HTML

Saba sadat Faghih Imani

2023

Introduction

- The idea of WWW is document sharing
- Main question: How to define the structure of document?
  - Text, tables, figures, link, …

- HTML (Hyper Text Markup Language)
  - A language to define structure of web docs
  - Tags specify the structure

HTML

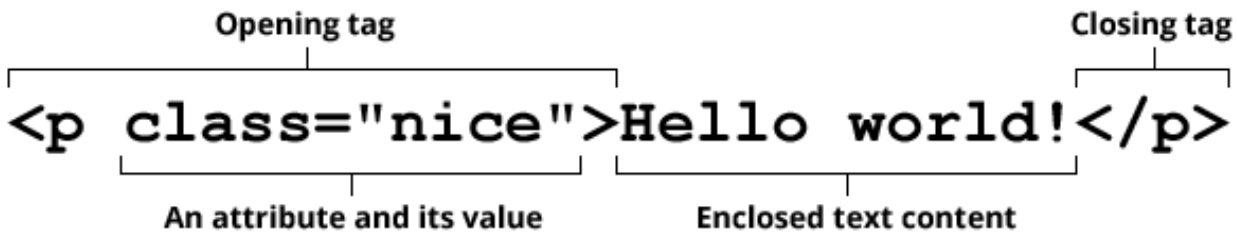
- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.
- HTML Is not a programming language and cannot be used to describe computations.
- HTML does/should not specify presentation . Font family, style, color, …

HTML Basics

- **HTML element:** An HTML element is defined by a start tag, some content, and an end tag.
  - Some HTML elements have no content (like the `<br>` element). These elements are called empty elements. Empty elements do not have an end tag!
  - All HTML documents consist of nested HTML elements.
  - For a complete list of all available HTML tags visit here.

- **HTML attributes:** HTML attributes provide additional information about HTML elements.
  - Attributes are always specified in **the start tag**
  - Attributes usually come in name/value pairs like: **name="value“**
  - Some examples: `href` attribute of `<a>`, `src` and `width` and `height` and `alt` attribute of `<img>`, `lang` attribute of `<html>`, `style` attribute , ….

