Assignment 2

LIB2024: Internet Applications

Martha Kohler
040864508
March 10, 2023

Contents

Introduction

The purpose of this assignment was to recreate a web page styled with CSS, requiring layered images, backgrounds, text wrapping, and grids. An external CSS stylesheet was linked to the HTML file in the metadata to format the page content.

Link to GitHub

https://github.com/kohl0028/kohl0028.github.io/tree/main/lib2024/Assignment%202%2 0Martha%20Kohler

Elements and Tricks

HTML: Selectors for CSS

The use CSS selectors was required to specify which parts of the HTML content to be styled, for example:

```
<div class="wrap">
    <div class="col1">
```

```
.wrap {
  max-width: 1280px;
```

For example, <class="wrap"> were used inside the <div> and identified in the stylesheet to format the text. ID could also have been used in the situations where formatting was unique to a single element, however for this assignment the use of class seemed sufficient. In a more complex application, ID would be helpful as well to remain organized.

HTML: Headers

For this document, there were 3 clear sections of formatting, so I used headers to add a title to each section of the page, using <h1>, <h2>, and <h3>. I was able to add borders and format the size, font colour and alignment of the text in CSS for each respective element. For <h1>, I assigned it a class "center" and applied the styling:

```
<section>
    <h1 class="center">DOG BREEDS</h1>
        <div class="wrap">
            <div class="col1">
            <img class="img" src="dogs1.jpg"
```

```
.center {
    color: IndianRed;
    text-align: center;
    border: 5px solid IndianRed;
    padding: 10px;
    margin: auto;
    font-family: Verdana, "sans serif";
    font-size: 35px;
    max-width: 1280px;
}
```

CSS: Float, Clear, Shape

To force the text content to wrap around the circle effectively, the float and clear functions were used extensively. This section was tricky and I used a template found on Codepen to help configure the styling (O'Brien). The text content was copied from Wikipedia and American Kennel Club and arranged into 2

columns inside <div class=wrap>. The text was styled with floats to the left and right, margins, and padding to leave white space evenly around the circular centre image. Within <p>, I added additional line breaks using <br> between paragraphs to add white space and visual appeal and make the text easier to read.

```
<div class="inner">
        <p id="first-parag">A dog breed is a particular strain
urposefully bred by humans to perform specific tasks, such as
```

```
.col1,
.col2 {
  padding: 10px;
  text-align: justify;
  width: calc(50% + 220px);
}
```

The shape of the circle was accomplished using border-radius and shape-outside. The image size was assigned in the HTML so it remain statics as the window size changes. The image also floats along with the text to maintain margins and not overlap if the screen size changes.

```
<div class="col2">
        <img class="img" src="dogs1.jpg" width="400" height ="400" alt ="Dogs Various">
<div class="inner">
```

```
.img {
  float: right;
  clear: right;
  margin: 20px 0 20px 20px;
  border-radius: 50%;
  border: 10px solid #fff;
  shape-outside: circle(50%);
}
```

```
.col2 img {
  float: left;
  margin: 20px 20px 20px 0;
  clear: left;
}
.col1 .inner {
  margin-right: 220px;
}
```

CSS: Position Property & Z-index

Absolute and relative positioning were key to the success to layer images in the middle watermark section, as well as Z-index. It was tough to centre the images and stack them simultaneously. I tried to apply the watermark image as a background initially but wasn't having success moving it to the centre of the page or rotating it as the example showed. I also didn't know how to create repeating initials using text content so I used a wallpaper image and adjusted the opacity down to achieve the desired affect. This <div> took advantage of the parent-child relationship of elements within a div to achieve the desired effect. Here is the HTML:

```
<div class="parent">
        <img class="child" src="lab-transparent.png" alt="Labrador Retriever">
        <img class="watermark" src="bone-paw-background.png" alt="Watermark Image">
</div>
```

