Shape Up! Case Study for *Murach's HTML5 and CSS3 (4th edition)*

This case study consists of a series of exercises that will guide you through the development of the pages for a health-oriented website.

Guidelines for doing the Shape Up! Case Study

Unlike the exercises in the book, the exercises for this case study don't give you step-by-step instructions. Instead, the description for each exercise includes an image of how the page should appear in a browser, along with specifications that you can complete in whatever sequence you prefer.

These specifications are detailed enough for you to complete each exercise. However, you'll need to use your best judgment on how to code many of the details.

Before you do the first exercise, your instructor will give you the starting files for the website. These files are stored in a folder named shape_up_exercises and include starting HTML and CSS files for the home page, along with all supporting files such as image, media, JavaScript, and text files. HTML files with "under construction" messages are also included for all of the other pages that can be accessed from the menu that you'll add in exercise 7, so you can refer to these pages from the menu. The exercises will instruct you about when to use each of the starting files.

Shape Up! 3   Create the home page

In this exercise, you'll code the HTML for the home page. When you're through, the page should look like this:

Specifications

- Create the home page from the index.html file that's in the root folder for the website. This file includes just the basic HTML elements.

- Add the HTML5 semantics to structure the page so it includes a header, the main content, and a footer.

- Add an image and two levels of headings to the header. You'll find the image, named shape_up_logo.png, in the images folder. You'll find the text for the header as well as for the other areas of the page in the home.txt file in the text folder.

- Add three levels of headings and the required text for the main content. The level-3 headings should contain links to other pages named index.html that can be found in the strength, cardio, stress, and diet folders.

- Code the quote in the second paragraph of text as a block quote.

- Code the additional resources as an unordered list with links to external pages that you locate.

- Add the required text to the footer, using a character entity for the copyright symbol.

- Add the link element for the favicon to the head element of the HTML document. You'll find the favicon in the images folder.

Shape Up! 4   Format the home page

In this exercise, you'll create an external style sheet that you can use to format the Shape Up! home page that you created in exercise 3. When you're through, the page should look similar to this:
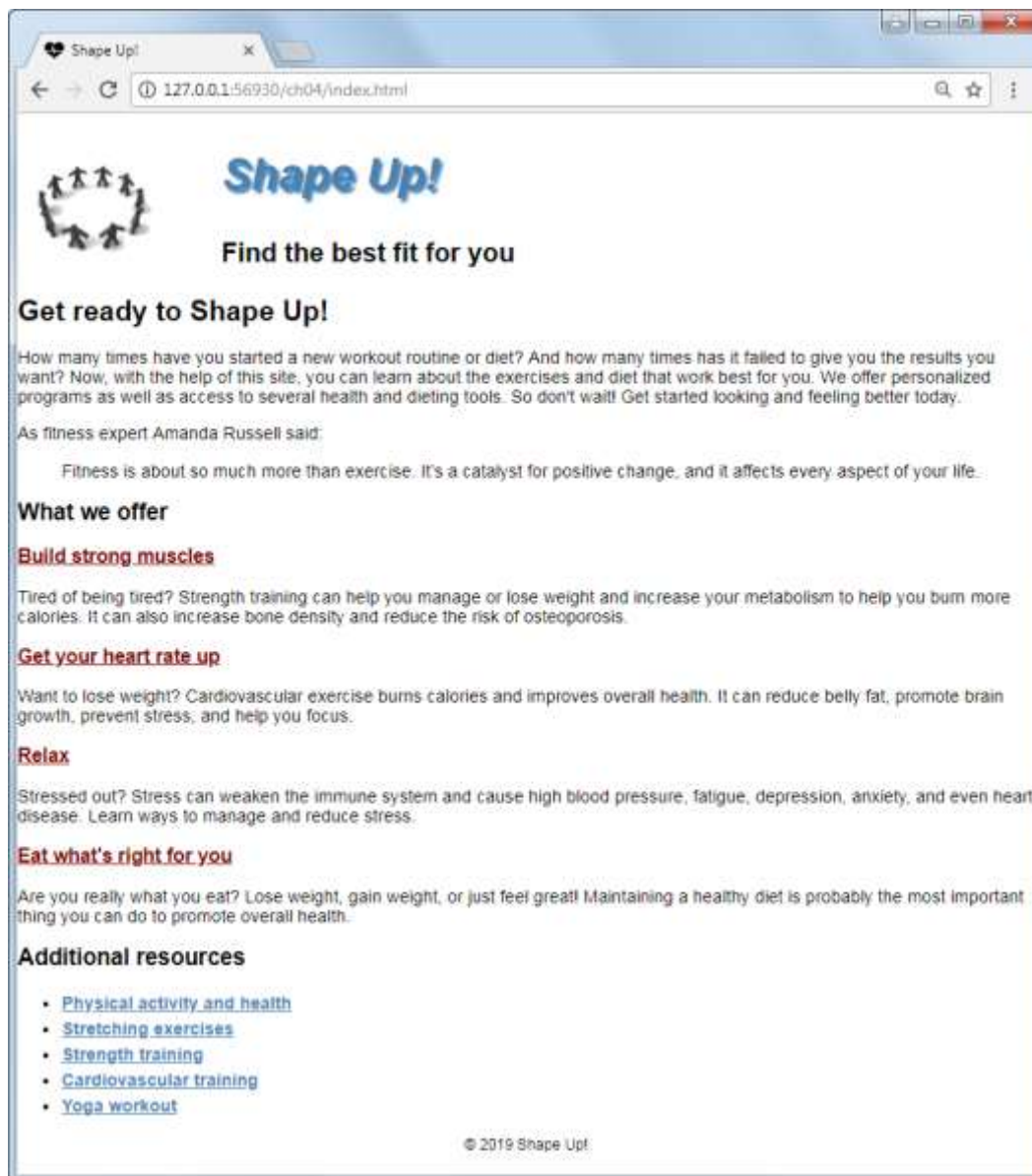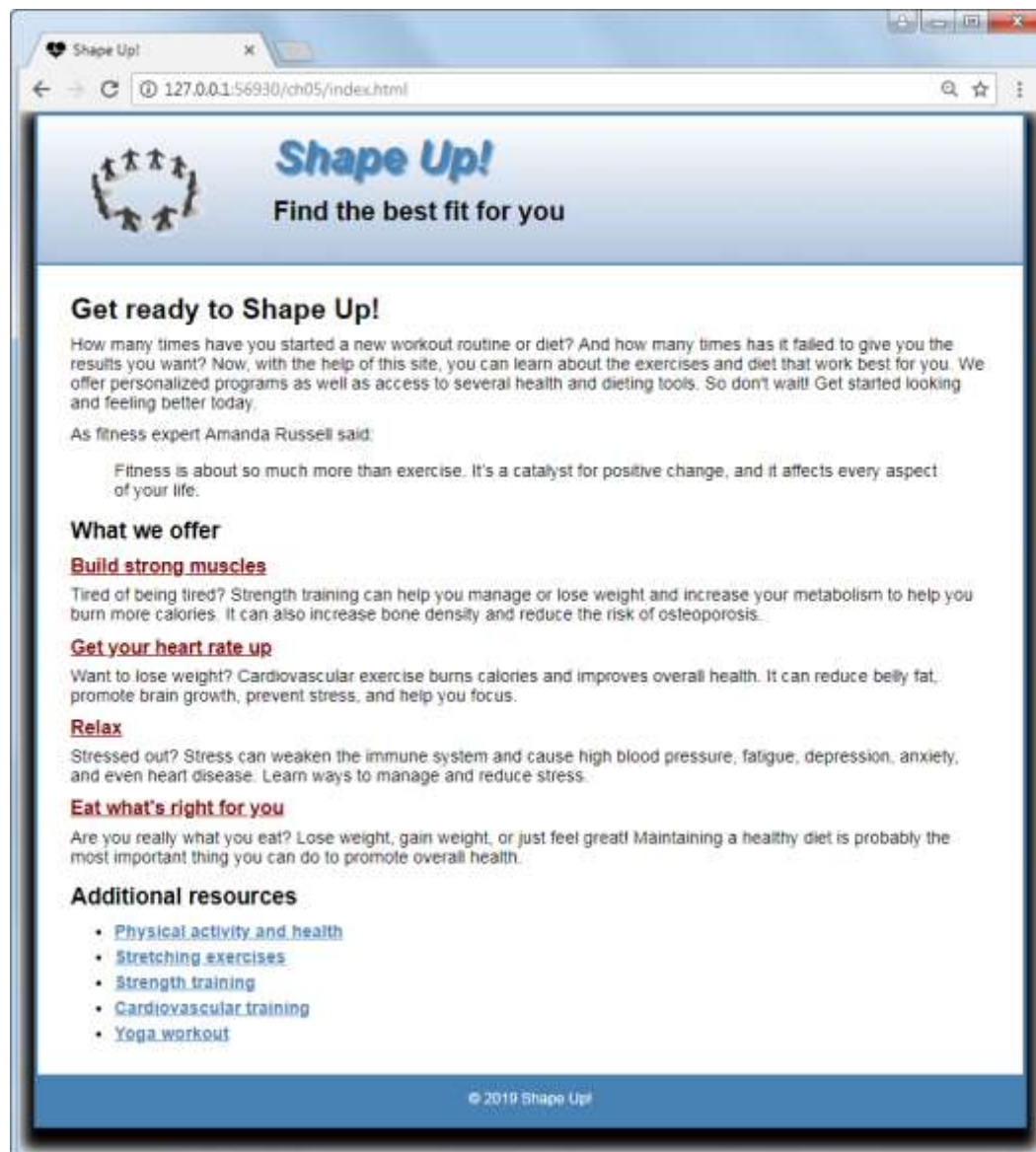

