Class 5: CSS Typography

What is typography?
CSS typography
Web typography in transition
Font services
Serif and sans-serif fonts
How to choose a typeface
Alignment, line spacing, measure and vertical rhythm
Units

References
Learning Web Design by Jennifer Robbins (5th Edition)
HTML & CSS by Jon Duckett
On Web Typography by Jason Santa Maria
Web Typography by Richard Rutter
Flexible Typesetting by Tim Brown
A Practical Guide to Designing for the Web by Mark Boulton
What is Typography? by David Jury
http://webtypography.net/
https://practicaltypography.com/
https://designingfortheweb.co.uk/part3/chapter11/ (Part 3: Chapters 11-14)
https://www.webstyleguide.com/wsg3/8-typography/index.html
https://alistapart.com/article/how-we-read
https://www.youtube.com/watch?v=6ardZEhjvV0 – Tips for great typography tutorial
https://www.pierrickcalvez.com/journal/a-five-minutes-guide-to-better-typography
https://cssreference.io/typography/

Class 5 Homework
Read: Chapter 12 of Learning Web Design
            or Chapter 8 of Head First HTML and CSS
            or Chapter 12 of HTML & CSS
   Part 3 of A Practical Guide to Designing for the Web
            and/or Chapter 5 of Above the Fold
            Any or all of On Web Typography
View: Beginner's guide to styling text with CSS by Kevin Powell (26 mins)

Task 1: Continue the development of your three designed objects website, consider the use of CSS to control and design the typography. Consider concepts such as visual hierarchy, vertical rhythm, measure, and alignment to improve the readability of your mini site. For this project, look at the range of typefaces available on the Google Fonts service and select one or more (two maximum) that are appropriate to your project site. Develop a typographic style and apply it to your site.

Course materials: Design for Web Content
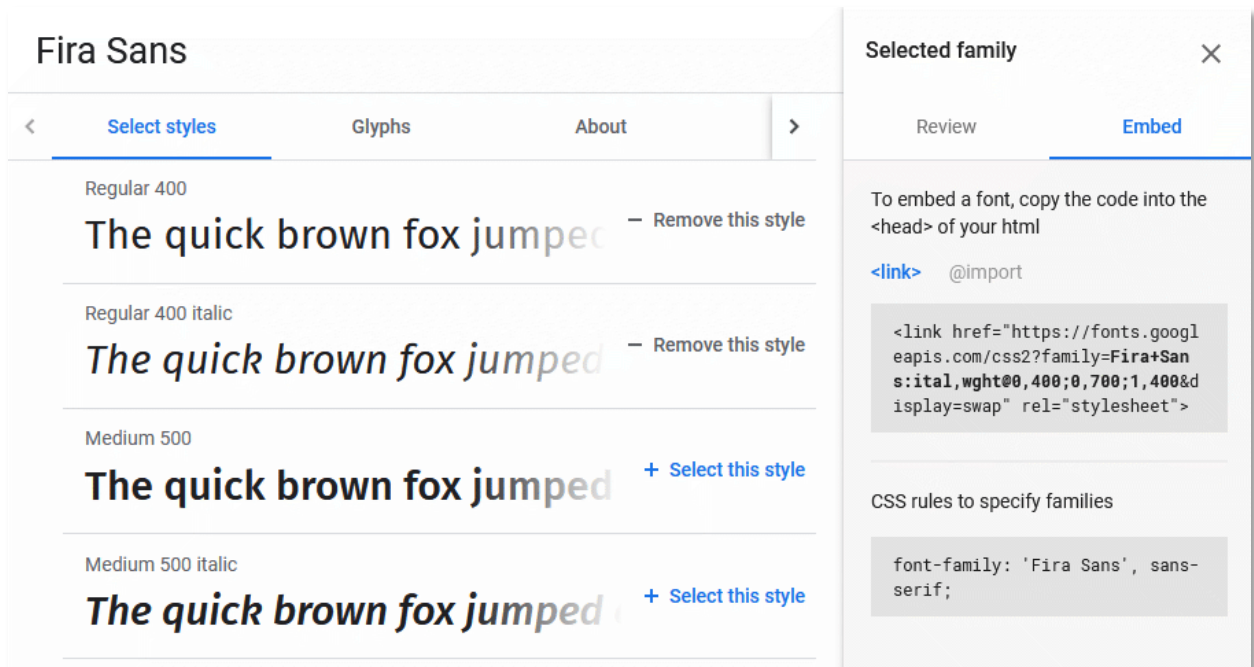
Don't forget that to use Google Fonts on your web pages, you must link to the fonts using a `<link>` element in HTML and then specify the font in CSS using the `font-family` property. Fortunately, Google Fonts makes this very easy. All you need to do is select the fonts you want to use, click the "Embed" tab and the code snippets are provided for you. You'll have to add additional font fall-backs to your font stack, using "web safe" fonts, but that's about it.

If you'd like to experiment more with web fonts, use the CSS Workshop files. The Google fonts service is a good option to begin your exploration of web fonts.

Course materials: Design for Web Content

Take care to ensure your CSS is well organised and use comments to create section headings. It's typical to maintain a separate section for your typographic rules:

```css
/*======= TYPOGRAPHY =======*/
body {
  font-family: "Lato", Calibri, Arial, Helvetica, sans-serif;
  font-weight: 400;
  color: #090909;
  background-color: #e5edf6;
}
h1, h2 {
  font-family: "Playfair Display", Georgia, "Times New Roman", Times, serif;
  color: #194a82;
}
h1 {
  font-size: 4.0em;
}
h2 {
  font-size: 2.4em;
  line-height: 1.2;
  margin-top: 1.8em;
}
p, li {
  font-size: 1.2em;
  line-height: 1.4em;
  margin-top: 0.4em;
}
p + p {
  margin-top: 1.6em;
}
```

Minimise your CSS code by taking advantage of font property inheritance. In theory, you shouldn't need to use the `font-family` property more than twice on a simple site, once as a default for body and once as an override for headings.

Top Tips
When designing your type, avoid using colour initially. Select an appropriate typeface for the content. Focus on the balance of text and whitespace, the contrast between headings and body text. Carefully consider the line height of your body text to provide a balanced visual density and good readability. Consider the line length (measure), and vertical rhythm. Spacing is crucial, especially the space between headings and paragraphs. Headings should be visually associated with the text they belong to. Visual hierarchy is also critical in allowing readers to understand document structure.

Using the next-sibling combinator
The next-sibling combinator (also referred to as the adjacent sibling selector) can be incredibly useful for typographic design. It selects only elements that immediately follow the specified element. For example, you can select all paragraphs that immediately follow another paragraph by using `p + p`. This has the effect of selecting all paragraphs except the first. This is useful if you are setting the space *between* paragraphs using `margin-top`. See the example above.

Course materials: Design for Web Content

CSS has [many selector types](#) and not just the [type selector](#) that we've been using so far. We'll cover other selector types in subsequent weeks, but for this exercise, see if you can style the content using only type, next-sibling combinator, and [descendant combinator](#) selectors. In other words, do not use classes or IDs.

For your blog
Task 2: Find 3 examples of sites with beautiful typography and explain why you think they work. Add any useful references for web typography to your blog.

Course materials: [Design for Web Content](#)