HTML Basics: Tag & Attribute & Element



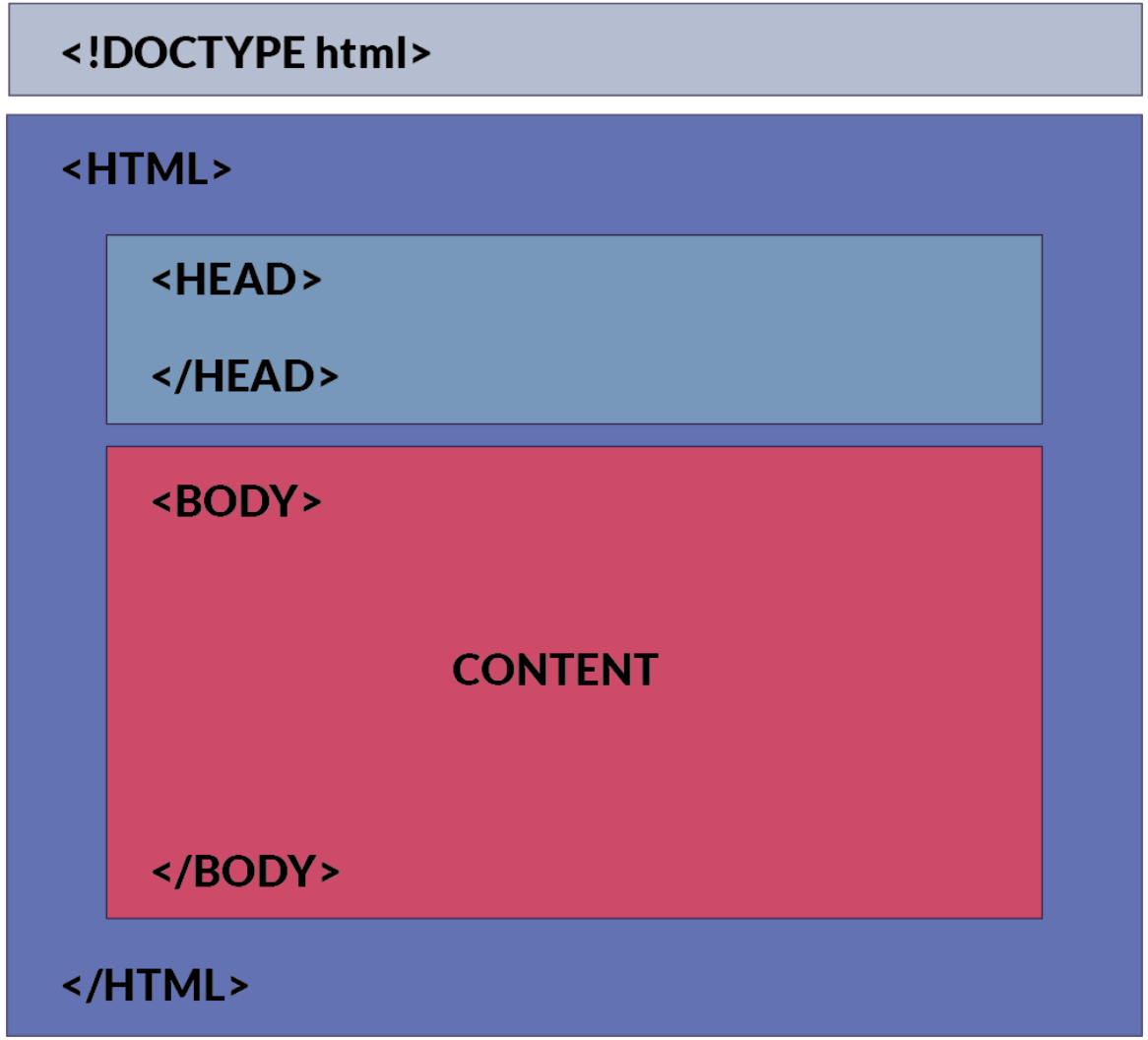
HTML Processing

- HTML is just a text file; How does it work?
- It is *processed* by applications for a specific purpose!
- Search engine objectives:
  - Analyze page, extract elements, prioritize, ranking, …
  - Each tag has meaning, used for ranking
    - E.g., paragraphs are not as important as headings
- Web browser objectives:
  - Display the document to client
  - Rendering
    - Generate layout for the document
    - Display elements

HTML Processing: Rendering

- The processing of displaying HMTL in browser
- Not all tags are to be displayed
  - E.g. Tags in `<head>`
- For tags which should be displayed
  - Tags by themselves are not displayed
  - Each tag has its own default presentation
- If tag has content, the presentation is applied to content
  - E.g. `<i>`This is italic`</i>`
- If tag has not content, the presentation is displayed (if it is needed)
  - E.g. `<br>`

Document Structure



HTML Body: <body> </body>

- The content of the document to be shared on Internet
  - To display for user in web browser
  - To be searched and ranked by search engines

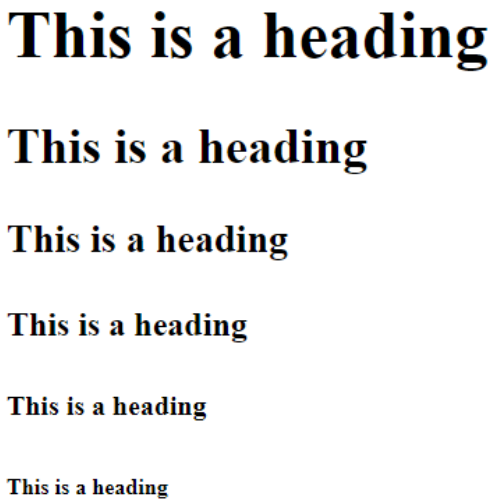
- Which contents?
  - Headings, paragraphs, lists, tables, Images, links, forms, text presentations, spaces, …

HTML Headings

- HTML headings are titles or subtitles that you want to display on a webpage.
- HTML offers 6 levels of heading
  - `<h1>` to `<h6>` : `<h1>` defines the most important heading. `<h6>` defines the least important heading.
- Headings are important.
  - Search engines use the headings to index the structure and content of your web pages.
  - Users often skim a page by its headings.
  - It is important to use headings to show the document structure.

