



# CoSc3081

## Web Programming

---

Instructor: Zinabu H.

[zinabuscholar@gmail.com](mailto:zinabuscholar@gmail.com)  
[zinabu@aku.edu.et](mailto:zinabu@aku.edu.et)

2024

# Chapter 2

## Hyper Text Markup Language

- Introduction to HTML
- HTML tags



# What is HTML?

---

- HTML (HyperText Markup Language) is used to **structure** and format the content of websites on the World Wide Web.
- Web Developers use it to create a **skeleton** of modern websites and web apps.



# What is HTML?

---

- ▣ **HyperText**

HyperText is a way of organizing text that allows the reader to easily navigate and access related information.

- ▣ **Markup language**

A markup language is a computer language that is used to add structure and formatting to a text document.



# HTML Structure

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

```
<!DOCTYPE html>
<html>
  <head></head>
  <body></body>
</html>
```

- Tags have attributes

```

```

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



# HTML

---

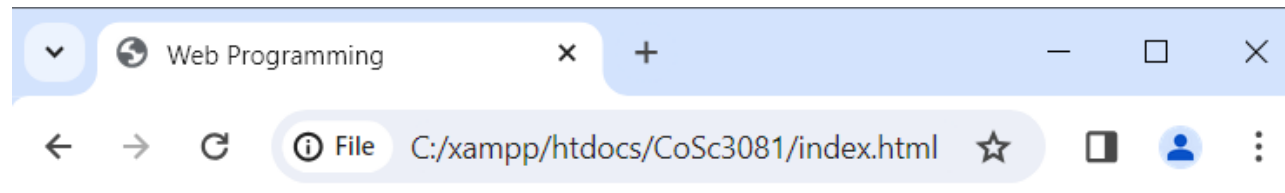
- ❑ The HTML source code should be formatted to increase readability and facilitate debugging.
- ❑ Every block element should start on a new line.
- ❑ Every nested (block) element should be indented.
- ❑ Browsers ignore multiple whitespaces in the page source, so formatting is harmless.



# First HTML Page

index.html

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Web Programming</title>
  </head>
  <body>
    <p>This is some text that will appear on
    the web page</p>
  </body>
</html>
```



This is some text that will appear on the web page



# First HTML Page: Tags

index.html

```
<!DOCTYPE  
<html>
```

Opening Tag

Closing Tag

```
<head>
```

```
<title>Web Programming</title>
```

```
</head>
```

```
<body>
```

```
<p>This is some text that will appear on  
the web page</p>
```

```
</body>
```

```
</html>
```

An HTML element consists of an opening tag, a closing tag and the content inside.





# Meta Tag

- In HTML, the `<meta>` tag is used to provide metadata about the HTML document.
- Metadata is data about the HTML document itself, such as the character encoding, viewport settings for responsive design, authorship information, and more.
- Here are some common uses of the `<meta>` tag:
  - ▣ **Character Encoding:** Specifies the character encoding for the document.  
`<meta charset="UTF-8">`
  - ▣ **Description:** Provides a short description of the document.  
`<meta name="description" content="Description of the HTML document">`



# Meta Tag...

- Here are some common uses of the <meta> tag:
  - ▣ **Keywords:** Specifies keywords related to the document.  
`<meta name="keywords" content="keyword1, keyword2, keyword3">`
  - ▣ **Author:** Specifies the author of the document.  
`<meta name="author" content="Author Name">`
  - ▣ **Refresh:** Automatically refreshes or redirects the page after a specified time.  
`<meta http-equiv="refresh" content="5;url=http://example.com">`



# Comments: <!-- --> Tag

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

<!-- AKU Logo (a JPG file) -->



<!-- Hyperlink to the web site -->

<a href="<http://www.aku.edu.et/>"> AKU</a>

<!-- Show the news table -->

<table class="newstable">

...



# Hyperlinks: <a> Tag

- Link to a document called form.html on the same server in the same directory:

```
<a href="form.html">Fill Our Form</a>
```

- Link to a document called cs.html on the same server in the parent directory:

```
<a href="../cs.html">Computer Science</a>
```

- Link to a document called courses.html on the same server in the subdirectory cs:

