Tania Sacoto and Katelyn Villacres
Matthew Fried
CSCI 355: Internet and Web Technologies
February 28, 2023

Project 1: Gray Paper

Home Page

1. Navbar

HTML:

```
<!--- NAVBAR -->
       <nav class = "nav">
           <div class = "container">
               <h1><a href = "index.html"><i class = "fas fa-mountain-sun"
id = "logo"></i>Adventure Peak</a></h1>
               <ul>
                   <li><a href = "index.html" class = "active">Home</a></li>
                   <li><a href = "programs.html">Programs</a></li>
                   <li><a href = "faq.html">FAQ</a></li>
                   <li><a href = "contact.html">Contact Us</a></li>
               </ul>
           </div>
       </nav>
```

The HTML provided here creates the basic framework for the site’s navbar. Inspired by Udemy’s Sticky Navbar project, we implemented a simple navbar with a mountain logo imported from Font Awesome and the words “Adventure Peak.” Everything within the navbar is contained within a div of class “container, so that the content could be padded and formatted easier in CSS. The different tabs are displayed as items of an unordered list, with the items “Home,” “Programs,” “FAQ,” and “Contact Us” being shown on every page of the website, and provide direct links to one another. Additionally, clicking on the mountain logo or the words “Adventure Peak” will take you directly to the “Home” page, regardless of which page you’re on. Finally, the class “active” indicates which page is active, and in the code above it indicates that the “Home” tab is currently active.

CSS:

```css
#logo {
   padding: 20px;
}

.nav {
   font-family: 'Shantell Sans';
   font-size: 20px;
   position: fixed;
   z-index: 1;
   background-color: #00896f;
   top: 0;
   left: 0;
   right: 0;
   transition: all 0.3s ease-in-out;
}

.nav .container {
   font-family: 'Shantell Sans';
   display: flex;
   justify-content: space-between;
   align-items: center;
   padding: 5px 0;
   transition: all 0.3s ease-in-out;
}

.nav ul {
   font-family: 'Shantell Sans';
   display: flex;
   list-style-type: none;
   align-items: center;
   justify-content: center;
}

.nav a {
   font-family: 'Shantell Sans';
   color: #fff;
   text-decoration: none;
   padding: 7px 25px;
   transition: all 0.3s ease-in-out;
}
```

The CSS above is inspired directly by the Sticky Navbar project. The #logo ID ties directly into the main icon of the page, the Font Awesome Mountain logo, and gives the logo padding on all sides. The first class, .nav, defines the font type “Shantell Sans,” which is the font that is used throughout the entire site. It also sets the font size, positioning of the navbar, the background color, and a small transition when you hover over any clickable elements of the navbar.

However, we encountered some trouble with the formatting, as some elements would go above the navbar, so we implemented a z-index of 1 to indicate that it would always be the topmost element. The remaining .nav classes handle the padding, alignment and justification of the list elements and their text. Specifically, the .nav .a class has a transition element that gives each list element a change in color when hovered on.

This section does not have any tied-in JavaScript.

2. Image carousel

HTML:

```
<!--- ROW 1: IMAGE CAROUSEL-->
       <div class = "wrapper">
           <h1 id = "adventure-text">Adventure Awaits at Adventure Peak Summer Camp!</h1>
           <div class = "carousel">
               <div class = "image-container" id = "imgs">
                   <img src = "assets/girl.jpg" alt = "image1">
                   <img src = "assets/ladders.jpg" alt = "image2">
                   <img src = "assets/artsandcrafts2.jpg" alt = "image3">
                   <img src = "assets/oldercampers.jpg" alt = "image4">
                   <img src = "assets/artsandcrafts3.jpg" alt = "image6">
                   <img src = "assets/tugofwar.jpg" alt = "image7">
                   <img src = "assets/artsandcrafts4.jpg" alt = "image8">
               </div>
           </div>
       </div>
```

The HTML for this section provides the basic framework for the image carousel. This section is also directly inspired by Udemy, and provides a carousel of various images of camp-related activities. The entire image carousel is wrapped in a div tag with a class of “wrapper.” It then has a heading, and another div tag with a class of “carousel.” Then, within the carousel, is another div tag with a class of “image-container” and id of “imgs” that holds all of the images. Within the carousel is 7 individual images, each with an individual alt tag of the image number.

CSS:

```
.wrapper{
   font-family: 'Shantell Sans';
   position: relative;
   text-align: center;
```

```css
   justify-content: center;
   padding: 20px;
}

.carousel{
   margin-left: auto;
   margin-right: auto;
   box-shadow: 2px 2px 5px rgba(0,0,0,0.3);
   width: 700px;
   height: 500px;
   overflow: hidden;
}

img{
   width: 700px;
   height: 500px;
   object-fit: cover;
}

.image-container{
   display: flex;
   transform: translateX(0);
   transition: transform 0.5s ease-in-out;
}
```

The CSS for this section allows the image carousel to look uniform rather than a list of jumbled pictures. The entire wrapper class, which contains text and the image carousel, is aligned centrally and is given a padding of 20px. The carousel class is given a width of 700px and height of 500px, a box shadow, is centered, and hides any overflowing content. Each img tag is given the same width and height as the carousel. Finally, the .image-container class is given a flexbox display, meaning that none of the other images will be displayed. The transform value is initialized at translateX(0), meaning that it will slide along the X-axis and will start at the first image (0px). The transition is set to create a sliding effect as the carousel moves from one image to the next, and is emphasized in the implementation of the JavaScript.

JavaScript:

```javascript
const imgs = document.getElementById('imgs');
const leftBtn = document.getElementById('left');
const rightBtn = document.getElementById('right');

const img = document.querySelectorAll('#imgs img');

console.log(img);
```

```
let idx = 0;

let interval = setInterval(run, 4000);

function run() {
   idx++;
   changeImage();
}

function changeImage(){
   if(idx > img.length - 1){
       idx = 0;
   } else if(idx < 0){
       idx = img.length - 1;
   }

   imgs.style.transform = `translateX(${-idx * 700}px)`
}
```

The purpose of the JavaScript is to shift from one image to the next. Since every image is 700px long, we shift in increments of 700px. If we reach the end of the images, it will circulate back to the first image in the list.

