CS-ELECTIVE -1

RCS-E12: WEB TECHNOLOGIES

Unit -2

Web Page Designing: HTML: List, Table, Images, Frames, forms, CSS, Document type definition, XML: DTD, XML schemes, Object Models, presenting and using XML, Using XML Processors: DOM and SAX, Dynamic HTML

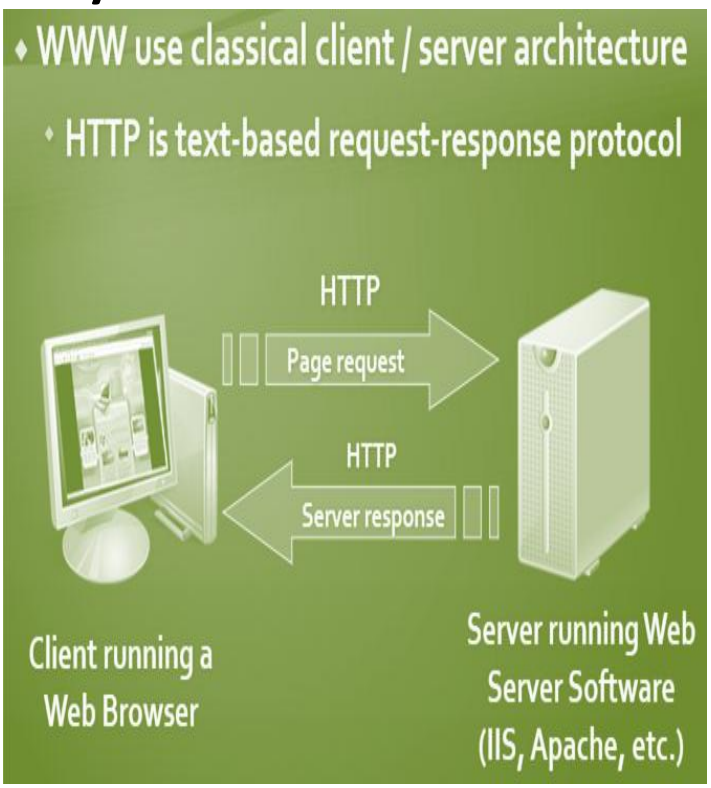
What is HTML ?

- HTML stands for Hypertext Markup Language
- HTML is not a programming Language but a Markup Language that uses **<tags>** like this.
- Language of the Browser

- It allows users to make web pages
- These web pages can include text, graphics and links to other web pages.
- The **World Wide Web** Consortium creates the standards of HTML.

Working of HTML

- By embedding all markup commands within each HTML file and standardizing them, it becomes possible for any web browser to read and reformat any web page
- HTML has some predefined tags to simplify the work of users



Structure of an HTML document

All HTML documents follow the same basic structure. They have the root tag as <html>, which contains <head> tag and <body> tag. The head tag is used for control information by the browser and the body tag contains the actual user information that is to be displayed on the screen. The basic document is shown below.

- **<html>** Marks the beginning of your HTML
  - **<head>** Begins the heading section of an HTML document
    - **<title> ... </title>** Gives an HTML document a title that appears on the browser menu bar, also will appear on search engines or bookmarks referencing your site (must appear between the <HEAD> ... </HEAD> tags; should be straight text, no tags
  - **</head>** Defines the end of the heading
  - **<body>** Defines the body of an HTML document (text contained within the <BODY> … </BODY> tags appears in the main browser window). Can be used with "BGCOLOR", "TEXT", "LINK", and "VLINK" attributes
  - **</body>**
- **</html>** Defines the end of your HTML document

Types of HTML Tags

There are two different types of tags:->

- **Container Element:** Container Tags contains start tag & end tag

  i.e. <HTML>… </HTML>

- **Empty Element:** Empty Tags contains start tag

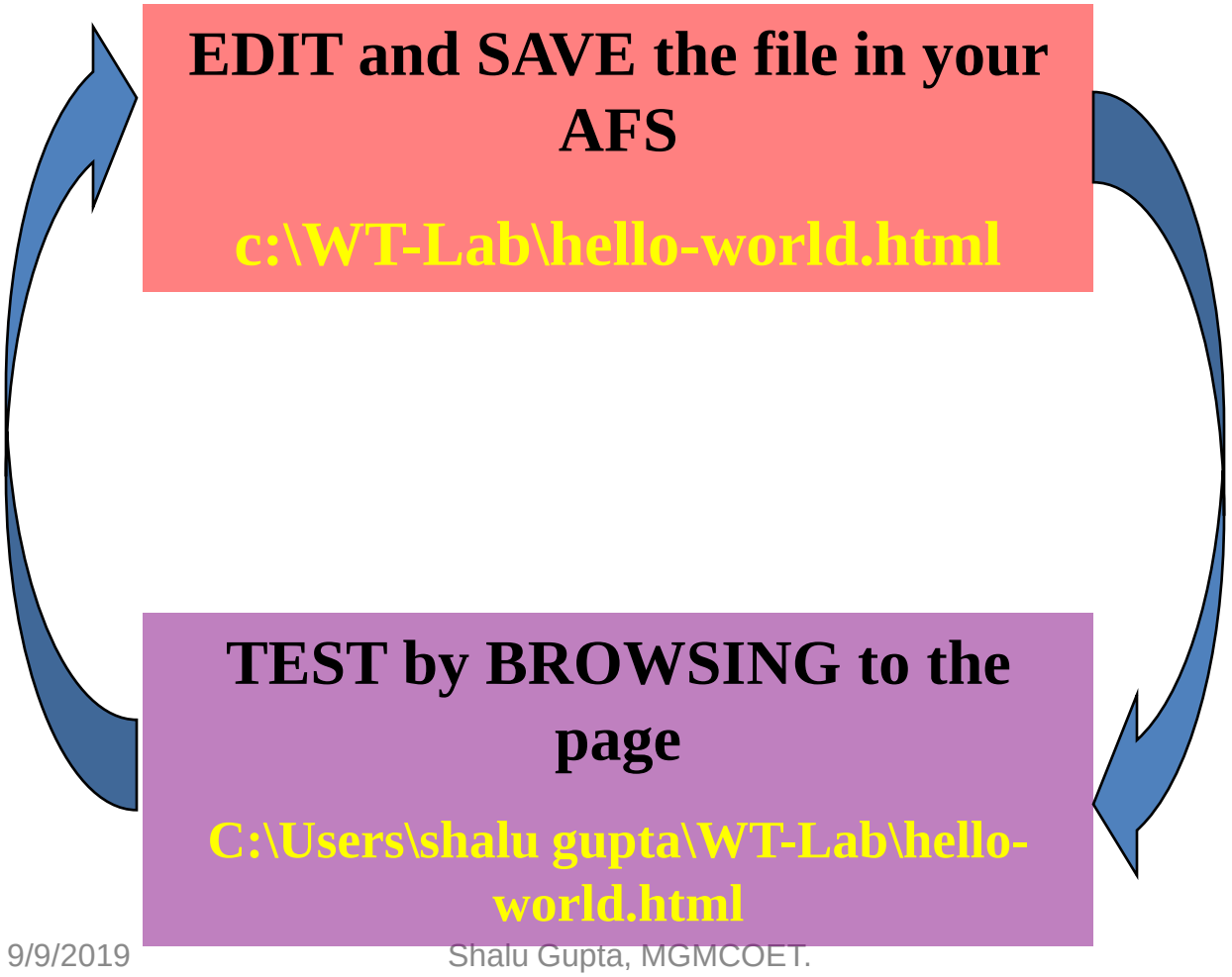
  i.e.