```
<a href="cs/courses.html">Courses</a>
```



# Hyperlinks: <a> Tag ...

- Link to an external Web site

```
<a href="http://www.ait.edu.et" target="_blank"> AiT</a>
```

- Always use a full URL, including "http://", not just [www.ait.edu.et](http://www.ait.edu.et)
- Using the target="\_blank" attribute opens the link in a new window
- Link to an e-mail address:

```
<a href="mailto:bugs@ait.edu.et?subject=Bug+Report">  
Please report bugs here (by e-mail only)</a>
```



# Hyperlinks: <a> Tag ...

- Link to a document called apply-now.html
  - ▣ On the same server, in same directory
  - ▣ Using an image as a link button:

```
<a href="apply-now.html">  
      
</a>
```

- Link to a document called index.html
  - ▣ On the same server, in the subdirectory english of the parent directory:

```
<a href=" ../english/index.html">  
    Switch to English version  
</a>
```



# Text Formatting Tag

- Text formatting tags modify the text between the opening tag and the closing tag
- ♦ Ex. `<b>Hello</b>` makes “**Hello**” bold

<code>&lt;b&gt;&lt;/b&gt;</code>	<b>bold</b>
<code>&lt;i&gt;&lt;/i&gt;</code>	<i>italicized</i>
<code>&lt;u&gt;&lt;/u&gt;</code>	<u>underlined</u>
<code>&lt;sup&gt;&lt;/sup&gt;</code>	Sample <sup>superscript</sup>
<code>&lt;sub&gt;&lt;/sub&gt;</code>	Sample <sub>subscript</sub>
<code>&lt;strong&gt;&lt;/strong&gt;</code>	<b>strong</b>
<code>&lt;em&gt;&lt;/em&gt;</code>	<i>emphasized</i>
<code>&lt;pre&gt;&lt;/pre&gt;</code>	Preformatted text
<code>&lt;blockquote&gt;&lt;/blockquote&gt;</code>	Quoted text block
<code>&lt;del&gt;&lt;/del&gt;</code>	Deleted text – <del>strike through</del>



# HTML image inserting tag

- The HTML **<img>** tag embeds an image within the HTML web page

```

```

- The HTML image tag has 2 important attributes:
  - The **src** attribute
  - The **alt** attribute
- The src attribute is a **required attribute** for the **<img>** tag. It specifies the path (URL) to the image. It tells the browser where to look for the image.
- **Note:** The **<img>** tag is an empty tag, i.e. It doesn't require a closing tag.:





# Inserting image map tag

- With HTML, image maps, you can create clickable areas on an image.
- The HTML `<map>` tag defines an image map
- The areas are defined with one or more `<area>` tags
- The image is inserted using the `<img>` tag, the only difference from other images is that you must add a `usemap` attribute
- The usemap value starts with a hash tag `#` followed by the name of the image map, and is used to create a relationship between the image and the image map

Example : Onboard



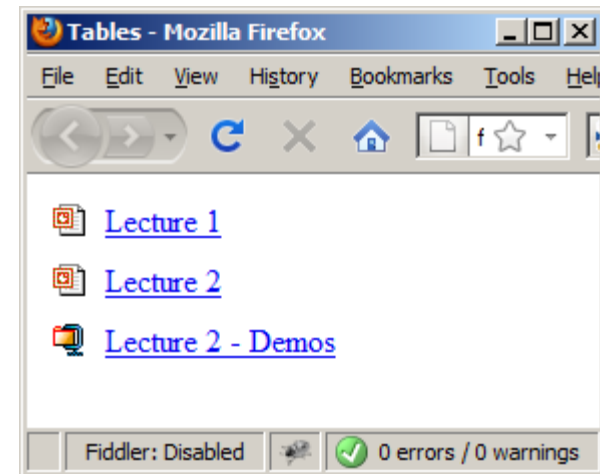
# HTML Tables

- Tables represent tabular data
  - ▣ A table consists of one or several rows
  - ▣ Each row has one or more columns
- Tables comprised of several core tags:
  - `<table></table>` : begin / end the table
  - `<tr></tr>` : create a table row
  - `<td></td>` : create tabular data (cell)
- Tables should not be used for layout. Use CSS floats and positioning styles instead



# Simple HTML Tables – Example

