Beginner Course Syllabus - HTML/CSS & jQuery

Pre-course Preparation: **Welcome**

- Welcome to web dev & code first girls community
- Basic intro to web languages with mention of HTML, CSS, JS (jQuery) & how they are related (i.e. web pages & web servers)
    - Not expected to understand the syntax, but we will demonstrate what each language does to a web page
- Introduction to the tech community, resources & how to study, developer news
    - open-source, collaboration (GitHub), stack overflow, w3 schools for online references
    - How can i quickly get help? how can i meet other developers?
    - what can these skills prepare me for? where do I go after this course?
- To Do 1: Create a github account
- To Do 2: installing software: Chrome, Sublime Text, GitHub Desktop Client (https://desktop.github.com/)
- To Do 3: GA Dash 1

Session 1: **Getting going**

- software installation troubleshooting (10 minutes)
- Intro to webpages & web servers - cover this in Prep, and do a Q&A (10-15 mins)
- more webserves & URL (10-15 mins) - link to DNS video
- Creating a HTML page (15 mins)
- HTML syntax - use a demo to talk through it (10-15 mins)
- Homework: GA Dash 2, create your own website locally, read something on CSS?/ figure out how to use GitHub (reading about GitHub)

Session 2: CSS

- talk about using it in <head> tags, but get them to use CSS in a separate file.
- CSS, Selectors and Attributes, Stylesheets
- how to use GitHub - what is version control? & Q&A (basic concepts, commits, pulls, forks, etc) (10 mins)
- GitHub Pages (https://pages.github.com/), hosting your website (+ talking about competition) (10 mins)
    - mention other things available on GitHub (https://education.github.com/pack for students), GitHub Gist
- Competition: Explain competition criteria, ask everyone to form teams by week 3 and brainstorm ideas - collaborate on Fb/HowCloud.

- Homework: GA Dash 3, host your site on github pages, read up on frameworks & libraries

Session 3: Bootstrapping it up

- GitHub Pages troubleshooting
- Introduction to frameworks & libraries (in CSS & JS later)
- Using Twitter Bootstrap to improve presentation of webpages
- Homework: make your website responsive
- Competition: Spend 10 mins at the end of the session ask the participants to get into their teams. Find teams for those who do not yet have a team. Write down names of teams and ideas. Explain competition criteria again to students (outlined below)

Session 4: jQuery

- jQuery: what? - talk about how it's different from JavaScript
- how to use? + resources for JS
- manipulating CSS with jQuery
- demo syntax on CodePen
- Last 15 mins: using google analytics and embedding google forms. - walk through it
- Homework: install Analytics & HTML Forms, add JS to your website
- Competition: Encourage groups to meet up outside class to work on the project

Session 5: Review and Competition project week

- Work on group projects for the CF:G Competition
- optional (do a poll from students?) - introduce concepts:
  - what is the command line? - *recommended*
  - or how do I link my GitHub page to a domain name? (CNAME stuff) - not as important

Session 6: Competition project final touches and presentation to class

- Spend the first 45 minutes finalising project websites.
- Spend 1 hour on group presentations (5-10 mins/group).
- Instructors to choose a winner and announce it at the end of the session.

Competition Guidelines

The CF:G competition is a chance for students to put into practice the skills learnt on the course. Students can work individually or in groups of 2-3 to create a landing page for a website. Aim to form teams by Week 3.

The criteria for the competition are here:

- A visually appealing design - good use of CSS and HTML elements, Twitter Bootstrap
- Good formatting
    - Code split into the appropriate files
    - Files indented properly
- A live website (Github page, Heroku or own domain)
- Extras e.g:
    - A contact form (for example name and email)
    - Social buttons
    - Widgets
    - As many different HTML elements you can manage

  Interactive elements (like forms) on the website don't need to be functional, but should be present if they need to be for the visual aspect of the design.
- (optional) Good organisation
    - Version control using GitHub
    - Sensible commit messages

Some of the winning entries from last term's competition can be found here.

There'll be prizes of Amazon vouchers for the winning team and a chance to be pitched alongside the winning entries from courses around the U.K. for a grand prize.