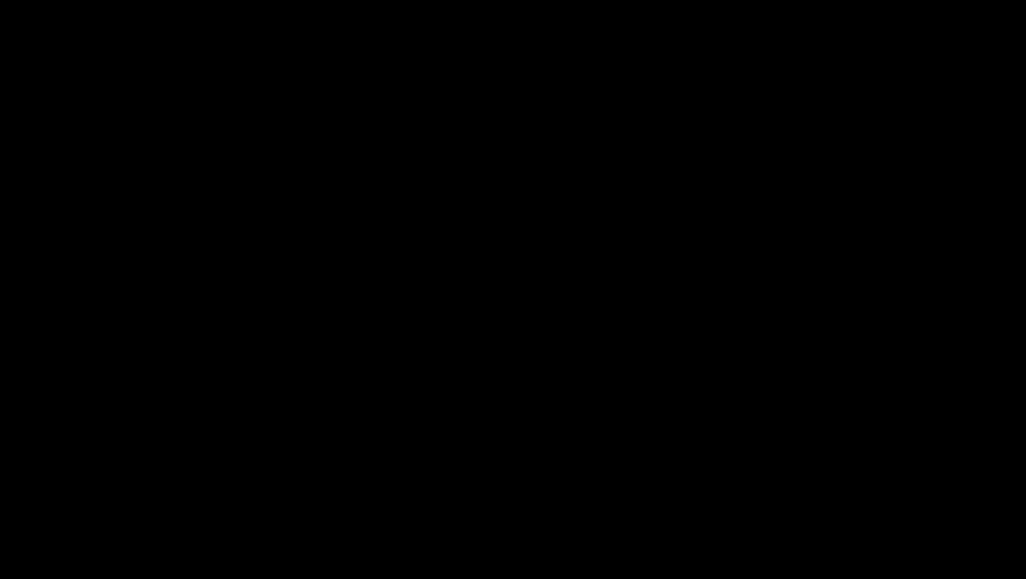

<body> Section: Introduction

- The <body> section describes the viewable portion of the page
- Starts after the <head> </head> section
- Begins with <body> and ends with </body>

```
<html>
  <head><title>Test page</title></head>
  <body>
    <!-- This is the Web page body -->
  </body>
</html>
```

Example

```
<!DOCTYPE html>
<html>
  <head>
  <meta charset="UTF-8">
   <title>Webage</title>
</head>
<body>
  My First HTML script
</body>
</html>
```



HTML Tags

- <h1>…</h1> – H1 to H6 where larger number means smaller heading
- .. – Paragraph Tag
- <hr> horizontal rule
-
 Break/New line

HTML Text Formatting Tags

****	**bold**
****	*italicized*
<u></u>	underlined
****	Samplesuperscript
****	Samplesubscript
****	**strong**
****	*emphasized*
<pre></pre>	`Preformatted text`
<blockquote></blockquote>	`Quoted text block`
<del></del>	Deleted text – ~~strike through~~

<b> vs

- **Bold** is a style that makes **letters thicker** so it stands out among other text but it has no semantic meaning, for example for voice browsers, screen readers, and other types of ways to access the Web. A device like Kindle Paperwhite that renders text differently, might not pick up the bold.
- **Strong** is an indication of how something should be. It looks like bold in a browser, but it could mean **'speak with urgency or seriousness'** when reading text aloud. It is semantic in the sense, that we instruct it to be stronger than the text it surrounds which is different from giving instructions on how the text should look in the case of <b>. It represents importance, seriousness, or urgency for its contents.

<i> vs

- ***Italics*** slants text. We usually **italicize** names of magazine, books, TV shows etc. Just like the bold tag, since it is meant purely for presentation purposes, it means nothing to someone who cannot read the text.
- ***Emphasis*** is used to **stress emphasis of its contents**. The word in a sentence you emphasize can change the whole meaning. Try reading the sentences below out loud, stressing on the emphasized words: 'you' and 'store'.

You have to go to the store.

Not me. That's your job!

You have to go to the *store*.

To the store. Not the arcade.

Special HTML Tags

- < → <
- > → >
- & → &
-   → space

- nbsp – non-breaking space
- amp – ampersand

HTML Font

```
<font size="5" face="arial" color="red">
This paragraph is in Arial, size 5, and in
    red text color.
</font>
```

```
<font size="10" face="Times New Roman"
    color="#aa0000">
This paragraph is in Times New Roman, size
    10, and in red text color.
</font>
```


HTML Image

- HTML images are defined with the <img> tag.
- Example
- File in the local folder
  - ✓
- File in the some other system
  - ✓

src	**Location of image file (relative or absolute)**
alt	**Substitute text for display (e.g. in text mode)**
height	**Number of pixels of the height**
width	**Number of pixels of the width**
border	**Size of border, 0 for no border**

HTML Background Image and Colour

Body Background Image

```
<body background="myimage.jpg">
```

Body Background Color

```
<body bgcolor="skyblue">
```


HTML Link

- HTML links are defined with the Anchor<a> tag.
- Example
  - ✓ ** A link **
- target=_blank -> The example below will open the linked document in a new browser window or a new tab
  - ✓ **Visit Jay website!**

- ✓ _blank - Opens the linked document in a new window or tab
- ✓ _self - Opens the linked document in the same window/tab as it was clicked (this is default)
- ✓ _parent - Opens the linked document in the parent frame
- ✓ _top - Opens the linked document in the full body of the window

HTML Link: Email Link

✓ mailto – makes a link as an email link

```
<a href="mailto:jayakumars@vit.ac.in">
        Send an email to Jay now! </a>
```


HTML Name Anchor Tags

- You create an anchor link that the person clicks on initially. Here is the HTML code for the anchor link.
  - ✓ `<a href="#anyword">Jump to any place</a>`

- You need to create a named anchor in the spot that you are going to jump to. This anchor will require the following syntax.
  - ✓ `<a name="anyword">Jump to this place</a>`

HTML LIST

HTML Lists

 An unordered list
 An ordered list
 list

```
<p>Unordered List</p>
<ul>
    <li>Coffee</li>
    <li>Milk</li>
</ul>
<hr>
<p>Ordered List</p>
<ol>
    <li>Coffee</li>
    <li>Milk</li>
</ol>
```

Unordered List

- Coffee
- Milk

Ordered List

1. Coffee
2. Milk

```
<ul type="square">
      <li>Coffee</li>
      <li>Milk</li>
            <ol type="i" start="3">
                  <li>Coffee</li>
                  <li>Milk</li>
            </ol>
</ul>
```

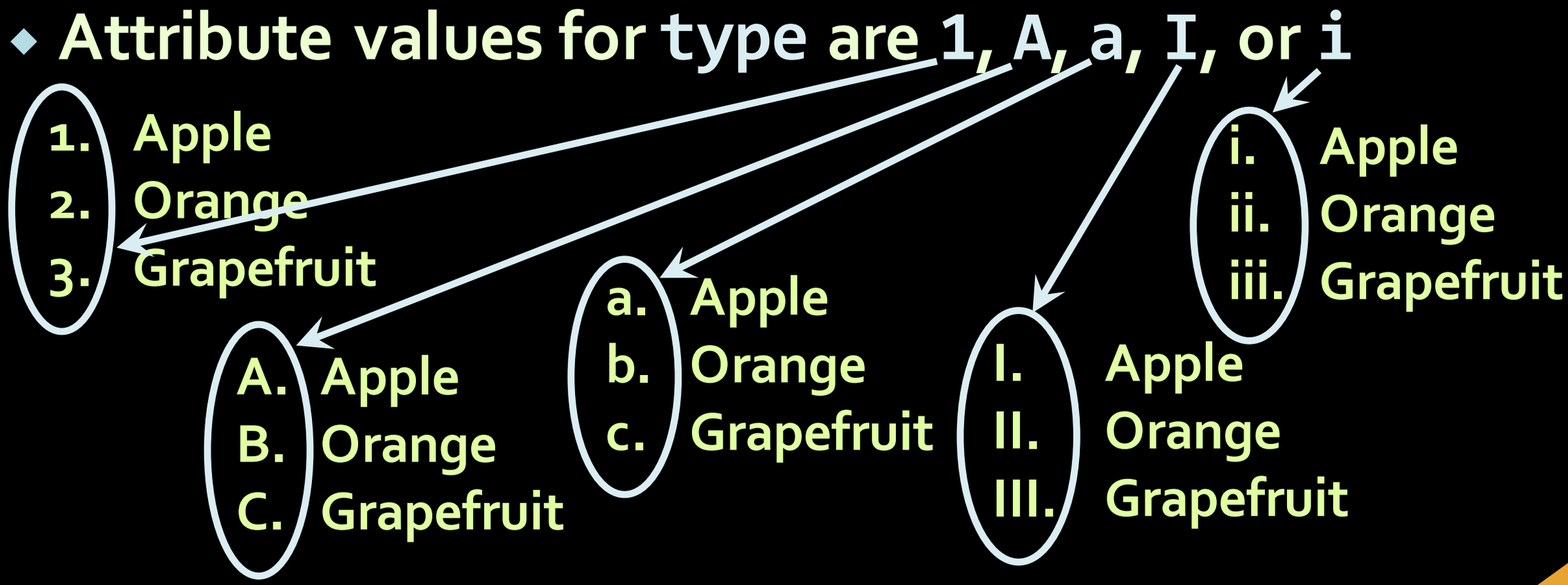
- Coffee
- Milk
    iii. Coffee
    iv. Milk

Ordered Lists: <ol> Tag

- **Create an Ordered List using :**

```
<ol type="1">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ol>
```

- **Attribute values for type are 1, A, a, I, or i**

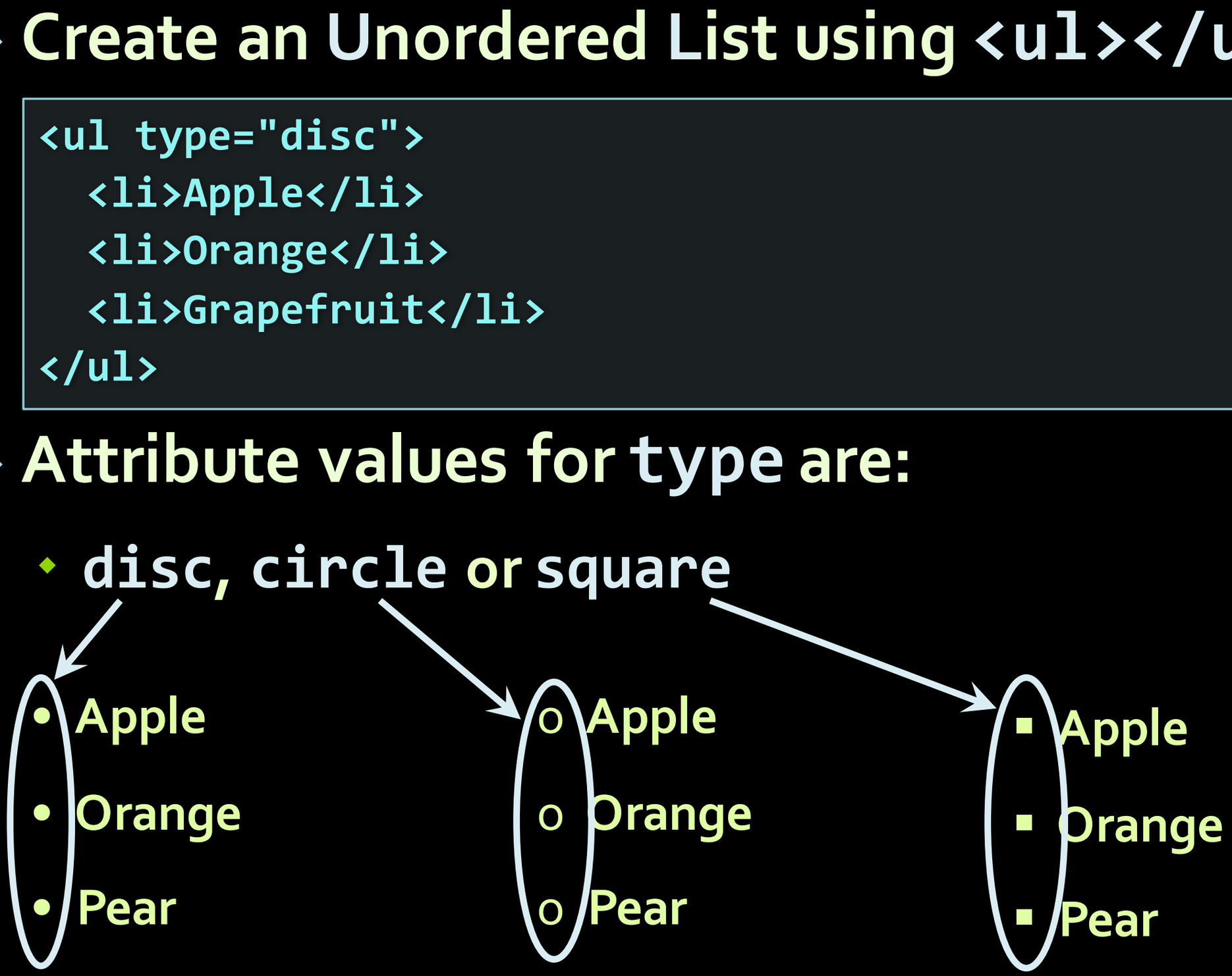


Unordered Lists: <ul> Tag

- **Create an Unordered List using `<ul></ul>`:**

```
<ul type="disc">
  <li>Apple</li>
  <li>Orange</li>
  <li>Grapefruit</li>
</ul>
```

- **Attribute values for `type` are:**
  - **`disc`, `circle` or `square`**



Definition lists: <dl> tag

- Create definition lists using `<dl>`
  - Pairs of text and associated definition; text is in `<dt>` tag, definition in `<dd>` tag

```
<dl>
   <dt>HTML</dt>
   <dd>A markup language …</dd>
   <dt>CSS</dt>
   <dd>Language used to …</dd>
</dl>
```

  - Renders without bullets
  - Definition is indented

Lists – Example

```
<ol type="1">
   <li>Apple</li>
   <li>Orange</li>
   <li>Grapefruit</li>
</ol>

<ul type="disc">
   <li>Apple</li>
   <li>Orange</li>
   <li>Grapefruit</li>
</ul>

<dl>
   <dt>HTML</dt>
   <dd>A markup language for Web Development</dd>
</dl>
```

1. Apple
2. Orange
3. Grapefruit

- Apple
- Orange
- Grapefruit

HTML
A markup language for Web Development

Boolean attribute **reversed**

```
<body>
    <!--Without boolean attribute 'reversed' -->
<ol>
  <li>HTML5</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ol>
<p>----------------------------</p>
   <!--With boolean attribute 'reversed' -->
<ol reversed>
  <li>HTML5</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ol>
```