```
<table cellpadding="0" cellspacing="5">
  <tr>
    <td></td>
    <td><a href="lecture1.ppt">Lecture 1</a></td>
  </tr>
  <tr>
    <td></td>
    <td><a href="lecture2.ppt">Lecture 2</a></td>
  </tr>
  <tr>
    <td></td>
    <td><a href="lecture2-demos.zip">
      Lecture 2 - Demos</a></td>
  </tr>
</table>
```



# Complete HTML Tables

- Table rows split into three semantic sections: header, body and footer
  - `<thead>` denotes table header and contains `<th>` elements, instead of `<td>` elements
  - `<tbody>` denotes collection of table rows that contain the very data
  - `<tfoot>` denotes table footer but comes BEFORE the `<tbody>` tag
  - `<colgroup>` and `<col>` define columns (most often used to set column widths)



# Complete HTML Tables

- Table rows split into three semantic sections: header, body and footer
  - `<thead>` denotes table header and contains `<th>` elements, instead of `<td>` elements
  - `<tbody>` denotes collection of table rows that contain the very data
  - `<tfoot>` denotes table footer but comes BEFORE the `<tbody>` tag
  - `<colgroup>` and `<col>` define columns (most often used to set column widths)



# Complete HTML Table: Example

<table>

<colgroup>

**columns**

<col style="width:100px" /><col />

</colgroup>

<thead>

**header**

**th**

<tr><th>Column 1</th><th>Column 2</th></tr>

</thead>

**footer**

<tfoot>

<tr><td>Footer 1</td><td>Footer 2</td></tr>

</tfoot>

**Last comes the body (data)**

<tbody>

<tr><td>Cell 1.1</td><td>Cell 1.2</td></tr>

<tr><td>Cell 2.1</td><td>Cell 2.2</td></tr>

</tbody>

</table>



# Complete HTML Table: Example

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

```
<colgroup>
```

```
width:100px" /><col />
```

By default, header text is bold and centered

```
<thead>
```

```
<tr>
```

```
</thead>
```

```
<tfoot>
```

```
<tr>
```

```
</tfoot>
```

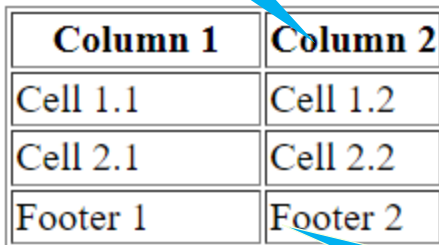
```
<tbody>
```

```
<tr><td>Cell 1.1</td>
```

```
<tr><td>Cell 2.1</td>
```

```
</tbody>
```

```
</table>
```



Column 1	Column 2
Cell 1.1	Cell 1.2
Cell 2.1	Cell 2.2
Footer 1	Footer 2

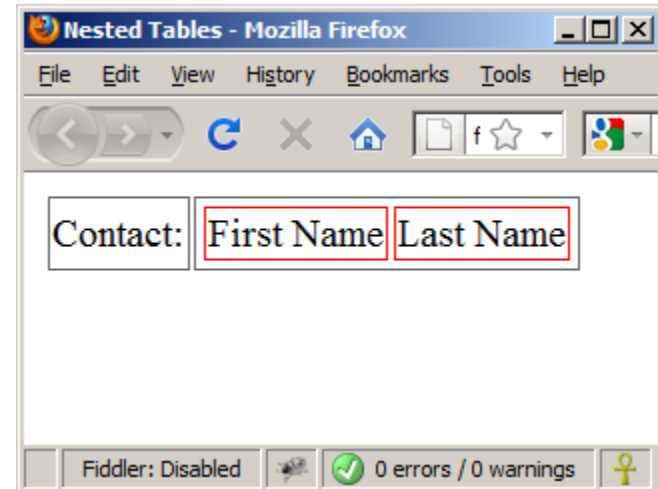
Although the footer is before the data in the code, it is displayed last



# Nested Tables

- Table data “cells” (<td>) can contain nested tables (tables within tables):