HTML Headings

```
<h1>This is a heading</h1>
<h2>This is a heading</h2>
<h3>This is a heading</h3>
<h4>This is a heading</h4>
<h5>This is a heading</h5>
<h6>This is a heading</h6>
```



HTML Paragraphs

- `<p> </p>`
- A paragraph always starts on a new line, and browsers automatically add some white space (a margin) before and after a paragraph.
- With HTML, you cannot change the display by adding extra spaces or extra lines in your HTML code.
- `<hr>` : defines a thematic break in an HTML page, and is most often displayed as a horizontal rule.
- `<br>` : defines a line break.
- `<pre> </pre>` : defines preformatted text. The text inside a `<pre>` element is is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks

HTML Text Formatting

- Formatting elements were designed to display special types of text:
- `<b>` : Bold text
- `<strong>` : Important text
- `<i>` : Italic text
- `<em>` : emphasized text
- `<mark>` : Marked text
- `<small>` : Smaller text
- `<del>` : Deleted text
- `<ins>` : Inserted text
- `<sub>` : Subscript text
- `<sup>` : Superscript text

```
<b>Bold Text</b> <br>
<strong>Important Text</strong><br>
<i>Italic Text</i><br>
<em>Emphasized Text</em><br>
look at <mark>Some marked text</mark><br>
<small>Small text</small><br>
This is <sub>subscripted</sub> text. <br>
This is <sup>superscripted</sup> text.<br>
My name is <del>Saman</del> <ins>Arash</ins>
```

Bold Text
Important Text
Italic Text
Emphasized Text
look at Some marked text
Small text
This is $_{\text{subscripted}}$ text.
This is $^{\text{superscripted}}$ text.
My name is ~~Saman~~ Arash

HTML Quotation and Citation Elements

- `<blockquote>:` defines a section that is quoted from another source.
- `<q>:` defines a short quotation.
- <abbr>: defines an abbreviation or an acronym, like "HTML", "CSS", "Mr.", "Dr.“.
- <address>: defines the contact information for the author/owner of a document or an article.
- <cite>: defines the title of a creative work (e.g. a book, a poem, a song, a movie, a painting, a sculpture, etc.).

HTML Links

- `<a href="URL">link name</a>`
- A link does not have to be text. A link can be an image or any other HTML element!
- The `target` attribute: By default, the linked page will be displayed in the current browser window. To change this, you must specify another target for the link.
  - `_self`: Default. Opens the document in the same window/tab as it was clicked
  - `_blank`: Opens the document in a new window or tab
  - `_parent`: Opens the document in the parent frame
  - `_top`: Opens the document in the full body of the window

```
<a href="https://www.wikipedia.org/" target="_blank">Visit Wikipedia!</a>
```

HTML Links

- The `title` attribute: specifies extra information about an element. The information is most often shown as a tooltip text when the mouse moves over the element.
- Absolute URLs and Relative URLs:
  - Use a full URL to link to a web page:
    ```
    <a href="https://www.w3schools.com/html/default.asp">HTML tutorial</a>
    ```
  - Link to a page located in the html folder on the current web site:
    ```
    <a href="/html/default.asp">HTML tutorial</a>
    ```
  - Link to a page located in the same folder as the current page:

    ```
    <a href="default.asp">HTML tutorial</a>
    ```

HTML Links

- **Image as a Link**

```
<a href="https://ubuntu.com/"><img src="./img/ubunto1.jpg" alt="Ubuntu" style="width:50px;height:50px;"></a>
```

- **Button as a Link:** To use an HTML button as a link, you have to add some JavaScript code. JavaScript allows you to specify what happens at certain events, such as a click of a button:

```
<button onclick="document.location='https://ubuntu.com/'">Ubuntu Website</button>
```

HTML Images

- The `<img>` tag is empty, it contains attributes only, and does not have a closing tag. It has two required attributes:
  - `src`: Specifies the path to the image
  - `alt`: Specifies an alternate text for the image
- Image Size: You can use the `style` attribute to specify the width and height of an image. Alternatively, you can use the `width` and `height` attribute.

```
<img src="./img/ubunto1.jpg" alt="Ubuntu" width="128" height="128">

<p>The second image uses the style attribute to set the width to 128 pixels,
   this will not be overridden by the style in the head section:</p>

<img src="./img/ubunto1.jpg" alt="Ubuntu" style="width:128px;height:128px;">
```

HTML Images

- Background Image: To add a background image on an HTML element, use the HTML `style` attribute and the CSS `background-image` property. You can also specify the background image in the `<style>` element, in the `head` section.

```
<p style="background-image: url('./img/ubunto1.jpg');
 background-repeat: no-repeat; background-size: 100% 100%;">
Ubuntu is a Linux distribution based on Debian and <br>
composed mostly of free and open-source software.<br>
Ubuntu is officially released in multiple editions:<br>
 Desktop, Server, and Core for Internet of things devices and robots.<br>
All of the editions can run on a computer alone, or in a virtual machine.
</p>
```

HTML Lists

- Unordered list: `<ul> </ul>`
  - list elements => `<li> </li>`
- Ordered list: `<ol> </ol>`
  - list elements => `<li> </li>`
- Description list: `<dl> </dl>`
  - list elements => term : `<dt> </dt>` description of term: `<dd> </dd>`
- List can be nested.