1. HTML5
2. CSS
3. JavaScript

---

3. HTML5
2. CSS
1. JavaScript

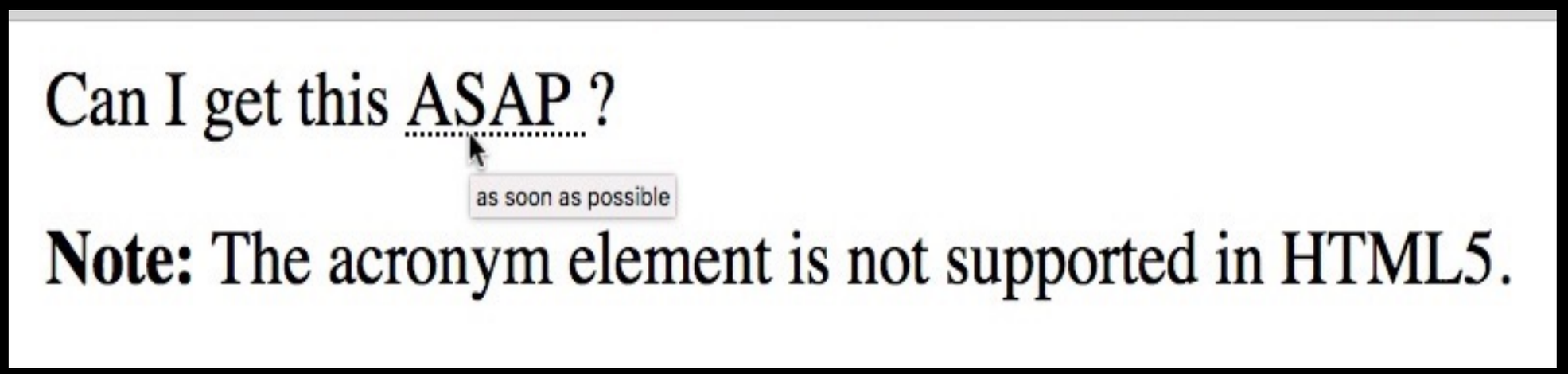
HTML Special Characters

Symbol Name	HTML Entity	Symbol
Copyright Sign	©	©
Registered Trademark Sign	®	®
Trademark Sign	™	™
Less Than	<	<
Greater Than	>	>
Ampersand	&	&
Non-breaking Space		
Em Dash	—	—
Quotation Mark	"	"
Euro	€	€
British Pound	£	£
Japanese Yen	¥	¥

Acronym Tag

An acronym is made of the initial letters of some words (e.g.Hypertext Mark-up Language => HTML)

```
<body>
Can I get this
<acronym title="as soon as possible"> ASAP </acronym>?
<p><strong>Note:</strong> The acronym element is not supported in HTML5.</p>
</body>
```

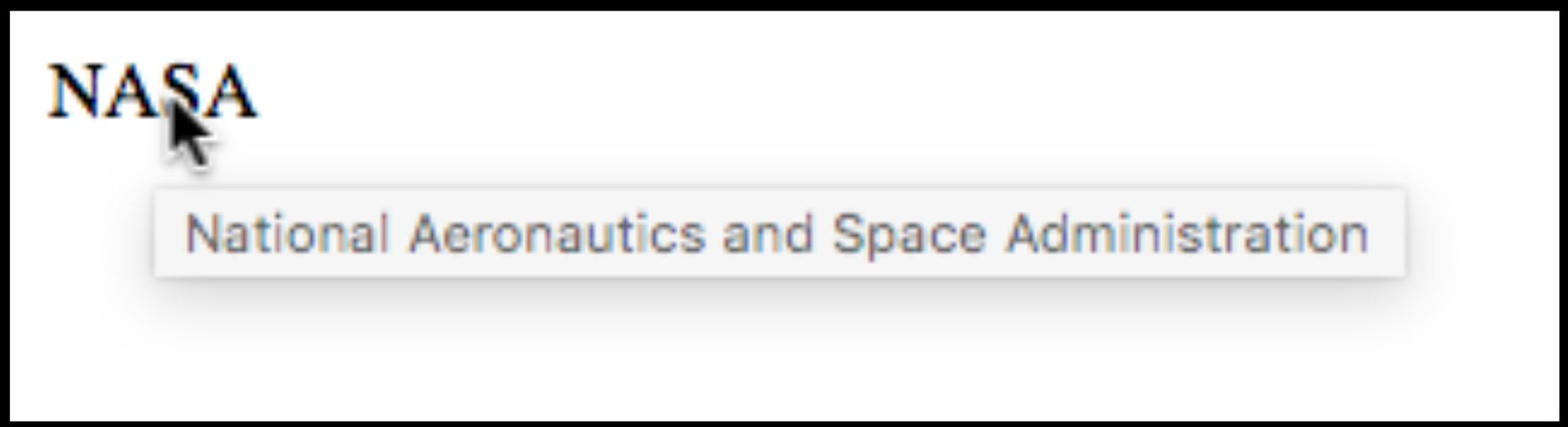


abbreviation<abbr> Tag

An abbreviation is a shortened form of a word (e.g. abbreviation => abbr).

Supported in HTML5

```
<abbr title="National Aeronautics and Space Administration”>NASA</abbr>
```



Using <DIV> and <SPAN> Block and Inline Elements

Block and Inline Elements

- Block elements add a line break before and after them
  - ✓ is a block element
- Other block elements are <table>, <hr>, headings, lists, <p> and etc.

- Inline elements don’t break the text before and after them
  - ✓ is an inline element
- Most HTML elements are inline, e.g. <a>

Tag

- The tag defines a division or a section in an HTML document.
- The tag is used to group block-elements to format them with styles.
  - ✓ examples: id, title, style, width, class, and height
- Container of other tags, grouping other tags together

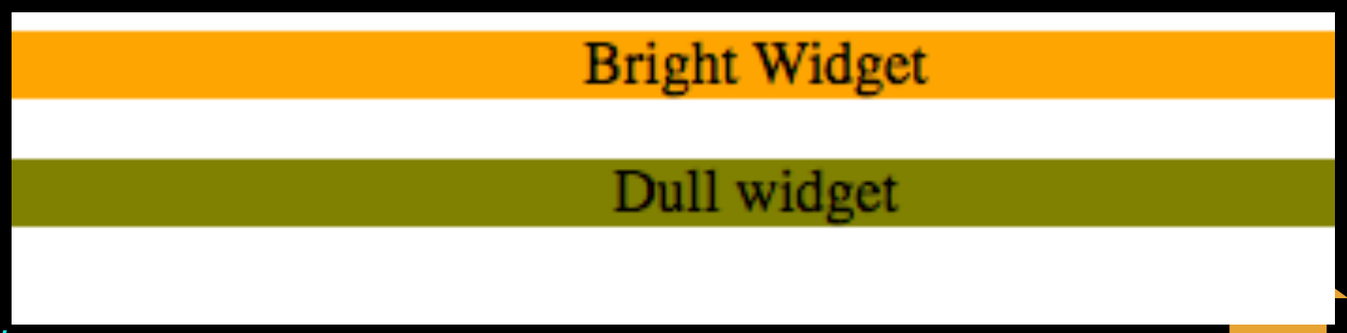
```
<body>
<div style="background: green">
<h5 >Jay website</h5>
<a href="http://jayakumars.in">Jay Site!!</a>
</div>
</body>
```

Attributes

- Standard Attributes: class, id, style, and title
- align (value: left, right, center, justify)
  - ✓ Use styles instead.
- Specifies the alignment of the content inside a element.

```
<div style="background-color:orange; text-
    align:center">
  <p>Bright Widget</p>
</div>

<div style="background-color:olive; text-
    align:center”>
  <p>Dull widget</p>
</div>
```



Tag

- Useful for modifying a specific portion of text
- Don't create a separate area (paragraph) in the document
- Very useful with CSS

```
<p>This one is <span style="color:red; font-
    weight:bold">only a test</span>
</p>
<p>This one is another <span style="font-size:32px;
    font-weight:bold">TEST</span>
</p>
```

This one is only a test

This one is another TEST

&

```
<body>
<p>This is a paragraph
     <div style="font-size:24px; color:red">DIV
                  example</div>
for DIV </p>

<p>This is a paragraph
    <span style="color:red; font-
                 weight:bold">TEST SPAN</span>
for span. </p>
</body>
```

This is a paragraph

DIV example
for DIV

This is a paragraph **TEST SPAN** for span.

Div Two Column Vertical

```
<div>
   <h1> Welcome to ITA1007 </h1>
</div>
<div style="float:left; width:20%">
   <p> welcome to Ita1008
</div>
<div style="float:left; width:80%" >
   <p>  Hello welcome to ITA1007 <br/>
   Class starts @11am monday
</div>
```

Div Style Float Left

```
<div style="float:left; width:20%">
   <p> welcome to Ita1008 – Left
</div>
<div style="float:right; width:40%;
                color:red" >
  <p>  Use LEFT to keep the Div as you code.
</div>
```

```
<div style="float:left; width:20%">
   <p> welcome to Ita1008 – left
</div>
```

Welcome to ITA1007

welcome to Ita1008 - Left

welcome to Ita1008 - left

Use LEFT to keep the Div as you code.

HTML Tables

HTML5 Tables

- An HTML table is defined with the **<table> </table>** tag.
- Each table row is defined with the **<tr> </tr>** tag.
- A table header is defined with the **<th> </th>** tag.
- By default, table headings are bold and centered. A table data/cell is defined with the **<td> </td>** tag.
- **<colgroup>** - Column Group
- **<thead>** - Table head
- **<tbody>** - Table body
- **<tfoot>** - Table foot

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
 <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
</table>
```

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94

HTML Table - Styles

```
<table style="width:100%">
<table align="right">
<table border="1" bordercolor="red">
<tr bgcolor="blue">
```


HTML Align and VAlign

```
<table border=1 width="100%">
        <tr>
            <td align="left">Left</td>
            <td align="center">Center</td>
            <td align="right">right</td>
        </tr>
        <tr style="height:100px">
            <td valign="top" align="right">Top</td>
            <td valign="middle">Middle</td>
            <td valign="bottom">bottom</td>
        </tr>
</table>
```

Left	Center	right
Top	Middle	bottom

```
<table style="width:100%" border=1>
<caption>Monthly savings</caption>
<tr>
 <th>Month</th>
 <th>Savings</th>
</tr>
<tr>
 <td>January</td>
 <td>$100</td>
</tr>
<tr>
 <td>February</td>
 <td>$50</td>
</tr>
</table>
```

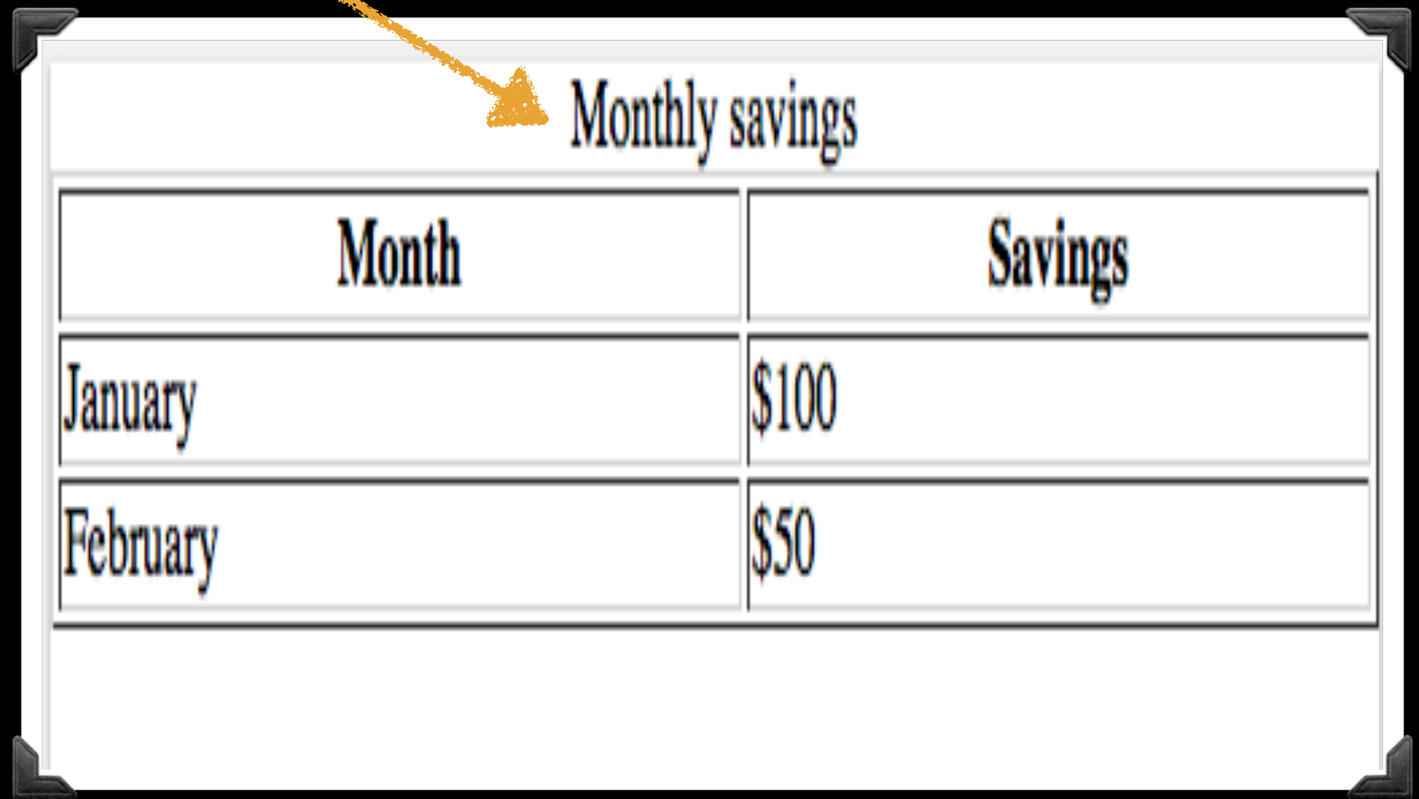


Table - Cellpadding

```
<p>Table without cellpadding:</p>
<table border=1>
<tr><th>Month</th><th>Savings</th><
<tr> <td>January</td>
<td>$100</td>  </tr>
</table>
<p>Table with cellpadding:</p>
<table border=1 cellpadding="10">
<tr>
<th>Month</th>
<th>Savings</th></tr>
<tr> <td>January</td> <td>$100</td> </tr>
</table>
```

Table without cellpaddin

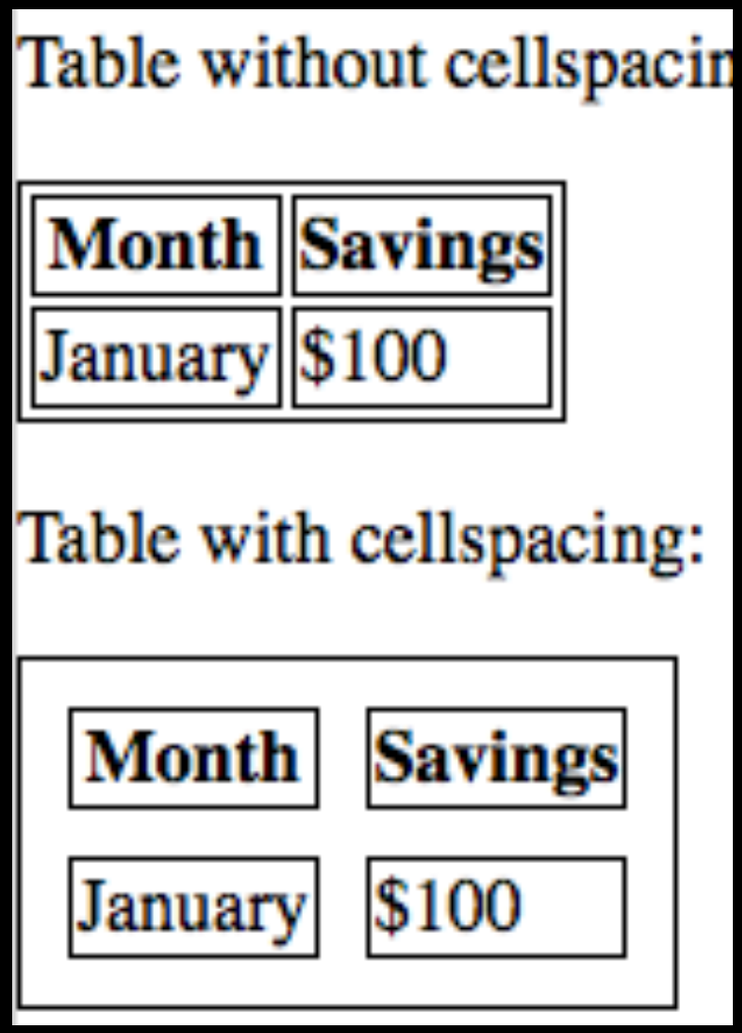
Month	Savings
January	$100

Table with cellpadding:

Month	Savings
January	$100

Table - Cellspacing