Specifications

- Create the external style sheet for the home page from the main.css file that's in the styles folder. This file is currently blank.

- Set the default font for the document to a sans-serif font.

- Adjust the font sizes for all of the headings so they look as shown above.

- Format the links in the section so they're displayed in maroon whether or not they've been visited. If a link has the focus or the mouse is hovering over it, though, it should be displayed in steelblue.

- Format the links in the aside so they're displayed in boldfaced steelblue whether or not they've been visited. If a link has the focus or the mouse is hovering over it, though, it should be displayed in maroon.

- In the header, float the image to the left and indent both headings. In addition, change the color of the first heading to steelblue, italicize it, and add a steelblue shadow.

- Format the list to increase the space between the list items.

- Reduce the font size for the footer and center it.

- Link the main style sheet that you just created as well as the normalize style sheet that's in the styles folder to the home page.

Shape Up! 5   Enhance the home page formatting

In this exercise, you'll use the CSS box model to add spacing, borders, and backgrounds to the Shape Up! home page that you formatted in exercise 4. When you're through, the page should look similar to this:

Specifications

- Format the body of the page so it's 900px wide, has a steelblue border with a black shadow, and is centered in the browser window.

- Add a steelblue border below the header, add a steelblue background to the footer, and change the color of the footer text to white.

- Add a gradient to the header. The gradient should range from white at the top to lightsteelblue at the bottom.

- Adjust the margins and padding for the elements on the page so it looks as shown above.

Shape Up! 6   Use 2- and 3-column layouts

In this exercise, you'll enhance the formatting of the home page that you worked on in chapter 5 so it uses a 2-column layout. Then, you'll add a third column with new content to the page.

Specifications for the 2-column layout
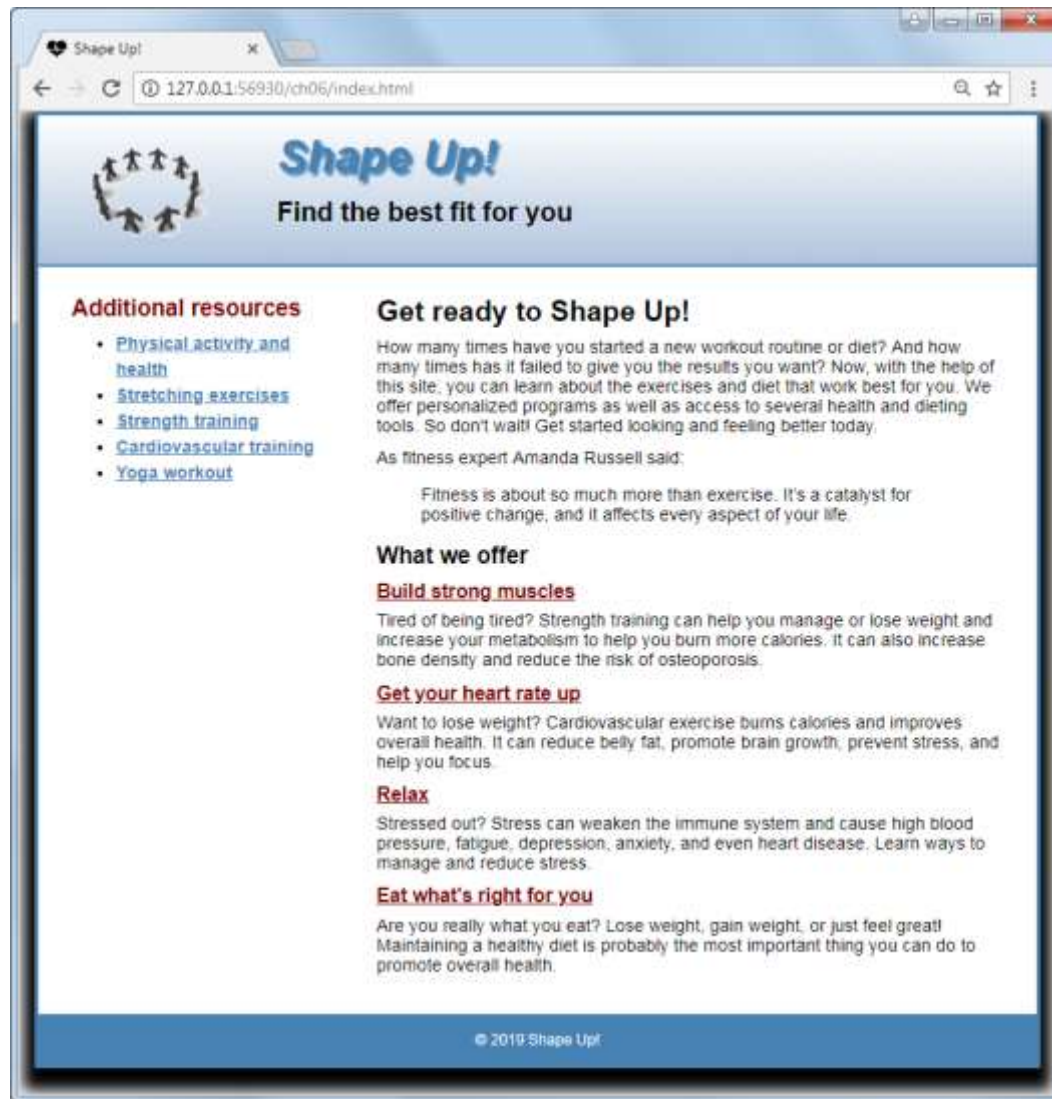
- Move the unordered list and its heading to a sidebar that's floated to the left of the remaining content in the main element, which should be coded in a section. The width of the sidebar should be 245px. Be sure to clear the footer so it's not displayed below the list.

