Browser Controlling Mechanism

- HTML
  - Hyper Text Markup Language
- ML in HTML gives us
  - a way to **"mark up"** our text with **INSTRUCTIONS (tags)** that tell the browser how your text is structured.
- There is also the **Hyper Text** aspect of HTML
  - which can offer us **break free single page environment** with linkages to other pages.

Browser Decisions to construct a Web Page

- HTML tells our browser about the **structural placement of the various components** of our document
  - Which component will stand as heading
  - How to construct the paragraphs
  - what text needs emphasis
  - And what not !!!
- However browser has its built in interpretation and default rules to display each of these elements.
- Thus HTML and Browser need to collaborate with each other.

Hyper Text Mark-Up Language

- HTML Page is written in the form of HTML elements
  - consisting of *tags* enclosed in angular braces (like <html>), within the web page content.
  - And occasionally accompanied by attributes with specific values
- HTML tags most commonly come in pairs
  - like <h1> and </h1>,
- Although some tags, known as ***empty elements***, are unpaired, for example <img>.

Hyper Text Mark-Up Language

- The first tag in a pair is the *start tag*, the second tag is the *end tag* (they are also called *opening tags* and *closing tags*).
- In between these tags web designers can add text, tags, comments and other types of text-based content.
- The purpose of a web browser is to read HTML document and compose them into visible / audible web pages.
- The browser does not display the HTML tags,
  - **Rather it displays the effects of tags**.

Hyper Text Mark-Up Language

- HTML elements form the building blocks of all websites.
- HTML allows images and objects to be embedded and can be used to create interactive forms.
- It provides a means to create documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.
- It can embed scripts in languages such as JavaScript, jQuery, AJAX which provides dynamic behavior for HTML web pages.

Hyper Text Mark-Up Language

- Web browsers can also refer to Cascading Style Sheets (CSS) to define the appearance and layout of text and other material.
- The W3C, maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicitly presentational HTML markup
- **World Wide Web Consortium** - main international standards organization for the World Wide Web

Hello world – The HTML Code


```
<html>
    <head>
        <title>Hello HTML</title>
    </head>
        <body>
            <p>
                Hello World!
            </p>
        </body>
</html>
```

- The text between <html> and </html> describes the web page
- The markup text '<title>Hello HTML</title>' defines the browser page title.
- Typically, the text between <body> and </body> is the visible page content.

The <html> tag tells the browser your document is actually HTML. While some browsers will forgive you if you omit it,

Other HTML Tags

- <FONT Face = "****" Style = "****" Size = 10 >HA HA HA </FONT>
  - Determines the font type for text with other attributes such as color & size
- <H1>
  - Highest level of Headings among H1.....H6
- <P>
  - Breaks the text into new paragraph or new line.
- <B> <I> <U>

  Provides Bold, Italic, Underline effects for text

Lists – Bulleted & Numbered


```
<OL>
    <LI> PLT </LI>
    <LI> DBMS</LI>
    <LI> OOP with C++</LI>
    <LI> Web Concepts</LI>
    <LI> SAP Modules </LI>
</OL>
```

Lists – Bulleted & Numbered

The Numbered List s with --- Normal numbering Character Numbering Roman Numbering

```
<OL>
    <LI> PLT </LI>
    <LI> DBMS</LI>
    <LI> OOP WithC++</LI>
    <LI> Web Concepts</LI>
</OL>
```

```
<OL Type = "A">
    <LI> PLT </LI>
    <LI> DBMS</LI>
    <LI> OOP WithC++</LI>
    <LI> Web Concepts</LI>
</OL>
```

```
<OL Type = "I">
    <LI> PLT </LI>
    <LI> DBMS</LI>
    <LI> OOP WithC++</LI>
    <LI> Web Concepts</LI>
</OL>
```