```
<p>Table without cellspacing:</p>
<table border=1>
<tr><th>Month</th>
<th>Savings</th></tr>
<tr><td>January</td>
<td>$100</td>  </tr>
</table>
<p>Table with cellspacing:</p>
<table border=1 cellspacing="10">
<tr><th>Month</th>
<th>Savings</th></tr>
<tr><td>January</td>
<td>$100</td> </tr>
</table>
```



HTML <col> Tag

```
<table border=2 bordercolor=black>
<colgroup>
 <col span="2" style="background-
                color:red">
<col style="background-color:yellow">
</colgroup>
<tr> <th>ISBN</th> <th>Title</th>
 <th>Price</th></tr>
 <tr>
 <td>3476896</td>
 <td>My HTML</td>
 <td>$53</td> </tr>
 <tr>
 <td>5869207</td> <td>My CSS</td>
 <td>$49</td></tr></table>
```

ISBN	Title	Price
3476896	My HTML	$53
5869207	My CSS	$49

HTML <colspan> Tag

```
<table border="1">
 <tr>
  <td colspan="2">Row 1 – Col 1 and Col 2</td>
 </tr>
 <tr>
  <td>Row 2 – Col 1</td>
  <td>Row 2 – Col 2</td>
 </tr>
</table>
```

Row 1 - Col 1 & Col 2	
Row 2 - Col 1	Row 2 - Col 2

HTML <rowspan> Tag

```
<table border="1">
 <tr>
  <td>Row 1 - Col 1</td>
  <td rowspan="2">Row 1 and Row 2 - Col 2</td>
 </tr>
 <tr>
  <td>Row 2 - Col 1</td>
 </tr>
</table>
```

Row 1 - Col 1	Row 1 & Row 2 - Col 2
Row 2 - Col 1	

```
<table border="1">
<tr>
   <td>Row 1 - Col 1</td>
   <td>Row 1 - Col 2</td>
   <td>Row 1 - Col 3</td>
   <td rowspan="4">Row 1, 2, 3 and 4 - Col 4</td>
</tr>
<tr>
   <td>Row 2 - Col 1</td>
   <td colspan="2" rowspan="3">Row 2, 3 and 4 - Col 2 and 3</td>
</tr>
<tr>
   <td>Row 3 - Col 1</td>
</tr>
<tr>
   <td>Row 4 - Col 1</td>
</tr>
<tr>
   <td colspan="2">Row 5 - Col 1 and 2</td>
   <td colspan="2">Row 5 - Col 3 and 4</td>
</tr>
<tr>
   <td colspan="4">Row 6 - Col 1, 2, 3 and 4</td>
</tr>
</table>
```

Output - Col & Row Span

<table>
<tr><td>Row 1 - Col 1</td><td>Row 1 - Col 2</td><td>Row 1 - Col 3</td><td rowspan="4">Row 1, 2, 3 & 4 - Col 4</td></tr>
<tr><td>Row 2 - Col 1</td><td colspan="2" rowspan="3">Row 2, 3 & 4 - Col 2 & 3</td></tr>
<tr><td>Row 3 - Col 1</td></tr>
<tr><td>Row 4 - Col 1</td></tr>
<tr><td colspan="2">Row 5 - Col 1 & 2</td><td colspan="2">Row 5 - Col 3 & 4</td></tr>
<tr><td colspan="4">Row 6 - Col 1, 2, 3 & 4</td></tr>
</table>

```
<table border="5" cellspacing="0" cellpadding="4" width="500" align="right" bgcolor="white">
   <caption align="top"><b>Race Results</b></caption>
   <tr bgcolor="yellow">
      <th colspan="2">Runner</th>
      <th>Time</th>
      <th>Origin</th>
   </tr>
   <tr>
      <td rowspan="3" valign="top" bgcolor="lightblue">Men</td>
      <td>1. Peter Teagan</td>
      <td align="right">2:12:34</td>
      <td>San Antonio, Texas</td>
   </tr>
   <tr>
      <td>2. Kyle Wills</td>
      <td align="right">2:13:05</td>
      <td>Billings, Montana</td>
   </tr>
   <tr>
      <td>3. Jason Wu</td>
      <td align="right">2:14:28</td>
      <td>Cutler, Colorado</td>
   </tr>
   <tr>
      <td rowspan="3" valign="top" bgcolor="lightgreen">Women</td>
      <td>1. Laura Blake</td>
      <td align="right">2:28:21</td>
      <td>Park City, Colorado</td>
   </tr>
   <tr>
      <td>2. Kathy Lasker</td>
      <td align="right">2:30:11</td>
      <td>Chicago, Illinois</td>
   </tr>
   <tr>
      <td>3. Lisa Peterson</td>
      <td align="right">2:31:14</td>
      <td>Seattle, Washington</td>
   </tr>
</table>
```

Specifying Table, Row, and Cell Colors

Specifying Table, Row, and Cell Colors

Race Results

Runner		Time	Origin
Men	1. Peter Teagan	2:12:34	San Antonio, Texas
	2. Kyle Wills	2:13:05	Billings, Montana
	3. Jason Wu	2:14:28	Cutler, Colorado
Women	1. Laura Blake	2:28:21	Park City, Colorado
	2. Kathy Lasker	2:30:11	Chicago, Illinois
	3. Lisa Peterson	2:31:14	Seattle, Washington

HTML5 Table

```
<table border="1" width="100%">
<thead align="center">
<tr>
 <td>Row 1, Head 1</td>
 <td>Row 1, Head 2</td>
 <td>Row 2, Head 3</td>
</tr>
</thead>
<tfoot align="center">
<tr>
 <td>Row 5, Foot 1</td>
 <td>Row 5, Foot 2</td>
 <td>Row 5, Foot 3</td>
</tr>
</tfoot>
```

Row 1, Head 1	Row 1, Head 2	Row 2, Head 3
Row 2, cell 1	Row 2, cell 2	Row 2, cell 3
Row 3, cell 1	Row 3, cell 2	Row 3, cell 3
Row 4, cell 1	Row 4, cell 2	Row 4, cell 3
Row 5, Foot 1	Row 5, Foot2	Row 5, Foot3

```
<tbody>
<tr>
   <td>Row 2, cell 1</td>
   <td>Row 2, cell 2</td>
   <td>Row 2, cell 3</td>
</tr>
<tr>
   <td>Row 3, cell 1</td>
   <td>Row 3, cell 2</td>
   <td>Row 3, cell 3</td>
</tr>
<tr>
   <td>Row 4, cell 1</td>
   <td>Row 4, cell 2</td>
   <td>Row 4, cell 3</td>
</tr>
</tbody>
</table>
```

Row 1, Head 1	Row 1, Head 2	Row 2, Head 3
Row 2, cell 1	Row 2, cell 2	Row 2, cell 3
Row 3, cell 1	Row 3, cell 2	Row 3, cell 3
Row 4, cell 1	Row 4, cell 2	Row 4, cell 3
Row 5, Foot 1	Row 5, Foot2	Row 5, Foot3

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.

HTML Frames - Row

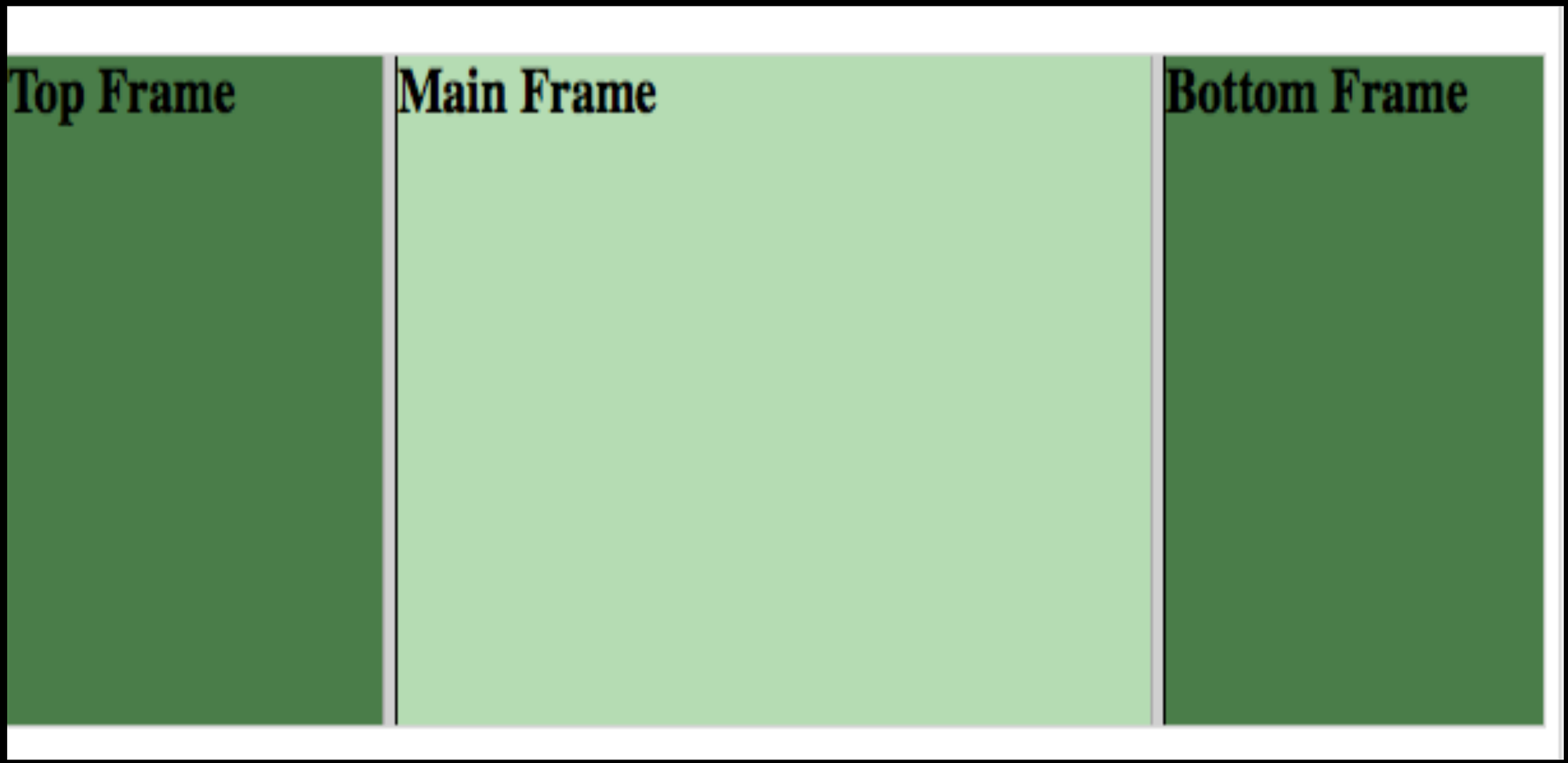
```
<!DOCTYPE html>
<html>
<head>
    <title>HTML Frames</title>
</head>
<frameset rows="15%,70%,15%">
    <frame name="top" src="top_frame.html"></frame>
    <frame name="main" src="main_frame.html"></frame>
    <frame name="bottom" src="bottom_frame.html"></frame>
<noframes>
    <body>Your Browser does not support frames
  </body>
</noframes>
</frameset>
</html>
```

HTML Frames - Row

HTML Frames - Cols

```
<!DOCTYPE html>
<html>
<head>
    <title>HTML Frames</title>
</head>
<frameset cols=“20%,60%,20%">
    <frame name="left" src="top_frame.html"></frame>
    <frame name="center" src="main_frame.html"></frame>
    <frame name="right" src="bottom_frame.html"></frame>
<noframes>
    <body>Your Browser does not support frames
  </body>
</noframes>
</frameset>
</html>
```

HTML Frames - Cols



Frame's name and target attribute

```
<!--FileName:  test.htm -->

<!DOCTYPE html>
<html>
    <head>
        <title>HTML Target Frames</title>
    </head>
    <frameset cols = "200, *">
        <frame src = "/html/menu.htm" name = "menu_page" />
        <frame src = "/html/main.htm" name = "main_page" />
        <noframes>
            <body>Your browser does not support frames.</body>
        </noframes>
    </frameset>
</html>
```


Frame's name and target attribute

```
<!--FileName:  menu.htm -->

<!DOCTYPE html>
<html>

    <body bgcolor = "#4a7d49">
        <a href = "http://google.com" target = "main_page">Google</a>
        <br />
        <br />
        <a href = "http://microsoft.com" target =
            "main_page">Microsoft</a>
        <br />
        <br />
        <a href = "http://bbc.co.uk" target = "main_page">BBC News</a>
    </body>
</html>
```


Frame's name and target attribute

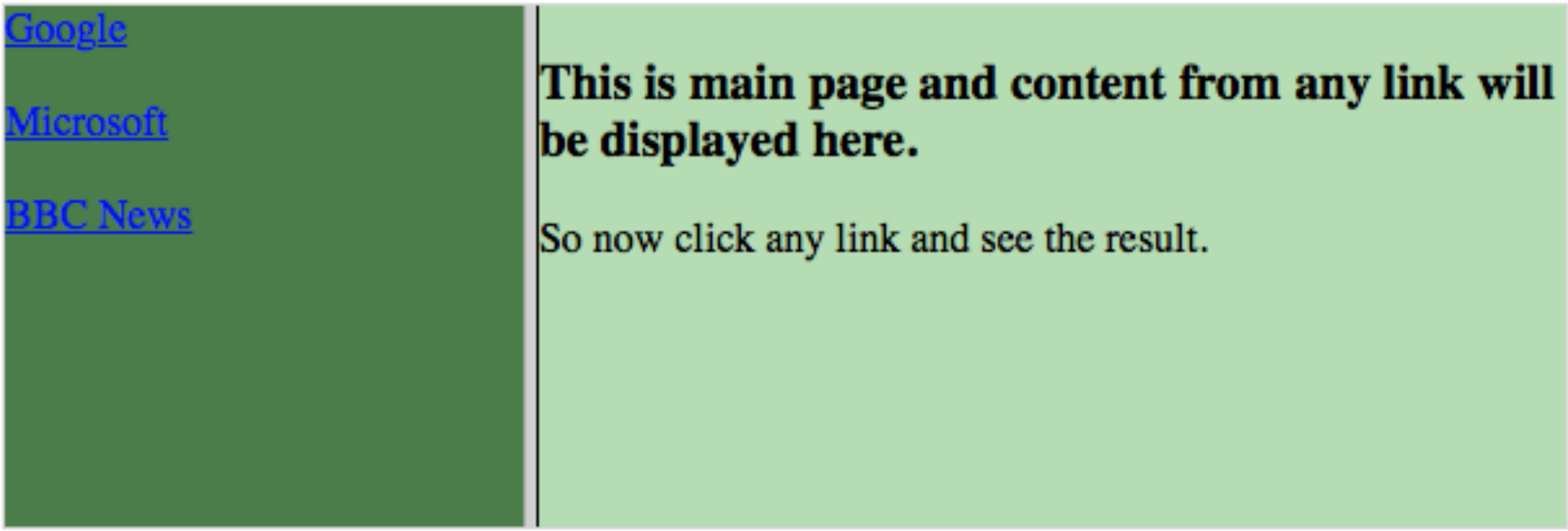
```
<!--FileName:  main.htm -->