- Adjust the margins and padding for the sidebar and section as necessary. (Note: because the unordered list is given some left padding by default, you don't need to apply any additional margin or padding to it.)

- Change the color of the h2 heading in the sidebar so it's the same as the color of the h3 headings in the section.

Specifications for the 3-column layout
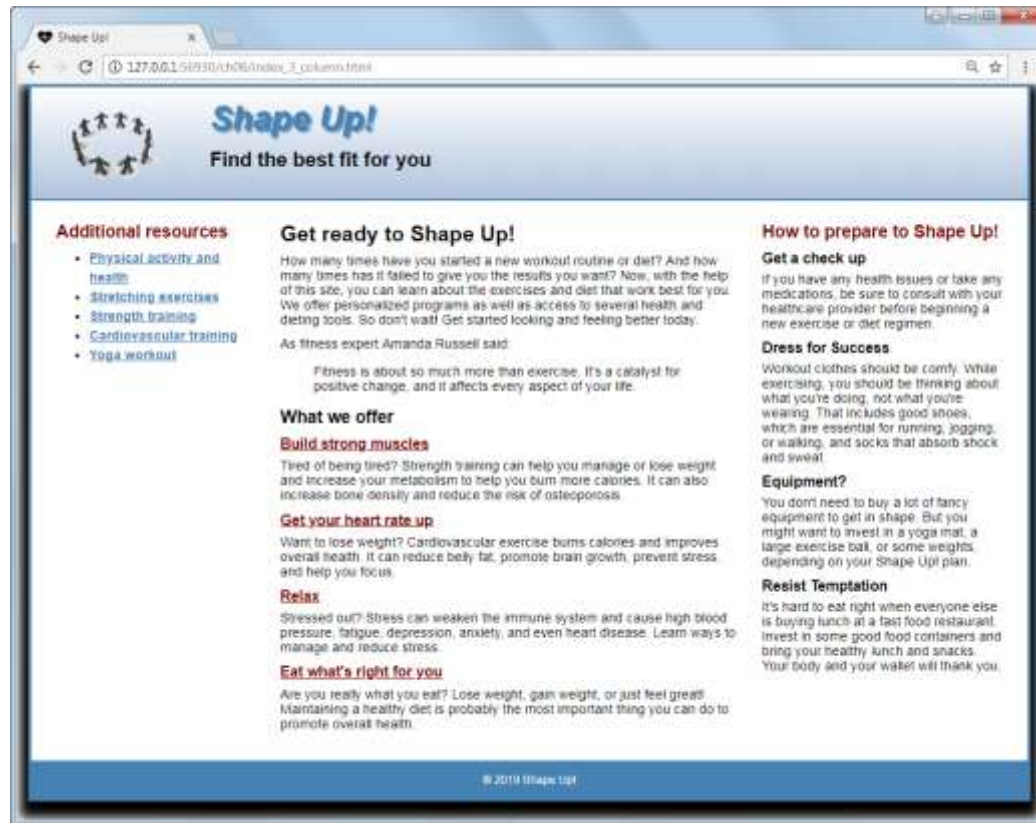
- To create the 3-column home page, you can copy the index.html page you just completed and rename it index_3_column.html. Then, you can copy the main.css file and rename it main_3_column.css

- Modify the link element for the style sheet in the 3-column home page so it refers to the new style sheet.

- Move the existing sidebar before the section in the HTML, and assign an id to this sidebar. Then, modify the CSS so it refers to the sidebar by this id.

- Add another sidebar after the section, and add the content shown above to this sidebar. You'll find this content in the prepare.txt file in the text folder. Be sure to assign an id to this sidebar so you can use it to refer to the sidebar in the CSS. Then, format the sidebar as shown above.

- Increase the width of the page to 1200px. Then, set the widths and spacing for the section and two sidebars and the floating for the new sidebar so the page appears as shown above.

Shape Up! 7   Add a navigation menu and more

In this exercise, you'll enhance the home page you worked on in exercise 6 so it includes a two-tier horizontal navigation menu and an image link. You'll also format the list in the sidebar so the items aren't indented, and you'll remove the underlines from the links. When you're through, the home page should look similar to this:

Specifications

- Add a 2-tier navigation menu. The main menu and Stress Relief submenu should include the links shown above. In addition, a Healthy Diets submenu should be included with the links "Why a Healthy Diet?", "Plan Your Meals", "Count Your Calories", and "Calculate Your BMI". Be sure all the links refer to the correct pages. (The "What is Stress?" link should refer to the index.html page in the stress folder, and the "Why a Healthy Diet?" link should refer to the index.html page in the diet folder).

- Format the navigation menu so the background color is steelblue, so the link for the page that's currently displayed (in this case, Home) has black text, and so the link that the mouse is hovering over or has the focus (in this case, Meditation) has a lightsteelblue background.

- Remove the links from the h3 headings in the section, since these pages can now be accessed from the menu.

- Modify the logo in the header so it's a link that displays the home page.

- Format the list in the sidebar so there's no space to the left of the list items. In addition, remove the underlines from the links in the list items.

Shape Up! 8   Use Responsive Web Design

In this exercise, you'll enhance the home page you worked on in exercise 7 so it uses Responsive Web Design.