The Bulleted List s with --- Normal Bullet Disc Bullet Circular Bullet

```
<UL>
    <LI> PLT </LI>
    <LI> DBMS</LI>
    <LI> OOP WithC++</LI>
    <LI> Web Concepts</LI>
</UL>
```

```
<UL Type = "Circle">
<LI> PLT </LI>
    <LI> DBMS</LI>
    <LI> OOP WithC++</LI>
    <LI> Web Concepts</LI>
</UL>
```

```
<UL Type = "Square">
    <LI> PLT </LI>
    <LI> DBMS</LI>
    <LI> OOP WithC++</LI>
    <LI> Web Concepts</LI>
</OL>
```

Linking Web Pages

- Hyperlinks are used to link another documents
- HTML uses the <A> Tag to create link to other document – Anchor Tag
- The anchor can point to any resource on Web
  - HTML Page, Image, Sound or other animation file.
- <A href = "the URL"> Text to be displayed </A>
  - href Attribute – To provide Link to specific URL

Handling Images in Web Pages

- In HTML, images are defined with the <img> tag – An Empty Tag
- To display an image on a page, you need to use the src attribute. Src stands for "source". The value of the src attribute is the URL of the image you want to display.
- The URL points to the location where the image is stored.
- The required alt attribute specifies an alternate text for an image, if the image cannot be displayed.
- <img src = "E:\LinkFiles\Desktop.GIF" Alt = "OOPs. Unable to Display my Desktop">

Handling Images in Web Pages

- **Set Height and Width of an Image**
- The height and width attributes are used to specify the height and width of an image.
- The attribute values are specified in pixels by default:
  - <img src="pulpit.jpg" alt="Pulpit rock" width="304" height="228”>
- When a web page is loaded, it is the browser, at that moment, that actually gets the image from a web server and inserts it into the page. Therefore, make sure that the images actually stay in the same spot in relation to the web page, otherwise your visitors will get a broken link icon. The broken link icon is shown if the browser cannot find or support the image.

Handling Tables in Web Pages

- HTML will handle tables with the <table> tag.
- A table is divided into rows (with the <tr> tag).
- Each row is divided into data cells (with the <td> tag).
- td stands for "table data," and holds the content of a data cell. <td> tag can contain text, links, images, lists, forms, other tables, etc.
- If you do not specify a border attribute, the table will be displayed without border.
- Column Headings in a table are defined with the <th> tag.
- Column Headings are displayed in BOLD and CENTER ALIGNED.

HTML Table Tags

Tag	Description
<table>	Defines a table
<th>	Defines a table header
<tr>	Defines a table row
<td>	Defines a table cell data

Handling Frames in Web Pages

- Until now, each web page opened takes over the entire browser screen.
- Now we can divide browser screen into two or more regions by using <FRAMESET> Tag. Each unique region is referred as frame.
- Each frame can be loaded with a different document and hence allow multiple HTML documents to be seen concurrently.

Handling Frames in Web Pages


```
<FRAMESET Rows = “50%,50%">
                                        <!– Splits screen into 2 equal ROWS -->
        <FRAMESET Cols = "50%,50%">
                                        <!– Splits 1st Row into 2 equal Columns -->
            <frame src = “LIST.HTML" > <!– Loads List.HTML in Frame1 -- >
            <frame src = “Links.HTML" > <!– Loads List.HTML in Frame2 -- >
        </FRAMESET>
        <FRAMESET Cols = "50%,50%">
                                        <!– Splits 2nd Row into 2 equal Columns -->
            <frame src = “Tables.HTML" > <!– Loads Tables.HTML in Frame3 -- >
            <frame src = “Images.HTML" > <!– Loads Imag.HTML in Frame4 -- >
        </FRAMESET>
</FRAMESET>
```

Handling Forms in Web Pages

- **HTML Form**
- HTML forms are used to accept and pass data to a server
- A form can contain **input elements** like text fields, checkboxes, radio-buttons, submit buttons and more.
- A form can also contain select lists, text area, field set, legend, and label elements.
- An **INPUT** element is used to select different types of user information.