```
<table>
  <tr>
    <td>Contact:</td>
    <td><table>
      <tr>
        <td>First Name</td>
        <td>Last Name</td>
      </tr>
    </table>
  </td>
</tr>
</table>
```

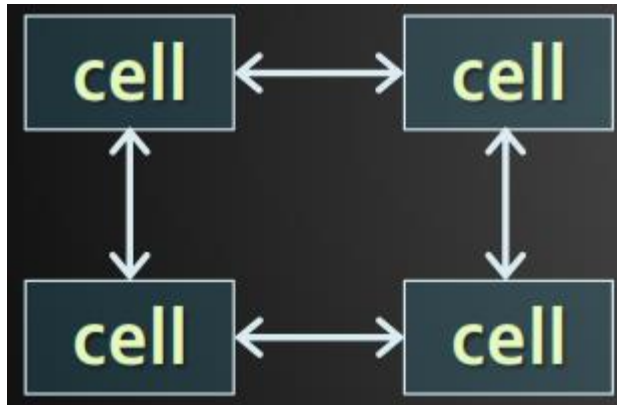




# Cell Spacing and Padding

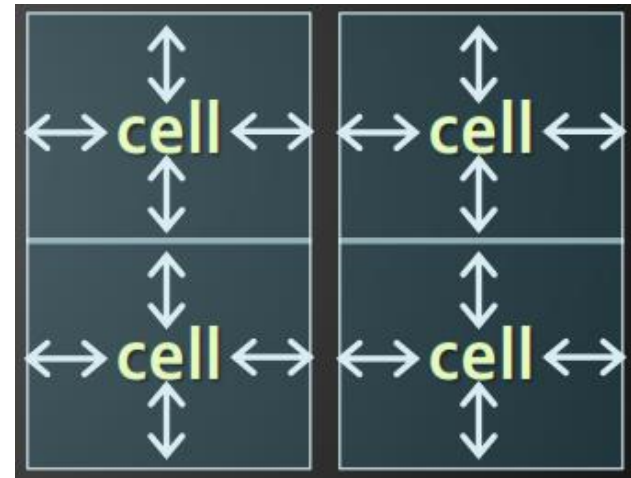
- Tables have two important attributes:

## cellspacing



Defines the  
empty space  
between cells

## cellpadding

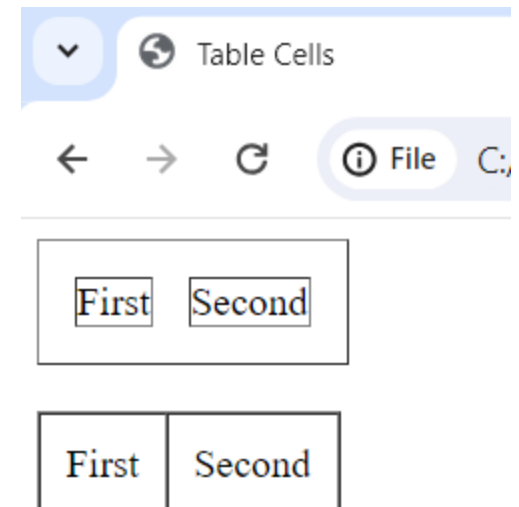


Defines the empty  
space around the cell  
content



# Cell Spacing and Padding

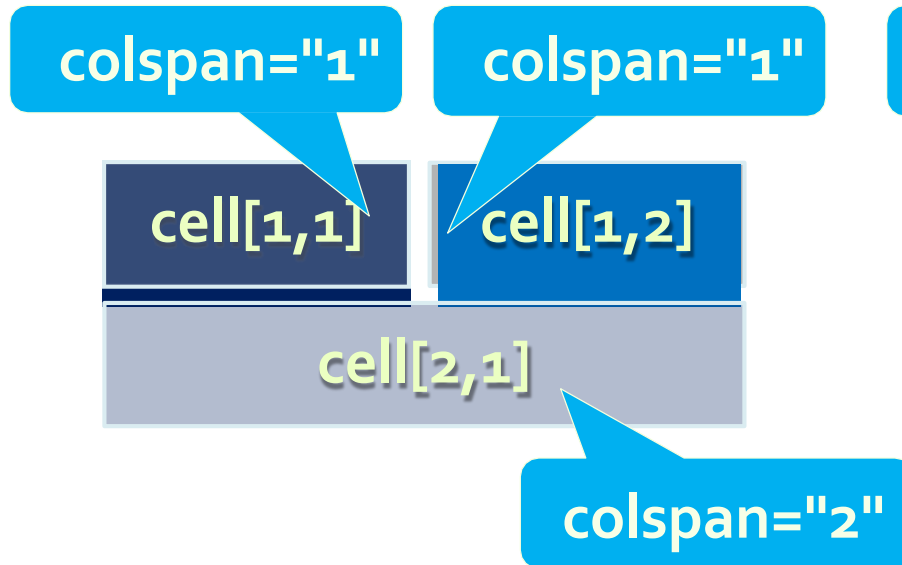
```
<html>
  <head><title>Table Cells</title></head>
  <body>
    <table cellpadding="15" cellspacing="0">
      <tr><td>First</td>
      <td>Second</td></tr>
    </table>
    <br/>
    <table cellpadding="10" cellspacing="0">
      <tr><td>First</td><td>Second</td></tr>
    </table>
  </body>
</html>
```



# Column and Row Span: Table

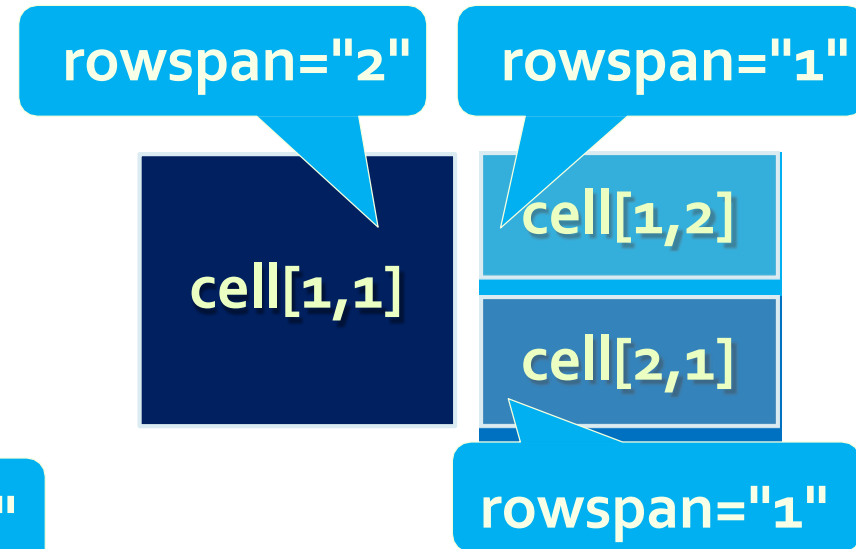
- Table cells have two important attributes:

## ◆ colspan



Defines how many columns the cell occupies

## ◆ rowspan



Defines how many rows the cell occupies



# Column and Row Span – Example

## table-colspan-rowspan.html