3. Expanding cards

HTML:

```
<!-- ROW 3: EXPANDING CARDS -->
       <div class = "wrapper">
           <h2 style = "text-align: center;">What We Offer</h2>
           <div class="container">
               <div class="panel" style = "background-image: url(assets/friends.jpeg)">
                   <h3>Lifelong Memories</h3>
               </div>

               <div class="panel" style="background-image: url('assets/tugofwar.jpg')">
                   <h3>Friendly Competition</h3>
               </div>

               <div class="panel" style="background-image: url('assets/parachute.jpg')">
                   <h3>Fun and Exciting Activities</h3>
               </div>

               <div class="panel" style="background-image: url('assets/sailing.jpg')">
                   <h3>Survival Skills</h3>
                </div>
               <div class="panel active" style="background-image: url('assets/camp.jpeg')">
                   <h3>Leadership Opportunities</h3>
                </div>
           </div>
       </div>
```

The HTML above creates the different expanding cards, inspired by the Expanding Cards Udemy project. Each image is contained within a div of class “panel”, and uses inline CSS to set the background image using images within the assets folder. The last image is given a class of “active,” which will tell the JavaScript code which card is expanded.

CSS:

```
.panel{
   background-size: auto 100%;
   background-position: center;
   background-repeat: no-repeat;
   height: 80vh;
   border-radius: 50px;
   color: #fff;
   cursor: pointer;
   flex: 0.5;
   margin: 10px;
   position: relative;
   transition: flex 0.7s ease-in;
}

.panel h3{
   font-size: 24px;
   position: absolute;
   bottom: 20px;
   left: 20px;
   margin: 0;
   opacity: 0;
}

.panel.active{
   flex: 5;
}

.panel.active h3{
   opacity: 1;
   border-radius: 50px;
   transition: opacity 0.3s ease-in 0.4s;
}

@media(max-width: 480px){
   .container{
       width: 100vw;
```

```
    }

    .panel:nth-of-type(4), .panel:nth-of-type(4){
        display: none;
    }
}
```

The CSS above formats the expanding cards so that, when a panel is active, it is larger than the rest of the cards. It also creates rounded edges, centers the entire section, and gives each panel a smooth transition between active and inactive.

JavaScript:

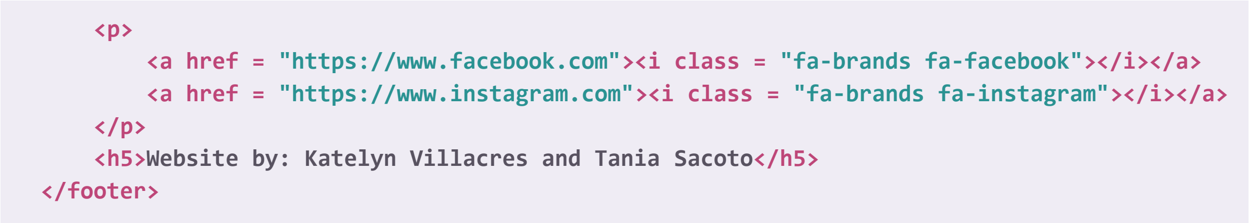
```
const panels = document.querySelectorAll('.panel')

panels.forEach(panel => {
    panel.addEventListener('click', () => {
        removeActiveClasses()
        panel.classList.add('active')
    })
})

function removeActiveClasses() {
    panels.forEach(panel => {
        panel.classList.remove('active')
    })
}
```

The JavaScript makes the expanding cards section more dynamic, by waiting for a click event listener to indicate whether or not to remove the active class tag from the other cards, then make the most recently clicked item the active class. This, combined with the CSS, makes for a smooth transition between elements in the expanding cards row.

4. Footer

HTML:

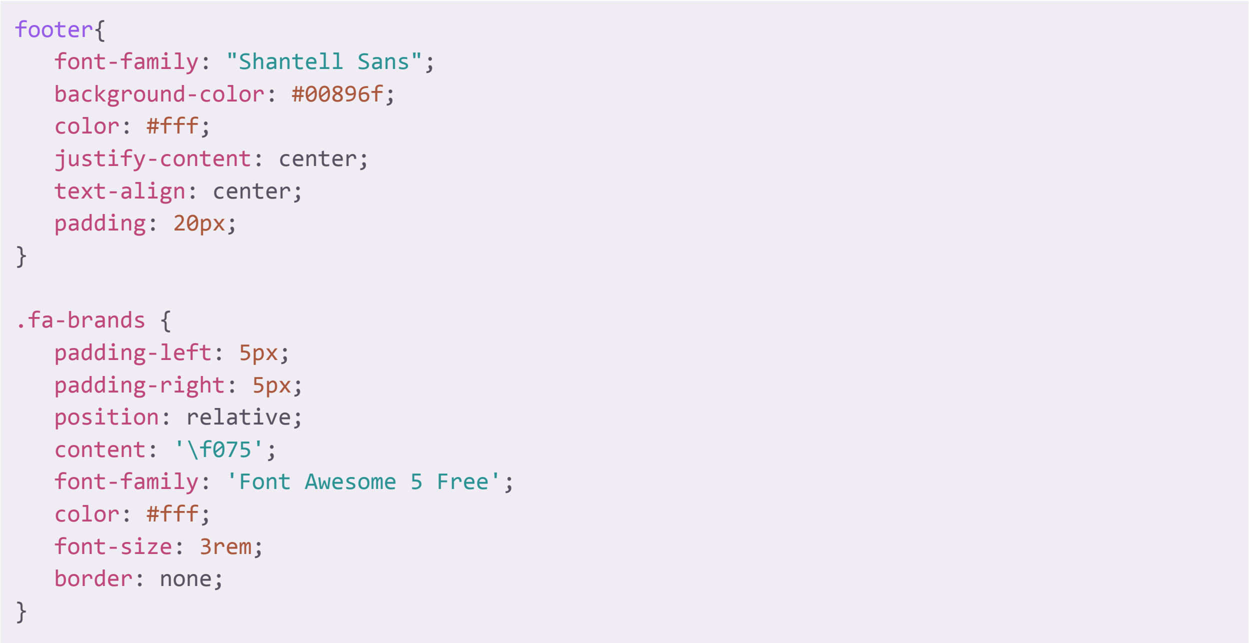
```
 <!-- Footer -->
    <footer>
      <h3>Adventure Peak</h3>
      <p>Sign Up Today!</p>
```



```
      <p>
          <a href = "https://www.facebook.com"><i class = "fa-brands fa-facebook"></i></a>
          <a href = "https://www.instagram.com"><i class = "fa-brands fa-instagram"></i></a>
      </p>
      <h5>Website by: Katelyn Villacres and Tania Sacoto</h5>
  </footer>
```

Similarly to the Navbar section, the footer section is consistent with every page of the website. The HTML creates a simple, yet effective footer, prompting users to sign up for Adventure Peak. The name of our site is wrapped in an h3 tag, while the “Sign Up Today!” prompt is wrapped in a p tag. Within another p tag are two icons from Font Awesome: Facebook and Instagram. Our intention was to have the website link to the Facebook and Instagram pages of Adventure Peak, but since we don’t have any accounts set up, the icons currently link to the homepages of the respective social media platforms. Finally, at the very bottom of the footer is an h5 tag with the names of the site creators, Katelyn Villacres and Tania Sacoto.