Some More Standard HTML Tags

- <hn>..</hn> - delimits level n heading (h1 to h6)
- .. - set in boldface
- .. - set in italics
- .. - set in center horizontally
- .. - unordered list (bullets)
- .. - ordered list (numbers)
- .. - an item on the list
-
 - line break
- <p> - paragraph
- <hr> - horizontal rule (line)
- <img src=“”> - displays an image
- .. - defines a hyperlink

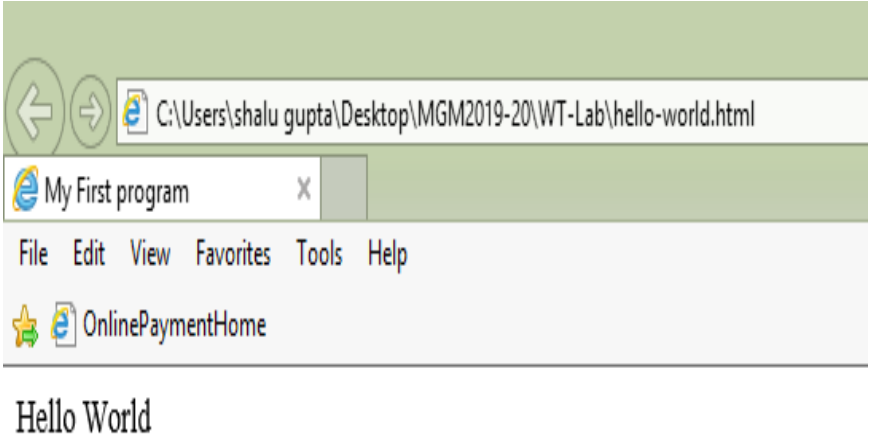
Web Page Development Cycle

Open a Notepad window. Write the HTML tags to the Notepad. Save the file with (.Html)/(.Htm) extension. View the page in any web browser viz. INTERNET EXPLORER, NETSCAPE NAVIGATOR etc. The purpose of a web browser (like internet explorer or firefox) is to read html documents and display them as web pages.



Lab Program: Write simple HTML file to print "Hello World".

- **Code:**
- <html>
- <head><title>My First program</title></head>
- <body
- <h2>Hello World</h2>
- </body>
- </html>



File Editors

- HTML files can be created with text editors:
  - NotePad, NotePad ++, PSPad
- Or HTML editors (WYSIWYG Editors):
  - Microsoft FrontPage
  - Macromedia Dreamweaver
  - Netscape Composer
  - Microsoft Word
  - Visual Studio

Some Simple Tags

- Hyperlink Tags

```
<a href="http://www.telerik.com/"
  title="Telerik">Link to Telerik Web site</a>
```

- Image Tags

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

- Text formatting tags

```
This text is <em>emphasized.</em>
<br />new line<br />
This one is <strong>more emphasized.</strong>
```

Some Simple Tags - Example

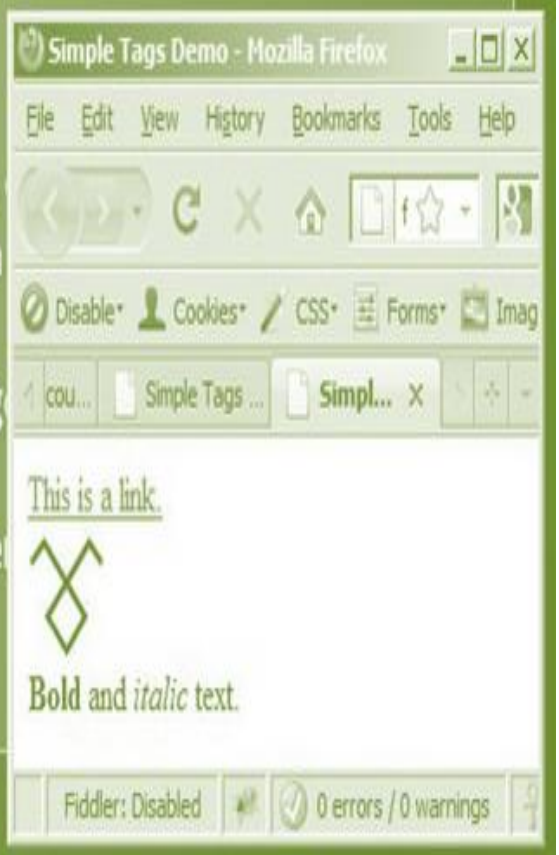
some-tags.html

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Simple Tags Demo</title>
</head>
<body>
<a href="http://www.telerik.com/" title=
    "Telerik site">This is a link.</a>
<br />
<img src="logo.gif" alt="logo" />
<br />
<strong>Bold</strong> and <em>italic</em> text.
</body>
</html>
```

Some Simple Tags - Example

some-tags.html

```
<!DOCTYPE HTML>
<html>
<head>
  <title>Simple Tags Demo</title>
</head>
<body>
<a href="http://www.telerik.
    "Telerik site">This is a
<br />
<img src="logo.gif" alt="log
<br />
<strong>Bold</strong> and <e
</body>
</html>
```



Tag Attributes

- Tags can have attributes
  - Attributes specify properties and behavior
  - Example:

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

  - Few attributes can apply to every element:
    - `id`, `style`, `class`, `title`
    - The `id` is unique in the document
    - Content of `title` attribute is displayed as hint when the element is hovered with the mouse
    - Some elements have obligatory attributes

Headings and Paragraphs

- Heading Tags (h1 – h6)