```
<table cellpadding="0">
  <tr class="1"><td>Cell[1,1]</td>
    <td colspan="2">Cell[2,1]</td></tr>
  <tr class="2"><td>Cell[1,2]</td>
    <td rowspan="2">Cell[2,2]</td>
    <td>Cell[3,2]</td></tr>
  <tr class="3"><td>Cell[1,3]</td>
    <td>Cell[2,3]</td></tr>
</table>
```

Cell[1,1]	Cell[2,1]	
Cell[1,2]	Cell[2,2]	Cell[3,2]
Cell[1,3]		Cell[2,3]

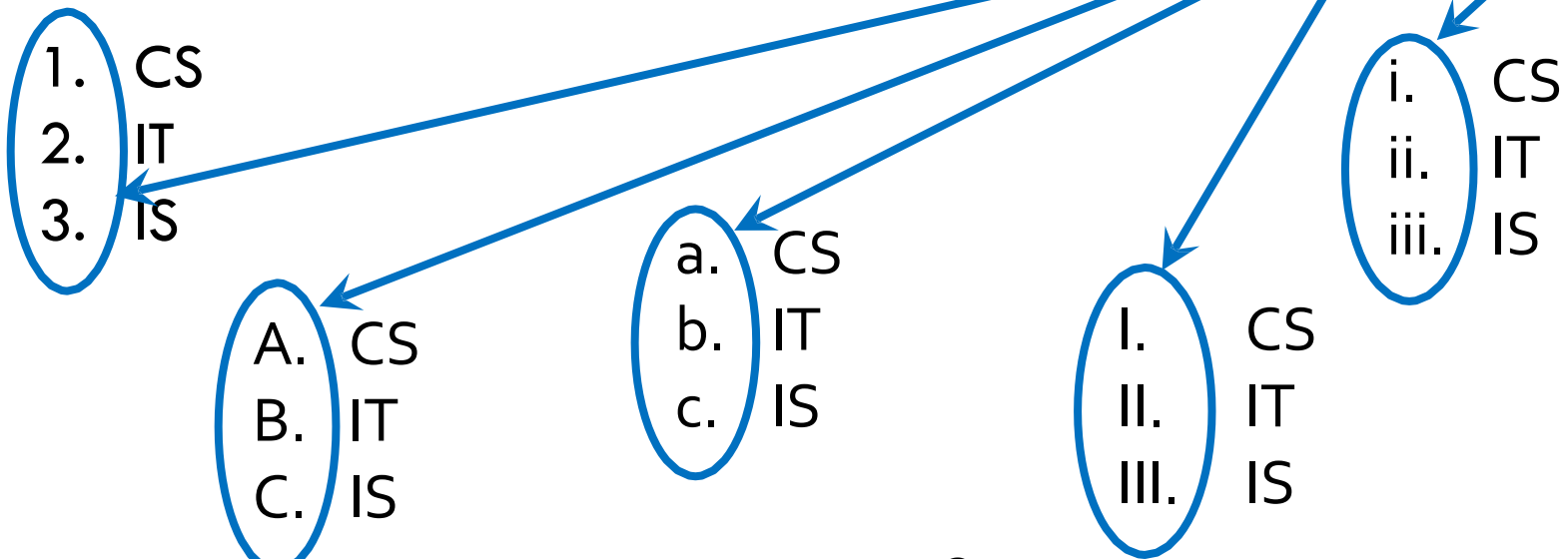


# Ordered Lists: <ol> Tag

- Create an Ordered List using `<ol></ol>`:

```
<ol type="1">  
  <li>CS</li>  
  <li>IT</li>  
  <li>IS</li>  
</ol>
```

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

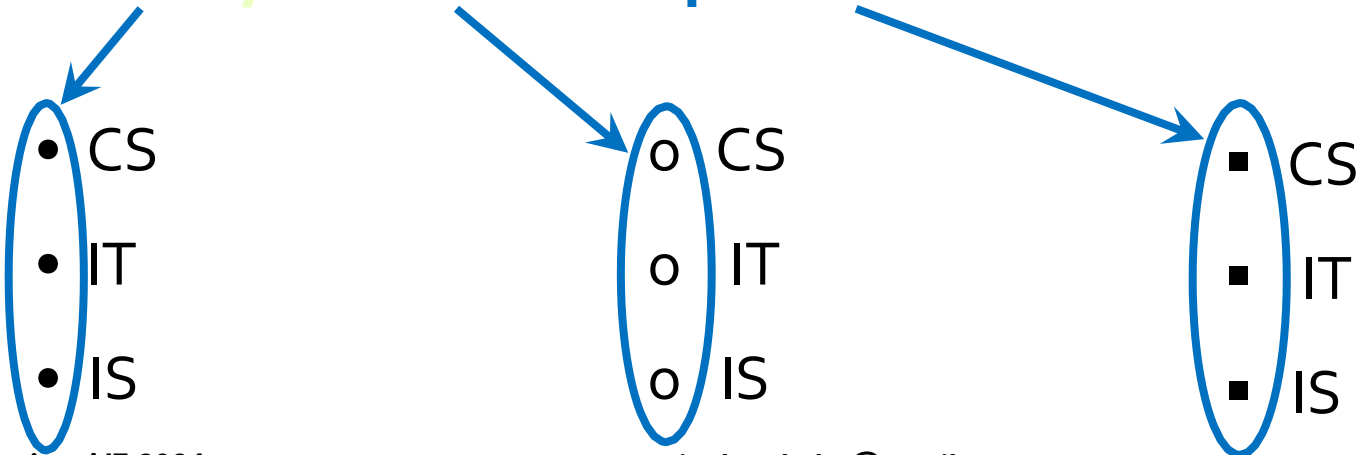


# Unordered Lists: <ul> Tag

- Create an **U**nordered **L**ist using **<ul></ul>**:

