

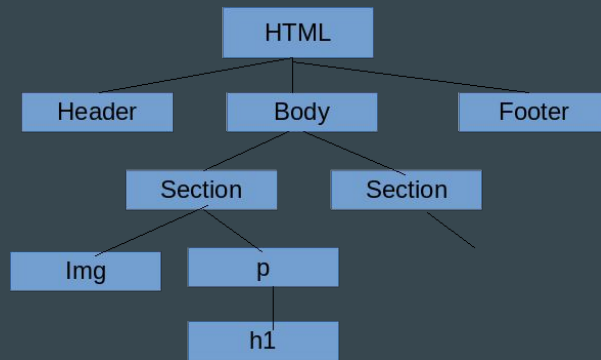
CodeSoc: Web development

...

Class 3: More HTML elements & style

Last time

- Considered the way webpages should be structured
- Used `<div>` and `` so we could group items
- Thought of webpages as a tree - and how styling affects all descendants if you apply to a parent element
- Worked through a blog example



Outline

1. Introduction and creation of simple example website
2. (Today) The HTML document - Page structure, content types & best practises
3. More HTML elements & adding style (part i)
4. More HTML elements & adding style (part ii)
5. Multi-page sites & navigation
6. External style
7. Advanced CSS
8. Introduction to Javascript

HTML Recap

- The content of the website - titles, paragraphs, images etc.
- Tell the browser the type of content by using *tags*
 - `<type> content </type>` or `<type> (
)`
 - Tags may have *attributes* ``, ` `
- Each tag creates an *element*
- Elements may contain other elements
 - This nesting is a crucial part of HTML, it gives structure to the page
 - Key idea of CS - split the problem/design into smaller subproblems & combine
 - We want to use this nesting often as it's often easier to think about smaller parts
- Special tags `<div>` and `` used for grouping elements
- In-line styling though the `style= "..."` attribute

Style recap (CSS)

- Used **padding** to create spacing that improved the look of our content
- Briefly went over using borders and margins (to be covered)

Today

- Today we will make a site with more varied elements
 - Think lists, tables etc..
- Will be introducing more ways to style elements

Lists

- Come in two types - ordered and unordered
- `...` for unordered `...` for ordered
- A list element contains many list-item elements denoted by ` item `
- Try adding both an ordered and unordered list
- What happens if one of the list-items is itself another list element?

Pretty lists

- We can use some inline-styles to make your list look much better
- In this notation | means “or”
- `list-style-type: circle | square | upper-roman | lower-alpha | none`
 - This changes the type of the marker for each list item
- `list-style-image: url(“...”)`
 - Changes the marker to an image

Tables

- Similar to lists, but a bit more going on
- You make table row by row, starting at the top
- `<table> ... </table>` is the tag used for the whole table element
- `<tr>... </tr>` is the tag for a row
- Inside your row you define the table data using `<td> ... </td>`
- If you want a heading for each column, you use `<th>` instead of `<td>`
- One last thing... you can use an attribute called `rowspan= "X"` inside of `<td>`
 - See what this does! (set X to 2)

Pretty tables

- We can use some inline-styles to make your table look much better
- `border-style: solid | dashed | double | inset | none`
 - `border-top-style: solid | dashed | ...`
 - `top` can be replaced with `left | right | bottom`
- `border-width: 5px` - changes the width
- `border-color: red | #777777` (try using any 6 digit number...)
- `border-radius: 3px` - makes the border rounded
- For all you can write `border-X-color` where `X = top | bottom | left | right`

Pretty tables

- Change the color of each row
 - Use the style: `background-color: blue | #eeeeee`
 - What happens when you give other elements this style?

Key points

- It's hard to remember all the ways of styling things
 - Use online resources (like these slides) to find ones you forget / I haven't covered
- To work with lists and tables you just need to remember how they are structured

Next time

- A couple more types of content
- More ways of styling things we already know!