HTML Lists

```
<h3>Unordered list</h3>
<ul>
    <li> Item 1 </li> <li> Item 2 </li>
    <ul>
        <li> Nested 1</li>
    </ul>
</ul>
<h3>Ordered list</h3>
<ol>
    <li> Item 1 </li>
    <ol>
        <li> Nested 1 </li> <li> Nested 2 </li>
    </ol>
    <li> Item 2 </li>
</ol>
<h3>Definition list</h3>
<dl>
    <dt>Item 1 </dt> <dd> This def. of item 1 </dd>
    <dt>Item 2 </dt> <dd> This def. of item 2 </dd>
</dl>
```

Unordered list

- Item 1
- Item 2
    - Nested 1

Ordered list

1. Item 1
    1. Nested 1
    2. Nested 2
2. Item 2

Definition list

Item 1
: This def. of item 1

Item 2
: This def. of item 2

HTML Tables

- Tables are created by `<table> </table>`
- Each row is created by `<tr> </tr>`
- Each column inside a row is created by `<td> </td>`
- Sometimes you want your cells to be table header cells. In those cases use the `<th>` tag instead of `<td>` tag.
- Caption is by `<caption> </caption>`
- To add a border, use the CSS `border` property on `table`, `th`, `td` elements.

HTML Tables

- `<tr>` attributes
  - `align:` text align in row: "left", "right", "center"
  - `valign:` text vertical align: "top", "middle", …
  - `bgcolor:` Row background color
- `<td>` or `<th>` attributes
  - `align, valign, bgcolor, height , width`
  - `colspan:` Span multiple columns
  - `rowspan:` Span multiple rows

HTML Tables

```
<table>
    <tr>
        <th></th>
        <th>Heading of column 1</th>
        <th>Heading of column 2</th>
        <th>Heading of column 3</th>
    </tr>
    <tr align="center">
        <th>Center</th>
        <td>1</td>
        <td>2</td>
        <td rowspan="2">3</td>
    </tr>
    <tr align="left" bgcolor="red">
        <th>Left</th>
        <td valign="bottom">1</td>
        <td bgcolor="blue">2 <br /> 2</td>
    </tr>
    <tr align="right">
        <th>Right</th>
        <td height="50" width="300">1</td>
        <td colspan="2">2</td>
    </tr>
</table>
```

	Heading of column 1	Heading of column 2	Heading of column 3
Center	1	2	3
Left	1	2 2	
Right	1	2	

HTML Forms

- An HTML form is used to collect user input. The user input is most often sent to a server for processing.
- HTML is only responsible to *gather* the information .It does not responsible to process
  - Data are processed by server side scripts
  - However, some preprocessing can also be performed in client side
- Major form components
  - The form element
  - Inputs
    - Text input, Checkboxes and radio buttons, Select boxes, File select
  - Buttons
    - submit, cancel, …

HTML Forms

- The `<input>` element: An `<input>` element can be displayed in many ways, depending on the `<type>` attribute. Here are some examples:

Type	Description
`<input type="text">`	Displays a single-line text input field
`<input type="radio">`	Displays a radio button (for selecting one of many choices)
`<input type="checkbox">`	Displays a checkbox (for selecting zero or more of many choices)
`<input type="submit">`	Displays a submit button (for submitting the form)
`<input type="button">`	Displays a clickable button

HTML Forms

- A from is composed of `<input>` elements
- Each form must have `action` and `method` attributes
  - `action` is a URL
    - Server side script that process the data
  - `method` is a HTTP method used to send data
    - `get`: User input data is sent through the *query part* of URL by HTTP GET method
    - `post`: User input data is sent as the *body* of HTTP message by HTTP POST method
- Each component has `type`, `name`, and `value` attributes
  - `type` specifies the type of component
  - `name` is the name of the component
  - `value` (except buttons)
    - If not empty, is the default value