CSS:



```
footer{
   font-family: "Shantell Sans";
   background-color: #00896f;
   color: #fff;
   justify-content: center;
   text-align: center;
   padding: 20px;
}

.fa-brands {
   padding-left: 5px;
   padding-right: 5px;
   position: relative;
   content: '\f075';
   font-family: 'Font Awesome 5 Free';
   color: #fff;
   font-size: 3rem;
   border: none;
}
```

The CSS for this section is divided into two groups. The first is the footer tag, which uses the same brand font for the text, sets the background color to a forest-green and text color to white, justifies the content so that everything is centered, and gives the footer a padding of 20px. The .fa-brands class handled the two imported icons from Font Awesome, which is the font-family of the section. This section provides padding to each icon, sets the icon color to white, removes the borders from each icon so that it’s just the images shown, and makes the icons larger.

There is no JavaScript for this section.

Courses Page

HTML:

```
<body>
    <!--- NAVBAR -->
    <nav class = "nav">
      <div class = "container">
          <h1><a href = "index.html"><i class = "fas fa-mountain-sun" id =
"logo"></i>Adventure Peak</a></h1>
          <ul>
              <li><a href = "index.html" class = "active">Home</a></li>
              <li><a href = "programs.html">Programs</a></li>
              <li><a href = "courses.html">Courses</a></li>
              <li><a href = "faq.html">FAQ</a></li>
              <li><a href = "contact.html">Contact Us</a></li>
          </ul>
      </div>
  </nav>
```

Within this snippet of code we have the navbar defined in the HTML of the courses page. It is one of the elements that is consistently used within all of our pages. This is one of the udemy elements that we decided would be a good idea to implement in our website. All the links to the different pages are within an unordered list.

```
<!--- Title -->
    <header>
      <h1>Courses</h1>
    </header>
    <!--- Information -->
    <div class = "boxContainer1 box">
      <div class = "boxC11">
        <h2>First Aid</h2>
        <p>Our first aid course for teens covers a range of basic first aid skills to help
them handle common injuries and illnesses that may occur while at camp. Here is an outline
of what the course could include:</p>
          <ul>
            <li>Introduction to first aid: Introducing the concept of first aid and its
importance in emergency situations.</li>
            <li>Assessing the situation: Teach teens how to assess the situation before
offering any help. Explain the importance of safety first, and how to check the area for any
potential hazards or dangers.</li>
            <li>Bleeding and wound care: Teach teens how to handle bleeding and wound care,
including how to apply pressure to stop bleeding, how to clean and dress a wound, and how to
recognize signs of infection.</li>
            <li>Burns and heat-related illnesses: Explain how to recognize and treat burns,
including how to cool the burn with running water and how to apply a sterile dressing. Cover
heat-related illnesses such as heat exhaustion and heat stroke and the importance of staying
hydrated.</li>
```

```
            <li>And so much more...</li>
          </ul>
      </div>
      <div class = "boxC12">
        <img src = "assets/first aid.jpg" alt = "first aid">
      </div>
    </div>
    <div class = "boxContainer2 box">
      <div class = "boxC21">
        <img src = "assets/cpr.jpg" alt = "cpr">
      </div>
      <div class = "boxC22">
        <h2>CPR</h2>
        <p>Our CPR course for teens is a crucial training to equip them with the skills to
respond in emergency situations where someone may need cardiopulmonary resuscitation (CPR).
Here is an outline of what the course could include:</p>
        <ul>
          <li>Introduction to CPR: Start by introducing the concept of CPR and its
importance in emergency situations.</li>
          <li>Assessing the situation: Teach teens how to assess the situation before
offering any help. </li>
          <li>Chest compressions: Cover the basics of chest compressions, including how to
position the person, how to place the hands and how to perform compressions. Demonstrate the
proper techniques and allow teens to practice on CPR dummies.</li>
          <li>Rescue breaths: Teach teens how to perform rescue breaths, including how to
tilt the head back and lift the chin to open the airway and how to give breaths using
mouth-to-mouth or a barrier device.</li>
          <li>And so much more...</li>
        </ul>
      </div>
    </div>
    <div class = "boxContainer1 box">
      <div class = "boxC11">
        <h2>Lifeguarding</h2>
        <p>Our lifeguard course for teens is an important training to equip them with the
skills necessary to maintain safety in and around the water. Here is an outline of what the
course could include:</p>
        <ul>
          <li>Introduction to lifeguarding: Start by introducing the concept of lifeguarding
and its importance in maintaining safety in and around the water.</li>
          <li>Water safety and rescue techniques: Teach teens about water safety rules, how
to recognize and respond to different water emergencies, and how to perform rescue
techniques such as reaching, throwing, and towing.</li>
          <li>Lifeguarding equipment: Teach teens about the different types of lifeguarding
equipment, such as rescue tubes, life jackets, and backboards, and how to properly use
them.</li>
          <li>Surveillance and prevention: Cover the importance of surveillance and
prevention, including how to recognize signs of distress, and how to communicate effectively
with other lifeguards and swimmers.</li>
          <li>And so much more...</li>
        </ul>
      </div>
      <div class = "boxC12">
        <img src = "assets/lifeguard.jpg" alt = "lifegaurd">
      </div>
    </div>
    <!-- Footer -->
    <footer>
      <h3>Adventure Peak</h3>
      <p>Sign Up Today!</p>
      <p>
          <a href = "https://www.facebook.com"><i class = "fa-brands fa-facebook"></i></a>
          <a href = "https://www.instagram.com"><i class = "fa-brands fa-instagram"></i></a>
      </p>
      <h5>Website by: Katelyn Villacres and Tania Sacoto</h5>
  </footer>
  <script src="courses.js"></script>
</body>
```

In the rest of the code, the actual content of the page is defined. With the use of multiple divs, we effectively incorporated the use of flexbox in order to display the content in a clean and clear way. The titles of the pages “cpr”, “lifeguarding” and “first aid” are shown in h2 tags. Information about the courses is explained in bullet points that outline the curriculum and images for the user to see.