```
<h1>Heading 1</h1>
<h2>Sub heading 2</h2>
<h3>Sub heading 3</h3>
```

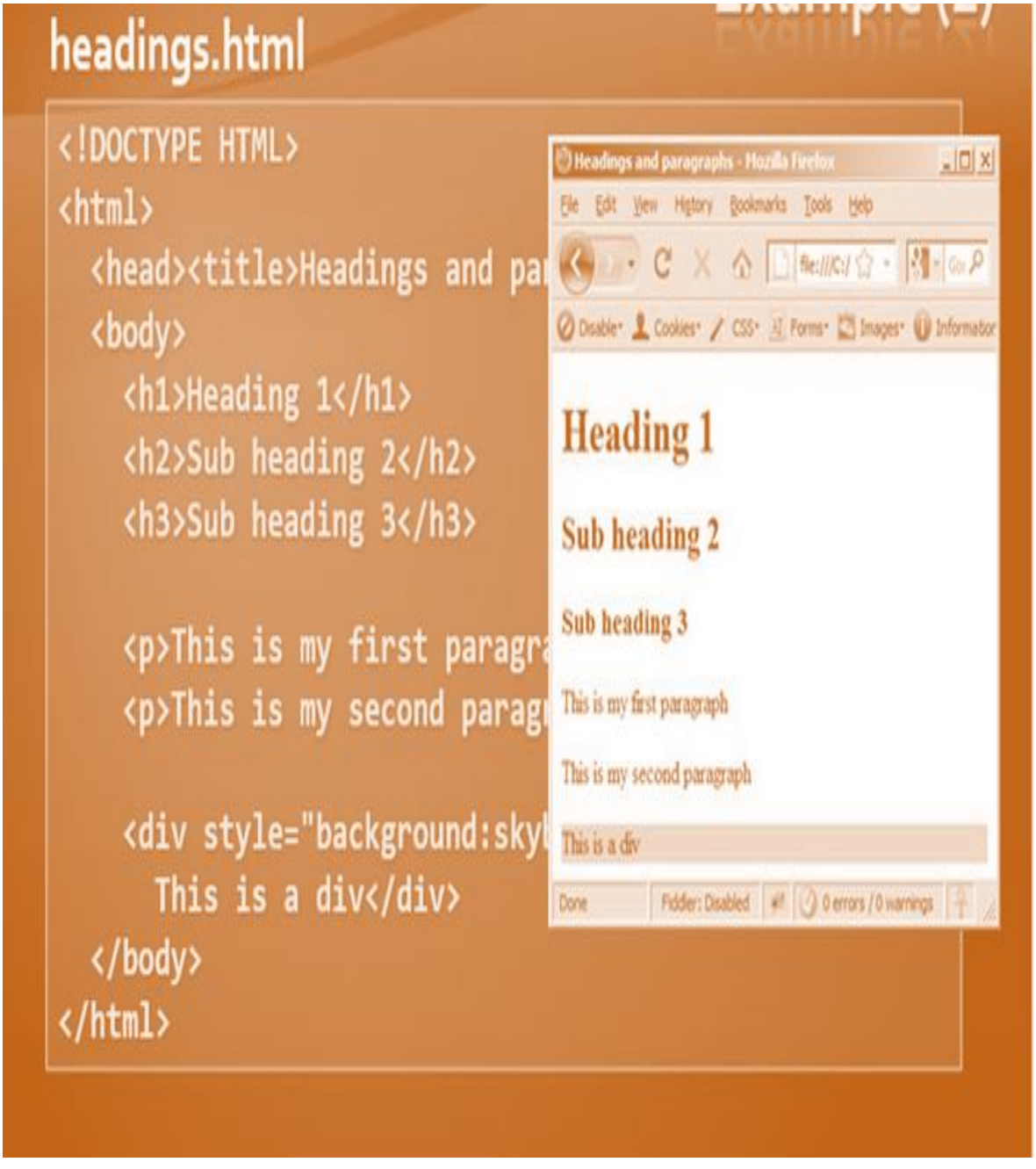
- Paragraph Tags

```
<p>This is my first paragraph</p>
<p>This is my second paragraph</p>
```

- Sections: div and span

```
<div style="background: skyblue;">
   This is a div</div>
```

Headings and Paragraphs - Example



The <!DOCTYPE> Declaration

- HTML documents must start with a document type definition (DTD)
  - It tells web browsers what type is the served code
  - Possible versions: HTML 4.01, XHTML 1.0 (Transitional or Strict), XHTML 1.1, HTML 5
- Example:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

  - See http://w3.org/QA/2002/04/valid-dtd-list.html for a list of possible doctypes

The<script> tag Example

- The <style> element embeds formatting information (CSS styles) into an HTML page

style-example.html

```
<html>
  <head>
    <style type="text/css">
      p { font-size: 12pt; line-height: 12pt; }
      p:first-letter { font-size: 200%; }
      span { text-transform: uppercase; }
    </style>
  </head>
  <body>
    <p>Styles demo.<br />
      <span>Test uppercase</span>.
    </p>
  </body>
</html>
```

<head> Section: <style>

- The <style> element embeds formatting information (CSS styles) into an HTML page

style-example.html

```
<html>
  <head>
    <style type="text/css">
      p { font-size: 12pt; line-height: 12pt; }
      p:first-letter { font-size: 200%; }
      span { text-transform: uppercase; }
    </style>
  </head>
  <body>
    <p>Styles demo.<br />
       <span>Test uppercase</span>.
    </p>
  </body>
</html>
```

Colour codes

Simple Internet HTML Color Table

with HTML Codes - 81 color set.

	Col. 1	Col. 2	Col. 3	Col. 4	Col. 5	Col. 6	Col. 7	Col. 8	Col. 9
Row 1	FFFFFF	000000	333333	666666	999999	CCCCCC	CCCC99	9999CC	666699
Row 2	660000	663300	996633	003300	003333	003399	000066	330066	660066
Row 3	990000	993300	CC9900	006600	336666	0033FF	000099	660099	990066
Row 4	CC0000	CC3300	FFCC00	009900	006666	0066FF	0000CC	663399	CC0099
Row 5	FF0000	FF3300	FFFF00	00CC00	009999	0099FF	0000FF	9900CC	FF0099
Row 6	CC3333	FF6600	FFFF33	00FF00	00CCCC	00CCFF	3366FF	9933FF	FF00FF
Row 7	FF6666	FF6633	FFFF66	66FF66	66CCCC	00FFFF	3399FF	9966FF	FF66FF
Row 8	FF9999	FF9966	FFFF99	99FF99	66FFCC	99FFFF	66CCFF	9999FF	FF99FF
Row 9	FFCCCC	FFCC99	FFFFCC	CCFFCC	99FFCC	CCFFFF	99CCFF	CCCCFF	FFCCFF

Comments: <!-- -->Tag

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

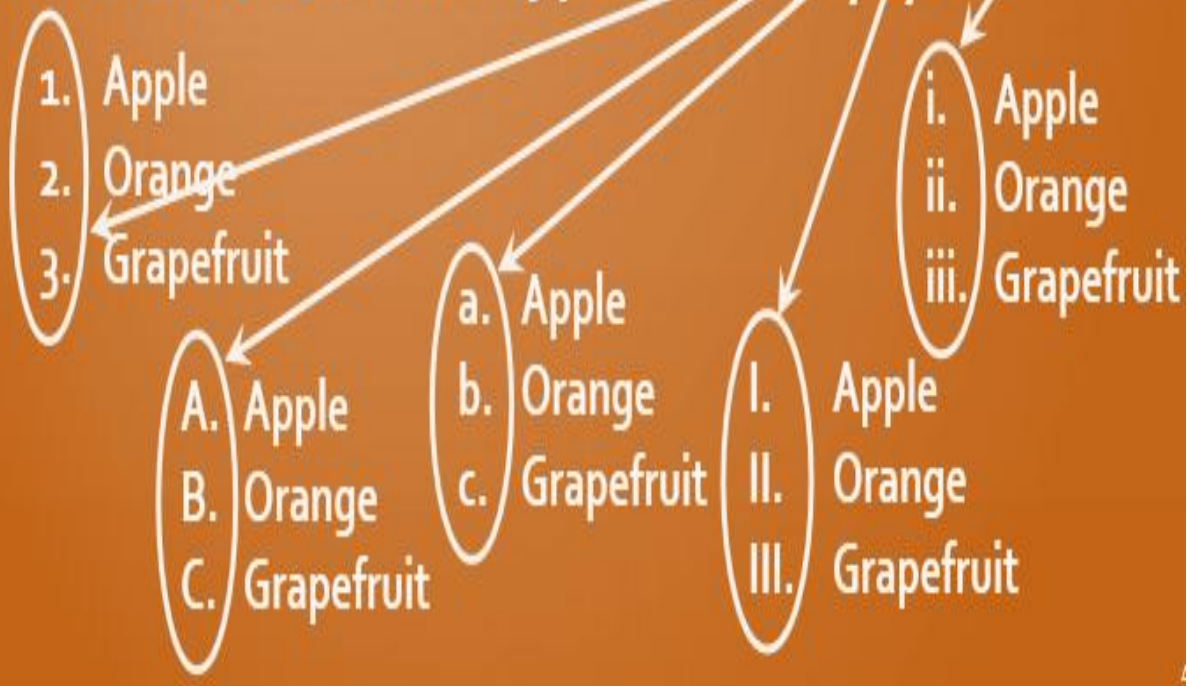
```
<!-- Telerik Logo (a JPG file) -->
<img src="logo.jpg" alt="Telerik Logo">
<!-- Hyperlink to the web site -->
<a href="http://telerik.com/">Telerik</a>
<!-- Show the news table -->
<table class="newstable">
...
```

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

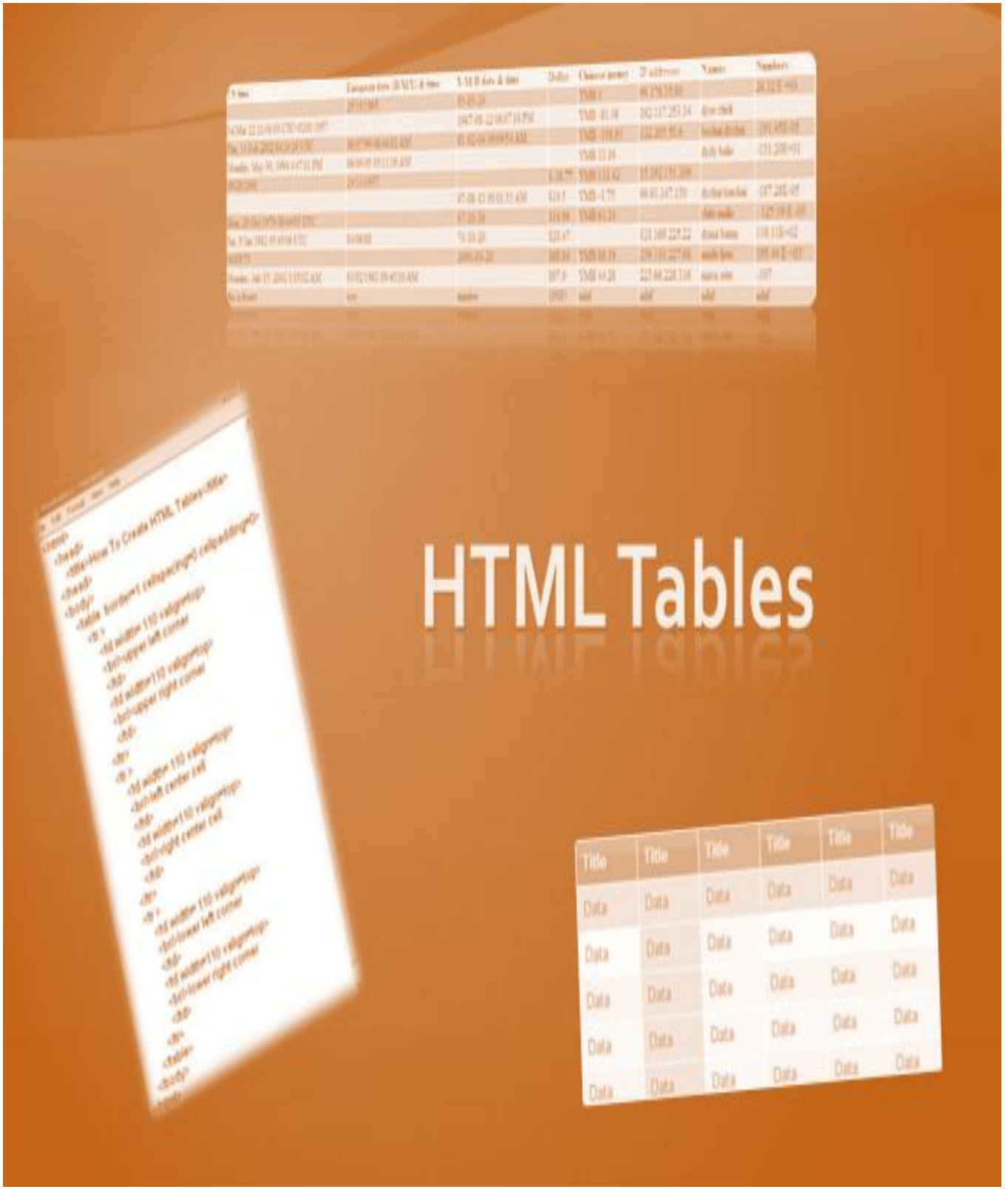
telerik

Unordered Lists: <ul> Tag

- Create an Unordered List using :

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

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


HTML Tables
Title
Data

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

HTML Tables

- Start and end of a table