HTML Forms: Buttons

- Buttons: `<input type=“T” value=“L”/>`
  - Predefined buttons:
    - To submit data to server: `type="submit"`
    - To reset all inputs to default values: `type="reset"`
  - To run client side script: `type="button"`
- Attribute `value` is the label of button
- `<input type=“T” value=“L”/>` can be replaced by `<button type=“T”> L </button>`
- Using image as a button
  - `type="image" src="image path" alt="text"`
- Attribute `name` is required if more than same type button in a form

HTML Forms: Text Input

- Single-line text
  - `type="text"`
- Password (instead of real input, other character is shown)
  - `type="password"`
- Multi-line text
  - Instead of `<input>`, we use `<textarea> </textarea>`
  - `cols` & `rows` specifies # of columns & rows
- `name=`*`value`* of component is sent to server
  - Password in plain text format!!!

HTML Forms: Text Input

```
<h2>HTML Forms</h2>

<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="Saba"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Imani"><br>
  <label for="pass">Password:</label><br>
  <input type="password" id="pass" name="password" value=""><br><br>
  <input type="submit" value="Submit">
  <input type="reset" value="Reset">
</form>

<p>If you click the "Submit" button,
    the form-data will be sent to a page called "/action_page.php".</p>
```

HTML Forms

First name:
Saba
Last name:
Imani
Password:

Submit Reset

If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".

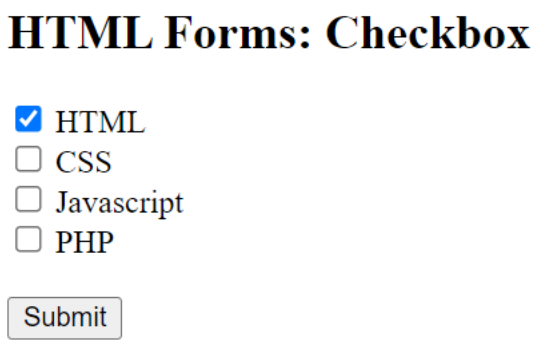
HTML Forms: Checkboxes

- The `<input type="checkbox">` defines a checkbox.
- Checkboxes let a user select ZERO or MORE options of a limited number of choices.
- If checked, its `name=value` is sent to server
  - User cannot change/enter value
- The `value` attribute is needed in most cases
  - If not given, it is assumed "`on`"
- To be checked by default:
  - `checked="checked"`

HTML Forms: Checkboxes

```
<h2>HTML Forms: Checkbox</h2>

<form action="/action_page.php">
        <input type="checkbox" id="skill_1" name="skill_1" value="HTML" checked="checked">
        <label for="skill_11"> HTML</label><br>
        <input type="checkbox" id="skill_2" name="skill_2" value="CSS">
        <label for="skill_2"> CSS</label><br>
        <input type="checkbox" id="skill_3" name="skill_3" value="JavaScript">
        <label for="skill_3"> Javascript</label><br>
        <input type="checkbox" id="skill_4" name="skill_4" value="PHP">
        <label for="skill_4">PHP</label><br><br>
        <input type="submit" value="Submit">
  </form>
```



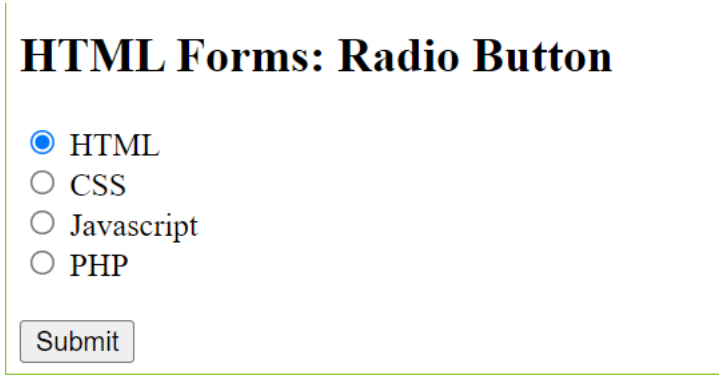
HTML Forms: Radio Buttons

- The `<input type="radio">` defines a radio button.
- Radio buttons let a user select ONE of a limited number of choices.
- Only one of button can be selected in a group of buttons with the same `name`
- `name=value` of the selected button will sent
  - Again, user cannot change/enter value
  - If the `value` attribute is missing, the default value is "`on`"
  - The `value` attribute is (almost always) needed

