User Interface

HTML

HTML Introduction :

- HTML is the standard markup language for creating Web pages.
- HTML stands for Hyper Text Markup Language
- HTML describes the structure of a Web page
- HTML elements tell the browser how to display the content
- HTML elements are represented by tags
- Latest Version(HTML5)

HTML Editors :

1. Notepad(Windows)
2. Eclipse
3. Text Editor/GEdit(Ubuntu)
4. Sublime Text Editor

How to Save HTML File?

We can create the html file in any of the mentioned editors(previous slide).

To save an html file, we need to use filename.html extension. Or .htm

Tags

Open Tag--->
Example : <html>

Close Tag :---->Closing tag is having forward slash before name of tag.
Example :</html>

Open tag and is the root tag for html file

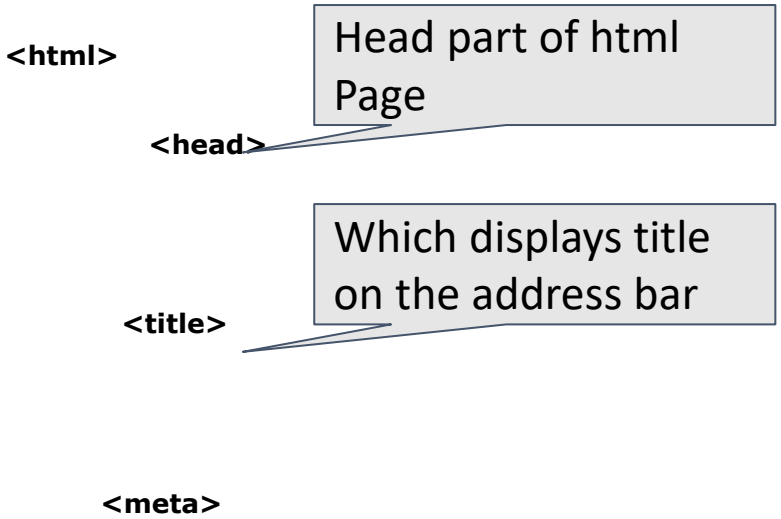
```
<html>
    <head>
        <title> Welcome Page </title>
    </head>

    <body>
        <h1>Welcome to TalentSprint</h1>
    </body>

</html>
```

Close tag of the root element

Basic Structure of HTML Document :



```
        <script>



 </HEAD>

 <BODY>
the remaining HTML elements are contained within these tags.
 </BODY>
</HTML>
```

Basic Body Tags :

Header Tags

Heading tags are designated with the letter **h**. Each **h** tag also has a number after the **h**. They range from **<h1> to <h6> .**

->The font size of text decreases from **<h1>** tag to **<h6>** tag .

```
<h1></h1>
<h2></h2>
<h3></h3>
<h4></h4>
<h5></h5>
<h6></h6>
```

Example html file :

```
<!DOCTYPE html>
<html>
<head>
<title>My First Web Page</title>
</head>
<body>
<h1>My First Heading</h1>
<p>My first paragraph.</p>
</body>
</html>
```

Paragraph tag :

-> you can use paragraphs to separate your content into blocks. You can create a paragraph by surrounding your content with the **<p>** tags.

Line Break :

-> Now, if you want to separate your content onto multiple lines, but you don’t want that space that comes with a paragraph, you can use a line break, or a **
** tag.

Html Formatting Elements :

Tags to format your text . Formatting elements were designed to display special types of text:

- **<b> - Bold text**
- ** - Important text**
- **<i> - Italic text**
- ** - Emphasized text**
- **<mark> - Marked text**
- **<small> - Small text**
- **<del> - Deleted text**
- **<ins> - Inserted text**

- **<sub> - Subscript text**
- **<sup> - Superscript text**
- **<u>-Underline text**

Horizontal Rule :

-> The horizontal rule tag will create a horizontal line on your web page that goes all the way across.

Syntax : **<hr>**

**Note : There are some unpaired tags in html like
,<img>,<hr>.(no closing tags)**

Anchor Link :

-> The <a> tag defines a hyperlink, which is used to link from one page to another.