```
<table> ... </table>
```

- Start and end of a row

```
<tr> ... </tr>
```

- Start and end of a cell in a row

```
<td> ... </td>
```

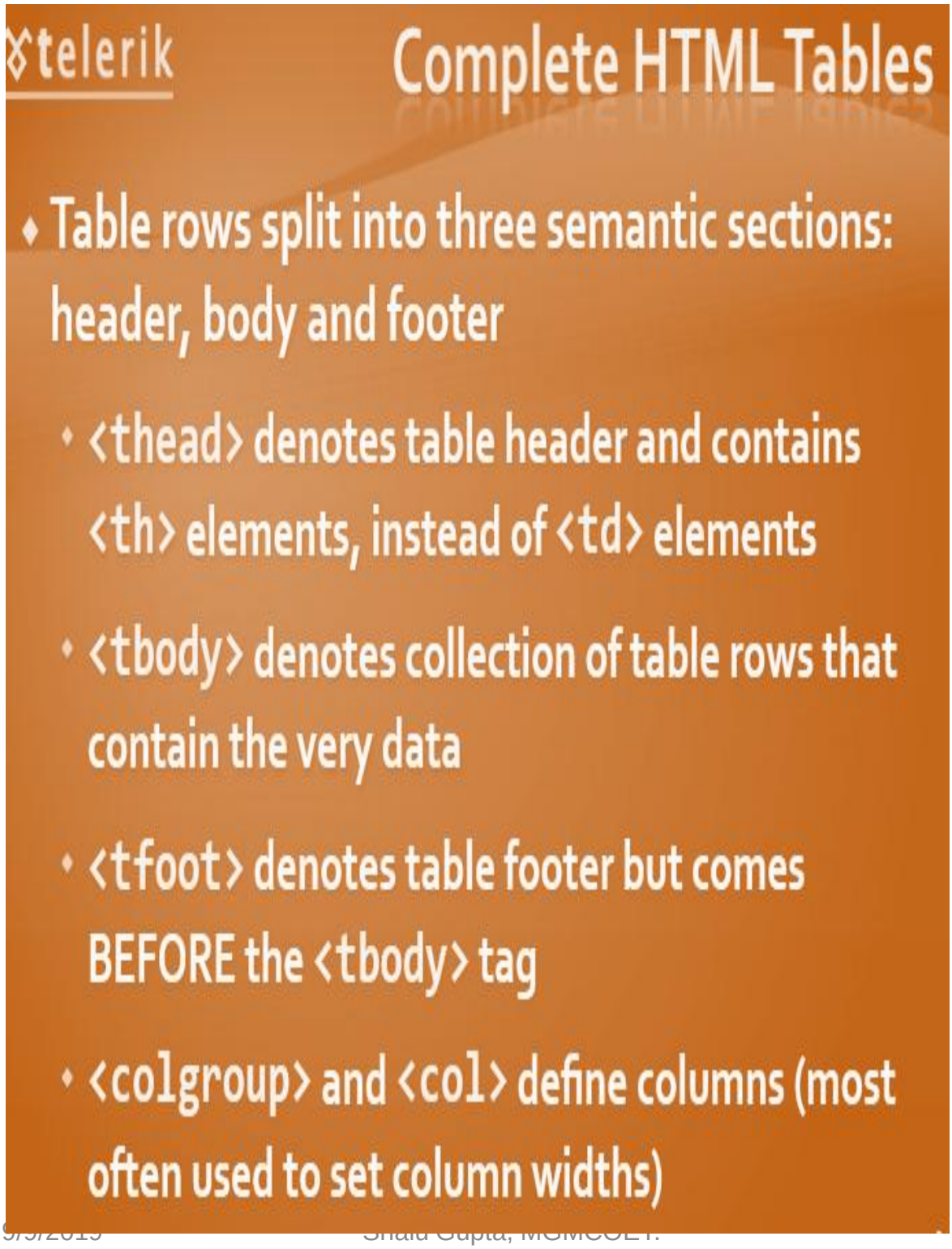
Simple HTML Tables - Example

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

Simple HTML Tables – Example2

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

Complete HTML Tables



Complete HTML Table: Example