Specifications for a fluid design
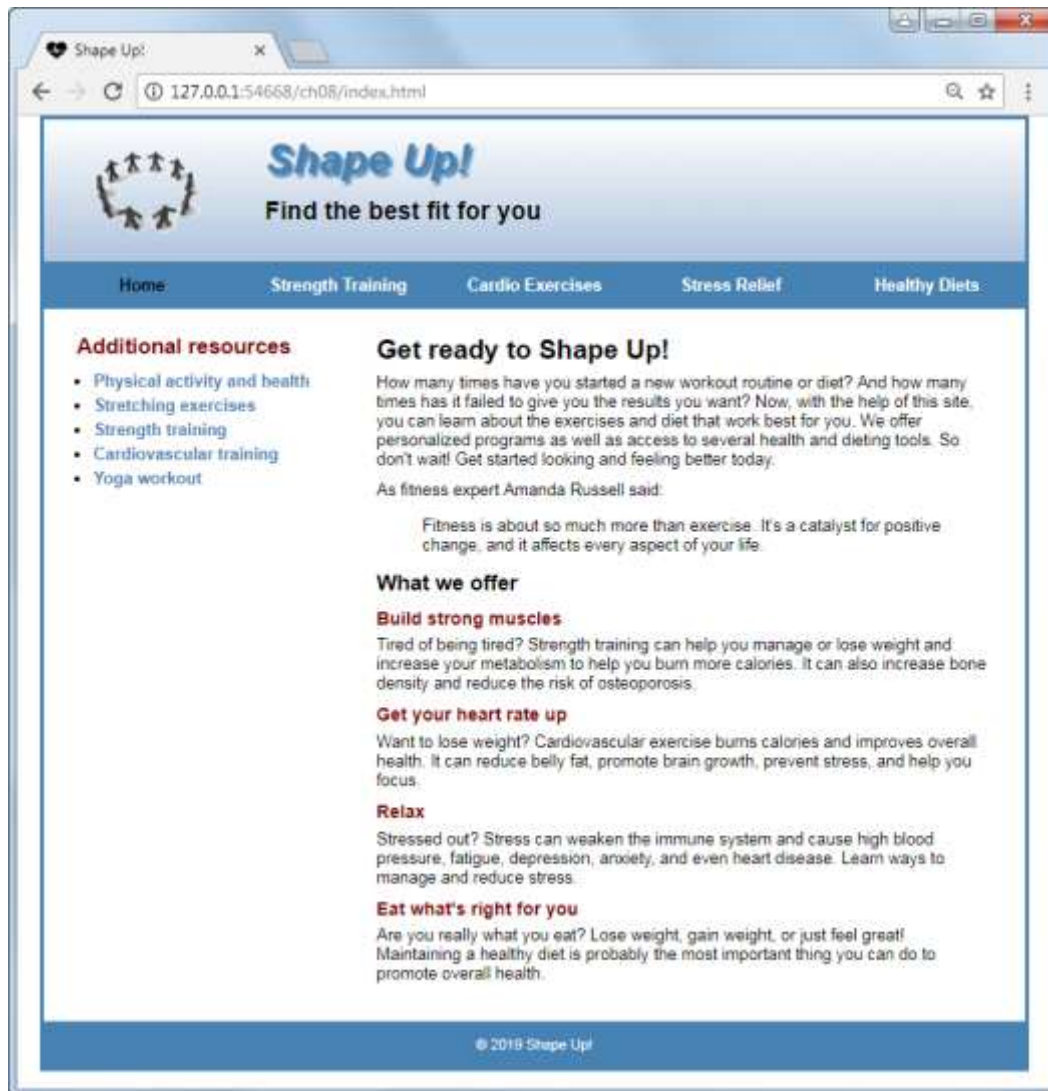
- Make a copy of the main.css style sheet and name it main_rwd.css. Then, modify the HTML for the page so it uses this style sheet.

- Modify the page so it uses a fluid layout. The maximum width for the page should be 1024px, and the page should occupy 95% of the browser window. Be sure to make the left and right margins and padding as well as the structural elements fluid.

- If you used pixels for any top or bottom margins or padding, convert them to ems. In addition, if you specified any of the font sizes using pixels, convert them to percentages or ems.

- Make the image in the header scalable. Make sure that the width of the image is limited to its native size of 140px, and make sure that the image can't be reduced to a width less than 70px.

Specifications for a media query for a tablet in portrait orientation

- Add a meta element to the page to control the viewport for media queries.

- Add a media query that will be applied to screens with a maximum width of 900px. Then, add styles to remove the shadow from the page and reduce the size of all the fonts on the page to 90%.

Specifications for a media query for a mobile phone in landscape orientation
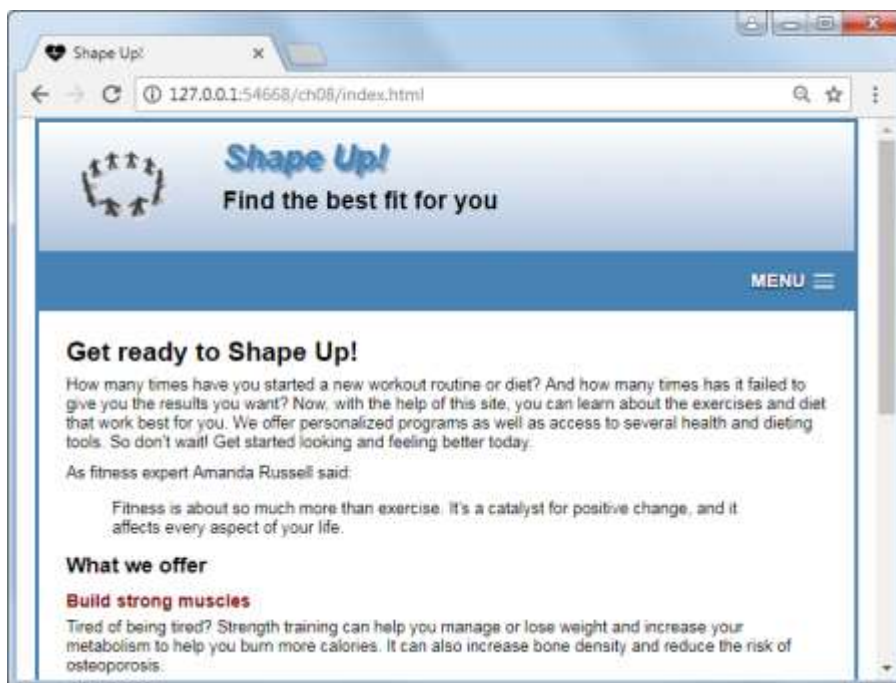
- Add a media query that will be applied to screens with a maximum width of 767px. Then, add the HTML and CSS required to hide the navigation menu and display a SlickNav menu in its place. You'll find the slicknav.css file in the styles folder, you'll find the slicknav.min.js file in the js folder, and you can get the jQuery library from the jQuery Content Delivery Network.

- Change the background color of the SlickNav menu to steelblue, and change its padding to 3px.

- Change the background color of the ul and <a> elements for the mobile menu to steelblue. In addition, when the mouse hovers over a link in the menu, the link should be displayed in lightsteelblue.

- Change the font size of the first heading in the header to 200%, and format the page so it appears in one column.

Note: The page shown above was captured on a desktop so it is an approximation.

