

# CodeSoc: Web development

...

Class 5: Multi-page sites & navigation

# Last time

- Consolidated knowledge of HTML & elements

# 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) - consolidating what we know
5. Multi-page sites & navigation
6. External style
7. Advanced CSS
8. Introduction to Javascript

# Today

- Learn how to structure our HTML files for multi-page websites
- Use the `<nav>` element to provide navigation between our sites

# Quick review

- The content of the website - titles, paragraphs, images etc.
- Tell the browser the type of content by using *tags*
  - `<type> content </type>` or `<type>` (`<br>`)
  - Tags may have *attributes* ``, `<a href="www...">` `</a>`
- 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 `<span>` used for grouping elements
- In-line styling though the `style= "..."` attribute

# First things

- Create a new folder which will act as the *root* of the website collection
- Now copy all of the sites you have made in previous classes to this folder
  - You might want to rename them to things like “class1.html”... “class2.html”
- Next create a new website called “home.html”
- Finally open the home.html file & add in html and body tags

# Navigation

- A `<nav>` tag is used to denote a navigation element
- All a navigation element is a collection of (local) links to other webpages
  - You can also use non-local links inside it
- It's sometimes a good idea to add a pipe character (“|”) between links to make it look better
- `<nav>`

```
<a href="home.html"> Home </a> |  
<a href="class1.html"> Class 1 </a>  
</nav>
```

# Navigation

- Add in links to all of the sites you have made so far
- After it should look like the following (maybe without the heading I added)

## **My collection of sites!**

[Home](#) | [Class 1](#) | [Class 2](#) | [Class 3](#) | [Challenge 1](#) | [Challenge 2](#) | [Challenge 3](#)



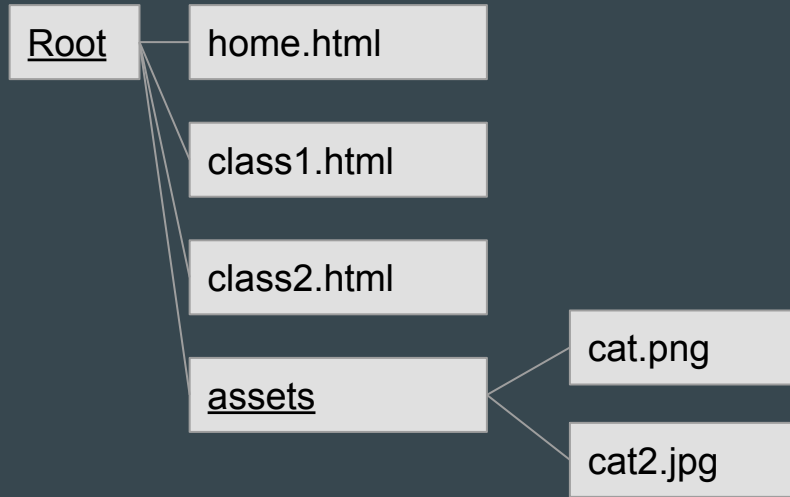
# Navigation

- We have one problem - when a link is clicked, we lose our navigation!
  - This is because your other websites don't have the `<nav>` element!
- Fortunately this is easy to resolve - just copy and paste the `<nav>` element into a sensible place on the other sites

# Making photos local

- We can link to pictures the same way we have linked to other pages
- Make a new folder inside the *root* folder called *assets*
  - This is where we will add the pictures we use
- Download all the pictures you have used previously to the *assets* folder
  - It'll help to give them semi-sensible names
- Recall our image tags have the **src** attribute
- We set this to “assets/imageName.**fileType**”
  - **fileType** may be png, jpeg, jpg...

# File structure



# Final remarks

- Next time we will be using other assets - CSS style sheets
- These will be added in a similar way and provide site-wide styling