```
<table>
<colgroup>
  <col style="width:100px" /><col />
</colgroup>
<thead>
  <tr><th>Column 1</th><th>Column 2</th></tr>
</thead>
<tfoot>
  <tr><td>Footer 1</td><td>Footer 2</td></tr>
</tfoot>
<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-full.html

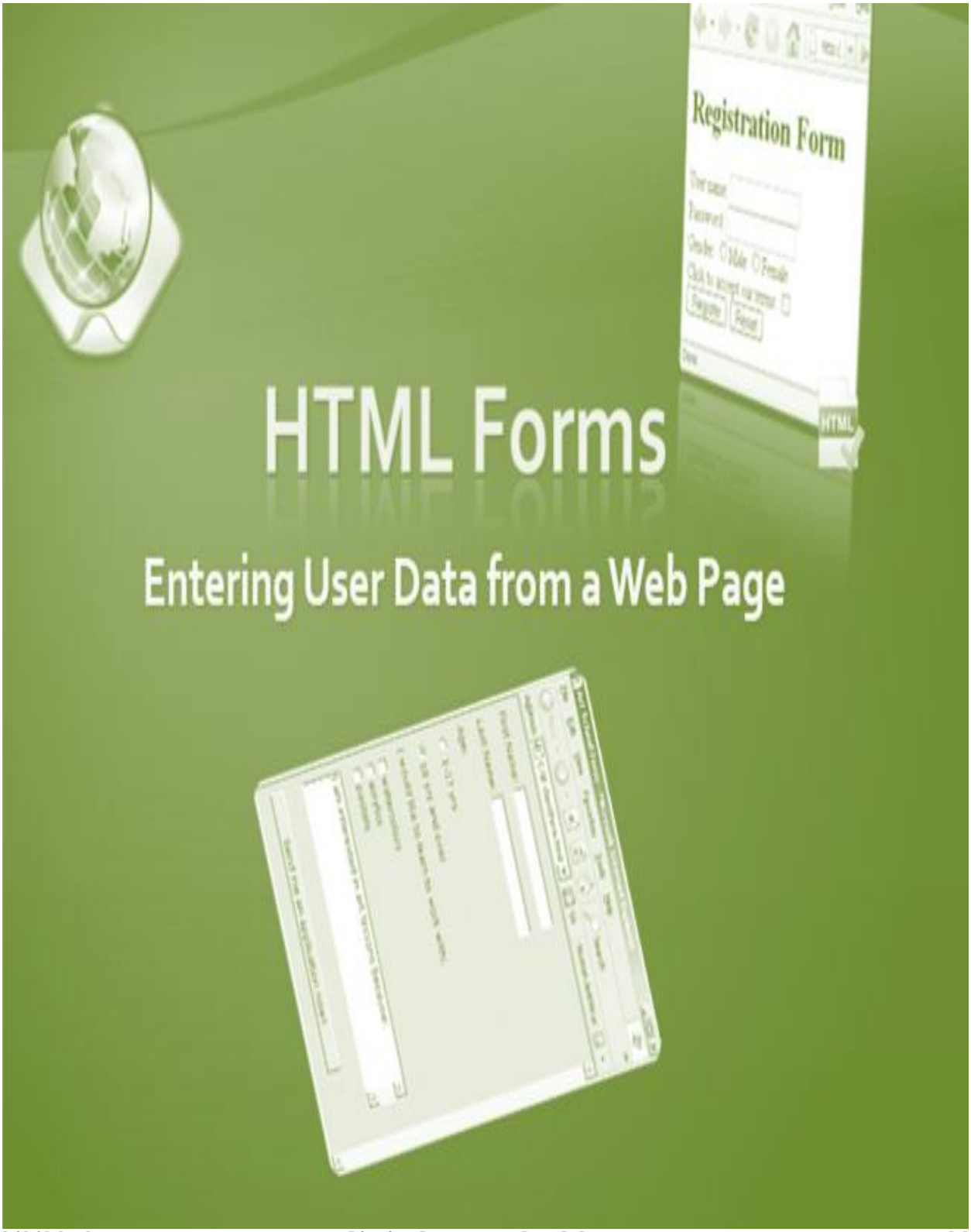
```
<table>
<colgroup>
  <col style="width
</colgroup>
<thead>
  <tr><th>Column 1<
</thead>
<tfoot>
  <tr><td>Footer 1</td><td>Footer 2</td></tr>
</tfoot>
<tbody>
  <tr><td>Cell 1.1</td><td>Cell 1.2</td></tr>
  <tr><td>Cell 2.1</td><td>Cell 2.2</td></tr>
</tbody>
</table>
```

Nested Tables

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

nested-tables.html

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


Registration Form
HTML Forms
Entering User Data from a Web Page

HTML Forms

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

```
<form></form>
```

- Example:

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

Form Fields

- Single-line text input fields:

```
<input type="text" name="FirstName" value="This
is a text field" />
```

- Multi-line textarea fields:

```
<textarea name="Comments">This is a multi-line
text field</textarea>
```

- Hidden fields contain data not shown to the user:

```
<input type="hidden" name="Account" value="This
is a hidden text field" />
```

  - Often used by JavaScript code

Fieldsets

- Fieldsets are used to enclose a group of related form fields:

```
<form method="post" action="form.aspx">
   <fieldset>
      <legend>Client Details</legend>
      <input type="text" id="Name" />
      <input type="text" id="Phone" />
   </fieldset>
   <fieldset>
      <legend>Order Details</legend>
      <input type="text" id="Quantity" />
      <textarea cols="40" rows="10"
         id="Remarks"></textarea>
   </fieldset>
</form>
```

- The <legend> is the fieldset's title.

