CSS Typography Review

Introduction to Typography

- **Definition**: Typography is the art of choosing the right fonts and format to make text visually appealing and easy to read. "Type" refers to how the individual characters are designed and arranged.
- **Typeface Definition**: A typeface is the overall design and style of a set of characters, numbers, and symbols. It's like a blueprint for a family of fonts.
- **Font Definition**: A font is a specific variation of a typeface with specific characteristics, such as size, weight, style, and width.
- **Serif Typeface**: This typeface has a classical style with small lines at the end of characters. Serif typefaces are commonly used for printed materials, like books.
- **Sans Serif Typeface**: This typeface has a more modern look, without the small lines at the end of characters. Sans Serif typefaces are commonly used in digital design because they are easy to read on screen. Some examples include Helvetica, Arial, and Roboto.
- **Font Weight**: This is the thickness of the characters, including light, regular, bold, and black.
- **Font Style**: This is the slant and orientation of the characters, like italic and oblique.
- **Font Width**: This is the horizontal space occupied by characters, like condensed and expanded.
- **Baseline**: This is the imaginary line on which most characters rest.
- **Cap Height**: This is the height of uppercase letters, measured from the baseline to the top.
- **X-height**: This is the average height of lowercase letters, excluding ascenders and descenders.
- **Ascenders**: These are the parts of lowercase letters that extend above the x-height, such as the tops of the letters 'h', 'b', 'd', and 'f'.
- **Descenders**: These are the parts of lowercase letters that extend below the baseline, such as the tails of 'y', 'g', 'p', and 'q'.

- **Kerning**: This is how space is adjusted between specific pairs of characters to improve their readability and aesthetics. For example, reducing the space between the letters A and V.
- **Tracking**: This is how space is adjusted between all characters in a block of text. It's essentially the distance between the characters. It affects how dense and open the text will be.
- **Leading**: This is the vertical space between lines of text. It's measured from the baseline of one line to the baseline of the next line.
- **Best Practices with Typography**: You should choose clear and simple fonts to make your designs easy to understand. This is particularly important for the main text of your website. Users are more likely to engage with your content if the font is easy to read. You should use two or three fonts at most to create a visual consistency. Using too many fonts can make the text more difficult to read and weaken your brand's identity. This can also impact the user experience by increasing the load time of the website. You can use font size to create a visual hierarchy for headings, subheadings, paragraphs, and other elements. For example, the main heading on a webpage should have a larger font, followed by subheadings with smaller font sizes. This will give every element in the hierarchy a specific font size that helps users navigate through the structure visually.

CSS font-family Property

- **Definition**: A font family is a group of fonts that share a common design. All the fonts that belong to the same family are based on the same core typeface but they also have variations in their style, weight, and width. You can specify multiple font families in order of priority, from highest to lowest, by separating them with commas. These alternative fonts will act as fallback options. You should always include a generic font family at the end of the font-family list.

Example Code

```
font-family: Arial, Lato;
```

- **Common Font Families**: Here are some common font families used in CSS: serif, sans-serif, monospace, cursive, fantasy

Web Safe Fonts

- **Definition**: These fonts are a subset of fonts that are very likely to be installed on a computer or device. When the browser has to render a font, it tries to find the font file on the user's system. But if the font is not found, it will usually fall back to a default system font.

- **Web Safe Fonts**:
    - Sans-serif fonts are commonly used for web development because they don't have small "feet" or lines at the end of the characters, so they're easy to read on screen. Some examples of web-safe sans-serif fonts are: Arial, Verdana, and Trebuchet MS.
    - In contrast, serif fonts do have small "feet" at the end of the characters, so they're commonly used for traditional print. Web safe serif fonts include: Times New Roman and Georgia.

At-rules and the @font-face At-rule

- **Definition**: At-rules are statements that provide instructions to the browser. You can use them to define various aspects of the stylesheet, such as media queries, keyframes, font faces, and more.
- **@font-face**: This allows you to define a custom font by specifying the font file, format, and other important properties, like weight and style. For the @font-face at-rule to be valid, you also need to specify the src property. This property contains references to the font resources.

Example Code

```
@font-face {
 font-family: "MyCustomFont";
 src: url("path/to/font.woff2"),
    url("path/to/font.woff"),
    url("path/to/font.otf");
}
```

- **Font Formats**: For each font resource, you can also specify the format. This is optional. It's a hint for the browser on the font format. If the format is omitted, the resource will be downloaded and the format will be detected after it's downloaded. If the format is invalid, the resource will not be downloaded. Possible font formats include collection, embedded-opentype, opentype, svg, truetype, woff, and woff2.

Example Code

```
@font-face {
 font-family: "MyCustomFont";
 src: url("path/to/font.woff2") format("woff2"),
    url("path/to/font.otf") format("opentype"),
    url("path/to/font.woff") format("woff");
}
```

- **woff and woff2**: woff stands for "Web Open Font Format." It's a widely used font format developed by Mozilla in collaboration with Type Supply, LettError, and other organizations. The difference between woff and woff2 is the algorithm used to compress the data.
- **OpenType**: This is a format for scalable computer fonts developed by Microsoft and Adobe that allows users to access additional features in a font. It's widely used across major operating systems.
- **tech()**: This is used to specify the technology of the font resource. This is optional.

Example Code

```
@font-face {
 font-family: "MyCustomFont";
 src: url("path/to/font.woff2") format("woff2"),
    url("path/to/font.otf") format("opentype") tech(color-COLRv1),
    url("path/to/font.woff") format("woff");
}
```

Working With External Fonts

- **Definition**: An external font is a font file that is not included directly within your project files.They're usually hosted on a separate server. To include these external fonts inside your project, you can use a link element or @import statement inside your CSS.
- **Google Fonts**: This is a Google service that offers a collection of fonts, many of which are designed specifically for web development.

Here is an example using the link element:

Example Code

```
<link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=swap" rel="stylesheet">
```

Example Code

```
.roboto-thin {

  font-family: "Roboto", sans-serif;

  font-weight: 100;

  font-style: normal;

}
```

Here is an example using the @import statement:

Example Code

```
@import url('https://fonts.googleapis.com/css2?family=Roboto:ital,wght@0,100;0,300;0,400;0,500;0,700;0,900;1,100;1,300;1,400;1,500;1,700;1,900&display=swap');
```

- **Font Squirrel**: This is another popular resource that you can use for adding custom external fonts to your projects.

text-shadow Property

- **Definition**: This property is used to apply shadows to text. You need to specify the X and Y offset, which represent the horizontal and vertical distance of the shadow from the text, respectively. These values are required. Positive values of the X offset and Y offset will move the shadow right and down, respectively, while negative values will move the shadow left and up.

Example Code

```
p {

  text-shadow: 3px 2px;

}
```

- **Shadow Color**: You can also customize the color of the shadow by specifying this value before or after the offset. If the color is not specified, the browser will determine the color automatically, so it's usually best to set it explicitly.

Example Code

```
p {
 text-shadow: 3px 2px #00ffc3;
}
```

- **Blur Radius**: The blur radius is optional but will make the shadow look a lot smoother and more subtle. The default value of the radius blur is zero. The higher the value, the bigger the blur, which means that the shadow becomes lighter.

Example Code

```
p {
 text-shadow: 3px 2px 3px #00ffc3;
}
```

- **Applying Multiple Text Shadows**: The text can have more than one shadow. The shadows will be applied in layers, from front to back, with the first shadow at the top.

Example Code

```
p {
 text-shadow:
  3px 2px 3px #00ffc3,
  -3px -2px 3px #0077ff,
  5px 4px 3px #dee7e5;
}
```