CodeSoc: Web development

● ● ●

Presented by: Josh warwick

Introduction - About me

- 4th year CS undergrad (Balliol College)
- CS interests lie in programming languages & verification
- Coding for 5+ years
- Favourite language - Ruby
- Email: Joshua.warwick@balliol.ox.ac.uk

Introduction - This course

- Beginner class into programming
- Aimed for people with zero experience
- Learning the tech behind "front end web development"
- "learn from doing" style of teaching - bring your laptop!

Introduction - Today

- A condensed version of what the course will be teaching
- By the end of the lesson you will have created a basic website
- It may feel like we are going through things quickly
  - This is to show what sort of things we can do
  - The classes in future will explain what's happening in detail

HTML (HyperText Markup Language)

- The content of the webpage
- Tells the web browser what we are trying to show
  - Could be an image, paragraph, link..

CSS (Cascading Style Sheets)

- Transforms plain HTML into something more pretty
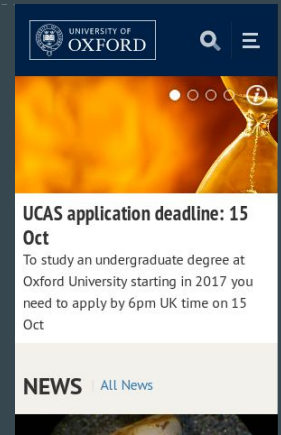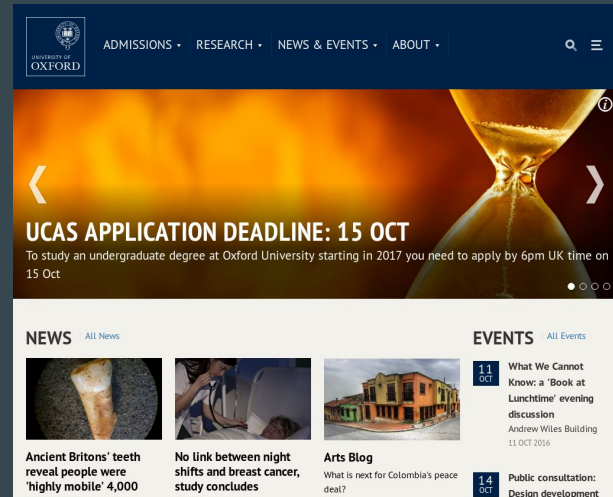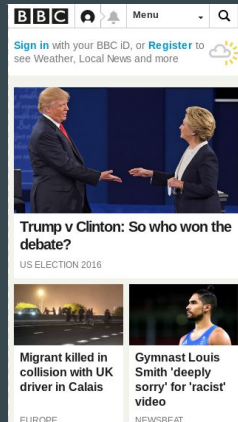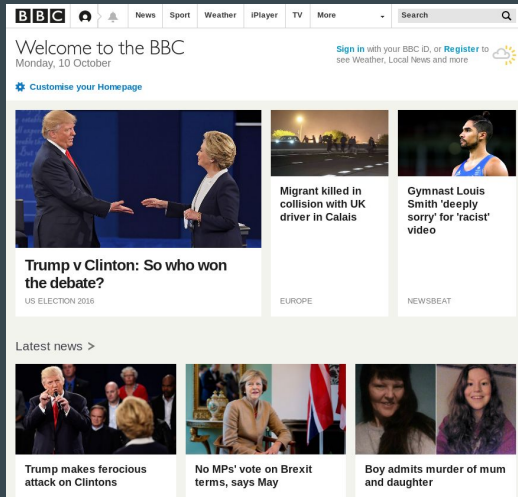- A description of how to *style* the webpage

JS (Javascript)

- The engine that powers dynamic content
- Every interaction with a website uses javascript
- A website with no javascript is comparable to a poster

HTML+CSS+JS = Every possible web design

- These three components make up every feasible website design
- It's possible to make a basic website very quickly

Okay let's get started

- First open a text editor
  - Windows users: Notepad
  - Mac users: TextEdit
  - Linux users: whatever you like
- Next open a web browser
- These two tools are all you need to create a website (and view it)

First website

- In your text editor write something you want to display
  - I'll use "Hello World"
- Now save your file to your desktop - call it "myFirstWebsite.html"
  - Make sure you include ".html" - this is so the browser knows it's a html document
- Next find the file you have just created
- Finally drag and drop the new file into your web browser
- Hooray - you should see the text on the screen!
  - If not say so now!

So what's going on here?

- Is this really a website??
  - Yes, it's really easy to make a simple page like this
- The browser knows to display this as a web page as we have the ".html" *extension*
- We can also specify this in the file - using *tags*

Tags

- Tags are labels that tell the browser what its reading
- The following tags tell the browser we are using HTML:

```
<html>
    Hello World!
</html>
```

- Here <html> is the tag that describes what content is about to follow
- <\html> is also a tag - it tells the browser that the *html* content is finished
- Generally tags are of the form: <start tag> content <end tag>
- Now add html tags to your website!

Elements

- Tags are used very often in html, they form what's called an *element*
- `<h1> Hello world! </h1>` as a whole is an example of an <u>element</u>
- Here "h1" means that the content to follow is a heading
    - The "1" just means we want the first type of heading (there are 6)
- A web page is wholly constructed out of these elements

- Now make your text on your website a heading!

Adding more content

- We can add a paragraph using the tag <p> ... </p>
- We can add a line break using
- Links
  - A special type of tag that has a *attribute*
  - Of the form Link text here
  - Links can go inside of paragraphs
    - This is called *nesting* and is a key part of using html
- Images - use a special tag that doesn't have an end
  - Of the form <img src="www.example.com/cat.png">
    - Notice that an image is in fact contained inside the tag itself

Making things pretty

- Currently the content of the page looks boring, all black and white
- We use a *style* tag to tell the browser how to style the webpage
  - For example we can make the header red with this:
  - <style> h1 { color: red; } </style>
  - There is a bit of new syntax here, which will be completely explained in upcoming classes

More pretty things...

- We can add some style to the paragraphs we have made
  - p { text-align: centre; } centres the text in paragraph
  - You can add multiple styles by spearing them with a semicolon
- We can add different styles that occur when we hover over things
  - h1:hover { color: blue; } makes titles blue when we hover over them
  - img: hover { width: 300px; } makes images 300pixels wide when we over over them

Ok, slow down

- We have covered very quickly, allot of new ideas
- Future classes will go through things more comprehensively
- The process of adding content, then style, is how we will progress through the term
- We don't expect you to remember any of what has been covered today

Closing remarks

- This has been a fast paced lesson to show what's possible using HTML
- We will be going into details of all the different types of content over the term
- My email: Joshua.warwick@balliol.ox.ac.uk
- Materials should be posted on the website
  - If not contact me and I'll email them to you
- Next class time: Monday 5:30pm
- In weeks 3-8 the class will be at 6:30pm on Mondays