-> The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

Visit this Link

Inserting Image Into Webpage :

To put an image on your web page, you can use the image tag as **<img>**

Syntax :

<img src="img1.jpg" alt=”Not Found”>

src attribute describes the location of image

alt attribute describes alternate text will be displayed if image location is not found

Types of Lists in HTML :

1. Ordered List
   - To create numbered list with alphabets(uppercase,lowercase),numerics,Roman numbers(uppercase,lowercase)
1. Unordered List
   - To create bulleted List with(square,circle,disc type bullets)
1. Description List
   - displays elements in definition form like in dictionary
   - **The 3 HTML description list tags are given below:**
     - **<dl> tag defines the description list.**
     - **<dt> tag defines data term.**
     - **<dd> tag defines data definition (description).**

1. Ordered lists

Ex:

```
<ol type="1">
  <li>A</li>
  <li>B</li>
  <li>C</li>
</ol>
```

Note :In the type attribute we can assign(a,A,i,I)

2.Unordered lists

Example :

```
<ul>
  <li>A</li>
  <li>B</li>
  <li>C</li>
</ul>
```

3.Description List

Example:

```
<dl>
<dt>Example :</dt>
<dd>Hello welcome</dd>
</dl>
```

Note : `<dt>` tag defines the term (name), and the `<dd>` tag describes each term

Table in Html :

We first need a **`<table>`** tag. Everything else in the table will be inside this tag.Inside the table we'll have table rows, table cells(table data), and table headers for the column headers.

<th> - table heading
<td> - table data
<tr> - table row

Example :

```
<table border="1 px" cellspacing="0" cellpadding="10">
                    <tr >
                            <th>Roll No.</th>
                            <th>Student Name</th>
                            <th>Subject Name</th>
                    </tr>
                    <tr>
```

```
            <td>1</td>
<td>Ram</td>
            <td>Physics</td>
</tr>
<tr>
            <td>2</td>
            <td>Shyam</td>
```

```
            <td>Math</td>
        </tr>
        <tr>
            <td>3</td>
            <td>Murli</td>
            <td>Chemistry</td>
        </tr>
</table>
```

Question :

Colspan attribute : To merge two or more columns into one column it is used.

Rowspan attribute : To merge two or more rows into one row it is used.

	Month	Rent	Utilities		Groceries	Eating Out	Entertainment
Fall	June	$1500	$100	$50	$350	$100	$50
	July	$1500	$100	$50	$350	$100	$50
	August	$1500	$100	$50	$350	$100	$50

Textarea tag :

1. The <textarea> tag defines a multi-line text input control.
2. A text area can hold an unlimited number of characters, and the text renders in a fixed-width font (usually Courier).
3. The size of a text area can be specified by the cols and rows attributes, or even better; through CSS' height and width properties.

Example :

```
<textarea name="message" rows="10" cols="30">

The cat was playing in the garden.

</textarea>
```

```
<textarea name="message" style="width:200px; height:600px;">

The cat was playing in the garden.

</textarea>
```

Html Button :

```
<button type="submit" value="Submit">Submit</button>
```

The <form> Element :

The HTML `<form>` element defines a form that is used to collect user input:

Form Elements in Html :

1. `<input name="firstname" type="text">` **(Input element)**
2. `<select name="cars">`

```
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
       <option value="audi">Audi</option>
</select>
```

(For creating dropdown list)

3. <textarea> element

4. <button> element

HTML Input Types : Here are the different input types you can use in HTML:

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`

- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`

Example :

```
<form>
  First name:<br>
  <input type="text" name="firstname"><br>
  Last name:<br>
  <input type="text" name="lastname">
</form>
```

Input attributes :

disabled	Specifies that an input field should be disabled
max	Specifies the maximum value for an input field
maxlength	Specifies the maximum number of character for an input field
min	Specifies the minimum value for an input field
pattern	Specifies a regular expression to check the input value against
readonly	Specifies that an input field is read only (cannot be changed)
required	Specifies that an input field is required (must be filled out)
size	Specifies the width (in characters) of an input field
step	Specifies the legal number intervals for an input field
value	Specifies the default value for an input field

CSS Introduction :

1. CSS stands for Cascading Style Sheets.
2. CSS describes how HTML elements are to be displayed on screen, paper, or in other media.
3. CSS saves a lot of work. It can control the layout of multiple web pages all at once.(Using External Style Sheets).
4. Latest Version(CSS3).

