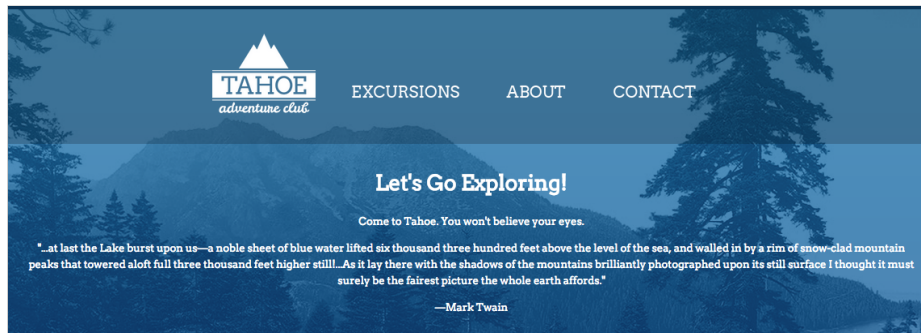


Exercise Preview



Exercise Overview

In this exercise, we will add content to the Tahoe Adventure Club page, including a background image. The image is a black & white photo that we will colorize using semi-transparent background gradients.

Getting Started

1. Open your code editor if it isn't already open.
2. We'll be switching to a new folder of prepared files for this exercise. Close all open files in your code editor to avoid confusion.
3. For this exercise we'll be working with the **Tahoe Backgrounds** folder located in **Desktop > Class Files > yourname-Web Dev Level 2 Class**. You may want to open that folder in your code editor if it allows you to (like Sublime Text does).
4. Open **index.html** from the **Tahoe Backgrounds** folder.
5. Preview **index.html** in a browser to remind yourself of where we left off in the last exercise. Hover over the top navigation links to see that their images change.
6. Leave the page open in the browser so we can come back to it and reload as we make changes.

Adding the Text Content

1. Return to **index.html** in your code editor.
2. To save you some time, we've provided you with some text content. Go to **Tahoe Backgrounds > snippets** and open **hero-text.txt**.
3. Hit **Cmd-A** (Mac) or **Ctrl-A** (Windows) to select all the text.
4. Hit **Cmd-C** (Mac) or **Ctrl-C** (Windows) to copy it.

5. Close the file.
6. You should be back in **index.html**. Paste the code below the header around line 31, as shown:

```
</header>
<div class="container">
  <article>
    <h1>Let's Go Exploring!</h1>
    <p>Come to Tahoe. You won't believe your eyes.</p>
    CODE OMITTED TO SAVE SPACE
    <p>&mdash;Mark Twain</p>
  </article>
</div>
</div>
```

7. Save the file.
8. Return to the browser and reload **index.html**. Check out the text content.

Adding the Background Image

We want to add a background image that will go behind both the text content and the nav header. We want it to span the entire page, so we'll put it in the banner-wrapper div.

1. In your code editor, open **main.css** from the **Tahoe Backgrounds > css** folder.
2. We want the background image to apply to any screen size, so we'll put it outside the media queries. At the bottom of the general styles, add the following new rule:

```
.banner-wrapper {
  background: url(../img/inspiration-point.jpg) no-repeat center top;
}
```

```
@media (max-width: 799px) {
```

NOTE: We specified that the image won't repeat (no-repeat), is centered, and starts at the top. The good thing about this image is that it won't really matter how it's cropped.

3. We want to make sure no matter what size the screen is, the image always stretches out and covers it all. To set this, add:

```
.banner-wrapper {
  background: url(../img/inspiration-point.jpg) no-repeat center top;
  background-size: cover;
}
```

RGBA & Background Gradients

4. Save the file.
5. Let's check out what this looks like so far. Go to **index.html** in your browser and reload it. You should see the image show up behind the text.
6. Try resizing the browser to see that the image resizes along with it.

Styling the Main Text & Adding Borders

We need to fix some things of course. Let's make the main text white and center it. Let's also add a border at the top and bottom of the banner-wrapper div to make it stand out more.

1. Return to **main.css** in your code editor.
2. Center align the text and make it white by adding the properties shown in bold:

```
.banner-wrapper {
  color: #fff;
  background: url(../img/inspiration-point.jpg) no-repeat center top;
  background-size: cover;
  text-align: center;
}
```

3. Save the file, then reload the page in your browser to see the text is white and centered.
4. Return to **main.css** in your code editor.
5. Add a border at the top and bottom of the banner-wrapper with the following code:

```
.banner-wrapper {
  color: #fff;
  background: url(../img/inspiration-point.jpg) no-repeat center top;
  background-size: cover;
  text-align: center;
  border-top: 4px solid #063456;
  border-bottom: 4px solid #063456;
}
```

6. Save the file, then reload the page in your browser. It's subtle but you should see a dark blue border above the header and below the photo.

RGBA Values

We would like to make the solid gray nav background semi-transparent so the photo is visible through it. The problem with using hexadecimal values for colors (as we've been doing so far) is that they are always completely opaque (no transparency).

Fortunately, we have an alternative we can use: RGBA values. **RGBA** (Red Green Blue Alpha) has the same color spectrum as hexadecimal values. Each of the Red, Green, and Blue values uses a number between 0 and 255. When all values are 0, the color is black. When all values are 255, the color is white. Any other number combination equals a different color. **A** stands for **Alpha** transparency, which specifies how transparent the color is. Alpha values range from 0 (100% transparent) to 1 (100% opaque). A decimal value specifies something in between.

1. Go to **main.css** in your code editor.
2. In the rule for **header**, around line 14, select the hexadecimal value for background and change it to an RGBA value as shown:

```
header {  
    background: rgba(0,0,0, 1);  
}
```

NOTE: 0,0,0 sets the color to pure black. 1 sets the opacity to 100%.