<!DOCTYPE html>
<html>
<body bgcolor = “#b5dcb3">
<h3>This is main page and content from any
        link will be displayed here.</h3>

<p>So now click any link and see the
        result.</p>
</body>
</html>
```

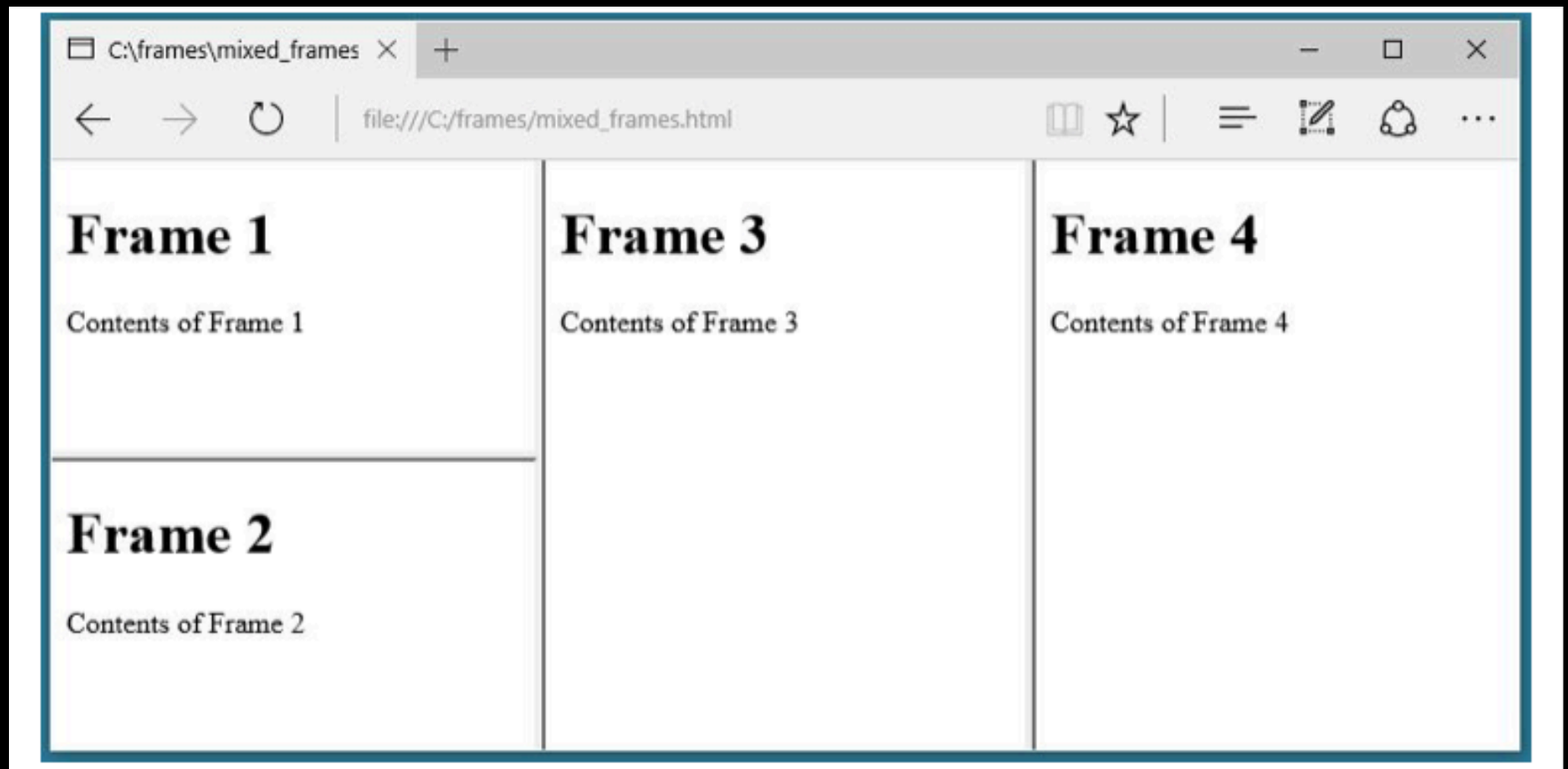
Frame's name and target attribute

When we load **test.htm** file, it produces following result –



Now you can try to click links available in the left panel and see the result. The *targetattribute* can also take one of the following values –

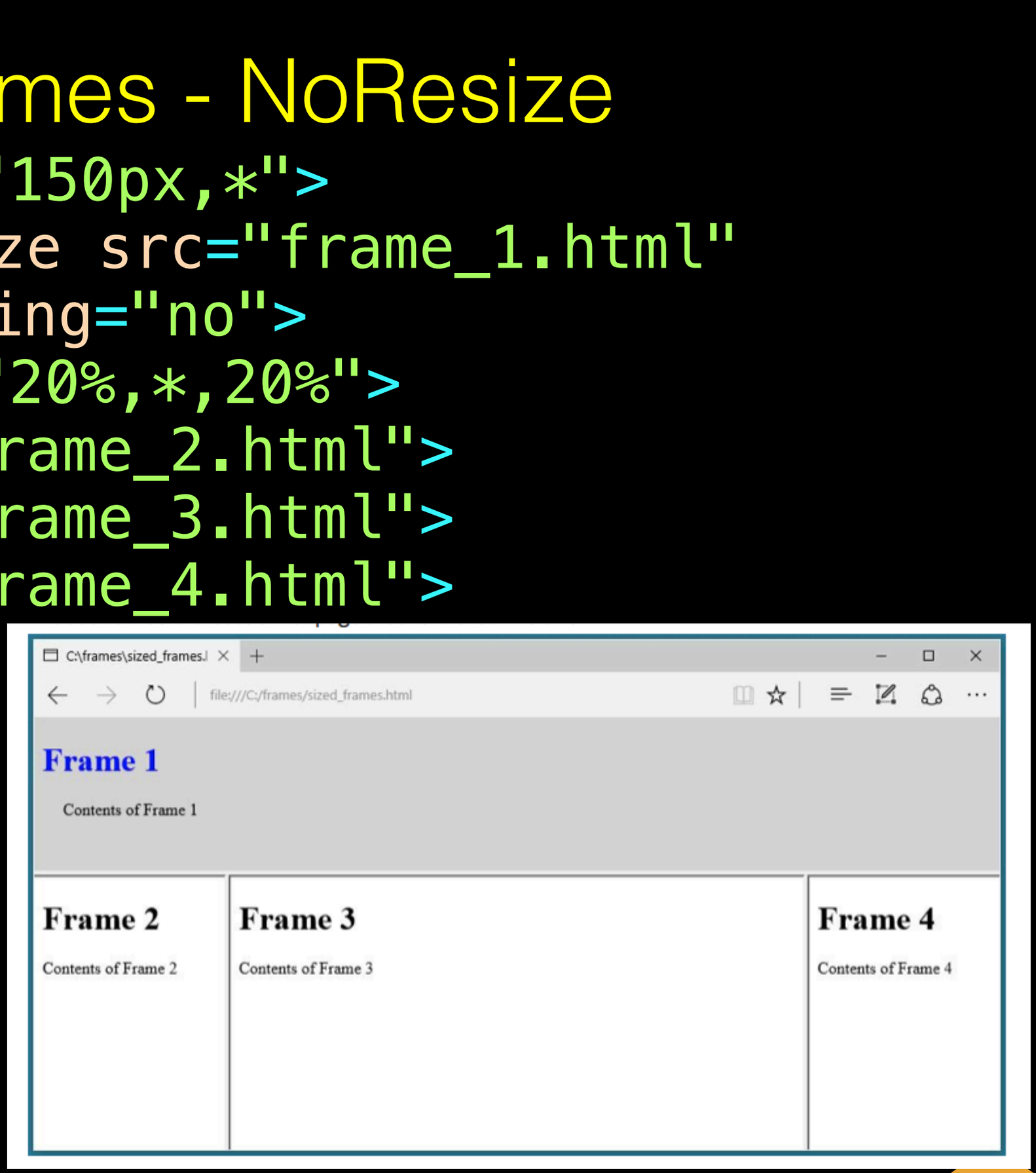
```
<frameset cols="*,*,*">
   <frameset rows="*,*">
         <frame src="frame_1.html">
         <frame src="frame_2.html">
   </frameset>
         <frame src="frame_3.html">
         <frame src="frame_4.html">
</frameset>
```



Frames Rows and Cols

Frames - NoResize

```
<frameset rows="150px,*">
   <frame noresize src="frame_1.html"
          scrolling="no">
<frameset cols="20%,*,20%">
   <frame src="frame_2.html">
   <frame src="frame_3.html">
   <frame src="frame_4.html">
</frameset>
</frameset>
```



HTML Frame Scrolling

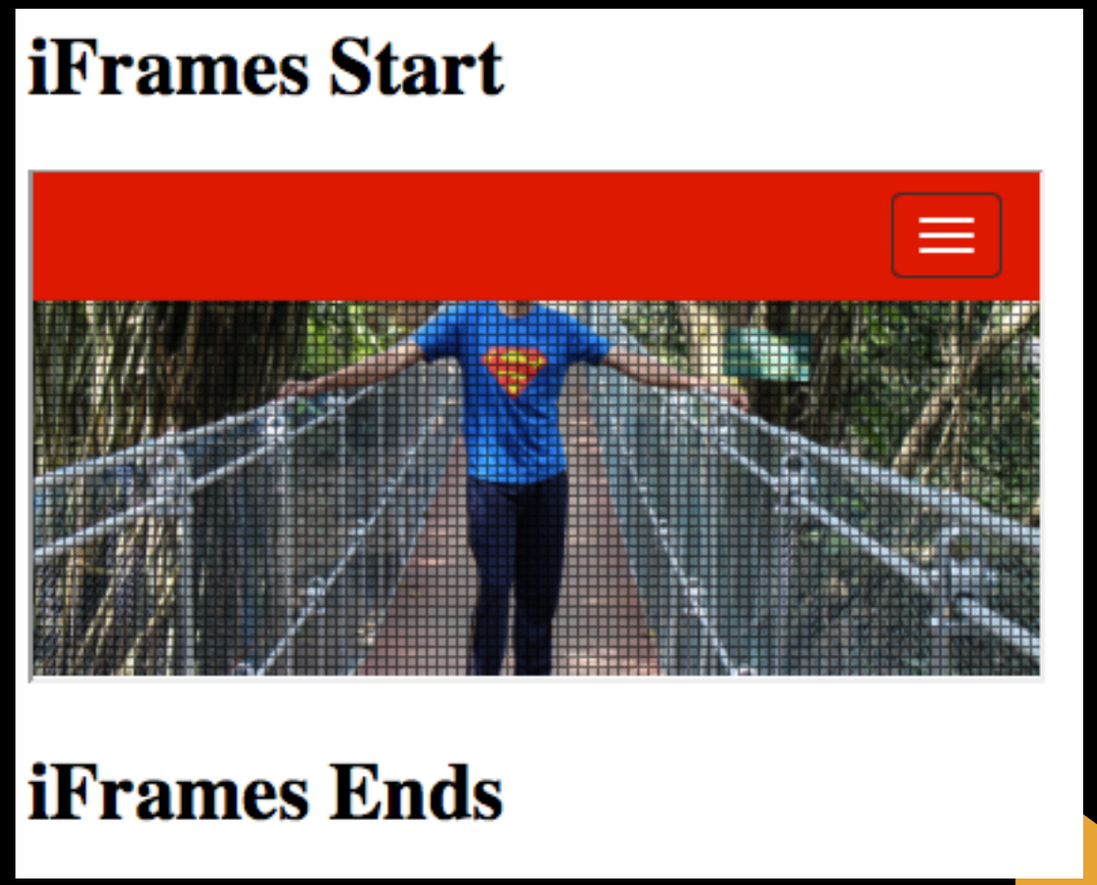
```
<frame scrolling="auto|yes|no"> <!-- MetaTag -->


<frameset cols="120,*" frameborder="0"
    border="0" framespacing="0">
   <frame src="menu.htm" name="menu" noresize
          scrolling=no>
   <frame src="frontf.htm" name="main" noresize
          scrolling=auto>

</frameset>
```


HTML iFrames

```
<html>
<body>
  <h1> iFrames Start </h1>
  <iframe src="http://www.jayakumars.in"
       height="200" width="400">
  <p>Your browser does not support iframes.</p>
  </iframe>
  <h1> iFrames Ends </h1>
</body>
</html>
```

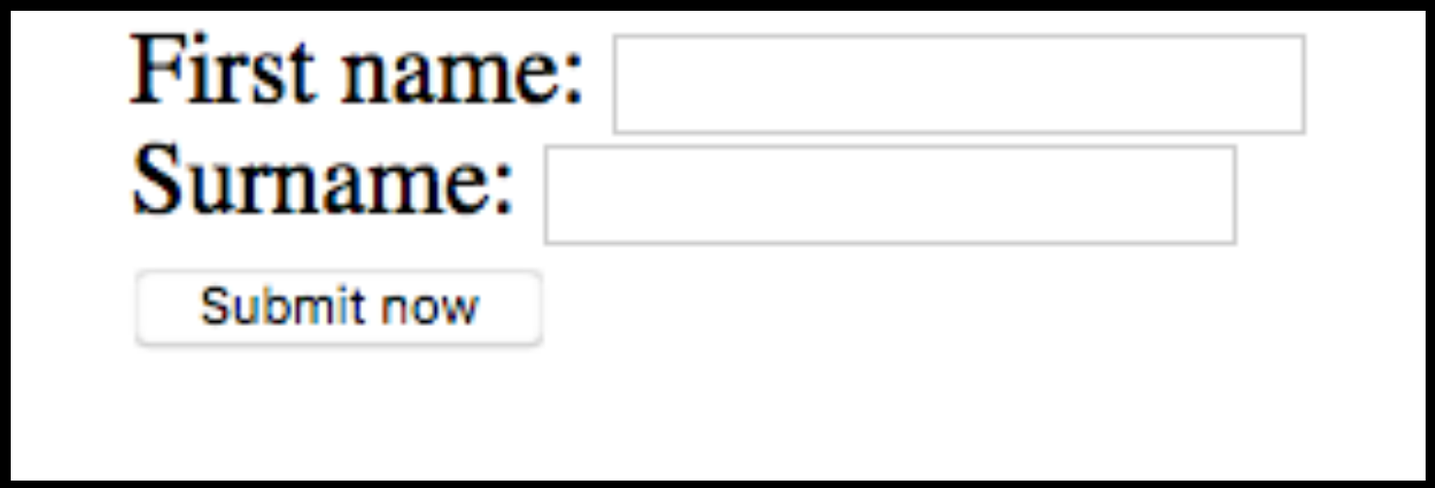


HTML Forms

HTML Forms