CSS:

```
@charset "utf-8";
@import url('https://fonts.googleapis.com/css2?family=Shantell+Sans&display=swap');

#logo {
  padding: 20px;
}
body{
  margin: 0px;
  overflow-x: hidden;
}
header{
  margin-top: 50px;
  padding-top: 60px;
}
.nav {
  font-family: 'Shantell Sans';
  font-size: 20px;
  position: fixed;
  z-index: 1;
  background-color: #00896f;
  top: 0;
  left: 0;
  right: 0;
  transition: all 0.3s ease-in-out;
}

.nav .container {
  font-family: 'Shantell Sans';
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 5px 0;
  transition: all 0.3s ease-in-out;
}

.nav ul {
  font-family: 'Shantell Sans';
  display: flex;
  list-style-type: none;
  align-items: center;
  justify-content: center;
}

.nav a {
  font-family: 'Shantell Sans';
  color: #fff;
  text-decoration: none;
  padding: 7px 25px;
  transition: all 0.3s ease-in-out;
}
```

In the CSS of the courses page, we styled the navbar the very same way we did in the other pages. Here we were able to display the icon as well as the nav links on opposite sides of each other in a clear way.

```
header > h1{
  font-family: "Shantell Sans";
  text-align: center;
  font-size: 45px;
}
.box{
  transform: translateX(200%);
  transition: transform 0.4s ease;
}
.box:nth-of-type(even){
  transform: translateX(-200%);
}
.box.show{
  transform: translateX(0);
}
.boxContainer1 {
  display: flex;
  width: 100%;
}
.boxC11{
  flex-basis: 60%;
  height: 600px;
  background-color: #E38F2D;
  margin: 2px 2px 2px 0;
}
img{
  width: 60%;
  height: 500px;
  padding: 50px 80px 50px 120px;
}
.boxC12{
  flex-basis: 40%;
  height: 600px;
  margin: 2px 0 2px 2px;
  background-color: #E38F2D;
}
.boxContainer2 {
  display: flex;
  width: 100%;
}
.boxC22{
  flex-basis: 60%;
  height: 600px;
  background-color: #E38F2D;
  margin: 2px 0 2px 2px;
}
.boxC21{
  flex-basis: 40%;
  height: 600px;
  margin: 2px 2px 2px 0;
  background-color: #E38F2D;
}
.boxC11 > h2{
  font-family: "Shantell Sans";
  margin-left: 60px;
  margin-top: 60px;
  font-size: 30px;
  color:white;
}
.boxC11 > p, ul{
  font-family: "Shantell Sans";
  margin: 0px 60px 0px 60px;
  padding-top: 5px;
  font-size: 18px;
  color: white;
}
.boxC22 > h2{
  font-family: "Shantell Sans";
  margin-left: 60px;
  margin-top: 60px;
  font-size: 30px;
  color:white;
}
.boxC22 > p, ul{
  font-family: "Shantell Sans";
  padding-top: 5px;
  margin: 0px 60px 0px 60px;
  font-size: 18px;
  color: white;
}
footer{
  font-family: "Shantell Sans";
  background-color: #00896f;
  color: #fff;
  justify-content: center;
  text-align: center;
  padding: 20px;
}

.fa-brands {
  padding-left: 5px;
  padding-right: 5px;
  position: relative;
  content: '\f075';
  font-family: 'Font Awesome 5 Free';
  color: #fff;
  font-size: 3rem;
  border: none;
}
```

For the content within the page, we had the heading displayed in a h1 tag and the page's content in boxes. The font size for the h1 tag was overridden since the original size felt too small. The flex basis lets us properly size the 2 boxes for the image and the description. We also utilized code from udemy here that styled the content boxes as the user scrolled down the page.

```
@media (max-width: 800px){
  .boxContainer1{
    flex-wrap: wrap;
  }
  .boxC11{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxC12{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxContainer2{
    flex-wrap: wrap-reverse;
  }
  .boxC21{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxC22{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxC22 > p{
    font-size: 24px;
  }
  .boxC11 > p{
    font-size: 24px;
  }
}
@media (max-width: 600px){
  .boxC22 > p{
    font-size: 20px;
    padding-top: 20px;
  }
  .boxC11 > p{
    font-size: 20px;
    padding-top: 20px;
  }
}
```

In this code snippet, we tried to incorporate more responsive features in the website as the browser got smaller. This allowed the website to stay visually appealing and clear as the user continues to shrink the window. For example, the flex-basis is altered to fill the whole page rather than just 60% of the window and the flex box now has the ability to wrap and be displayed as a column.

JS:

```
const boxes = document.querySelectorAll('.box')
window.addEventListener('scroll', checkBoxes)
function checkBoxes(){
  const triggerBottom = window.innerHeight / 5 * 4
  boxes.forEach(box => {
    const boxTop = box.getBoundingClientRect().top
    if(boxTop < triggerBottom){
      box.classList.add('show')
    }
    else{
      box.classList.remove('show')
    }
  })
}
```

In the JavaScript file, we had used the code given in udemy for a slide animation. We have the event listener utilize the checkBoxes() function as the user scrolls through our page. This page is then called into the HTML page's body to be used on the content that is being acknowledged.

Programs Page

HTML:

```
<body>
     <!--- NAVBAR -->
     <nav class = "nav">
      <div class = "container">
          <h1><a href = "index.html"><i class = "fas fa-mountain-sun" id =
"logo"></i>Adventure Peak</a></h1>
          <ul>
              <li><a href = "index.html" class = "active">Home</a></li>
              <li><a href = "programs.html">Programs</a></li>
              <li><a href = "courses.html">Courses</a></li>
              <li><a href = "faq.html">FAQ</a></li>
              <li><a href = "contact.html">Contact Us</a></li>
          </ul>
      </div>
  </nav>
     <!--- Title -->
    <header>
      <h1>Programs</h1>
    </header>
    <!--- Information -->
    <div class = "boxContainer1 box">
      <div class = "boxC11">
        <h2>Canoeing</h2>
        <p>Canoeing is a wonderful activity for kids that involves exploring bodies of
water, building teamwork skills, and enjoying the great outdoors. During canoeing
excursions, camp counselors and instructors will typically guide kids on safe and enjoyable
routes through local waterways, pointing out interesting wildlife and natural features along
the way. This can provide a unique and educational experience for kids, helping them to
learn about the environment and the importance of conservation. </p>
      </div>
      <div class = "boxC12">
        <img src = "assets/canoeing.jpg" alt = "canoeing">
      </div>
    </div>
    <div class = "boxContainer2 box">
    <div class = "boxC21">
      <img src = "assets/soccer.jpg" alt = "soccer">
    </div>
    <div class = "boxC22">
      <h2>Sports</h2>
      <p>We offer a wide range of sports activities, including basketball, soccer, baseball,
flag football, tennis, and more. Campers may participate in friendly matches or games,
practice drills, and receive instruction and coaching from trained staff. sports can help to
build self-confidence and boost self-esteem. As children learn new skills and improve their
abilities, they may feel a sense of accomplishment and pride. Kids can learn the value of
teamwork, communication, and cooperation as they work with their peers to achieve common
goals.</p>
    </div>
    </div>
    <div class = "boxContainer1 box">
      <div class = "boxC11">
        <h2>Arts and Crafts</h2>
        <p>Our arts and crafts program provides a fun and creative outlet for
self-expression and learning new skills. Arts and crafts activities may include a wide range
of projects, such as painting, drawing, sculpture, printmaking, jewelry making, and more.
Campers may work individually or in groups, creating projects that reflect their own
interests and personalities. This can help to build friendships and a sense of community, as
children work together to create something they can be proud of.</p>
      </div>
      <div class = "boxC12">
        <img src = "assets/artsAndCrafts.jpg" alt = "arts and crafts">
      </div>
    </div>
    <div class = "boxContainer2 box">
      <div class = "boxC21">
        <img src = "assets/hiking.jpg" alt = "hiking">
      </div>
      <div class = "boxC22">
        <h2>Hiking</h2>
        <p>Hiking offers our pupils the opportunity to explore the outdoors, learn about
nature, and build physical endurance. At Adventure Peek, hiking involves guided trips
through local trails and parks, or overnight camping excursions where kids can experience
the great outdoors up close. During hikes, kids can learn about local wildlife and
ecosystems, as well as explore natural features like waterfalls, canyons, and forests.</p>
      </div>
    </div>
    <!-- Footer -->
    <footer>
      <h3>Adventure Peak</h3>
      <p>Sign Up Today!</p>
      <p>
          <a href = "https://www.facebook.com"><i class = "fa-brands fa-facebook"></i></a>
          <a href = "https://www.instagram.com"><i class = "fa-brands fa-instagram"></i></a>
      </p>
      <h5>Website by: Katelyn Villacres and Tania Sacoto</h5>
  </footer>
  <script src="programs.js"></script>
  </body>
```