Including CSS in HTML Documents

You can include CSS in an HTML document in three ways:

- **Inline styles** — Using the `style` attribute in the HTML start tag.
- **Internal styles** — Using the **`<style>`** element in the head section of a document.
- **External style sheets** — Using the **`<link>`** element, pointing to an external CSS file.

Note : In external style sheet styles are written using selectors in a file (saved with filename.css)

CSS selectors are used *to select the content you want to style*. Selectors are the part of CSS rule set. CSS selectors select HTML elements according to its id, class, type, attribute etc.

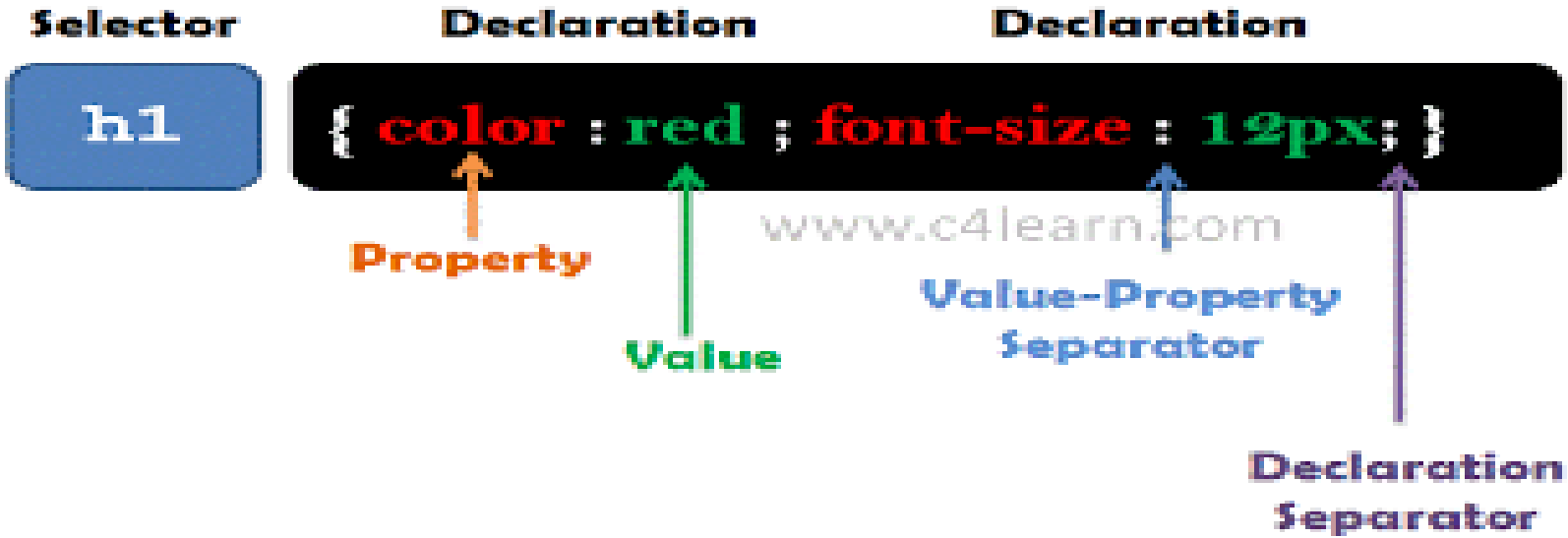
There are several different types of selectors in CSS.

1. Element Selector
2. Id Selector
3. Class Selector
4. Universal Selector
5. Grouping Selector

CSS Selectors

Selector	Example	Example description	CSS
.class	.intro	Selects all elements with class="intro"	1
#id	#firstname	Selects the element with id="firstname"	1
*	*	Selects all elements	2
element	p	Selects all <p> elements	1
element,element	div,p	Selects all elements and all <p> elements	1
element element	div p	Selects all <p> elements inside elements	1
element>element	div>p	Selects all <p> elements where the parent is a element	2
element+element	div+p	Selects all <p> elements that are placed immediately after elements	2

Element Selector Syntax :



2. ID Selector

- ID selector is used to specify a single, unique element.
- ID selector uses the id attribute of the HTML element .
- ID selector defines with “#”.
- Do not start an ID name with a number.
- <style>
- #param{
- test-align : center;
- font-size : 20px;
- }
- </style>
- ……