```
<ul type="disk"
  <li>CS</li>
  <li>IT</li>
  <li>IS</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



# HTML Frames

- **Frames** provide a way to show multiple HTML documents in a single Web page
- The page can be split into separate views (frames) horizontally and vertically
- Frames were popular in the early ages of HTML development, but now their usage is rejected
- Frames are not supported by all user agents (browsers, search engines, etc.)
  - A **<noframes>** element is used to provide content for non-compatible agents.





# HTML Frames : Demo

## □ Frames

```
<html>
<head>
  <title>Frames Example</title>
</head>
<frameset cols="180px,*,150px">
  <frame src="left.html" />
  <frame src="middle.html" />
  <frame src="right.html" />
</frameset>
</html>
```



# Inline Frames: <iframe>

- **Inline frames** provide a way to show one website inside another website:

```
<iframe name="iframeGoogle" width="600" height="400"  
src="http://www.google.com" frameborder="yes"  
scrolling="yes">  
</iframe>
```



# HTML Form and Form Controls

- Forms are the primary method for gathering data from site visitors
- Create a form block with

`<form> </form>`

- Example:

The "method" attribute tells how the form data should be sent – via GET or POST request

```
<form name="myForm" method="post"
action="path/to/some-script.php">
</form>
```

The "action" attribute tells where the form data should be sent



# HTML Form Elements

- A form contains special interactive elements that users use to send the input.
  - HTML <input> tag
  - HTML <label> tag
  - HTML <button> tag
  - HTML <select>, <option> and <optgroup> tags
  - HTML <textarea> tag
  - HTML <fieldset> tag
  - HTML <legend> tag
  - HTML <datalist> tag
  - HTML <output> tag

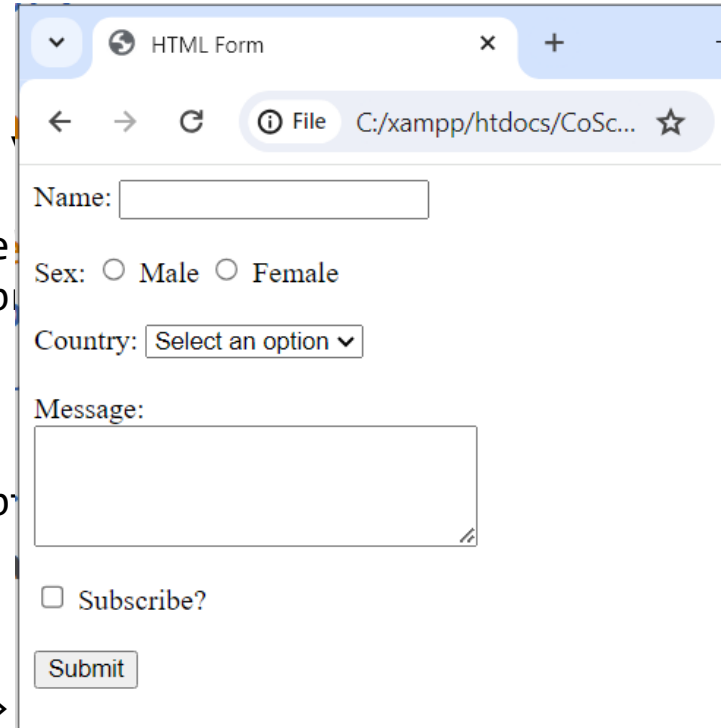


# HTML Form Elements

<form>

```
<label for="name">Name:</label>
<input type="text" name="name"><br><br>
<label for="sex">Sex:</label>
<input type="radio" name="sex" id="male"
<label for="male">Male</label>
<input type="radio" name="sex" id="female"
<label for="female">Female</label> <br><br>
<label for="country">Country: </label>
<select name="country" id="country">
  <option>Select an option</option>
  <option value="ethiopia">Ethiopia</op
  <option value="kenya">Kenya</option>
  <option value="sudan">Sudan</option>
</select><br><br>
<label for="message">Message:</label><br>
<textarea name="message" id="message" cols="30"
rows="4"></textarea><br><br>
<input type="checkbox" name="newsletter" id="newsletter">
<label for="newsletter">Subscribe?</label><br><br>
<input type="submit" value="Submit">
```