To style the elements, the container <div> specified the contents to align centre and also have an auto margin to allow the image to scale to whatever window size.   Each image element was then styled separately so they could be sized and stacked appropriately. The top image was given absolute position, block display to take the full width of the page and not allow the other image to sit beside it. A Z-index of 2 was assigned to ensure it would stay at the front. The watermark image was assigned a relative position and flex display so it scales with the page, and rotated to achieve the angled effect and diamond shape shown in the example. The CSS coding is as follows:

```
.child {
    position: absolute;
    display: block;
    margin-left: auto;
    margin-right: auto;
    width: 50%;
    z-index: 2;
}
```

```
.watermark {
    position: relative;
    display: flex;
    opacity: 0.2;
    rotate: -45deg;
    z-index: 1;
}
```

HTML: <hr> Element

The <hr> element was used to insert the dashed line break. According to W3 Schools, it is used to "define thematic changes in the content" which is appropriate; I needed the line to run the width of the page and provide a visual break between sections. In CSS, I applied dashed border styling, color and width, as well as a flex display to make it responsive to the page size:

```
</section>

<hr>|

<section>
```

```
hr {
    display: flex;
    border-style: dashed;
    color: IndianRed;
    border-width: 5px;|
}
```

CSS: Display Property, Object-Fit, Grid

This was the aspect of the assignment I had the most trouble with. I had a lot of trouble getting my images to fall into the correct 3x2 grid orientation, even when I (thought) I translated code over exactly from online templates.

In order to achieve this layout, I put images and captions inside 6 repeating <figure> attributes, which were inside a <div> to create the grid layout, which was then nested within another <div> to contain it all and apply a background:

```
<div class="container2">
    <div class="gallery">
        <figure class="gallery__item gallery__item--1">
            <img src="grid1.png" class="gallery__img" alt="Fre
            <figcaption class>2. French Bulldog</figcaption>
        </figure>
```

For the outermost div, it was styled with a blue background, padding and margins. In the nested div, the content was assigned a grid layout and specified the rows and columns as follows:

```
.container2 {                      .gallery {
    width: 90%;                        display: grid;
    margin: 2rem auto;                 grid-template-columns: repeat(3, 1fr);
    padding: 20px;                     grid-template-rows: repeat(2, 1fr);
    background-color: steelblue;       grid-gap: 5px;
}                                  }
```

The gallery items were then formatted, as well as the captions:

```
.gallery__img {                    figcaption {
    width: 100%;                       background: rgba(0, 0, 0, 0.5);
    height: 100%;                      color: white;
    object-fit: contain;               font-family: Verdana, sans-serif;
    display: block;                    font-size: 20px;
    background-color: lightsteelblue;  padding: .4em 0;
}                                      position: absolute;
                                       bottom: 0;
figure.gallery__item {                 text-align: center;
    display: inline-block;             width: 100%;
    position: relative;            }
}
```

Finally, each image was assigned a position in the grid. This was for the top left image; each of the other are arranged similarly.

```
.gallery__item--1 {
    grid-column-start: 1;
    grid-column-end: 2;
    grid-row-start: 1;
    grid-row-end: 2;
}
```

Search Engine Optimization & User Experience

There are multiple ways in which this web page meets SEO and UX standards, as well as accessibility standards. In the header, I included metadata like a title tag, description, keywords, and content tags to

```
<head>
    <meta charset="UTF-8">
    <meta name="description" content="All about dog breeds">
    <meta name="keywords" content="dogs, dog breeds, popular dogs, Kennel Club">
    <meta name="author" content="Martha Kohler">
    <link rel="shortcut icon" href=pawprint.png>
    <title>Dog Breeds</title>
    <link href="layout.css" rel="stylesheet" type="text/css">
</head>
```

ensure the page would be found on a search and relevant information would be displayed:

In the body, I kept header tags to a minimum and didn't repeat the use of <h1> (which is treated as a title). My headings were relevant to the following content. To meet accessibility standards, I included image alt text in case the images couldn't render properly, or to accommodate those using assistive devices like screen readers. I tried to use colors with good contrast between text and background, chose sans-serif fonts, and maintained a nice white balance and spacing so they page is visually

appealing and easily readable. Finally, all of the main elements are formatted to scale to width to accommodate most screen sizes without needing to scroll sideways to access content.