HTML Forms: Radio Buttons

```html
<h2>HTML Forms: Radio Button</h2>

<form action="/action_page.php">
        <input type="radio" id="skill_1" name="skill" value="HTML" checked="checked">
        <label for="skill_11"> HTML</label><br>
        <input type="radio" id="skill_2" name="skill" value="CSS">
        <label for="skill_2"> CSS</label><br>
        <input type="radio" id="skill_3" name="skill" value="JavaScript">
        <label for="skill_3"> Javascript</label><br>
        <input type="radio" id="skill_4" name="skill" value="PHP">
        <label for="skill_4">PHP</label><br><br>
        <input type="submit" value="Submit">
</form>
```



HTML Forms: Select Boxes

- The same functionality of radio buttons
  - However, to save spaces
- Created by
- `<select name="selname"> </select>`
- Options are given by `<option value="val"> text </option>`
- *`slename=val`* of the selected item is sent to server
- User cannot enter value; If the value attribute is missing, the "*`text`*" is assumed as the value

HTML Forms: Select Boxes

```
<form action="http://127.0.0.1/" method="get" name="frmColors">
    <label for="select">Select Color:</label>
    <select name="selColor" id="select">
        <option value="r">Red</option>
        <option value="g">Green</option>
        <option value="b">Blue</option>
    </select>
    <input type="submit" value="Submit" />
</form>
```

HTML Block vs Inline Elements

- Every HTML element has a default display value, depending on what type of element it is. There are two display values: block and inline.
- A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element. It always takes up the full width available (stretches out to the left and right as far as it can).
  - block-level elements in HTML: `<p>` `<div>` `<address>` `<h1>-<h6>` `<dl>` `<dt>` `<li>` `<hr>` `<ol>` `<ul>` `<pre>` …
- An inline element does not start on a new line.
  - inline elements in HTML: `<a>` `<abbr>` `<b>` `<big>` `<br>` `<button>` `<cite>` `<em>` `<img>` `<input>` `<span>` `<strong>` `<sub>` `<sup>` …

`<div>` Element

- `<div>` is a general block-level element
  - To create an element *without* any presentation
  - To group some existing block-level elements
- Nested `<div>` are used to define structure of complex pages.
- The `<div>` element has no required attributes, but `style`, `class`, and `id` are common.

`<span>` Element

- `<span>` is a general inline element
- Used to create an inline element *without* any presentation
- The `<span>` element has no required attributes, but `style`, `class`, and `id` are common.
- Behavior & Presentation of `<span>` & `<div>` are controlled by JavaScript & CSS

HTML HEAD: <head> </head>

- The elements (usually) not for displaying
  - Mainly, the info in head is not for user

- This element is additional information for
  - Web browsers: How to render the page
  - CSS rules definitions and inclusions
  - JavaScript codes
  - …
- Search engines: Control the ranking of the page
  - Keywords for the page
  - Extra description for the page

HTML HEAD: <head> </head>

- The HTML <head> element is a container for the following elements: <title>, <style>, <meta>, <link>, <script>, <base>
- <title>: defines a title in the browser toolbar and in search engine-results
  - The content of a page title is very important for search engine optimization (SEO)!
- <style>: is used to define style information for a single HTML page.
  - We will discussed in the next lecture.
- <link>: To link some other documents to this HTML file. (external CSS , …)
- <script>: is used to define client-side JavaScripts.
  - We will discussed in next lectures.

HTML HEAD: <head> </head>

- <meta>: is typically used to specify the character set, page description, keywords, author of the document, and viewport settings. The metadata will not be displayed on the page, but is used by browsers (how to display content or reload page), by search engines (keywords), and other web services.

```
<meta charset="UTF-8">
<meta name="description" content="Web Programming Course">
<meta name="keywords" content="HTML, CSS">
<meta name="author" content="Imani">
```

- <base>: specifies the base URL and/or target for all relative URLs in a page. This tag must have either an href or a target attribute present, or both.
  - <base href="http://www.abc.com"/>
  - link1 → http://www.abc.com/test.html
  - link2 → http://test.html

HTML Debugging

- Browser reads HTML document
  - Parses it -> tree
  - Document Object Model (DOM) tree
    - Shows how browser interprets your HTML file
- Google Chrome “Inspect element”
- Firefox developer edition
- Firefox extensions
  - Firebug
  - Web Developer toolbar

HTML Validation

- https://validator.w3.org/