3. Class Selector

- Class selector is used to specify a style for group of elements.
- Set a particular style for many HTML elements with the same class.
- Class selector defines with "."
- <style>
- .title{
- color : red ;
- font-size : 30px;
- }
- </style>
- ……

Universal Selector

- The universal selector is denoted by an asterisk (*)
- Universal selector represents ever possible element type

```
* { font-weight : bold }
```

- This specifies a value of *bold* for the *font-weight* property of every element in the document

Grouping Selectors

Group **the same selector** with different declarations together on one line.

```
h1 {color: black;}
h1 {font-weight: bold;}
h1 {background: white;}
```

Example of grouping selectors (both are correct):

```
h1 {
color: black;
font-weight: bold;
background: white;
}
```

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      h1, h2, p
       {
      text-align: center;
      color: red;
        }
    </style>
  </head>
    <body>
      <h1> Welcome to wikitechy</h1>
      <h2> Find solutions for All
      technology Errors</h2>
      <p>  Execute the code in
      wikitech.</p>
    </body>
</html>
```

Inline Styles :

Inline styles are used to apply the unique style rules to an element by putting the CSS rules directly into the start tag. It can be attached to an element using the `style` attribute.

The `style` attribute includes a series of CSS property and value pairs. Each `"property: value"` pair is separated by a semicolon (`;`), just as you would write into an internal or external style sheets. But it needs to be all in one line i.e. no line break after the semicolon, as shown here:

Example :

```
<h1 style="color:red; font-size:30px;">This is a heading</h1>

<p style="color:green; font-size:22px;">This is a paragraph.</p>

<div style="color:blue; font-size:14px;">This is some text
content.</div>
```

Embedded Style Sheets

- Embedded or internal style sheets only affect the document they are embedded in.
- Embedded style sheets are defined in the **`<head>`** section of an HTML document using the **`<style>`** element. You can define any number of **`<style>`** elements in a HTML document but they must appear between the **`<head>`** and **`</head>`** tags.

```
<!DOCTYPE html>
<html>
<head>
   <title>My HTML Document</title>
   <style type="text/css">
       body { background-color: Yellow; }
       p { color: red; }
   </style>
</head>
<body>
   <h1>This is a heading</h1>
   <p>This is a paragraph of text.</p>
</body>
</html>
```

External Style Sheets

An external style sheet is ideal when the style is applied to many pages of the website.

An external style sheet holds all the style rules in a separate document that you can link from any HTML file on your site. External style sheets are the most flexible because with an external style sheet, you can change the look of an entire website by changing just one file.

Linking External Style Sheets

Before linking, we need to create a style sheet first. Let's open your favorite code editor and create a new file. Now type the following CSS code inside this file and save it as "style.css".

```
body {
   background: yellow;
   font: 18px Arial, sans-serif;
}
h1 {
   color: orange;
}
```

An external style sheet can be linked to an HTML document using the **<link>** tag. The **<link>** tag goes inside the **<head>** section, as shown in this example:

```
<!DOCTYPE html>

<html lang="en">

<head>

   <title>My HTML Document</title>

   <link rel="stylesheet" href="style.css">

</head>

<body>

   <h1>This is a heading</h1>

  <p>This is a paragraph of text.</p>

</body>

</html>
```

Bootstrap

- Bootstrap is the most popular HTML, CSS and JavaScript framework for developing a responsive and mobile friendly website.
- It is absolutely free to download and use.
- It is a front-end framework used for easier and faster web development.
- It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many others.
- It can also use JavaScript plugins.
- It facilitates you to create responsive designs.

Components

Alerts :

Provide contextual feedback messages for typical user actions with the handful of available and flexible alert messages.