Conclusion

Styling with CSS allows for much more flexibility and creativity when designing web pages, but it also makes the process that much more complex! In the end, I feel I was able to accurately recreate the sample page. I found a lot of help online looking at templates to help me achieve the trickier elements of the page.

References

Barysevichm, Aleh. (2020, March 27). *8 of the Most Important HTML Tags for SEO.*
https://www.socialmediatoday.com/news/8-of-the-most-important-html-tags-for-seo/574987/

Codepen.(n.d.) *Image Gallery with Caption - html5-Tag, figure figcaption – responsive.*
https://codepen.io/ambassador/pen/XLvjVQ

Computer Hope. (2021, December 30). *How to center text in HTML*.
https://www.computerhope.com/issues/ch001474.html

Dog breed. (2023, 7 March). In *Wikipedia*.
https://en.wikipedia.org/w/index.php?title=Dog_breed&oldid=1143363389

Fitzgerald, Anna. (2022, 16 May). *How to Bold, Italicize & Format Text in HTML*. Hubspot.
https://blog.hubspot.com/website/how-to-bold-in-
html#:~:text=To%20italicize%20the%20text%20in,property%20set%20to%20%E2%80%9Citalic.
%E2%80%9D

FreeCodeCamp. (2018, 27 December) How to create an image gallery with CSS Grid.
https://www.freecodecamp.org/news/how-to-create-an-image-gallery-with-css-grid-
e0f0fd666a5c/

Freepik. (n.d.) *Pawprint icon*. https://www.flaticon.com/free-icon/pawprint_1076877

O'Brien, Paul. (n.d.) *Shape-outside centred image*. Codepen.
https://codepen.io/paulobrien/pen/RMrrBd

Outside. (2012, 23 October). *The Most Popular Dog Breeds in the U.S.* Retrieved from
https://www.outsideonline.com/wp-content/uploads/migrated-images_parent/migrated-
images_4/intro-slide_ph.jpg\

PNG All. (n.d.) *Beagle dog transparent.* https://www.pngall.com/beagle-png/download/39350

PNG All. (n.d.) *German Shepherd Dog PNG Download Image*. https://www.pngall.com/german-
shepherd-png/download/39407

PNG All. (n.d.) *Golden French Bulldog*. https://www.pngall.com/french-bulldog-png/download/39325

PNG All. (n.d.) Golden Retriever Dog Transparent. https://www.pngall.com/golden-retrievers-
png/download/39124PNG All. (n.d.) *Labrador Retriever PNG Photo*.
https://www.pngall.com/labrador-retriever-png/download/39460

PNG All. (n.d.) *Poodle PNG Free Download* https://www.pngall.com/poodle-png/download/43512

PNGitem. (n.d.) *Bulldog Png Transparent Images - English Bulldog Transparent Background, Png
Download*. https://www.pngitem.com/middle/TbRxwx_bulldog-png-transparent-images-
english-bulldog-transparent-background/

Pngtree. (n.d.) *Cute Dog Tiled Background Bone Claw Print.* https://pngtree.com/freebackground/cute-dog-tiled-background-bone-claw-print_1349644.html

Reisen, Jan. (2022, 15 March). *Most Popular Dog Breeds of 2021.* American Kennel Club. https://www.akc.org/expert-advice/dog-breeds/most-popular-dog-breeds-of-2021/

Unwin, Sean T. (n.d.) *3x3 grid with Flex*. Codepen. https://codepen.io/seantunwin/pen/zxEqad

W3 Schools. (n.d.) *CSS z-index Property*. https://www.w3schools.com/cssref/pr_pos_z-index.php

W3 Schools (n.d.) *How TO - Center Images* *https://www.w3schools.com/howto/howto_css_image_center.asp*

W3 Schools.(n.d.)*HTML <hr> Tag.* https://www.w3schools.com/tags/tag_hr.asp

W3 Schools. (n.d.). *HTML <meta> Tag.* https://www.w3schools.com/tags/tag_meta.asp