HTML Forms - The Input Elements

- **Text Fields**
- <input type="text" /> defines a one-line input field that a user can enter text into

```
<form>
        First Name : <input type="text" name="Firstname" ><br>
        Last Name: <input type="text" name="Lastname">
</form>
```

First Name :

Last Name :

HTML Forms - The Input Elements

- **Password Field**
- <input type="password" /> defines a password field

- ```
  <form>
        Password: <input type="password" name="pwd" />
  </form>
  ```
  ```

HTML Forms - The Input Elements

- **Checkboxes**
- <input type="checkbox" /> defines a checkbox.
- Checkboxes let a user select **ZERO** or **MORE** options of a limited number of choices.

```
<form>
    <input type="checkbox" name="Vehicle" value="Bike" /> I
        have a bike<br />
    <input type="checkbox" name="Vehicle" value="Car" /> I
        have a car
</form>
```

HTML Forms - The Input Elements

- **Radio Buttons**
- <input type="radio" /> defines a radio button.
- Radio buttons let a user select **ONLY ONE** of a limited number of choices

```
<form>
    <input type="radio" name="Gender" value="male" /> Male<br />
    <input type="radio" name=" Gender " value="female" /> Female
</form>
```

HTML Forms - The Input Elements

- **Submit Button**
- <input type="submit" /> defines a submit button.
- A submit button is used to send form data to a server. The data is sent to the page specified in the form's action attribute. The file defined in the action attribute usually does something with the received input

```
<form name="input" action="html_form_action.asp" method="get">
        Username: <input type="text" name="USER" />
        <input type="submit" value="Submit" />
</form>
```

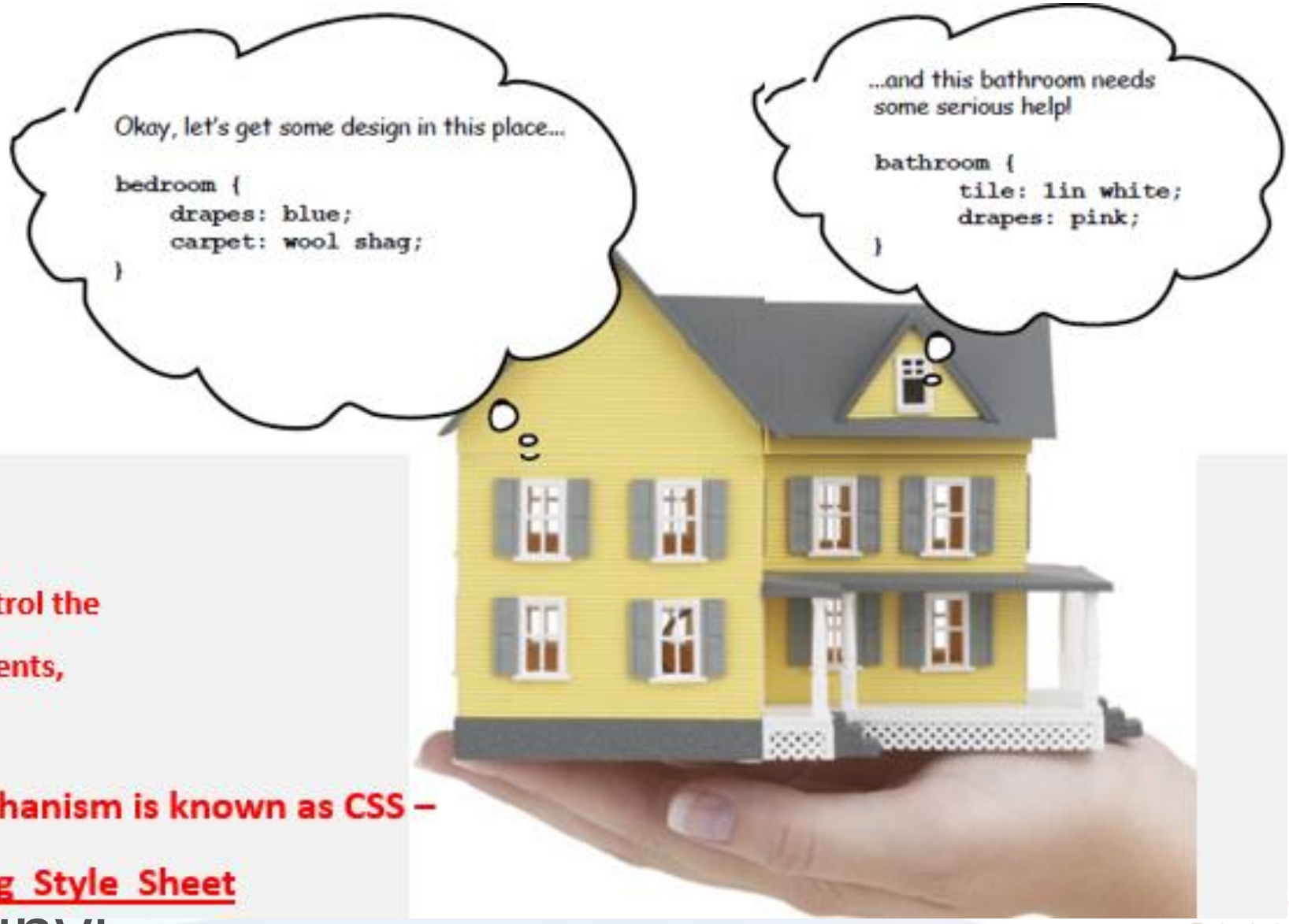
Generalized Look & Feel for Components

In the Browser Context, HTML doesn't have ROOMS.

HTML does have elements .

These elements are the locations where we will put up STYLING.

Here we are going to control the presentation of our elements,



This Mechanism is known as CSS –

Cascading Style Sheet

Using CSS with HTML

- Want to paint the walls of our **<p>** elements to red?
- No probs ..... Here is the way.....

We will select the element we want to STYLE

Then we will specify the property which we want to STYLE ... Here it is Background Color.

And we are going to set background color to RED