Form Input Controls

- Checkboxes:

```
<input type="checkbox" name="fruit"
value="apple" />
```

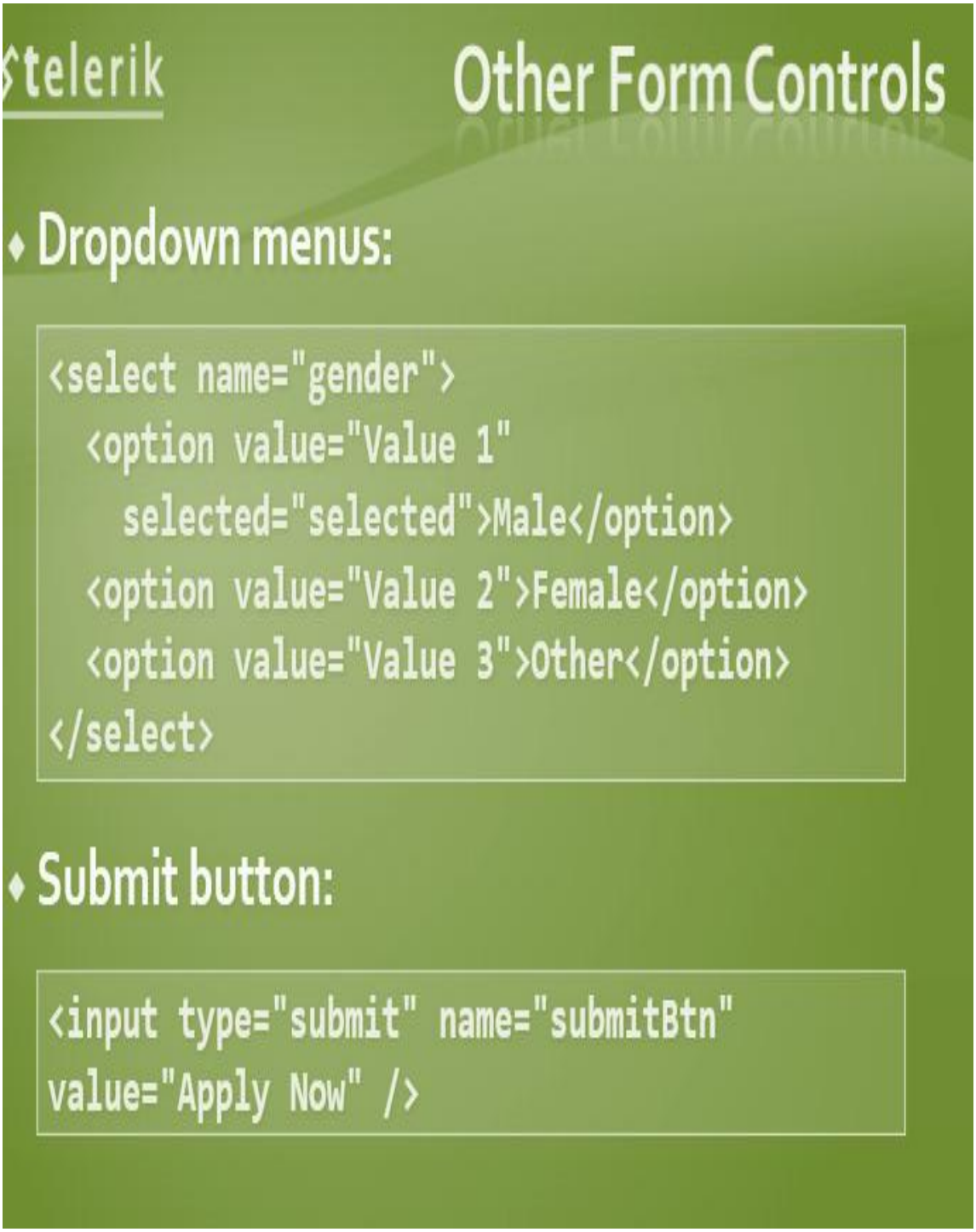
- Radio buttons:

```
<input type="radio" name="title" value="Mr." />
```

- Radio buttons can be grouped, allowing only one to be selected from a group:

```
<input type="radio" name="city" value="Lom" />
<input type="radio" name="city" value="Ruse" />
```

Other Form Controls



Other Form Controls

- Reset button – brings the form to its initial state

```
<input type="reset" name="resetBtn"
value="Reset the form" />
```

- Image button – acts like submit but image is displayed and click coordinates are sent

```
<input type="image" src="submit.gif"
name="submitBtn" alt="Submit" />
```

- Ordinary button – used for Javascript, no default action

```
<input type="button" value="click me" />
```

Other Form Controls

- Password input – a text field which masks the entered text with * signs

```
<input type="password" name="pass" />
```

- Multiple select field – displays the list of items in multiple lines, instead of one

```
<select name="products" multiple="multiple">
  <option value="Value 1"
    selected="selected">keyboard</option>
  <option value="Value 2">mouse</option>
  <option value="Value 3">speakers</option>
</select>
```

Labels

- Form labels are used to associate an explanatory text to a form field using the field's ID.

```
<label for="fn">First Name</label>
<input type="text" id="fn" />
```

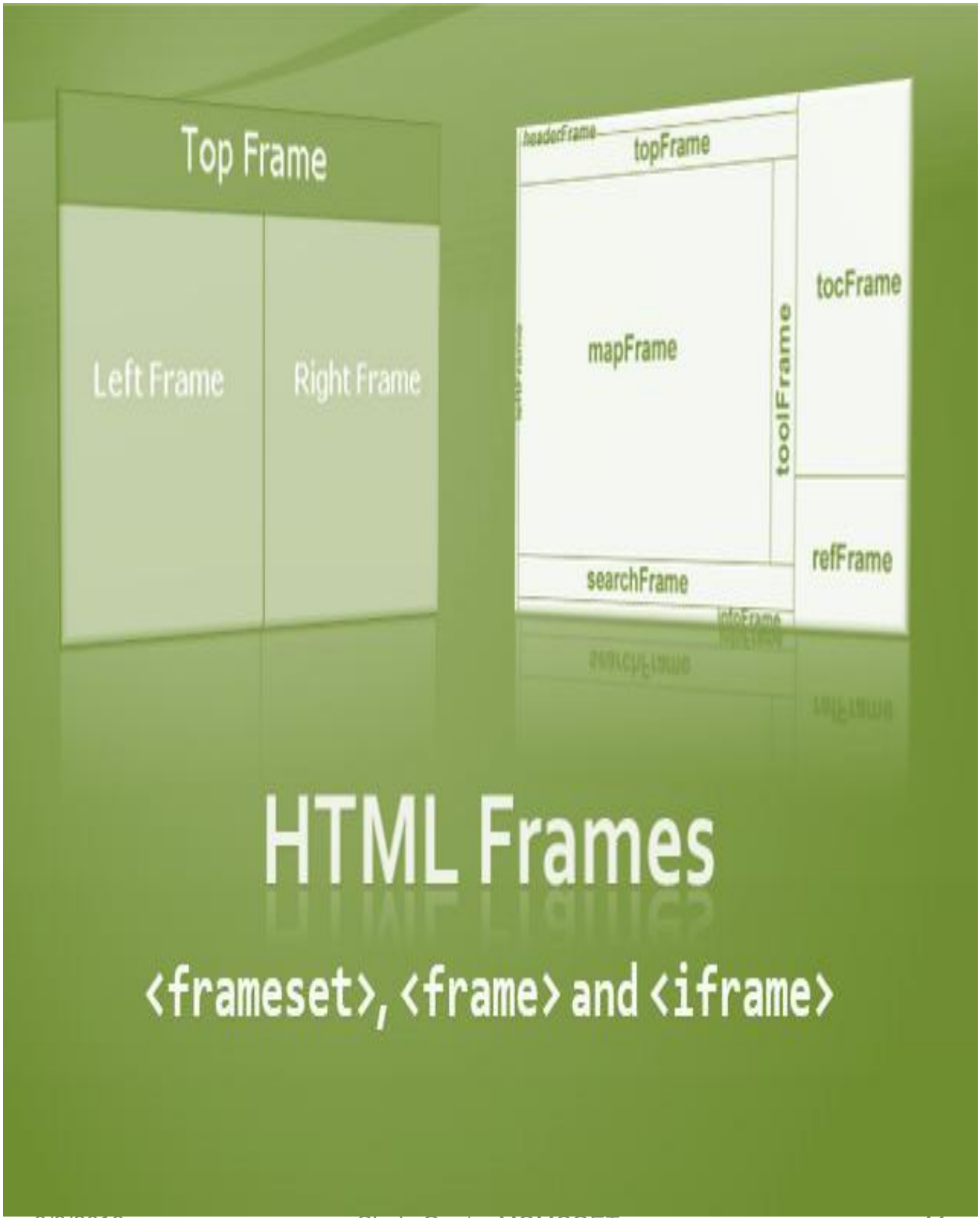
- Clicking on a label focuses its associated field (checkboxes are toggled, radio buttons are checked)
- Labels are both a usability and accessibility feature and are required in order to pass accessibility validation.