Specifications for a media query for a mobile phone in portrait orientation

- Add a media query that will be applied to screens with a maximum width of 479px. Then, reduce the font sizes of the headings in the header and the <p> elements in the section.
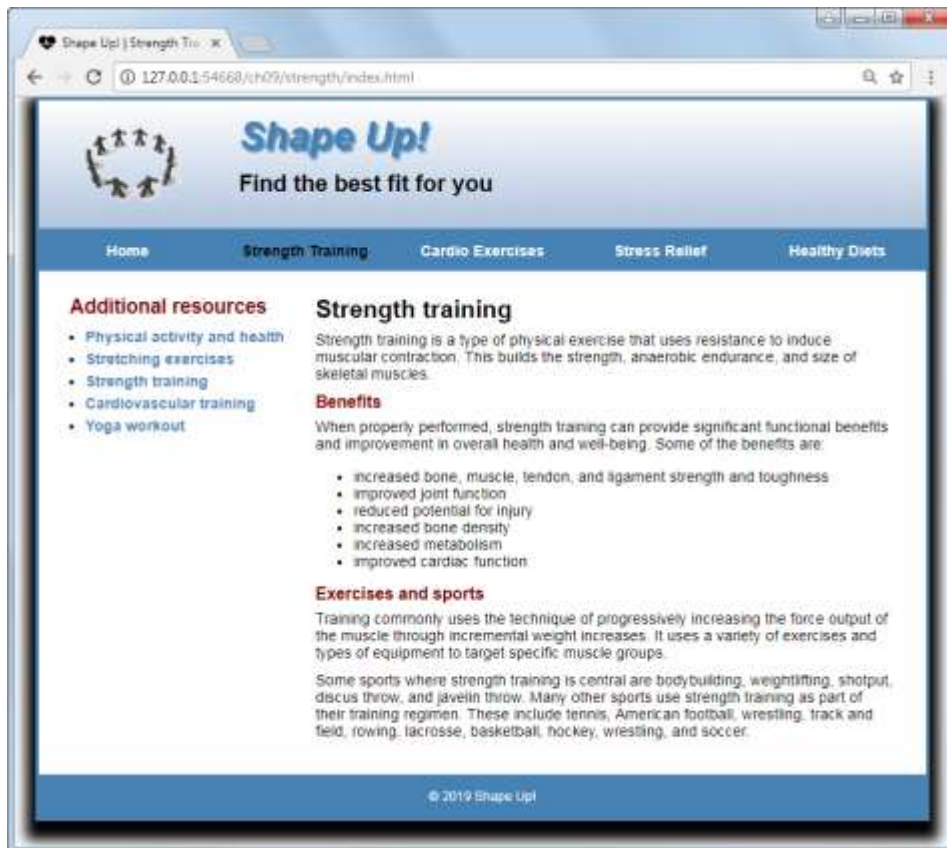
Note: The page shown above was captured on a desktop so you can see the layout of the entire page.

Shape Up! 9   Use Flexible Box Layout

In this exercise, you'll create a strength training page, and you'll use Flexible Box Layout to format portions of that page.

Specifications for the desktop layout

- Open the index.html file in the strength folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the meta element for the viewport, the link elements for the favicon and style sheets, and the script elements for the jQuery library and the SlickNav menu from the index.html page for exercise 8. In addition, copy the code for the header, footer, navigation menus, and sidebar from this page, and add a section element before the sidebar.

- Create a new style sheet named strength.css for the strength training page, and copy the styles you need from the main_rwd.css file to this style sheet. Then, modify the link element for the style sheet in the index.html file so it points to the correct style sheet.

- Delete the submenus from the navigation menu to make it easier to work with. Then, delete any CSS that applies to the submenus.

- Modify the navigation menu so it indicates that the strength training page is the current page, and modify the links in the menu as necessary.

- Add the content in the strength_training.txt file in the text folder to the section. Then, format this content so it looks as shown above.

- Modify the navigation menu so it uses flex. The menu should extend the full width of the body, each flex item should occupy the same amount of space within the flex container, and space should be allocated evenly between the flex items.

- Modify the main element so it uses flex. Allocate space to the sidebar and section so they're displayed as shown above.

- Reorder the section and the sidebar so the sidebar is displayed at the left side of the page.

- Adjust the spacing as necessary so the page looks as shown above.

Specifications for a media query for a mobile phone in landscape orientation
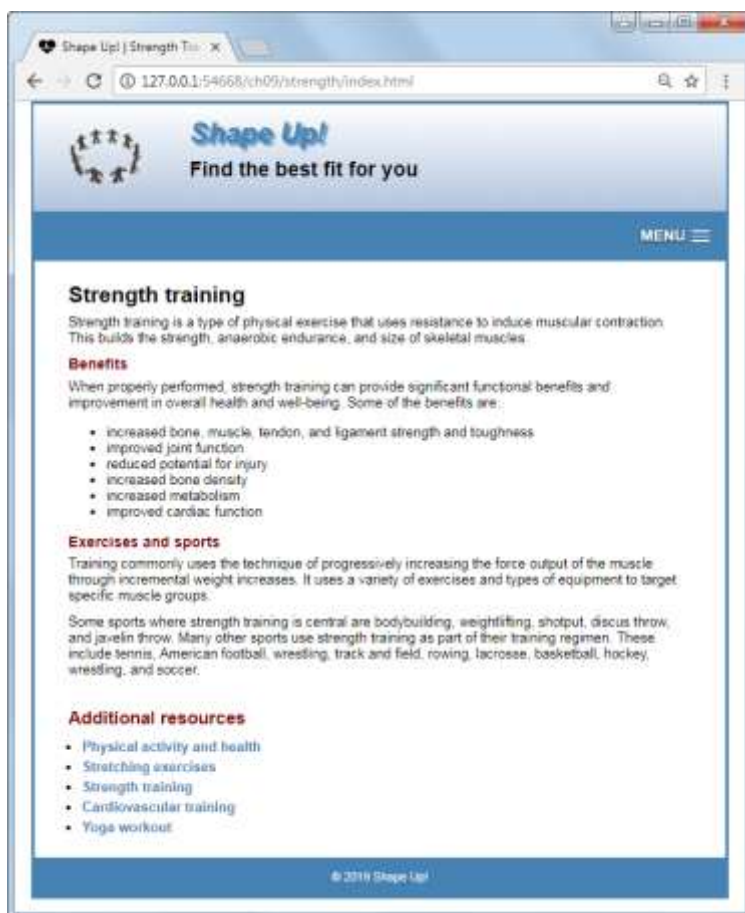

- Modify the main element so the section and sidebar are displayed in a column.

- Reorder the section and sidebar so the section is displayed above the sidebar.

- Adjust the spacing as necessary so the page is displayed as shown above.

Shape Up! 10   Use Grid Layout

In this exercise, you'll create a stress relief page that's formatted using Grid Layout.

Specifications for the desktop layout
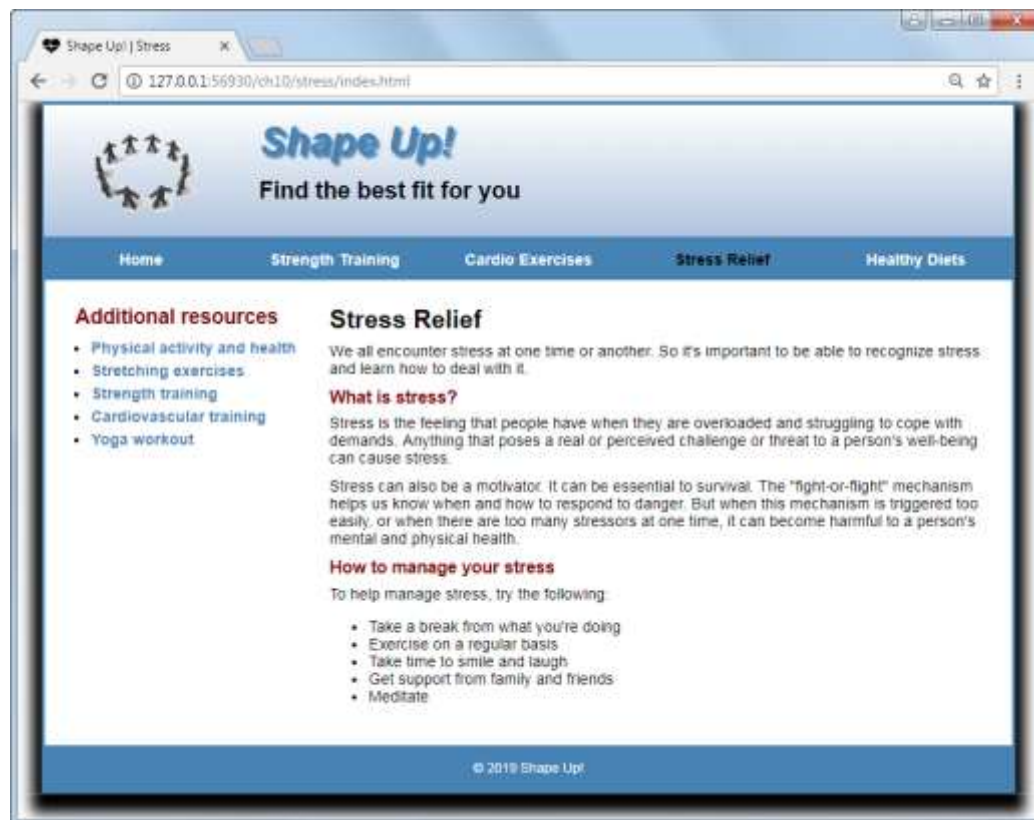
- Open the index.html file in the stress folder. This file contains the basic content for an "under construction" page.