```
p {
    background-color: red;
}
```

Place all the styles for <p> element in between { } Braces

There is a colon in between the property and its value

At the end, put the semi colon

Wanna add more style?

- We can add as many properties and values as we want in each CSS rule.

All we have to do is to add another property and value

```
p {
    background-color: red;
    border: 1px solid gray;
}
```

The <p> Element will have a border .....

That is 1 pixel thick, solid and gray

Styling HTML with CSS

- CSS can be added to HTML in the following ways
  - **Inline** - using the style **attribute** in existing HTML elements
  - **Internal** - using separate <style> **element** in the <head> section
  - **External** - using an external CSS **file**
- The obviously preferred way to add CSS to HTML, is to put CSS syntax in separate CSS files
- We can use a style sheet as a **Template** or **Master Page** so that multiple pages can use the same style sheet to implement same stylistic attributes.

Using the Style Sheets

- **Inline Styles**
- An inline style can be used if a unique style is to be applied to one single element.
- To use inline styles, use the style attribute in the relevant tag. The style attribute can contain any CSS property.
- The example below shows how to change the text color and the left margin of a paragraph

```
<p style="color:blue;margin-left:20px;">This is a paragraph.</p>
```

HTML Style - Background Color - Font, Color and Size


```
<html>
    <body  Style = " background-color : yellow ; " >
        <h2 Style = " background-color : red ; " >      This is a heading  </h2>
        <p  Style = " background-color : green ; " >This is a paragraph  </p>
    </body
<html>

<html>
    <body  Style = " background-color : yellow ; " >
        <h2 Style = " font-family : verdana ; " >       This is a heading  </h2>
        <p  Style = "font-family : arial ; color : red ; font-size : 20px;">
                                                        This is a paragraph  </p>
    </body
<html>
```

HTML Style Example - Text Alignment


```
<html>
    <body>
        <h1 style="text-align:center;">Center-aligned heading</h1>
            <p>This is a paragraph.</p>
    </body>

</html>
```

HTML Style Example - Text Alignment

- An internal style sheet can be used if one single document has a unique style. Internal styles are defined in the <head> section of an HTML page, by using the <style> tag, like this

```
<head>
    <style type="text / css">
        body { background-color : yellow ; }
        p { color : blue ; }
    </style>
</head>
```

External Style Sheet

- An external style sheet is ideal when the style is applied to many pages.
- With an external style sheet, we can change the look of an entire Web site by changing one file.
- Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the <head> section

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