Again in this page we are able to see the navbar coded at the start of the page. The rest of the code is where the actual content of the page is defined. Along with the courses page, we use multiple divs in order to implement the use of flexbox to display the content in a clean and clear way. The titles of the pages “hiking”, “arts and crafts”, “sports” and “canoeing” are shown in h2 tags. More in depth information about the courses is explained in the p tags below that. These paragraphs go in depth about what to expect from these activities and what they help students accomplish. The footer at the very bottom of the page includes our company's socials at the very bottom.

CSS:

```css
@charset "utf-8";
@import url('https://fonts.googleapis.com/css2?family=Shantell+Sans&display=swap');

#logo {
  padding: 20px;
}
body{
  margin: 0px;
  overflow-x: hidden;
}
header{
  margin-top: 50px;
  padding-top: 60px;
}
header > h1{
    font-family: 'Shantell Sans';
    text-align: center;
    font-size: 45px;
}
.nav {
  font-family: 'Shantell Sans';
  font-size: 20px;
  position: fixed;
  z-index: 1;
  background-color: #00896f;
  top: 0;
  left: 0;
  right: 0;
  transition: all 0.3s ease-in-out;
}

.nav .container {
  font-family: 'Shantell Sans';
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 5px 0;
  transition: all 0.3s ease-in-out;
}

.nav ul {
  font-family: 'Shantell Sans';
  display: flex;
  list-style-type: none;
  align-items: center;
  justify-content: center;
}

.nav a {
  font-family: 'Shantell Sans';
  color: #fff;
  text-decoration: none;
  padding: 7px 25px;
  transition: all 0.3s ease-in-out;
}
.box{
  transform: translateX(200%);
  transition: transform 0.4s ease;
}
.box:nth-of-type(even){
  transform: translateX(-200%);
}
.box.show{
  transform: translateX(0);
}
```

Here we can see the implementation of the JavaScript slide animation. As well as the very same styling of the navbar that we previously had. Our pages content comes in from opposite sides of the screen as shown in the .box:nth-of-type(even) class selector.

```
.boxContainer1 {
  display: flex;
  width: 100%;
}
.boxC11{
  flex-basis: 60%;
  height: 600px;
  background-color: #99BF72;
  margin: 2px 2px 2px 0;
}
img{
  width: 60%;
  height: 500px;
  padding: 50px 80px 50px 120px;
  }
.boxC12{
  flex-basis: 40%;
  height: 600px;
  margin: 2px 0 2px 2px;
  background-color: #99BF72;
}
.boxContainer2 {
  display: flex;
  width: 100%;
```

```
}
.boxC22{
  flex-basis: 60%;
  height: 600px;
  background-color: #99BF72;
  margin: 2px 0 2px 2px;
}
.boxC21{
  flex-basis: 40%;
  height: 600px;
  margin: 2px 2px 2px 0;
  background-color: #99BF72;
}
.boxC11 > h2{
  font-family: 'Shantell Sans';
  margin-left: 60px;
  margin-top: 60px;
  font-size: 35px;
  color:white;
}
.boxC11 > p{
  font-family: 'Shantell Sans';
  padding: 10px 0;
  margin: 0px 60px 0px 60px;
  font-size: 28px;
  color: white;
}
.boxC22 > h2{
  font-family: 'Shantell Sans';
  margin-left: 60px;
  margin-top: 60px;
  font-size: 35px;
  color:white;
}
.boxC22 > p{
  font-family: 'Shantell Sans';
  padding: 10px 0;
  margin: 0px 60px 0px 60px;
  font-size: 28px;
  color: white;
}
footer{
  font-family: "Shantell Sans";
  background-color: #00896f;
  color: #fff;
  justify-content: center;
  text-align: center;
  padding: 20px;
}

.fa-brands {
  padding-left: 5px;
  padding-right: 5px;
  position: relative;
  content: '\f075';
  font-family: 'Font Awesome 5 Free';
  color: #fff;
  font-size: 3rem;
  border: none;
}
```

In the CSS above we basically have the content centered and styled within the flex boxes. Some of the h tags were overridden in terms of font size. This is because we found the font to be too small for the title of the pages and wanted them to fit nicely within the page. We also have the footer styled to stay relative to the page.

```
 @media (max-width: 800px){
.boxContainer1{
    flex-wrap: wrap;
  }
  .boxC11{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxC12{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxContainer2{
    flex-wrap: wrap-reverse;
  }
  .boxC21{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxC22{
    flex-basis: 100%;
    margin: 2px 0 2px 0;
  }
  .boxC22 > p{
    font-size: 24px;
  }
  .boxC11 > p{
    font-size: 24px;
  }
}
@media (max-width: 600px){
  .boxC22 > p{
    font-size: 20px;
    padding-top: 20px;
  }
  .boxC11 > p{
    font-size: 20px;
    padding-top: 20px;
  }
}
```

In this code snippet, the responsive features from the courses page make a reappearance. We altered the font size of the content to have it actually fit the page nicely once the user has changed the browser size.

JS:

```
const boxes = document.querySelectorAll('.box')
window.addEventListener('scroll', checkBoxes)
function checkBoxes(){
  const triggerBottom = window.innerHeight / 5 * 4
  boxes.forEach(box => {
    const boxTop = box.getBoundingClientRect().top
    if(boxTop < triggerBottom){
      box.classList.add('show')
    }
    else{
      box.classList.remove('show')
    }
  })
}
```

In the JavaScript file, we used the code given in udemy again for a slide animation since both the programs page and courses page have generally the same layout. The function, checkBoxes(), is called as the user scrolls through our page.