- Delete the main element from the body of the document. Then, copy the meta element for the viewport, the link elements for the favicon and style sheets, and the script elements for the jQuery library and the SlickNav menu from the index.html page for exercise 8. In addition, copy the code for the header, footer, navigation menus, and sidebar from this page, and add a section element before the sidebar.

- Create a new style sheet named stress.css for the stress relief page, and copy the styles you need from the main_rwd.css file to this style sheet. Then, modify the link element for the style sheet in the index.html file so it points to the correct style sheet.

- Delete the submenus from the navigation menu to make it easier to work with. Then, delete any CSS that applies to the submenus.

- Modify the menu so it indicates that the stress relief page is the current page, and modify the links in the menu as necessary.

- Add the content in the stress_relief.txt file in the text folder to the section. Then, format this content so it looks as shown above.

- Format the body of the page so it uses grid. The grid should include grid areas for the header, the navigation menu, the sidebar, the section, and the footer. Use any technique you want to position elements in the grid.

- Format the navigation menu so it uses grid. The menu items should be sized proportionally.

Specifications for a media query for a mobile phone in landscape orientation
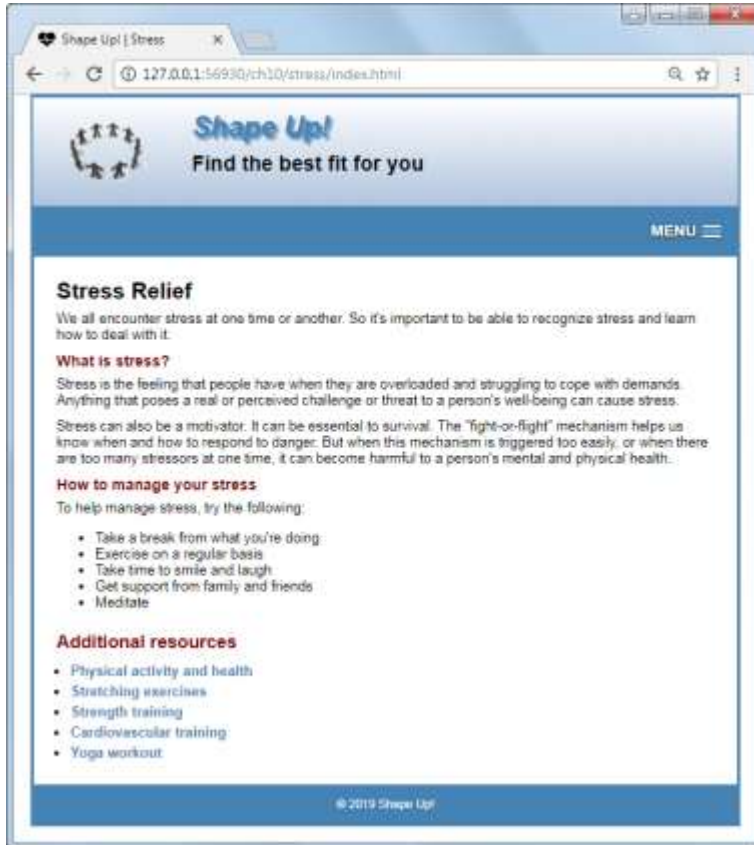
- Redefine the grid for the body of the page so the header, navigation menu, sidebar, section, and footer are displayed in a single column that's the full width of the screen. The section should be displayed before the sidebar.

- Position the mobile menu in the grid so it's displayed in place of the navigation menu.

- Adjust the spacing as necessary so the page looks as shown above.

Shape Up! 11   Create a stress tips page and add icons to the home page

In this exercise, you'll create a stress tips page that includes several images and that uses a logo with the WebP format. You'll also add icons to the home page.

Specifications for the stress tips page
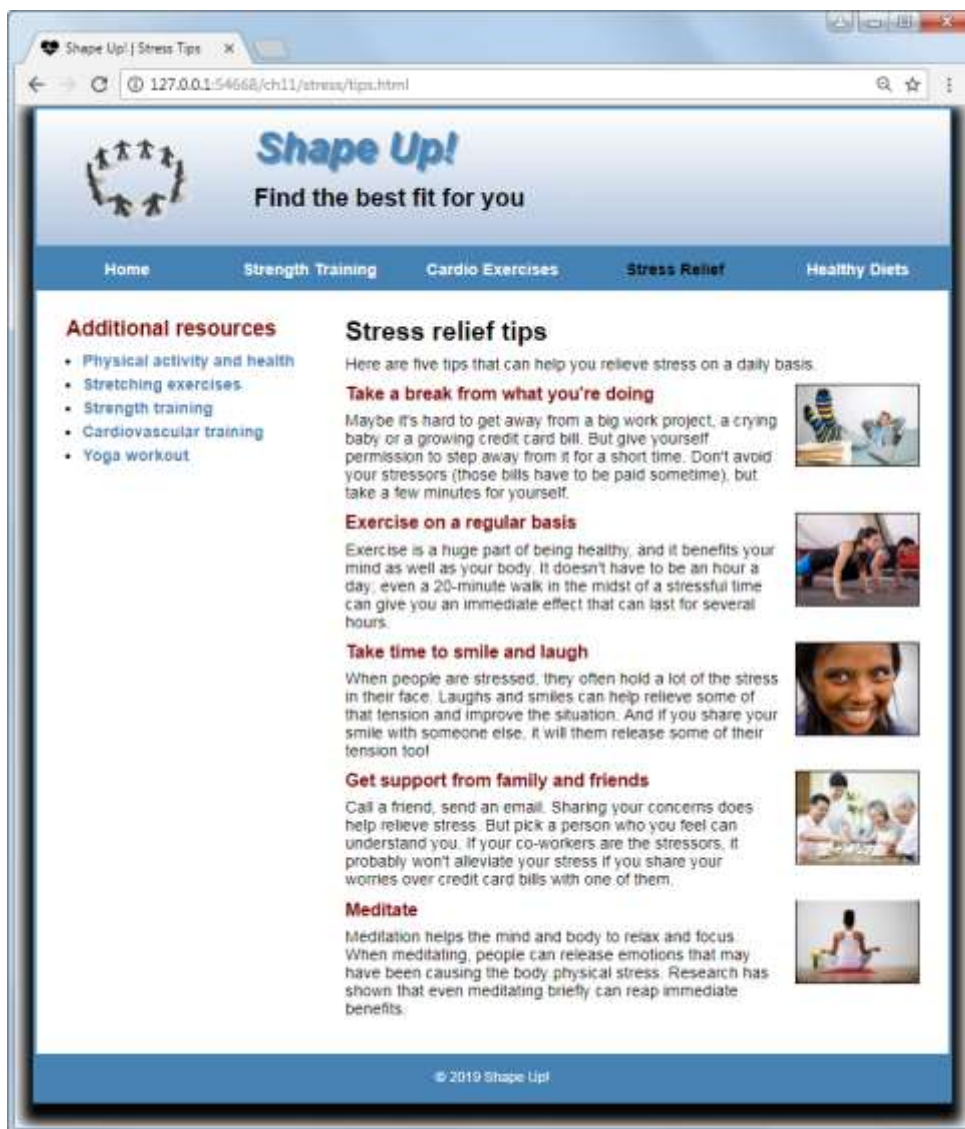
- Open the tips.html file in the stress folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the link elements for the favicon and style sheets, as well as the code for the header, footer, navigation menu, and sidebar, from the index.html page for exercise 7, and add a section element before the sidebar.