</form>



The screenshot shows a web browser window titled "HTML Form" with the address bar displaying "C:/xampp/htdocs/CoSc...". The rendered form includes a text input for "Name:", radio buttons for "Sex" (Male and Female), a dropdown menu for "Country" with the text "Select an option", a text area for "Message:", a checkbox for "Subscribe?", and a "Submit" button.



# HTML Form Attributes

- The HTML `<form>` element contains several attributes for controlling data submission. They are as follows:

## action

- The action attributes define the action to be performed when the form is submitted. It is usually the url for the server where the form data is to be sent.

```
<form action="/login">  
  <label for="email">Email:</label>  
  <input type="email" name="email"><br><br>  
  <label for="password">Password:</label>  
  <input type="password" name="password"><br><br>  
  <input type="submit" value="Submit">  
</form>
```



# HTML Form Attributes

## method

- The method attribute defines the HTTP method to be used when the form is submitted. There are **3** possible values for the *method* attribute:
- **post** - It is used to send data to a server to update a resource.
- **get**: It is used to request data from a specified resource.
- **dialog**: This method is used when the form is inside a `<dialog>` element. Using this method closes the dialog and sends a form-submit event.



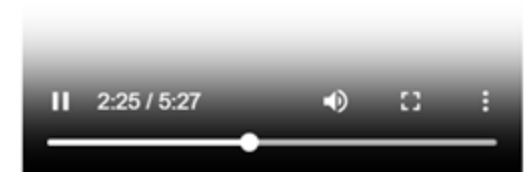
# Inserting Multimedia in HTML

## HTML Video

- The HTML `<video>` tag is used to embed a media player which supports video playback into the HTML page.
- We use the HTML `<video>` tag and the `<source>` tag to show the video
- For example

```
<video width="320" height="190" controls>  
  <source src="video.mp4" type="video/mp4">  
</video>
```

WITH AGILE  
CHANGES ARE WELCOME





# Inserting Multimedia in HTML

## HTML Audio

- The HTML `<audio>` tag is used to embed a media player which supports audio playback into the HTML page.
- We use the HTML `<audio>` tag along with the `<source>` tag to add the audio player.
- For example

```
<audio controls>  
  <source src="audio.mp3" type="audio/mp3">  
</audio>
```

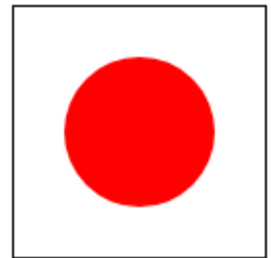


# HTML Graphics

## HTML SVG

- SVG (Scalable Vector Graphics) is used to create 2D diagrams such as shapes, logos, charts, etc
- The HTML `<svg>` tag is used to embed SVG graphics in a web page.
- Example

```
<svg width="100" height="100" style="border:1px solid black;">  
  <circle cx="50" cy="50" r="30" fill="blue" />  
</svg>
```

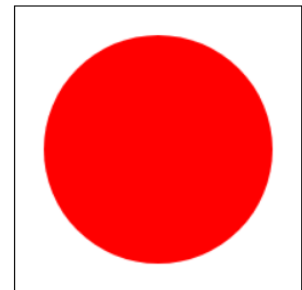


# HTML Graphics

## HTML Canvas

- HTML **<canvas>** is used to create graphics in HTML. We create the graphics inside the **<canvas>** using JavaScript.
- Example

```
<canvas id="circle-canvas" height="200" width="200" style="border: 1px solid;"></canvas>
<script>
  let canvas = document.getElementById("circle-canvas");
  let ctx = canvas.getContext("2d");
  ctx.beginPath();
  ctx.arc(100, 100, 80, 0, 2 * Math.PI, false);
  ctx.fillStyle = 'blue';
  ctx.fill();
</script>
```





# End of Chapter 2

---

**Aksum University- AIT**  
**2024**