Contact Page

HTML:

```
<body>
         <!--- NAVBAR -->
        <nav class = "nav">
            <div class = "container">
                <h1><a href = "index.html"><i class = "fas fa-mountain-sun" id = "logo"></i>Adventure Peak</a></h1>
                <ul>
                    <li><a href = "index.html" class = "active">Home</a></li>
                    <li><a href = "programs.html">Programs</a></li>
                    <li><a href = "courses.html">Courses</a></li>
                    <li><a href = "faq.html">FAQ</a></li>
                    <li><a href = "contact.html">Contact Us</a></li>
                </ul>
            </div>
        </nav>
        <!--- Title -->
        <header>
            <h1>Contact Us</h1>
            <h3>Thank you for reaching out to us. We will do our best to get back to you as soon as possible.</h3>
        </header>
        <hr class="solid">
        <br>
        <!--- Contact Form -->
        <div class = "contactForm">
            <form>
                <p>
                    <input class = "topBox1" type = "text" id = "firstName" placeholder = "First Name">
                    <input class = "topBox2" type = "text" id = "LastName" placeholder = "Last Name">
                </p>
                <p>
                    <input class = "email" type = "text" id = "email" placeholder = "Email">
                </p>
                <div class = "radioForm">
                    <div class = "title">
                        <h4>Choose One: </h4>
                   </div>
                   <div class = "radioButtons">
                        <div class = "buttons">
                           <input type="radio" name="radioOption" value="work with us" id = "Work">
                           <label class = "option" for="Work">Work with Us</label> <br>
                        </div>
                        <div class = "buttons">
                           <input type="radio" name="radioOption" value="Join the Camp" id = "join">
                           <label class = "option" for="join">Join the Camp</label> <br>
                        </div>
                        <div class = "buttons">
                           <input type="radio" name="radioOption" value="questions" id =
"questions">
                           <label class = "option" for="questions">Question</label> <br>
                        </div>
                    </div>
                </div>
                <p>
                    <textarea name="questionsComments" rows="6" cols="50" placeholder="Write
your question or comment here..."></textarea>
                </p>
                <div class = "submitButton">
                    <button>Submit</button>
                </div>
            </form>
        </div>

        <div class = "cards">
            <a class = "card1" href = "courses.html">
            <div class = "card">
                <div class = "card-header animated-bg" id = "header">
                  <img src = "assets/courses.jpg" alt = "courses">
                </div>
                <div class = "card-content">
                    <h3 class = "card-title animated-bg animated-bg-text" id =
"title">Courses</h3>
                    <p class = "class-excerpt" id = "excerpt">Come learn more about out
courses</p>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                </div>
            </div>
            </a>
            <a class = "card1" href = "programs.html">
            <div class = "card">
                <div class = "card-header animated-bg" id = "header">
                  <img src = "assets/programs.jpg" alt = "courses">
                </div>
                <div class = "card-content">
                    <h3 class = "card-title animated-bg animated-bg-text" id =
"title">Programs</h3>
                    <p class = "class-excerpt" id = "excerpt">Come learn more about out
programs</p>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                </div>
            </div>
            </a>
            <a class = "card1" href = "faq.html">
            <div class = "card">
                <div class = "card-header animated-bg" id = "header">
                     <img src = "assets/questions.jpg" alt = "courses">
                </div>
                <div class = "card-content">
                    <h3 class = "card-title animated-bg animated-bg-text" id =
"title">FAQ</h3>
                    <p class = "class-excerpt" id = "excerpt">Still have more questions?</p>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                    <span class = "animated-bg animated-bg-text"> &nbsp;</span>
                </div>
            </div>
            </a>
        </div>
        <script src="contact.js"></script>
    </body>
```

In the HTML file of the contact page, we again see the nav bar defined at the very top of the page and the footer defined at the bottom. Within this file we incorporated a contact form, that allows the user to have a way to contact our company. They have the option to join our camp, ask a question or work with us. They can pick only one radio button and then proceed to type in the text area. Once they are done, they can utilize the submit button. We also have incorporated cards from the udemy videos at the bottom of the page that encourage users to continue to explore our website.

CSS:

```
@charset "utf-8";
@import url('https://fonts.googleapis.com/css2?family=Shantell+Sans&display=swap');

#logo {
  padding: 20px;
}
body{
  margin: 0px;
}
header{
  margin-top: 50px;
  padding-top: 60px;
}
header > h1{
  font-family: "Shantell Sans";
  text-align: center;
  font-size: 45px;
```

```
}
h3{
  font-family: "Shantell Sans";
  text-align: center;
}
#excerpt, #excerpt1, #excerpt2{
  font-family: "Shantell Sans";
}
.nav {
  font-family: 'Shantell Sans';
  font-size: 20px;
  position: fixed;
  z-index: 1;
  background-color: #00896f;
  top: 0;
  left: 0;
  right: 0;
  transition: all 0.3s ease-in-out;
}

.nav .container {
  font-family: 'Shantell Sans';
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 5px 0;
  transition: all 0.3s ease-in-out;
}

.nav ul {
  font-family: 'Shantell Sans';
  display: flex;
  list-style-type: none;
  align-items: center;
  justify-content: center;
}

.nav a {
  font-family: 'Shantell Sans';
  color: #fff;
  text-decoration: none;
  padding: 7px 25px;
  transition: all 0.3s ease-in-out;
}
.solid{
  border-top: 2px solid #bbb;
  width: 50%;
  padding: auto;
}
.contactForm{
    display: flex;
    text-align: center;
    flex-direction: column;
    background-color: #00896f;
    border-radius: 15px;
    margin: auto;
    padding: 15px;
    width: 50%;
    height: 470px;
    box-sizing: border-box;
}
form{
    margin: auto;
    display: flex;
    flex-wrap: wrap;
    justify-content: center;
}
input{
    font-family: Arial, Helvetica, sans-serif;
    border-radius: 5px;
    height: 28px;
    border: 1px solid #ccc;
}
input::placeholder{
  font-size: 10px;
}
.topBox1{
  margin-right: 20px;
  width: 150px;
}
.topBox2{
  margin-left: 20px;
  width: 150px;
}
textarea::placeholder{
    padding: 5px 5px 5px 5px;
    font-size: 10px;
  }
.email{
    width: 350px;
}
textarea{
    font-family: Arial, Helvetica, sans-serif;
    border-radius: 5px;
    height: 150px;
    border: 1px solid #ccc;
}
.option{
  font-family: Arial, Helvetica, sans-serif;
  color: #fff;
  font-size: 15px;
}
.radioForm{
  width: 100%;
  display: flex;
  flex-direction: column;
}
.title > h4{
  font-family: Arial, Helvetica, sans-serif;
  color: #fff;
  float: left;
  margin: 5px 15px 0;
}
.radioButtons{
  display: flex;
  flex-direction: row;
  justify-content: center;
}
button{
  background-color: #66b8a8;
  border: none;
  color: white;
  padding: 15px 32px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  font-size: 16px;
  margin: 4px 2px;
  border-radius: 15px;
  cursor: pointer;
}
button:hover{
  background-color: #00362c;
}
img{
  max-width: 100%;
}
.cards{
  display: flex;
  flex-direction: row;
  margin: 40px auto 40px auto;
}
.card1{
  margin: 20px auto 20px auto;
  text-decoration: none;
  text-align: center;
}
/*JavaScript card*/
.card{
  box-shadow: 0 2px 10px rgba( 0, 0, 0, 0.2);
  border-radius: 10px;
  overflow: hidden;
  width: 325px;
}
.card1:visited {
  color: black;
```

```
}
.card-header{
  height: 200px;
}
.card-header img{
  object-fit: cover;
  height: 100%;
  width: 100%;
}
.card-content{
  background-color: #fff;
  padding: 30px;
}
.card-title{
  height: 20px;
  margin: 0;
}
.card-excerpt{
  color: #777;
  margin: 10px 0 20px;
  text-align: center;
}
.animated-bg {
  background-image: linear-gradient(
    to right,
    #f6f7f8 0%,
    #edeef1 10%,
    #f6f7f8 20%,
    #f6f7f8 100%
  );
  background-size: 200% 100%;
  animation: bgPos 1s linear infinite;
}
.animated-bg-text {
  border-radius: 50px;
  display: inline-block;
  margin: 0;
  height: 10px;
  width: 100%;
}
@keyframes bgPos {
  0% {
    background-position: 50% 0;
  }
  100% {
    background-position: -150% 0;
  }
}
footer{
  font-family: "Shantell Sans";
  background-color: #00896f;
  color: #fff;
  justify-content: center;
  text-align: center;
  padding: 20px;
}

.fa-brands {
  padding-left: 5px;
  padding-right: 5px;
  position: relative;
  content: '\f075';
  font-family: 'Font Awesome 5 Free';
  color: #fff;
  font-size: 3rem;
  border: none;
}
```

