

Day 1 of Notes and coding (13, Jan, 2023) [Target :Finish the HTML course and jump to CSS]

COLOR: → Easy, I know

COLOR: → Need to learn with examples

COLOR: → Topics

HTML notes:

From the first skillshare course, I learnt that xhtml and html work the same but there are some differences between them. Xhtml is generally strict and mostly used for xml languages.

`
` -> An example of HTML

`
` -> An example of XHTML

Here, we can clearly understand that XHTML has a closing /. Other examples can be like:

`<Input type="text" disabled >` → an example of HTML

`<Input type="text" disable="disabled" >` → an example of XHTML

In conclusion, xHTML is a little bit stricter than html.

About Internal, External and inline CSS

Learned about tags which I was unknown about. `<code>`, `<samp>`, `<kbd>`, `<pre>`, `<var>`

`<code>` → tag simply formats the line as per the previous line. It's used for file name. For eg:

`<p>Hello! This is Nikesh Acharya. </p>`

`<code>!! </code>`

`<samp>` → The `<samp>` tag is used for code formatting.

`<kbd>` → tag is used specially for keyboard words like:

`<kbd> ctrl </kbd> + <kbd> R</kbd>`

`<pre>` → represents a block of preformatted text.

`<var>` → The `<var>` tag is used to define a variable in programming or in a mathematical expression.

`<details> </details>` → It's mostly used to write about copyright

`<time> </time>` → show the time of publication, comments and more

`<mark> </mark>` → It is used to highlight the text. The color of highlighting text is yellow.

Bold and Italic Semantics:

Even though bold and strong tags give the same result, in fact, strong is more bold.

They are especially important for screen readers.

Italic and em tags follow bold and strong cases.

Head Elements:

`<meta charset = "UTF-8">` → Hey browser, make sure you are going to accept every sort of language there!! This is the globally acceptable format.

`<meta name="viewport" content="...">` → makes our website responsible

`<title>` → The title tag is the part of head element.

Also, we can use `<link>` tag with `<script>` tag on it. `<script>` tag is used for writing javascript code.

In short, the meta elements are used in more amounts than other elements. Meta elements do not have child elements.

In general, the head element doesn't go deep. The tags used inside the head element do not consist of child tags.

List of all meta tags

<https://gist.github.com/kevinsuttle/1997924>

Imp. , useful meta tags:

`<meta name="robots" content="index,follow">` → let google allow users to display a site in their site.

Linking Favicon:

`<link rel="icon" type="image/png" href="favicon/favicon.png">`

Here, note that if we are using favicon.jpg, then we change type to image/jpg and so on

Favicon doesn't work on the internet and Mozilla Explorer. The way to check if they are working or not is by going to inspect the element and clicking on the link. If it gives you an image. The favicon is working.

Ctrl + shift+ r → hard refresh

HTML Entities:

`<` → Less than symbol

`>` → greater than symbol

` ` → Non-breaking space

`©` → copyright symbol

`&` → ampercentage

`→` → right arrow

`←` → left arrow

`•` → bullet points

`»` → right double chevrons

`«` → left double chevrons

Form

`<form method="post/get" action=" // where actually we are sending this form to usual" >`

..

`<textarea/>`

`<select>`

`<option> ... </option>`

`</select>`

`<input/>`

`</form>`

Get method → It's one way of sending data to the server. It always takes whatever the name of the input is. Because of the get method, it will show the submitted info in the url.

```
<form method="get" action="">
<div>
<label for="firstname"> Enter your firstName: </label>
<input type="text" name="Firstname" >
</div>
<input type="submit" name="Submit">
```

- It is a bad way to send data. It's an insecure way of receiving data.
- On the other hand, post is the secure method to be used in the form.
- The post method does not show variables in the url.

All input types→

```
<input type="submit" name="Submit">
<input type="number" name="number" value="number" step="0.5" max="50"
min="0">
<input type="radio" name="Submit">
<input type="checkbox" name="Submit">
<input type="password" name="Password">
<input type="file" name="file" value="File-Upload">
<input type="url" name="Your url here">
```

About Textarea

- It's not as a input field.
- It does have an ending tag.

```
<textarea name="textarea"
placeholder="ghost Text"
id="t1"
cols="30"
rows="10"
minlength="5"
autofocus="autofocus"
maxlength="30"
Required
disabled</textarea>
```

Note that: <label for="// id here">
In above case,

<label for="t1"> Your text Area</label>

Autofocus, to immediately place the cursor in that textarea. We can use it in only one of the input fields in our webpage. This enhances the user-friendly behavior.

User can also disable any of these inputs and textarea by:

In XHTML disabled = "disabled"

In HTML just typing disabled in side the opening tag

Button

```
<form method="get" action="">
  <input type="submit" value="Submit">
    <button type="submit"> Submit</button>
</form>
```

These two are the same. One has value in the opening tag while the other has closing tag and value is given in between them.

DropDown →

IT can be made using <select> with its child tag <option>.

Video element→

Video tag allows us to display video. It has a child tag called source where we place the source for the video.

```
<video width="400" controls loop autoplay = true muted >
// Video was introduced in html5
```

```
<source src = ".." type="video/mp4">
<source src = ".." type="video/mp4">
</video>
```

Audio element →

```
<audio>
<source src= ".." type="audio/mpeg">
</audio>
```