```
<form>
First name:
 <input type="text" name="firstname"
      size="35" maxlength="140"/>
<br />
 Surname:<input type="text" name="surname" />
<br />
<input type="submit" value="Submit now" />

</form>
```



Submit & Reset Buttons

- The submit button is used whenever you want to submit a form and the markup looks like this:

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

- The reset button is used to clear al inputs by the user and the markup looks like this:

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

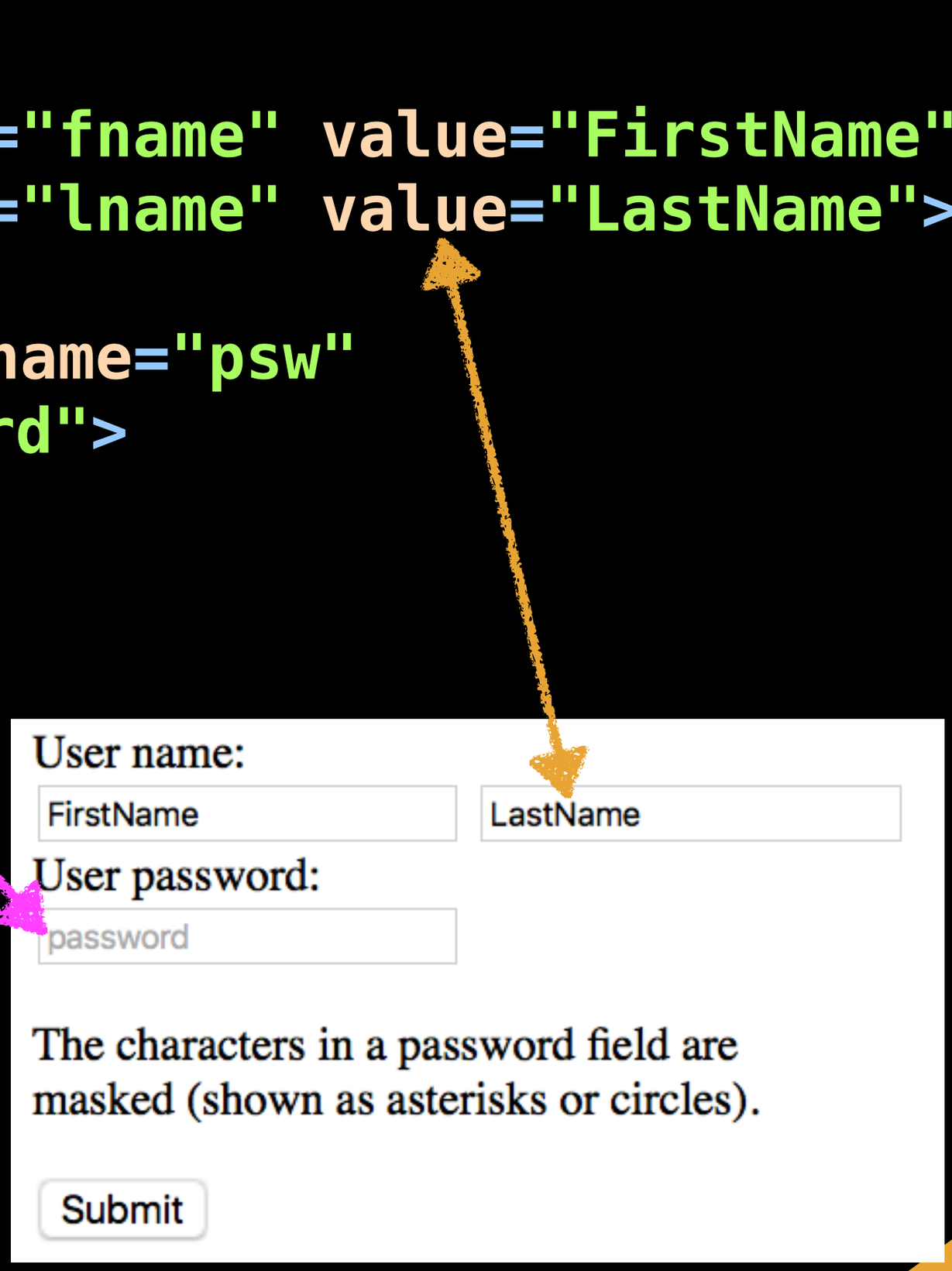
- Ordinary button – used for Javascript, no default action

```
<input type="button" value="Click Me"/>
```

Fieldset

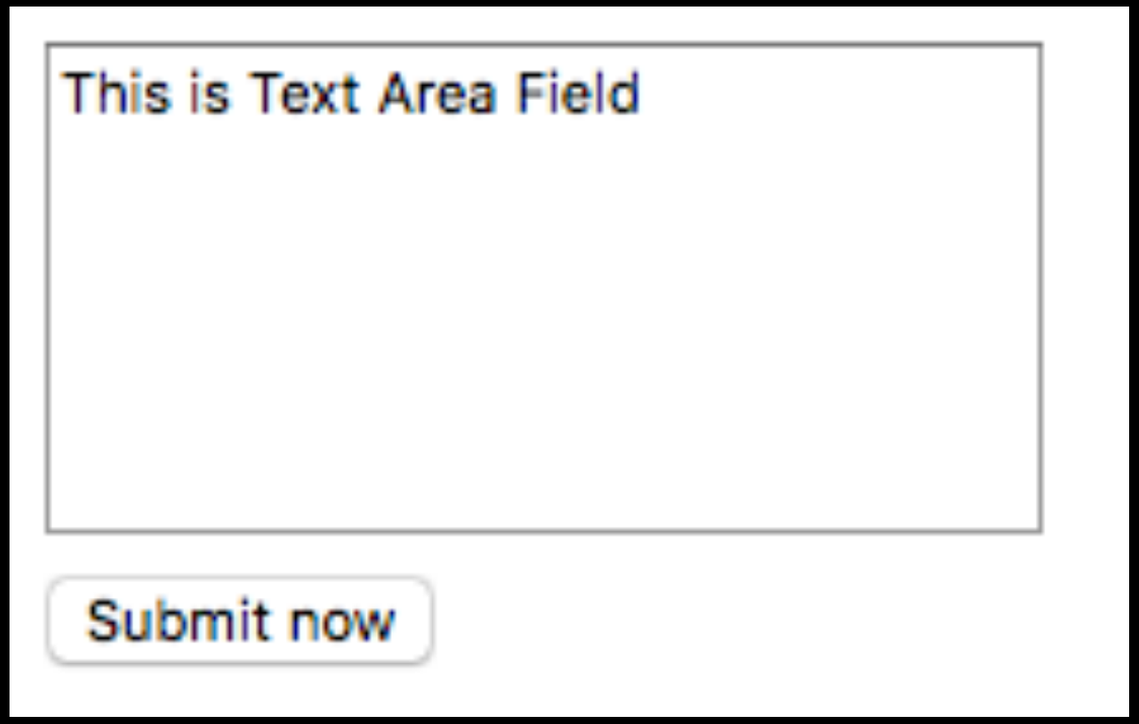
```
<form>
<fieldset>
  <legend>Contact Details</legend>
 <label for="name">Name:</label>
 <input id="name"><br />
 <label for="telephone">Telephone:</label>
 <input id="telephone"><br />
 <label for="user-email">Email:</label>
 <input id="user-email">
</fieldset>
<fieldset>
  <legend>User info</legend>
 <label for="username">Username:</lab
 <input id="username"><br />
 <label for="password">Password:</lab
 <input id="password"><br />
</fieldset>
<input type="submit" value="Submit no
</form>
```

```
<form action="">
User name:<br>
<input type="text" name="fname" value="FirstName">
<input type="text" name="lname" value="LastName">
<br>User password:<br>
<input type="password" name="psw"
    placeholder="password">
</form>
```



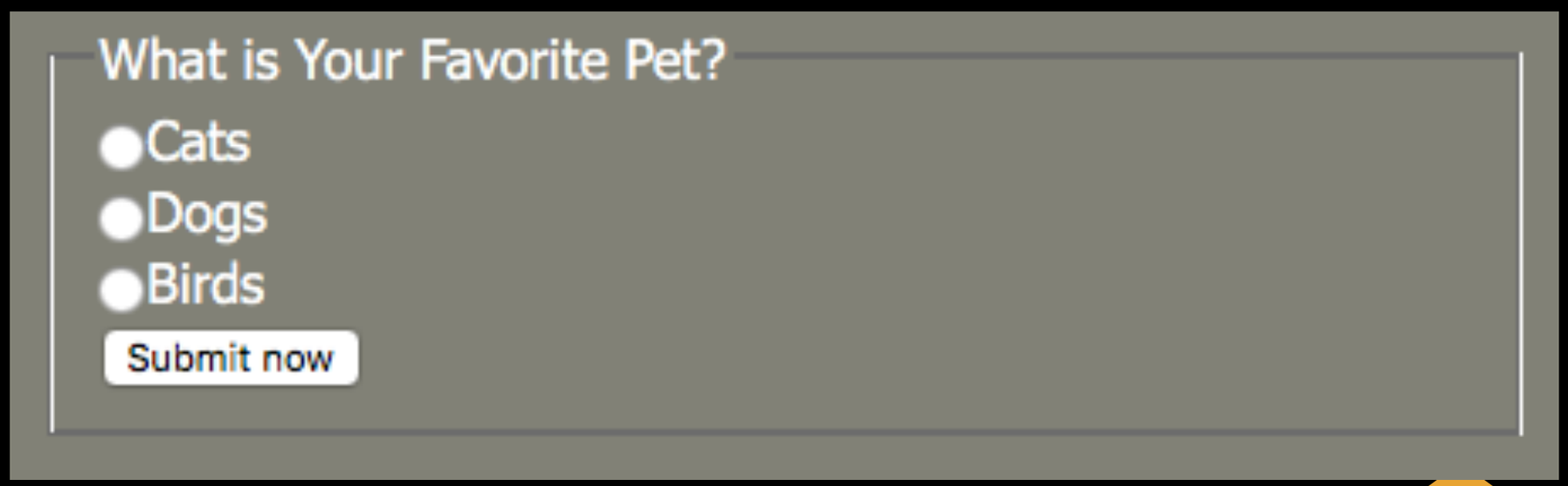
Form - Textarea

```
<form method="post">
   <textarea cols="25" rows="7">
        This is Text Area Field
   </textarea> <br />
   <input type="submit" value="Submit now" />
</form>
```



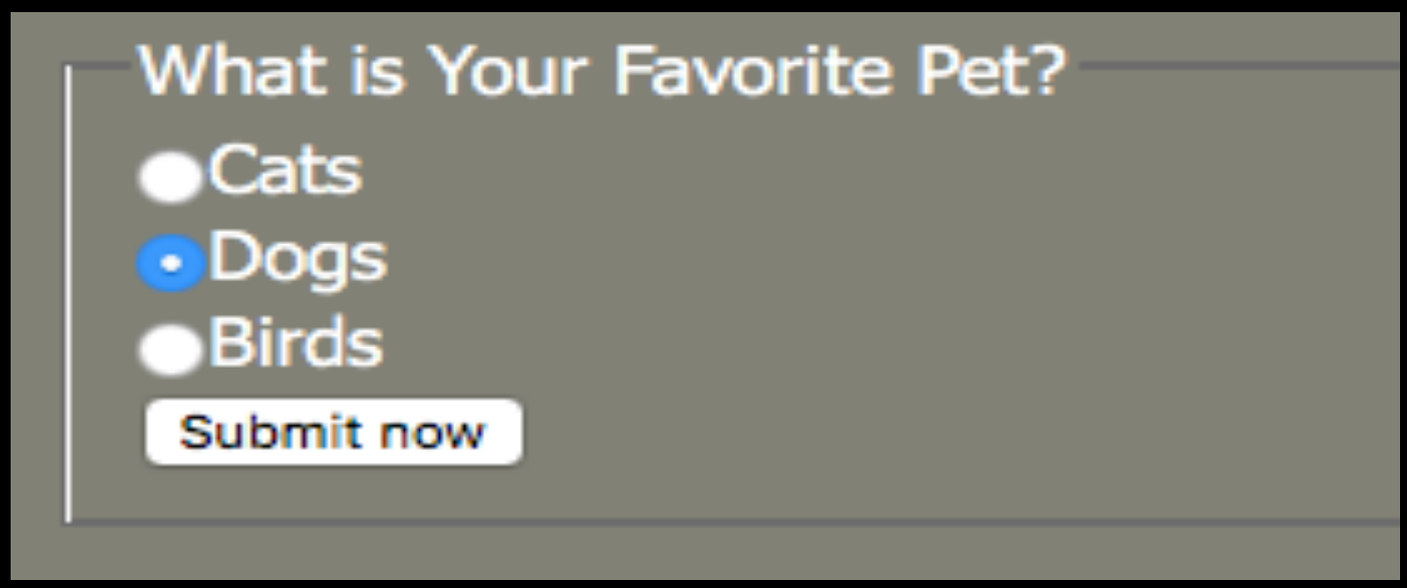
Radiobuttons

```
<form method="post">
<fieldset>
 <legend>What is Your Favorite Pet?</legend>
 <input type="radio" name="animal" value="Cat"/> Cats <br>
 <input type="radio" name="animal" value="Dog"/>Dogs <br>
 <input type="radio" name="animal" value="Bird"/>Birds
     <br>
 <input type="submit" value="Submit now" />
</fieldset>
</form>
```



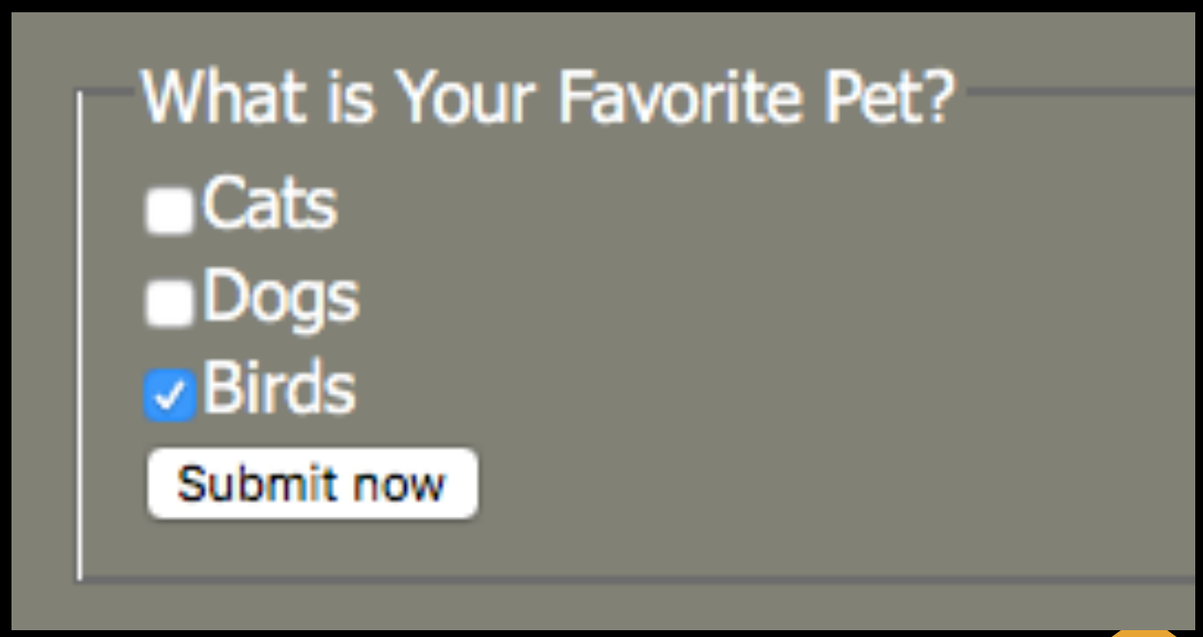
Radiobuttons - Checked