The CSS remains the same for both the navbar and footer as seen in previous pages. The contact form is stylized for clear reading and ease of use. The contact form also uses flexbox to help with responsiveness. We also utilize the card placeholder element from the udemy course.

```
/* responsive */
@media (min-width: 1200px){
  .contactForm{
    width: 30%;
  }
}
@media (max-width: 700px){
  textarea{
    width: 225px;
  }
  .email{
    width: 200px;
    justify-content: center;
  }
  form{
    flex-wrap: wrap;
  }
  .topBox1, .topBox2{
    margin: 5px 0;
    width: 200px;
  }
  .radioButtons{
    flex-direction: column;
  }
  .contactForm{
    height: 660px;
  }
  .option{
    font-size: 12px;
  }
  .title{
    font-size: 15px;
    padding: 20px;
  }
  .cards{
    flex-direction: column;
  }
}
```

Again we tried to make these pages as responsive as possible as mentioned previously. It makes it easier to use on mobile devices and such but there may be small glitches within as the screen shrinks.

JS:

```
const header = document.getElementById('header')
const title = document.getElementById('title')
const excerpt = document.getElementById('excerpt')
const header1 = document.getElementById('header1')
const title1 = document.getElementById('title1')
const excerpt1 = document.getElementById('excerpt1')
const header2 = document.getElementById('header2')
const title2 = document.getElementById('title2')
const excerpt2 = document.getElementById('excerpt2')

const animated_bgs = document.querySelectorAll('.animated-bg')
const animated_bg_texts = document.querySelectorAll('.animated-bg-text')

setTimeout(getData, 2500)

function getData() {
  header.innerHTML =
    '<img src="assets/courses.jpg" alt="courses" />'
  title.innerHTML = 'Courses'
  excerpt.innerHTML =
    'Come learn more about our courses'
    header1.innerHTML =
    '<img src = "assets/programs.jpg" alt = "programs">'
  title1.innerHTML = 'Programs'
  excerpt1.innerHTML =
    'Come learn more about our programs'
    header2.innerHTML =
    '<img src = "assets/questions.jpg" alt = "questions">'
  title2.innerHTML = 'FAQ'
  excerpt2.innerHTML =
    'Still have more questions?'


  animated_bgs.forEach((bg) => bg.classList.remove('animated-bg'))
  animated_bg_texts.forEach((bg) => bg.classList.remove('animated-bg-text'))
}
```

As stated above, this code is from the Udemy tutorial but the cards are altered to fit our website's overall design. The functionality remains the exact same but had the code get our site's information rather than what was listed as the screen begins to load.

FAQ Page

1. Navbar

HTML:

```
<!--- NAVBAR -->
   <nav class = "nav">
     <div class = "container">
       <h1><a href = "index.html"><i class = "fas fa-mountain-sun" id
= "logo"></i>Adventure Peak</a></h1>
       <ul>
         <li><a href = "index.html">Home</a></li>
         <li><a href = "#">Programs</a></li>
         <li><a href = "faq.html" class = "active">FAQ</a></li>
         <li><a href = "#">Contact Us</a></li>
       </ul>
     </div>
   </nav>
```

CSS:

```
/* Navbar */

#logo {
 padding: 20px;
}

body {
 padding-top: 60px;
 font-family: 'Shantell Sans';
 justify-content: center;
}

@media (max-width: 979px) {
 body {
   padding-top: 0px;
 }
}

.nav {
 font-family: 'Shantell Sans';
 font-size: 20px;
 position: fixed;
 z-index: 1;
 background-color: #00896f;
 top: 0;
 left: 0;
 right: 0;
 transition: all 0.3s ease-in-out;
}

.nav .container {
 font-family: 'Shantell Sans';
 display: flex;
 justify-content: space-between;
 align-items: center;
 padding: 5px 0;
 transition: all 0.3s ease-in-out;
}

.nav ul {
 font-family: 'Shantell Sans';
 display: flex;
 list-style-type: none;
 align-items: center;
 justify-content: center;
}

.nav a {
 font-family: 'Shantell Sans';
 color: #fff;
 text-decoration: none;
 padding: 7px 25px;
 transition: all 0.3s ease-in-out;
}
```

As in every other page, the HTML here creates the site's navbar. In the code above, it indicates that the "FAQ" tab is currently active. The CSS is consistent with the CSS formatting on every other page containing a navbar. ***There is no JavaScript for this section***