- Modify the navigation menu so it indicates that one of the stress relief pages is the current page, and modify the links in the menu as necessary.

- Add the content in the stress_tips.txt file in the text folder to the section. Then, format this content so it looks as shown above. This page should use the main.css style sheet that's also used by the home page.

- Add an image element before each level-2 heading in the section, and assign all the elements to the same class. You'll find the images, named break.png, exercise.png, smile.png, support.png, and meditate.png, in the images folder.

- Modify the main.css style sheet so it uses the class you assigned to the images you just added to float the images to the right of the content that follows. In addition, add a border around the images, and add any necessary spacing around the images.

- Modify the logo in the header so it will display an image with the WebP format if it's supported by the browser or the image with the PNG format otherwise. You'll find the shape_up_logo.webp file in the images folder.

Specifications for the home page
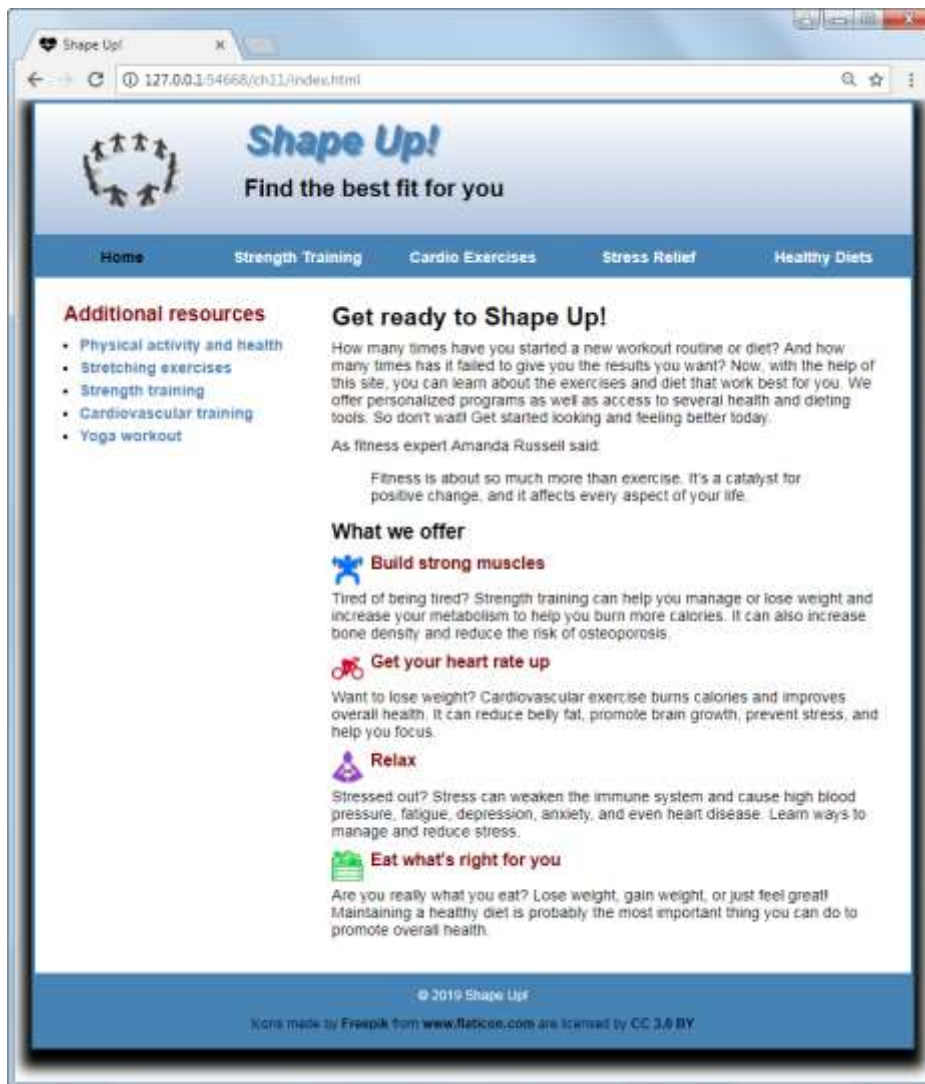
- Download icons like the ones shown above from the Flaticon library at www.flaticon.com. Save each image as a PNG file, choose whatever color you want for each icon, and select 32 pixels as the size. As you download the icons, be sure to note how you need to credit the author.

- Add the icons you just downloaded to the section, and float them to the left of the headings. Clear the left float from the paragraphs that follow the headings so they don't flow into the space to the right of the icons. Then, add space below the icons as shown above.

- Add the required credits for the icons to the footer after the copyright date. (In this case, a single credit can be used for all the icons.) Then, center the credits and format the links they contain so they're black whether or not they've been visited and so a link that the mouse is hovering over or has the focus is white. Adjust the spacing in the footer as necessary so it looks as shown above.