```
<form method="post">
<fieldset>
<legend>What is Your Favorite Pet?</legend>
<input type="radio" name="animal" value="Cat" />Cats
<input type="radio" name="animal" value="Dog" checked/>
                                                Dogs
<input type="radio" name="animal" value="Bird" />Birds
<input type="submit" value="Submit now" />
</fieldset>
</form>
```



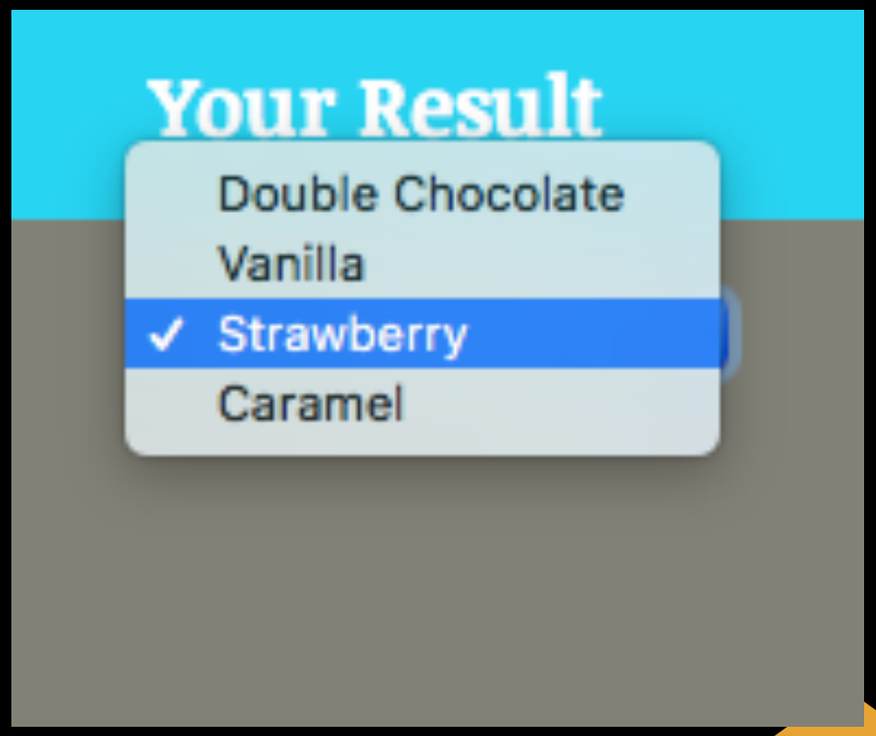
Checkboxes

```
<form method="post">
<fieldset>
<legend>What is Your Favorite Pet?</legend>
<input type="checkbox" name=“animal1" value="Cat" />Cats
<input type="checkbox" name="animal2" value="Dog" />Dogs
<input type="checkbox" name=“animal3" value="Bird"
                                    checked /> Birds
<input type="submit" value="Submit now" />
</fieldset>
</form>
```



DROPDOWN LIST

```
<form method="post">
 <select name="Icecream Flavours">
<option value="chocolate">Double Chocolate</option>
<option value="vanilla">Vanilla</option>
<option value="strawberry" selected>Strawberry</option>
<option value=“caramel”>Caramel</option>
</select>
</form>
```



HTML5 Input Types

HTML5 added several new input types:

- color
- date
- datetime-local
- email
- month
- number
- range
- search
- tel
- time
- url
- week

Input Type Number

```
<form action="">
   Quantity (between 10 and 5000):
  <input type="number" name=“quantity”
         min="10" max="5000" step=“2"
         value="30">
<input type="submit">
</form>
```

Depending on browser support:
Fixed steps will apply in the input field.

Quantity: 30 Submit

Note:type="number" is not supported in IE9 and earlier.

<input type="range">

```
<form action="">
Points:
<input type="range" name="points" min="0"
       max="10" value="6">
<input type="submit">
</form>
```

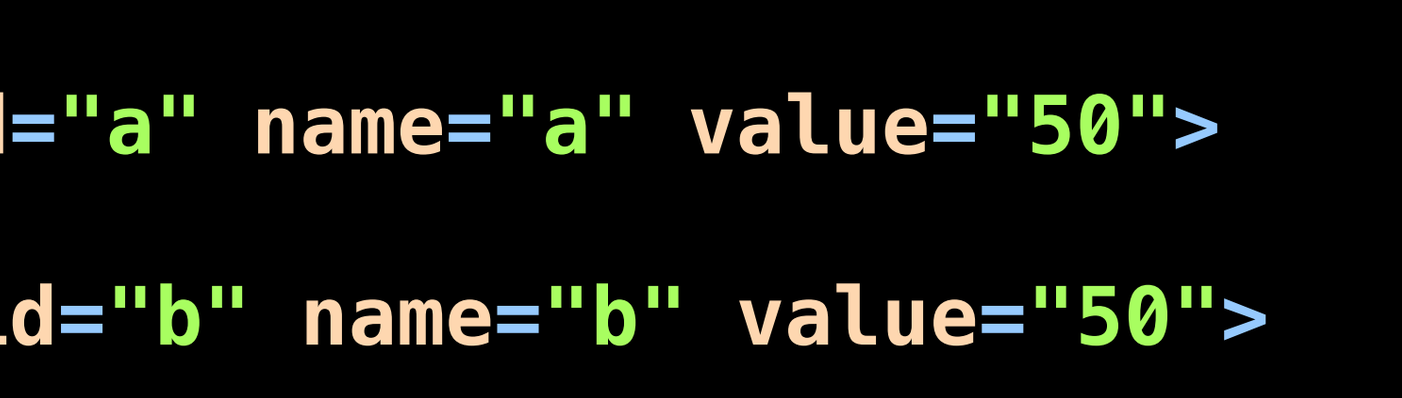
Depending on browser support:
The input type "range" can be displayed as a slider control.

Points: Submit

Note: type="range" is not supported in Internet Explorer 9 and earlier versions.

Output

```
<form action="" id="numform"
    oninput="x.value=parseInt(a.value)
    +parseInt(b.value)">
0
<input type="range" id="a" name="a" value="50">
100 +
<input type="number" id="b" name="b" value="50">
=
<input type="submit">
</form>
<output form="numform" name="x" for="a b"></
    output>
<p>The output element below is outside the form,
    but still a part of it.</p>
```



HTML5 validation

- The basic idea behind HTML5 validation is, that you tell the browser which fields you want validated but don't actually do the tedious implementation yourself.
- As you define what state your input field is in you also asks the browser to validate the field client-side based on the type of input field.

Required attribute

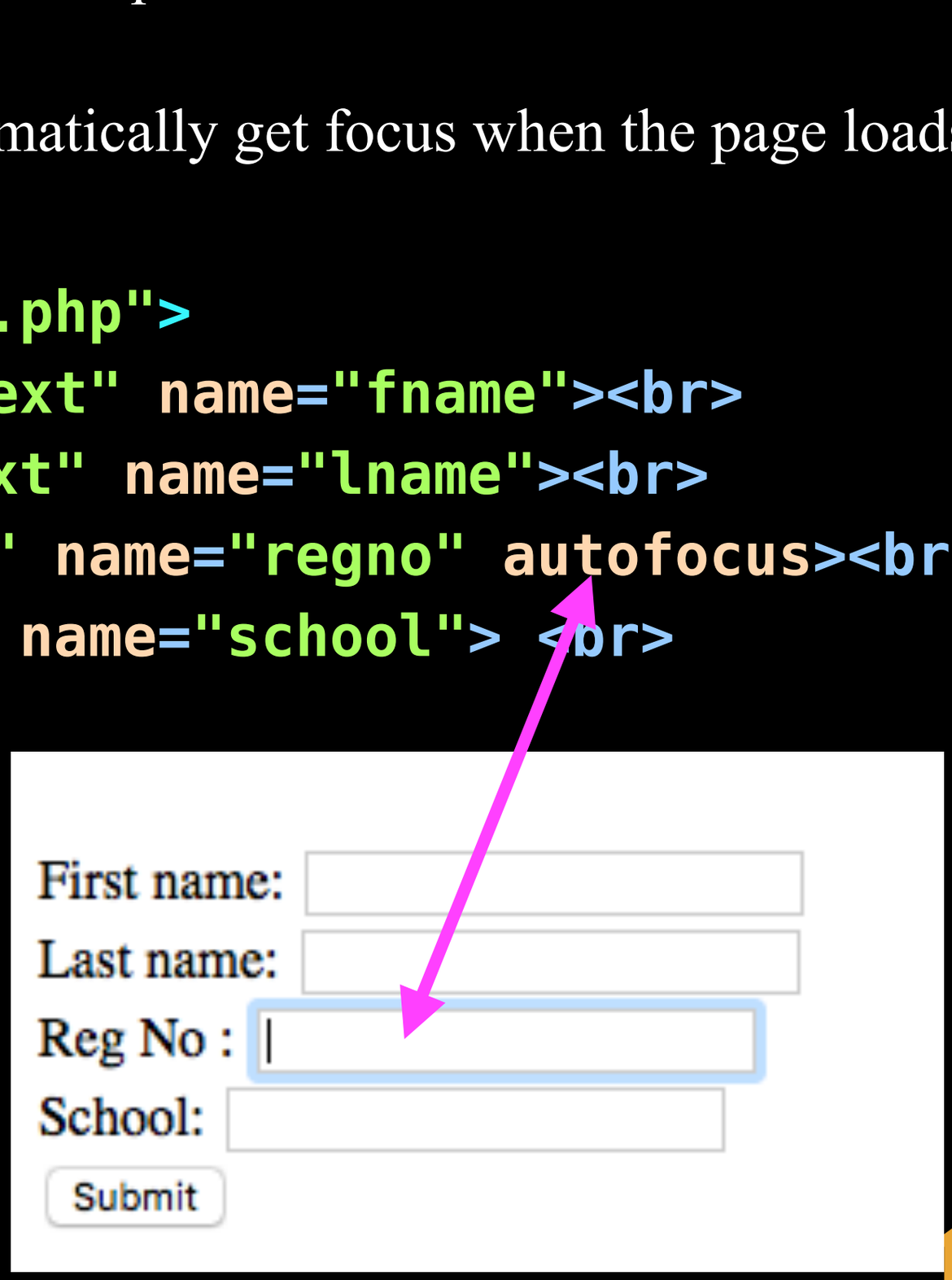
```
<input type="text" required />
```

- Using this required attribute the browser informs the user that he has to fill out this particular field before submitting the form.
- The required attribute do not take into consideration what kinds of data are typed into the input fields but you do have the opportunity to do this with some of the following input types.

Autofocus

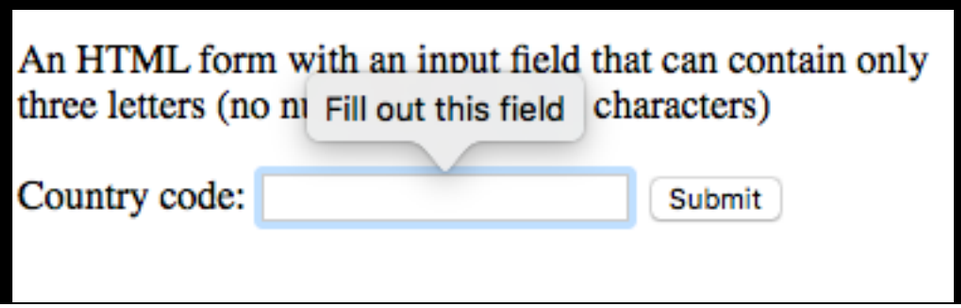
- When present, it specifies that an <input> element should automatically get focus when the page loads.
- Let the "**Reg No**" input field automatically get focus when the page loads

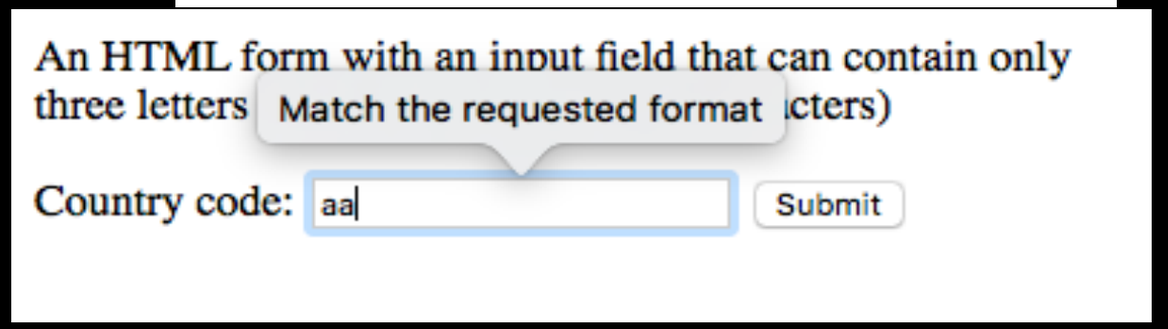
```
<form action="/action_page.php">
First name: <input type="text" name="fname"><br>
Last name: <input type="text" name="lname"><br>
Reg No : <input type="text" name="regno" autofocus><br>
School: <input type="text" name="school"> <br>
<input type="submit">
</form>
```

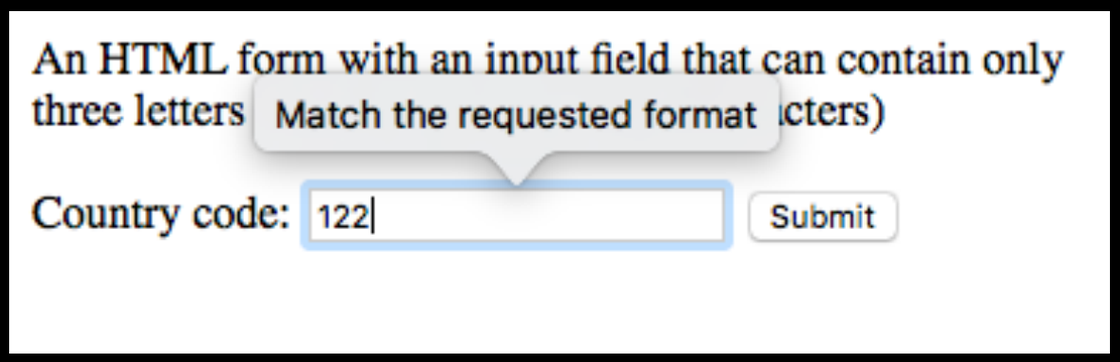


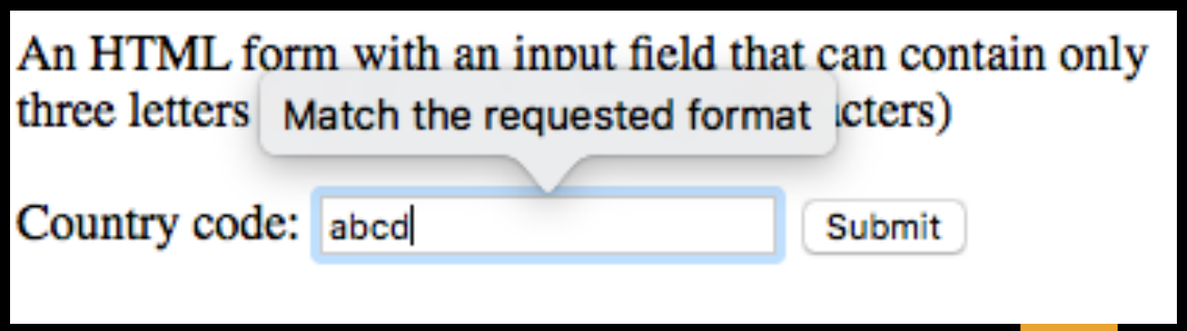
Form - Pattern