2. FAQ Container

HTML:

```
<!--- FAQ SECTION-->
   <h1 style = "text-align: center; size: 20px;">Frequently Asked Questions</h1>
   <div class="faq-container">

     <!--- What is Adventure Peak? -->
     <div class="faq">
       <h3 class="faq-title">What is Adventure Peak?</h3>
       <p class="faq-text">
```

```
           Adventure Peak is a fun, moderately competitive summer camp located in the
           Adirondack Mountains. We aim to provide 8-17 year olds with fun and engaging
           activities, while also providing leadership opportunities, social and emotional
           learning, survival skills, and most importantly, lifelong memories!
       </p>

       <button class="faq-toggle">
         <i class="fas fa-chevron-down"></i>
         <i class="fas fa-times"></i>
       </button>
     </div>

     <!--- Sleepaway Camp -->
     <div class="faq">
       <h3 class="faq-title">Is Adventure Peak a sleepaway camp?</h3>
       <p class="faq-text">
         Yes, Adventure Peak is a sleepaway camp. We ask that you please keep your child
         at camp for the entire duration of the program that they were signed up for.
       </p>
       <button class="faq-toggle">
         <i class="fas fa-chevron-down"></i>
         <i class="fas fa-times"></i>
       </button>
     </div>

     <!--- Camp Duration -->
     <div class="faq">
       <h3 class="faq-title">How long will my child be away at camp?</h3>
       <p class="faq-text">
         Depending on which program your child is signed up for, your child will be at camp
for either 6 weeks, 8 weeks, or 12 weeks.
         We have a full, fleshed-out curriculum for every part of the day, hence why we
strongly ask that your child remain enrolled in
         camp for the entire time.
       </p>
       <button class="faq-toggle">
         <i class="fas fa-chevron-down"></i>
         <i class="fas fa-times"></i>
       </button>
     </div>

     <!--- Early Dismissal -->
     <div class="faq">
       <h3 class="faq-title">
         What if my child needs to leave camp before the end of program?
       </h3>
       <p class="faq-text">
         Because of our heavy enrollment numbers, we ask that you only submit
         an enrollment form if you know your child will be available for the entire program,
         or consider enrolling in our shorter programs. We generally do not accommodate
         Requests for early dismissal from camp. In the event of an emergency, please
         contact our office at (123) 456-7890
```

```
      </p>
      <button class="faq-toggle">
        <i class="fas fa-chevron-down"></i>
        <i class="fas fa-times"></i>
      </button>
    </div>
```

For the faq.html file, there are far more sections than displayed above. However, since the code is redundant, only the first four questions are displayed. The HTML creates a basis for the Frequently Asked Questions page, separating each question into its own group with additional information. At the top of the page is a h1 tag that reads “Frequently Asked Questions.” The entire FAQ section beneath the header is contained in a div tag with a class “faq-container.” From there, each individual question follows the same format:

1. A div tag of class “faq” is created to contain the entire question and answer
2. An h3 tag of class “faq-title” is created within the faq div, displaying the frequently asked question
3. Under that h3 tag is a p tag of class “faq-text,” displaying the answer to the frequently asked question
4. Without CSS formatting, a button tag with class of “faq-toggle” is created directly underneath the p tag, with two buttons being displayed. The first is a down button, and the second is a close button. Again, without the CSS or JavaScript, these buttons do absolutely nothing.

All of the remaining questions follow the exact same framework, with the appropriate closing tags at the end.

CSS:

```
.faq-container {
 max-width: 600px;
 margin: 0 auto;
}

.faq {
 background-color: transparent;
 border: 1px solid #9fa4a8;
 border-radius: 10px;
 margin: 20px 0;
 padding: 30px;
 position: relative;
 overflow: hidden;
 transition: 0.3s ease;
}

.faq.active {
 background-color: #fff;
 box-shadow: 0 3px 6px rgba(0, 0, 0, 0.1), 0 3px 6px rgba(0, 0, 0, 0.1);
}

.faq.active::before,
.faq.active::after {
 content: '\f075';
 font-family: 'Font Awesome 5 Free';
 color: #2ecc71;
 font-size: 7rem;
 position: absolute;
 opacity: 0.2;
 top: 20px;
 left: 20px;
 z-index: 0;
}

.faq.active::before {
 color: #3498db;
 top: -10px;
 left: -30px;
 transform: rotateY(180deg);
}

.faq-title {
 margin: 0 35px 0 0;
}

.faq-text {
 display: none;
 margin: 30px 0 0;
}

.faq.active .faq-text {
 display: block;
}
```

The CSS formatting above gives the FAQ section a clean, sleek look, and is directly inspired by the Udemy course on collapsible FAQs. The .faq and .faq-container classes handle the simple look of the FAQs without the collapsible elements built in yet, giving each individual question a rounded border, transparent background, centered margins, and a smooth transition when the window is opened and closed.

The .faq.active class adds a white background color and a box shadow, while the .faq.active::before and .faq.active::after classes add two styled comment icons from the Font Awesome website. The z-index is set to 0 to keep it behind everything in the container. Then, for the .faq.active::before class, the comment icon is styled so that it’s transformed 180 degrees to face the opposite direction.

The faq.text class sets the display to none so that the answer is not shown initially, and is only shown if the class is active. If the faq text is active, then we format it so that the display is a block.

```
.faq-toggle {
 background-color: transparent;
 border: 0;
 border-radius: 50%;
 cursor: pointer;
 display: flex;
 align-items: center;
 justify-content: center;
 font-size: 16px;
 padding: 0;
 position: absolute;
 top: 30px;
 right: 30px;
 height: 30px;
 width: 30px;
}

.faq-toggle:focus {
 outline: 0;
}

.faq-toggle .fa-times {
 display: none;
}

.faq.active .faq-toggle .fa-times {
 color: #fff;
 display: block;
}

.faq.active .faq-toggle .fa-chevron-down {
 display: none;
}

.faq.active .faq-toggle {
 background-color: #9fa4a8;
}
```

The CSS above gives functionality to the buttons on the page, controlling whether the down button or the exit button is shown, as well as the positioning of each button. When a question is not active, it will show the down toggle, whereas if a question is active, the exit button will be displayed. Both buttons are set to the right of the container, and is formatted so that there's a slight change in color and size.

JavaScript:

```
const toggles = document.querySelectorAll('.faq-toggle')

toggles.forEach(toggle => {
   toggle.addEventListener('click', () => {
       toggle.parentNode.classList.toggle('active')
   })
})
```

The JavaScript above makes the buttons dynamic. By using the .faq-toggle class, the JavaScript waits for an interaction (in this case, a click). When this happens, it will use the toggle method to change the active class. So, when the down button is clicked, the faq-toggle will be set to active, and conversely, active will be removed from faq-toggle when the exit button is clicked.