HTML Form Examples

form.html

```
<form method="post" action="apply-now.php">
  <input name="subject" type="hidden" value="Class" />
  <fieldset><legend>Academic information</legend>
    <label for="degree">Degree</label>
    <select name="degree" id="degree">
      <option value="BA">Bachelor of Art</option>
      <option value="BS">Bachelor of Science</option>
      <option value="MBA" selected="selected">Master of
        Business Administration</option>
    </select>
    <br />
    <label for="studentid">Student ID</label>
    <input type="password" name="studentid" />
  </fieldset>
  <fieldset><legend>Personal Details</legend>
    <label for="fname">First Name</label>
    <input type="text" name="fname" id="fname" />
    <br />
    <label for="lname">Last Name</label>
    <input type="text" name="lname" id="lname" />
```

Form Example


Top Frame
Left Frame
Right Frame
topFrame
mapFrame
toolFrame
tocFrame
refFrame
searchFrame
HTML Frames
<frameset>, <frame> and <iframe>

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

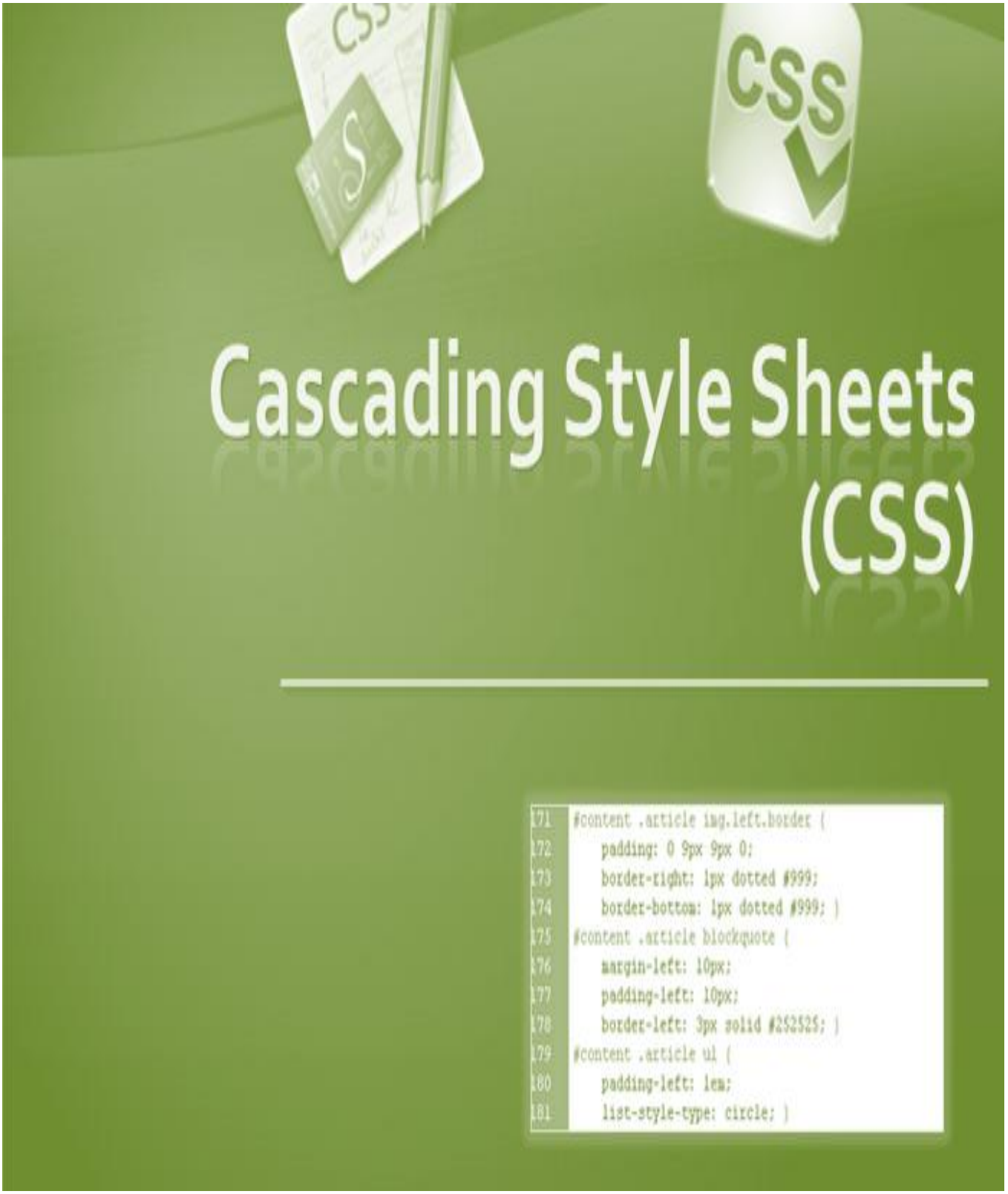
```
<html>

  <head><title>Frames Example</title></head>

  <frameset cols="180px,*,150px">
    <frame src="left.html" />
    <frame src="middle.html" />
    <frame src="right.html" />
  </frameset>

</html>
```

- Note the target attribute applied to the <a> elements in the left frame.


CSS
Cascading Style Sheets
(CSS)
#content .article img.left.border {
padding: 0 9px 9px 0;
border-right: 1px dotted #999;
border-bottom: 1px dotted #999; }
#content .article blockquote {
margin-left: 10px;
padding-left: 10px;
border-left: 3px solid #252525; }
#content .article ul {
padding-left: 1em;
list-style-type: circle; }