```
<p>An HTML form with an input field that can contain only three
    letters (no numbers or special characters)</p>
<form action="">
Country code:
<input type="text" name="country_code"
    pattern="[A-Za-z]{3}" title="Three letter
    country code" required>
<input type="submit">
</form>
```









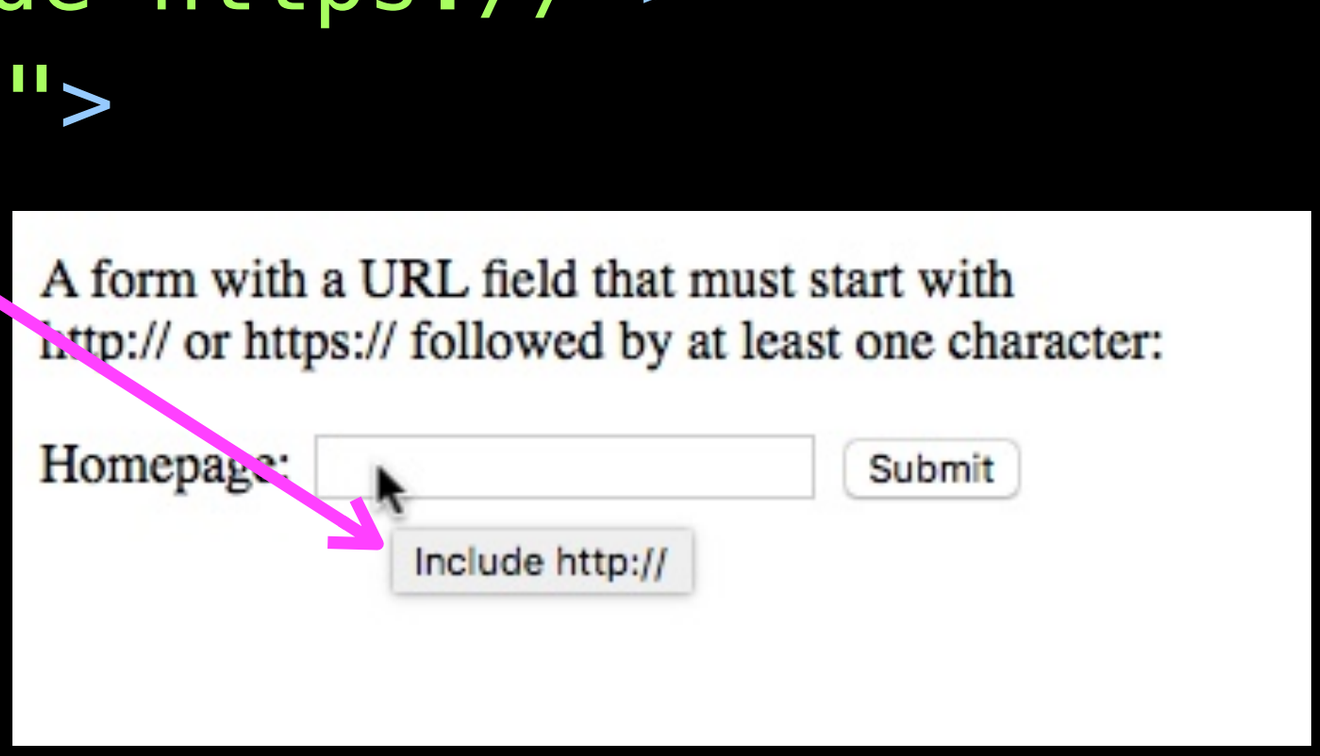
Form - email addresses

- Some browsers only look for the @ and other browsers look for at pattern consisting of a @ followed by at least one letter and a dot.

```
<form>
<input pattern="/^[a-zA-Z0-9._-]+@[a-zA-
    Z0-9-]+(\.[a-zA-Z0-9]){2,3}$/" required
    />
<br />
<input type="submit" value="Submit Now!">
</form>
```


Form - URL

```
<form action="/action_page.php">
Homepage:
<input type="url" name="website"
       pattern="https?://.+"
       title="Include https://">
<input type="submit">
</form>
```



Date

- The format for the date type is YYYY-MM-DD, which means that January 1st 2017 would be rendered 2017-01-01. This is a classic choice for anything from vacation dates to delivery dates.

```
<input type="date">
```

- Date input pattern (dd/mm/yyyy or mm/dd/yyyy):

```
<input type=“date" pattern="\d{1,2}/\d{1,2}/\d{4}">
```

- The format for the month type is YYYY-MM, which means that May 2012 would be rendered 2017-01-01

```
<input type=“month">
```

TIME

- Sometimes you need a specific time and the time type is renderes HH:mm:ss.ss but the seconds are optional.
- This means that 4.30 p.m. would be rendered 16:30 or if you choose to include seconds, 16:30:23.4

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

- **DATE & TIME**
  - ✓ The datetime type has a long format: YYYY-MM-DD THH:mm:ss.s and January 1st 2017would be rendered 2017-01-01 T16:30:23.4

```
<input type="datetime">
```

HTML5 <input> Attribute

- `<input type="color" name="color" value="#aa00cc">`

- An HTML form with a read-only input field:
  - ✓ `<input type="text" name="country" value="India" readonly>`

- `<input type="search" name="googlesearch">`

- The disabled attribute specifies that the input field is disabled.
- A disabled input field is unusable and un-clickable, and its value will not be sent when submitting the form:
- `<input type="text" name="firstname" value="John" disabled>`

HTML5 <input - Autocomplete> Attribute

- The autocomplete attribute specifies whether a form or input field should have autocomplete on or off.
- When autocomplete is on, the browser automatically complete the input values based on values that the user has entered before.

```
<form action="" autocomplete="on">
   First name:
   <input type="text" name="fname"><br>
   Last name:
   <input type="text" name="lname"><br>
   E-mail:
  <input type="email" name="email" autocomplete="off">
  <input type="submit">
</form>
```


HTML MULTIMEDIA

Media Tags

<video> - <audio> - <source> - <track>

Old Days of HTML

- Before HTML5, browsers could only natively display one type of multimedia – **animated GIF**
  - ★ <img src="omgcat.gif">
- Later, plugins appeared
  1991: Apple Quicktime (.mov)
  1994: MIDI (background music)
  1995: RealAudio (.ra, .ram)
  1997: RealVideo (H.263)
  1998: Windows Media Player
  1999: Quicktime for Windows (.mov, .mp3, .mp4, .swf)
- 2002: Macromedia Shockwave Flash (.swf)
  - ★ <embed type="application/x-shockwave-flash" src="flash.swf"/>
- 2007: Microsoft Silverlight (flash, vid, etc.)

HTML5 Multimedia

- In HTML5, you can embed audio or video using native HTML tags **audio** and **video**, and if the browser supports the tags, it will give users controls to play the file.
  - ✓ No plugins needed
  - ✓ Better performance Native
  - ✓ accessible controls
- The **audio** element is used for embedding an audio player inside a page for a particular audio file.
  - ✓ <audio src=“music.ogg" controls="true" preload=“true”></audio>
- The **video** element embeds a video player for a particular video file.
  - ✓ <video src="movie.ogv" controls width="390"></video>

Multimedia Codecs

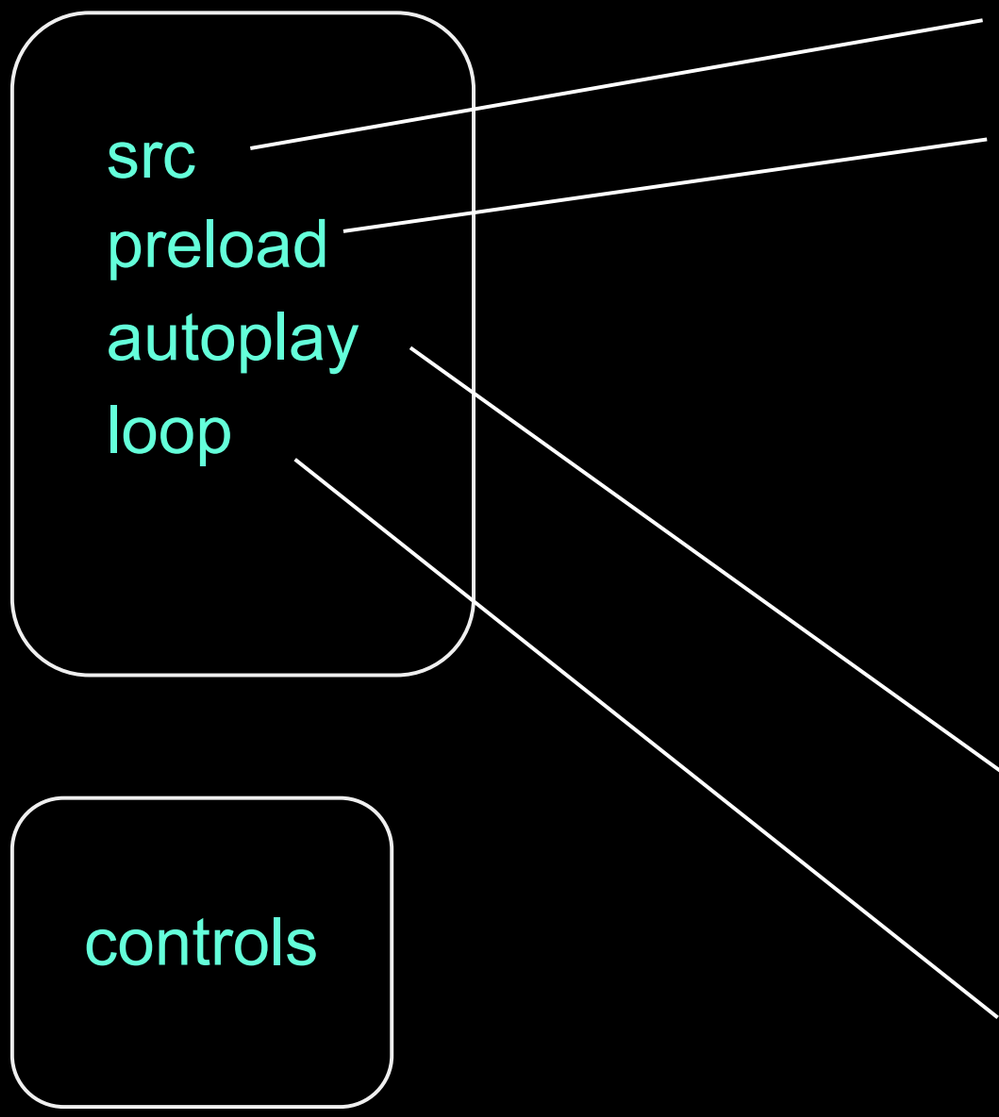
- Audio - An audio track is compressed, stored, and decoded according to a codec. The most relevant audio codecs are:
  - ✓ **MP3:** Patent-encumbered.
  - ✓ **AAC (Advanced Audio Coding):** Patent-encumbered. Used in Apple products.
  - ✓ **Ogg Vorbis:** Free, open-source, patent-free
- Video file, like an ".avi" file, is really a container for multiple related files that describe a video, like video track, one or more audio tracks with synchronization markers and metadata (title, album art, etc). The most popular video containers are:
  - ✓ **MPEG4:** .mp4, .m4v
  - ✓ **Ogg:** .ogv
  - ✓ **WebM:** .webm

<audio> element

The <audio> tag is new to HTML, like the <video> tag, and allows developers to embed music on their websites.

```
<h2>1st Method</h2>
<audio src="audio.mp3" controls></audio>
<br>
<h3> 2nd Method </h3>
<audio controls autoplay loop>
    <source src="audio.ogg" type="audio/ogg">
    <source src="audio.mp3" type=“audio/mpeg">
    <source src=“audio.wav" type="audio/wav">
    Your browser does not support the audio element.
</audio>
```

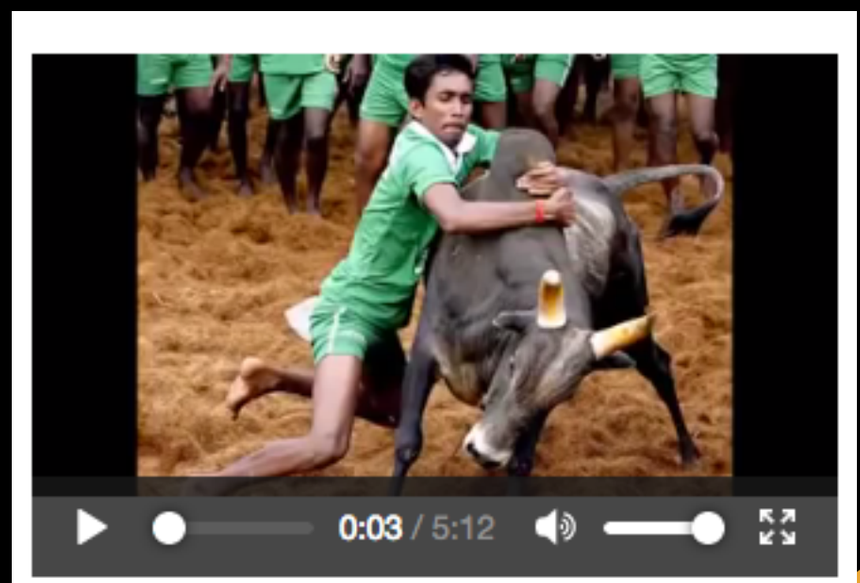
<audio> element



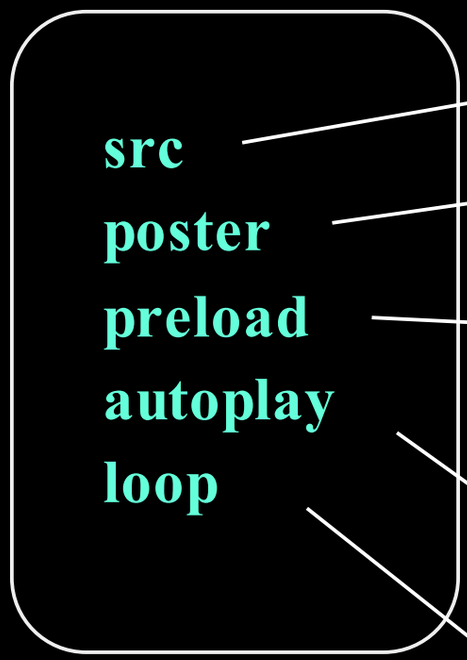
<video> element

In modern browsers, adding a video to your page is as easy as adding an image. No longer do you need to deal with special plug-ins or require crazy markup, you can do it with a single element.