3. Save the file.
4. In your browser, reload the page. Notice the header background is totally black and you can't see through it at all. We need some transparency.
5. Go back to **main.css** and change the alpha value to give it some transparency:

```
header {  
    background: rgba(0,0,0, 0.5);  
}
```

NOTE: This sets the opacity to 50%.

6. Save the file, then reload the browser. Nice, we can see the photo behind the header!
7. Let's make it just a bit less opaque. In **main.css** change the alpha value as follows:

```
header {  
    background: rgba(0,0,0, 0.2);  
}
```

8. Save the file and reload the browser to see our more subtle look.

CSS Gradients

Wouldn't it be great if we could put the same type of transparency on the background image to create a tinted look? Unfortunately, if you have a background-image and a background-color, the image is always going to stack on top of the color. The way around that is to use background gradients.

RGBA & Background Gradients

A CSS gradient is not a series of CSS color values. Instead, the CSS gradient function creates an **image** that represents a gradient of colors. Because the specification for CSS allows us to use multiple images for a single element, we can create a gradient color tint on top of the photographic image. Let's investigate.

We should get started by learning the basic syntax for creating a CSS gradient.

1. Return to **main.css** and comment out the background declaration as follows:

```
.banner-wrapper {
  color: #fff;
  /* background: url(../img/inspiration-point.jpg) no-repeat center top; */
  background-size: cover;
  text-align: center;
  border-top: 4px solid #063456;
  border-bottom: 4px solid #063456;
}
```

2. Add the following new background property:

```
.banner-wrapper {
  color: #fff;
  background: linear-gradient(to bottom, powderblue, darkblue);
  /* background: url(../img/inspiration-point.jpg) no-repeat center top; */
  background-size: cover;
  text-align: center;
  border-top: 4px solid #063456;
  border-bottom: 4px solid #063456;
}
```

This code will create a vertical linear gradient that starts with light blue at the top of the banner-wrapper and gradates **to** a darker blue the **bottom** of the element. The CSS gradient image has no set dimensions. Its size will match the size of the element it applies to.

3. Save the file and reload the browser to check out the gradient. Easy! Let's fine-tune the blue values a bit.
4. Return to **main.css** and edit the color values as follows:


```
background: linear-gradient(to bottom, #0066a5, #004187);
```
5. Save the file and reload the browser to see the subtle difference.

Making the Gradient Semi-Transparent

If we plan to see the photograph beneath the gradient, we need to use alpha transparency for the gradient. Let's replace the hex values with RGBA.

1. Return to **main.css** and edit the color values as follows:

```
background: linear-gradient(to bottom, rgba(0,102,165, 0.7), rgba(0,65,135, 0.7));
```

2. Go to your browser and reload **index.html** to see the gradient. It's hard to tell that the gradient has alpha transparency but we'll be able to see how effective this is once we layer in the background image.

Layering in the Background Image

Remember: CSS3 gradients function as part of the background-image property, and the spec for CSS3 also allows for multiple background values. This affords you the opportunity to layer more than one background image behind an element, or even layer images and gradients, as we're about to do here.

1. Return to **main.css** in your code editor. We need to add the image URL. You can copy the following code from the background property you commented out earlier (around line 44):

```
url(../img/inspiration-point.jpg) no-repeat center top
```

2. Paste the code as shown:

```
background: linear-gradient(to bottom, rgba(0,102,165, 0.7), rgba(0,65,135, 0.7)), url(../img/inspiration-point.jpg) no-repeat center top;
```

NOTE: Make sure that you include the comma before the url() value. Multiple backgrounds must be comma-delineated.

3. Save the file, then reload the page in your browser. Check out the image showing behind the color gradient! The tinting works well because the image is black & white. The great thing about tinting with CSS is that you can experiment as much as you want with the color and opacity without having to edit the photo in Photoshop and re-export.

As a best practice, let's go back and remove the commented code. While we're here, we can add a solid color as a fallback for older versions of Internet Explorer that do not support gradients or RGBA.

4. Return to **main.css** in your code editor.
5. **Delete** the commented background property around line 45.

RGBA & Background Gradients

6. Add the following hexadecimal background color **above** the other background properties, as follows:

```
.banner-wrapper {  
  color: #fff;  
  background: #0066a5;  
  background: linear-gradient(to bottom, rgba(0,102,165, 0.7), rgba(0,65,135,  
0.7)), url(../img/inspiration-point.jpg) no-repeat center top;  
  background-size: cover;  
  text-align: center;  
  border-top: 4px solid #063456;  
  border-bottom: 4px solid #063456;  
}
```

NOTE: The order of these rules matter. The first background declaration with the hex color value will be applied but then overridden by the second declaration with the gradient and image, if those properties are supported by the browser. Older browsers that lack support will ignore the second declaration.

Fixed Backgrounds

One additional thing we want to do is set the background image so it has fixed positioning. This means that, as the browser is resized, the image will stay in place while the content moves over it.

1. Around line 50, add the bold code:

```
background: linear-gradient(to bottom, rgba(0,102,165,0.7)  
0%,rgba(0,65,135,0.7) 100%), url(../img/inspiration-point.jpg) no-repeat  
center top;  
background-attachment: fixed;  
background-size: cover;
```

2. Save the file, then reload the page in your browser. To best see this effect, shrink the height of the browser so that some of the text content is cut off at the bottom.
 3. Scroll down to see the text scrolling while the background stays in place.
 4. If you have some extra time, try experimenting with changing the background gradient color.
-