```
<div class="alert alert-primary" role="alert">
  This is a primary alert—check it out!
</div>
<div class="alert alert-secondary" role="alert">
  This is a secondary alert—check it out!
</div>
<div class="alert alert-success" role="alert">
  This is a success alert—check it out!
</div>
<div class="alert alert-danger" role="alert">
  This is a danger alert—check it out!
</div>
<div class="alert alert-warning" role="alert">
  This is a warning alert—check it out!
</div>
<div class="alert alert-info" role="alert">
  This is a info alert—check it out!
</div>
<div class="alert alert-light" role="alert">
  This is a light alert—check it out!
</div>
<div class="alert alert-dark" role="alert">
  This is a dark alert—check it out!
</div>
```

This is a primary alert—check it out!

This is a secondary alert—check it out!

This is a success alert—check it out!

This is a danger alert—check it out!

This is a warning alert—check it out!

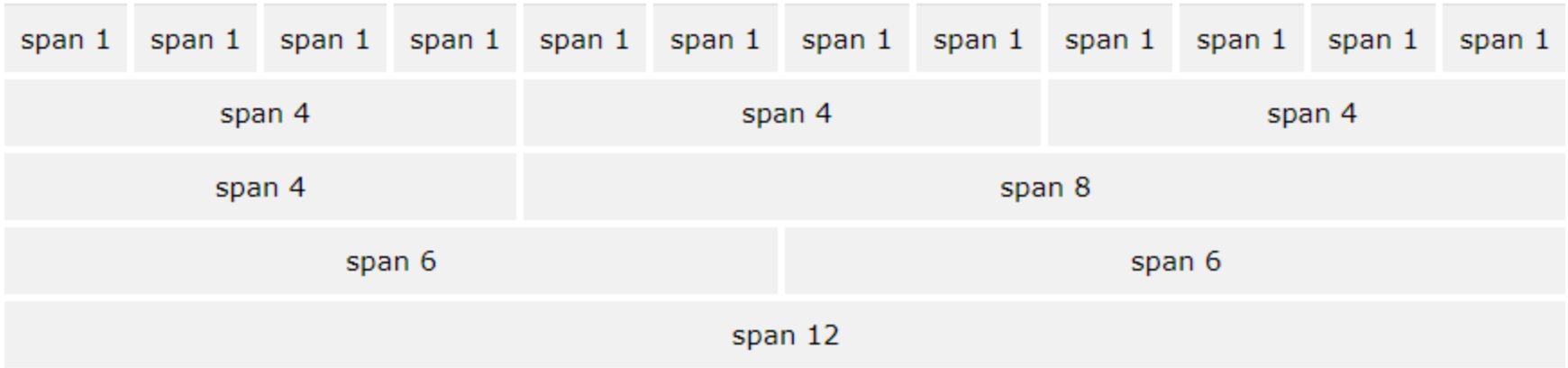
This is a info alert—check it out!

This is a light alert—check it out!

This is a dark alert—check it out!

Bootstrap Grids

- Bootstrap's grid system allows up to 12 columns across the page.
- If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:
- create a row (`<div class="row">`). Then, add the desired number of columns (tags with appropriate `.col-*-*` classes). Note that numbers in `.col-*-*` should always add up to 12 for each row.



Grid Classes

The Bootstrap grid system has four classes:

- `xs` (for phones - screens less than 768px wide)
- `sm` (for tablets - screens equal to or greater than 768px wide)
- `md` (for small laptops - screens equal to or greater than 992px wide)
- `lg` (for laptops and desktops - screens equal to or greater than 1200px wide)

Bootstrap image Shapes

Rounded Corners

The `.img-rounded` class adds rounded corners to an image

Circle

The `.img-circle` class shapes the image to a circle (IE8 does not support rounded corners):

Thumbnail

The `.img-thumbnail` class shapes the image to a thumbnail:

Modal

- Modals are built with HTML, CSS, and JavaScript. They’re positioned over everything else in the document and remove scroll from the `<body>` so that modal content scrolls instead.
- Clicking on the modal “backdrop” will automatically close the modal.
- Bootstrap only supports one modal window at a time. Nested modals aren’t supported as we believe them to be poor user experiences.

```
<div class="modal" tabindex="-1" role="dialog">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <h5 class="modal-title">Modal title</h5>
        <button type="button" class="close" data-dismiss="modal" aria-label="Close">
          <span aria-hidden="true">&times;</span>
        </button>
      </div>
      <div class="modal-body">
        <p>Modal body text goes here.</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-primary">Save changes</button>
        <button type="button" class="btn btn-secondary" data-dismiss="modal">Close</button>
      </div>   </div>  </div></div>
```

Bootstrap Container

In Bootstrap, container is used to set the content's margins dealing with the responsive behaviors of your layout. It contains the row elements and the row elements are the container of columns (known as grid system).

The **container class** is used to create boxed content.

There are two container classes in Bootstrap:

1. Container : class provides a responsive fixed width container
2. Container-fluid : class provides a full width container, spanning the entire width of the viewport