```
<video width="320" height="240" controls autoplay>
    <source src="video.mp4" type="video/mp4">
    <source src="video.ogg" type="video/ogg">
    <source src="video.WebM" type="video/webm">
    Your browser does not support the video tag.
</video>
```



<video> element



Video URL (needs to match a valid MIME Type)

Image URL that the user agent can show while no video data is available

none: Tells user agent that isn't necessary load media resource (minimise traffic).
metadata: Tells user agent that it could load basic informations from media resource (dimension, first frame, duration, etc).
auto: Tells user agent that it could load the entire media resource.

Starts the video automatically.

Repeat the video when it ends

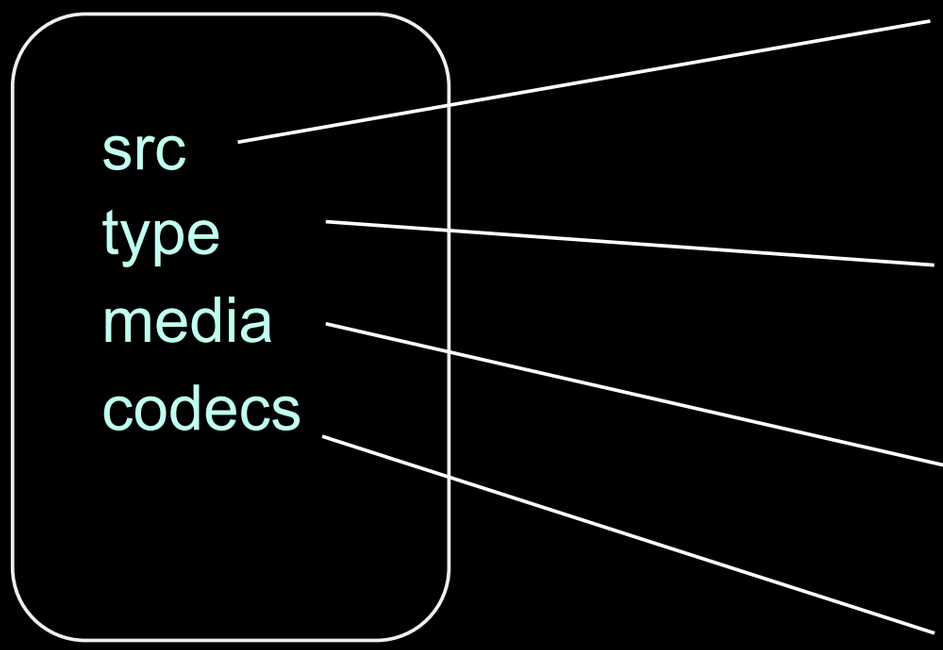
<source> element

An audio player with two source files. The browser should choose which file (if any) it has support for

```
<audio controls>
<source src="horse.ogg" type="audio/ogg">
<source src="horse.mp3" type="audio/mpeg">
Your browser does not support the audio element.
</audio>
```

```
<picture>
<source media="(min-width: 650px)"
        srcset="img_pink_flowers.jpg">
<source media="(min-width: 465px)"
        srcset="img_white_flower.jpg">
<img src="img_orange_flowers.jpg" alt="Flowers"
        style="width:auto;">
</picture>
```

<source> element



Audio/Video URL (needs to match a valid MIME Type)

Tells the user agent the exactly type of this file (ex: video/mp4).

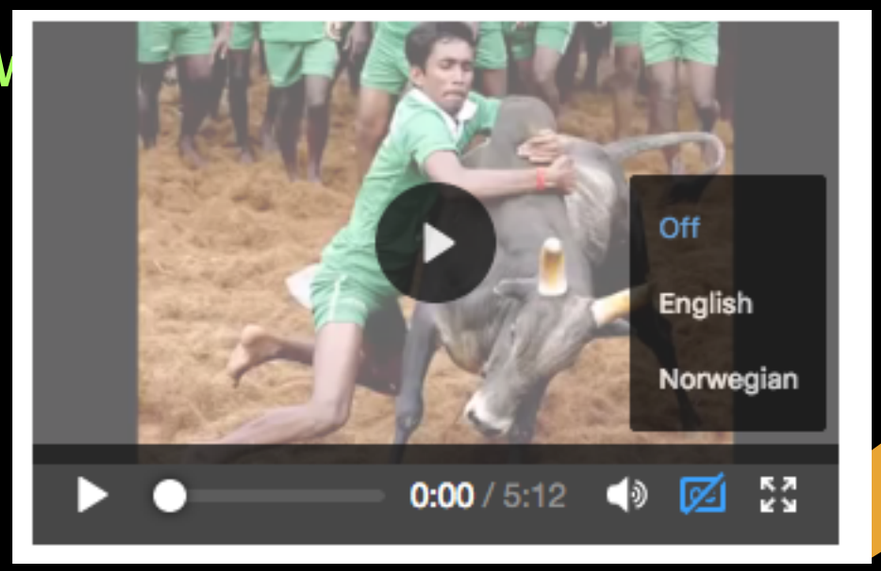
Media query (like in CSS).

Define the codec necessary for this file

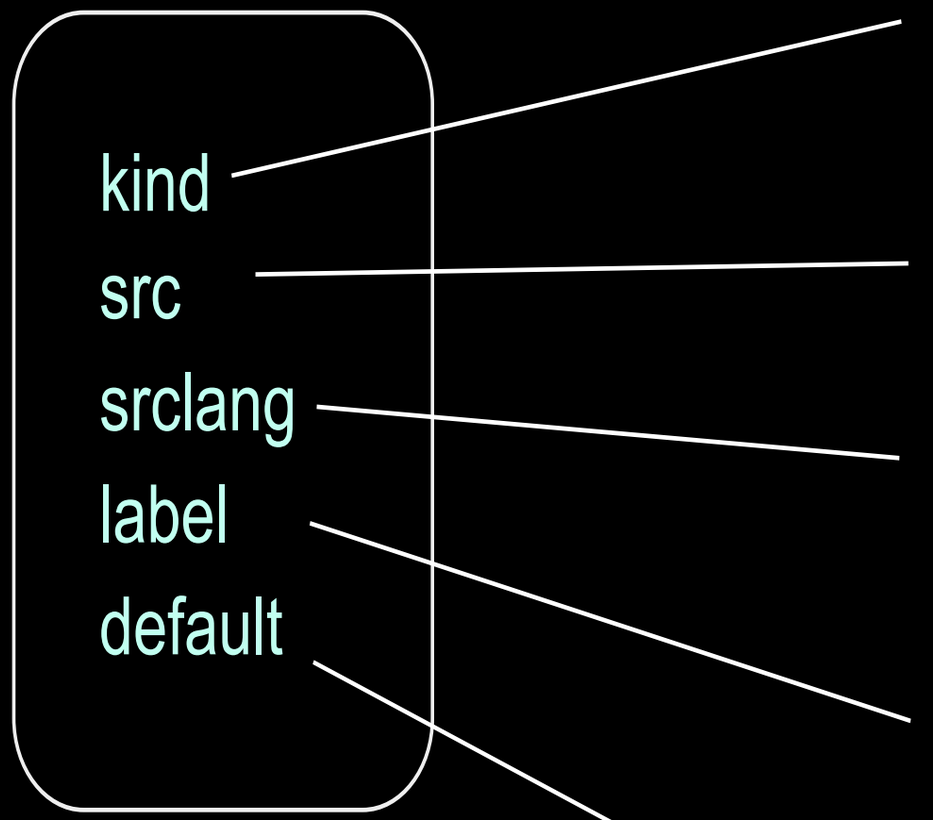
<track> element

- The **<track>** element provides a simple, standardized way to add subtitles, captions, screen reader descriptions and chapters to your video, which improves accessibility but also makes it possible for search engines to understand what's in the video.

```
<video width="320" height="240" controls>
    <source src="video.mp4" type="video/mp4">
    <source src="video.ogg" type="video/ogg">
    <track src="subtitles_en.vtt" kind="subtitles"
        srclang="en" label="English">
    <track src="subtitles_no.vtt" kind="subtitles"
        srclang="no" label="Norw
</video>
```



<track> element



Track type: **subtitles**, **captions**, **descriptions**, **chapters**, **metadata**

Track URL

The track language (spanish, portuguese, etc).

Define a title for Track (normally used when has more than one).

Default track when don't have a track matching user preference. (ex: language).

HTML5 Semantic Elements

Semantics is the study of the meanings of words and phrases in language.
Semantic elements are elements with a meaning.

What are Semantic Elements?

- A semantic element clearly describes its meaning to both the browser and the developer.
- Examples of **non-semantic** elements: and - Tells nothing about its content.
- Examples of **semantic elements**: <form>, <table>, and <img> - Clearly defines its content.

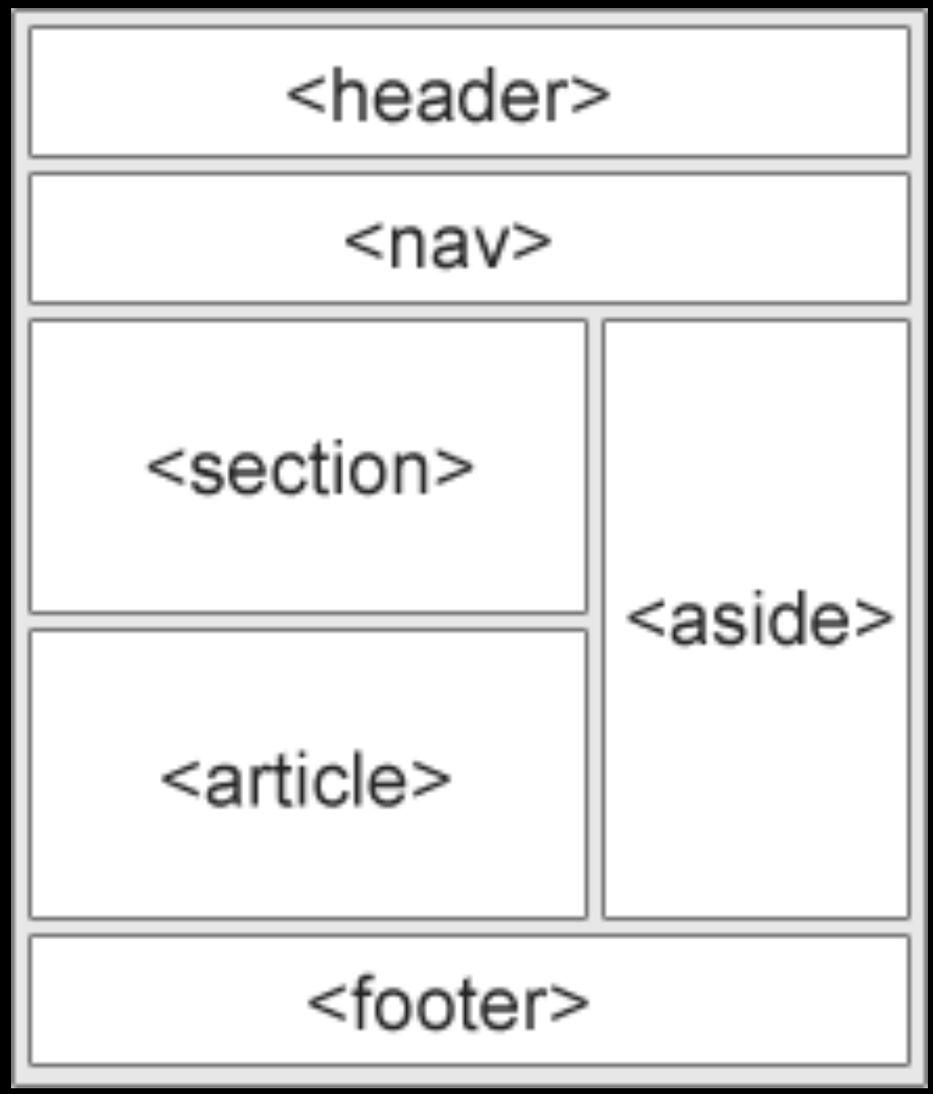
Why Semantic HTML5 Elements?

- With HTML4, developers used their own favorite attribute names to style page elements:
  - ✓ Header, top, bottom, footer, menu, navigation, main, container, content, article, sidebar, topnav, ...
- This made it impossible for search engines to identify the correct web page content.
- With HTML5 elements like:
  - ✓ <header> <footer> <nav> <section> <article>, this will become easier.
- "Allows data to be shared and reused across applications, enterprises, and communities."

New Semantic Elements in HTML5

HTML5 offers new semantic elements to define different parts of a web page:

- ✓ <article>
- ✓ <aside>
- ✓ <details>
- ✓ <figcaption>
- ✓ <figure>
- ✓ <footer>
- ✓ <header>
- ✓ <main>
- ✓ <mark>
- ✓ <nav>
- ✓ <section>
- ✓ <summary>
- ✓ <time>



HTML5 <section> Element

The <section> element defines a section in a document.

"A section is a thematic grouping of content, typically with a heading."

```
<section>
  <h1>Apple</h1>
  <p>Apple revolutionized personal
    technology with the introduction of the
    Macintosh in 1984. </p>
</section>
```


HTML5 <article> Element

- The **<article>** element specifies independent, self-contained content.
- An article should make sense on its own, and it should be possible to read it independently from the rest of the web site.

```
<article>
  <h1>What Does ARTICLE Do?</h1>
  <p>This is the ARTICLE tag in HTML 5.</p>
</article>
```


HTML5 <header> Element

- The **<header>** element specifies a header for a document or section.
- The **<header>** element should be used as a container for introductory content.
- You can have several **<header>** elements in one document.

```
<article>
   <header>
      <h1>What Does HEADER Tag Do?</h1>
      <p>Headers  mission:</p>
   </header>
   <p>This is Header tag In HTML 5<p>
</article>
```

HTML5 <footer> Element

- The **<footer>** element specifies a footer for a document or section.
- A <footer> element should contain information about its containing element.
- You can have several <footer> elements in one document.

```
<footer>
   <p>For Further Info contact us</p>
   <p>Contact information:
        <a href="mailto:jayakumars@vit.ac.in">
                Mail to Jay!!!</a>
   </p>
</footer>
```


HTML5 <nav> Element

- The **<nav>** element defines a set of navigation links.
- The <nav> element is intended for large blocks of navigation links. However, not all links in a document should be inside a <nav> element!

```
<nav>
  <a href="/html/">HTML</a>
  <a href="/css/">CSS</a>
  <a href="/js/">JavaScript</a>
  <a href="/jquery/">jQuery</a>
</nav>
```

Other HTML5 semantic elements

- The HTML5 spec also includes several other tags with more precise uses. With the exception of <figcaption>, which Webflow automatically adds to image captions, these aren't yet available in Webflow.
- <bdi> defines a section of text that might be formatted in a different direction from other text (for instance, a quote in Hebrew or Arabic in an otherwise-English article)
- <details> defines additional details that people can view or hide (like a tooltip)
- <dialog> defines a dialog box or window
- <figcaption> defines the caption for a <figure>

Other HTML5 semantic elements

- <mark> defines marked or highlighted text
- <menuitem> defines a command/menu item that the user can select from a popup menu
- <meter> defines a scalar measurement within a known range (a gauge)
- <progress> defines the progress of a task
- <rp> defines what to show in browsers that do not support ruby annotations

Other HTML5 semantic elements

- <rt> defines an explanation/pronunciation of characters (for east asian typography)

- <ruby> defines a ruby annotation (for east asian typography)

- <summary> defines a visible heading for a <details> element

- <time> defines a date/time
- <wbr> defines a possible line-break

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>My Title</title>
</head>
<body>
   <header>
        <nav>..navigation menu links here…</nav>
   </header>
   <article>
        <section>…</section>
        <section>…</section>
        <section>…</section>
   </article>
<aside>…</aside>
<footer>…</footer>
</body>
</html>
```

Reference

- *www.w3schools.com*
- *www.html5rocks.com*
- *www.tutorialspoint.com/html*

Thanks...