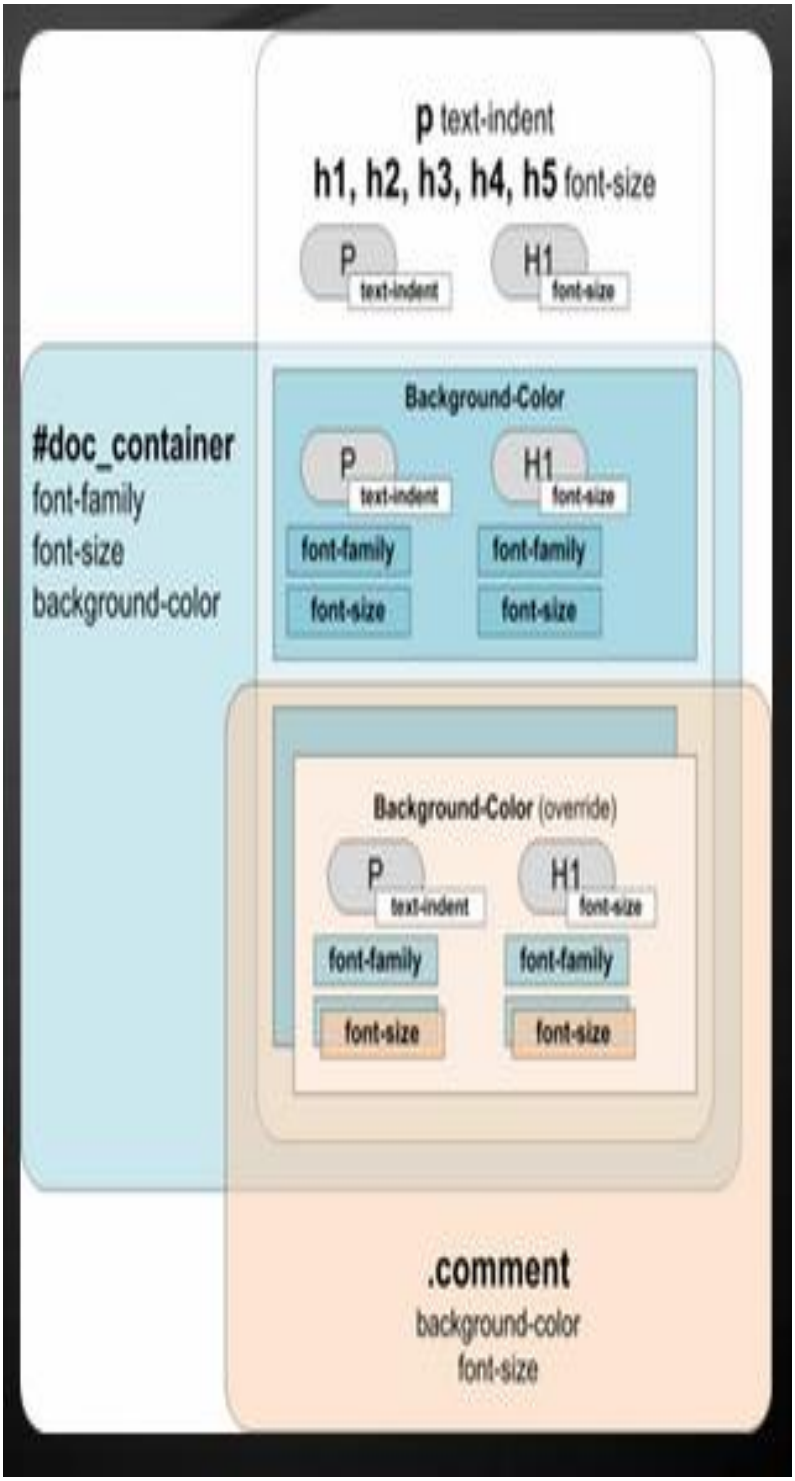
CSS Introduction

- Cascading Style Sheets (CSS)
  - Used to describe the presentation of documents
  - Define sizes, spacing, fonts, colors, layout, etc.
  - Improve content accessibility
  - Improve flexibility
- Designed to separate presentation from content
- Due to CSS, all HTML presentation tags and attributes are deprecated, e.g. font, center, etc.

CSS Introduction

- CSS can be applied to any XML document
  - Not just to HTML / XHTML
- CSS can specify different styles for different media
  - On-screen
  - In print
  - Handheld, projection, etc.
  - ... even by voice or Braille-based reader

Why “Cascading”

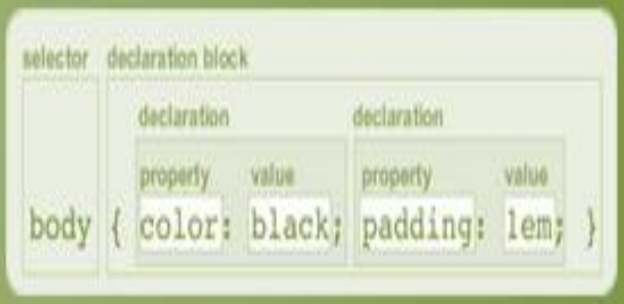


Why “Cascading”

- Some CSS styles are inherited and some not
  - Text-related and list-related properties are inherited - color, font-size, font-family, line-height, text-align, list-style, etc
  - Box-related and positioning styles are not inherited - width, height, border, margin, padding, position, float, etc
  - <a> elements do not inherit color and text-decoration

Style Sheets Syntax

- Stylesheets consist of rules, selectors, declarations, properties and values



http://css.maxdesign.com.au/

- Selectors are separated by commas
- Declarations are separated by semicolons
- Properties and values are separated by colons

```
h1,h2,h3 { color: green; font-weight: bold; }
```

Linking HTML and CSS

- HTML (content) and CSS (presentation) can be linked in three ways:
  - Inline: the CSS rules in the `style` attribute
    - No selectors are needed
  - Embedded: in the <head> in a <style> tag
  - External: CSS rules in separate file (best)
    - Usually a file with `.css` extension
    - Linked via <link rel="stylesheet" href=…> tag or `@import` directive in embedded CSS block

Linking HTML and CSS

- Using external files is highly recommended
  - Simplifies the HTML document
  - Improves page load speed as the CSS file is cached

Inline style

inline-styles.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml11/
DTD/xhtml11-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Inline Styles</title>
</head>
<body>
  <p>Here is some text</p>
<!--Separate multiple styles with a semicolon-->
  <p style="font-size: 20pt">Here is some
    more text</p>
  <p style="font-size: 20pt;color:
    #0000FF" >Even more text</p>
</body>
</html>
```

inline-styles.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/
DTD/xhtml1-tra
<html xmlns="h
<head>
  <title>Inlin
</head>
<body>
  <p>Here is s
<!--Separate m
  <p style="fo
    more text<
  <p style="fo
    #0000FF" >
</body>
</html>
```

Embedded Style

- Embedded in the HTML in the <style> tag:

```
<style type="text/css">
```

  - The <style> tag is placed in the <head> section of the document
  - type attribute specifies the MIME type
    - MIME describes the format of the content
    - Other MIME types include text/html, image/gif, text/javascript ...
- Used for document-specific styles

Embedded - Example

embedded-stylesheets.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <title>Style Sheets</title>
  <style type="text/css">
    em {background-color:#8000FF; color:white}
    h1 {font-family:Arial, sans-serif}
    p  {font-size:18pt}
    .blue {color:blue}
  </style>
<head>
```

External CSS Styles

- External linking
  - Separate pages can all use a shared style sheet
  - Only modify a single file to change the styles across your entire Web site (see http://www.csszengarden.com/)
- link tag (with a rel attribute)
  - Specifies a relationship between current document and another document

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

  - link elements should be in the <head>

External CSS Styles - Example

styles.css

```
/* CSS Document */

a       { text-decoration: none }

a:hover { text-decoration: underline;
          color: red;
          background-color: #CCFFCC }

li em   { color: red;
          font-weight: bold }

ul      { margin-left: 2cm }

ul ul   { text-decoration: underline;
          margin-left: .5cm }
```

Text – related CSS Properties

- color – specifies the color of the text
- font-size – size of font: xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger or numeric value
- font-family – comma separated font names
  - Example: verdana, sans-serif, etc.
  - The browser loads the first one that is available
  - There should always be at least one generic font
- font-weight can be normal, bold, bolder, lighter or a number in range [100 … 900]

Backgrounds

- background-image
  - URL of image to be used as background, e.g.:

```
background-image:url("back.gif");
```

- background-color
  - Using color and image and the same time
- background-repeat
  - repeat-x, repeat-y, repeat, no-repeat
- background-attachment
  - fixed/scroll

Benefits of using CSS

- More powerful formatting than using presentation tags
- Your pages load faster, because browsers cache the `.css` files
- Increased accessibility, because rules can be defined according given media
- Pages are easier to maintain and update