Shape Up! 12   Create a page that uses a table

In this exercise, you'll create a page that uses a table to present a calorie counter. When you're through, the page should look similar to this:

Specifications

- Open the calories.html file in the diet folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the link elements for the favicon and style sheets, as well as the code for the header, footer, navigation menu,

and sidebar, from the index.html page for exercise 7, and add a section element before the sidebar.
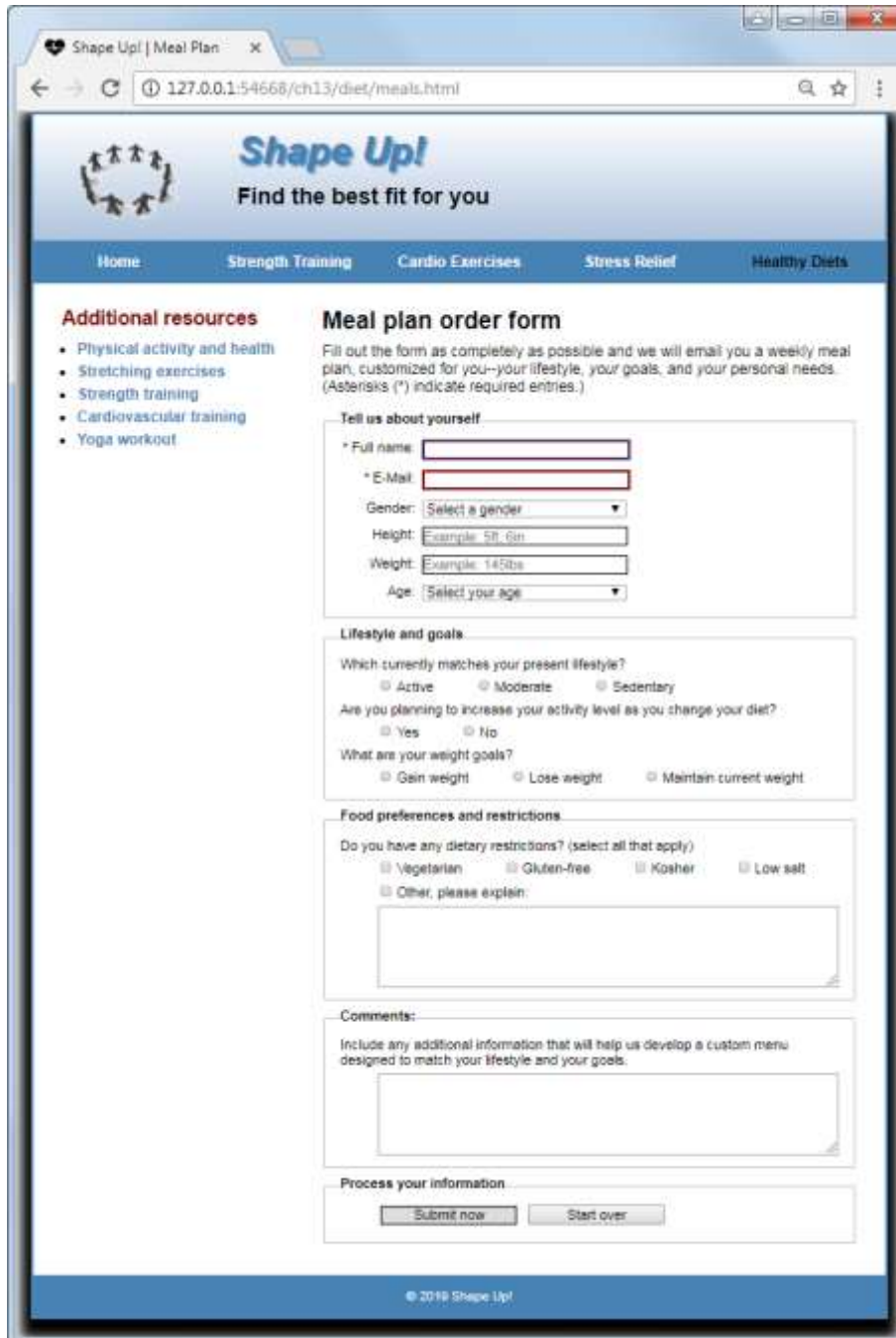
- Modify the navigation menu so it indicates that one of the healthy diets pages is the current page, and modify the links in the menu as necessary.

- Add the content of the calories.txt file in the text folder to the section. Then, add a table with a caption, a header, and a body that contains all the content of the section except for the last paragraph as shown above. Be sure to merge the rows in the first column for each category so the category name is displayed only in the first row.

- Create a new style sheet named calories.css for the calorie counter page, and copy the styles you need from the main.css file to this style sheet. Then, modify the link element for the style sheet in the calories.html file so it points to the correct style sheet.

- Align the caption, headings, and data, and apply any other required formatting as shown above.

Shape Up! 13   Create a page that uses a form

In this exercise, you'll create a page that includes a form for requesting a meal plan. In addition, you'll create a page that's displayed when the form is submitted.

Specifications for the meals form

- Open the meals.html file in the diet folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the link elements for the favicon and style sheets, as well as the code for the header, footer, navigation menu, and sidebar, from the index.html page for exercise 7, and add a section element before the sidebar.

- Modify the navigation menu so it indicates that one of the healthy diets pages is the current page, and modify the links in the menu as necessary.

- Create a new style sheet named meals.css for the meals page, and copy the styles you need from the main.css file to this style sheet. Then, modify the link element for the style sheet in the meals.html file so it points to the correct style sheet.

- Add the heading, text, and form element to the section. Code the form element so a form named meal_request.html in the diet folder is displayed when the Submit button is clicked.

- Add the label, input, and select elements for the form. The focus should be on the first text field when the form is first displayed.

- Align the controls as shown above.

- Use the data validation attributes to require entries in the first two text fields. In addition, add a 2px solid maroon border around these fields until valid data is entered. Then, display a 1px solid black border.

- Use placeholders to display messages in the Height and Weight fields as shown above.

- Add any other formatting as necessary so the page looks as shown above.

Specifications for the meal request form
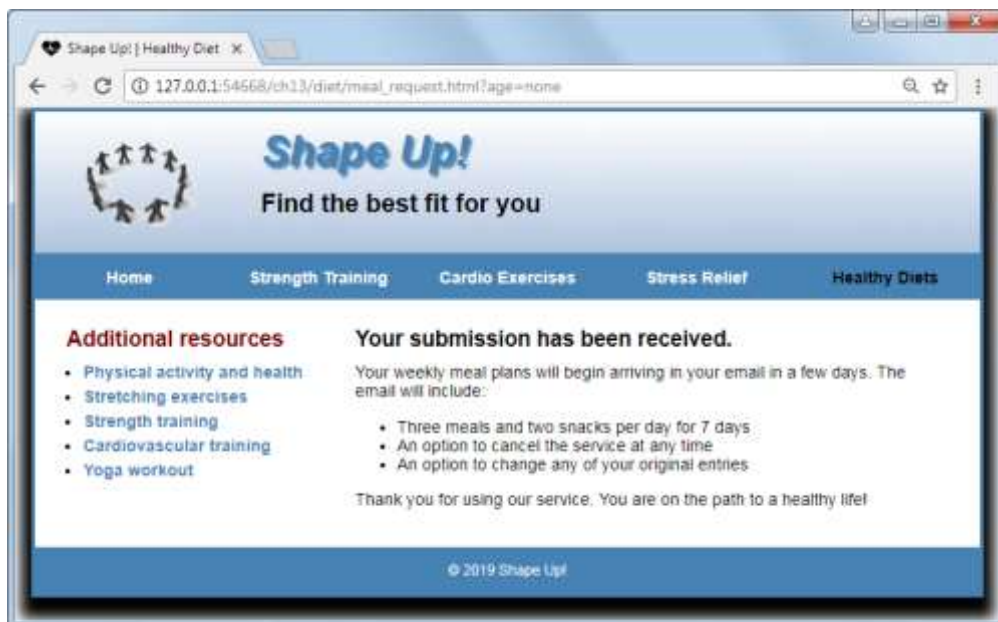
- Create a new page named meal_request.html in the diet folder. Then, copy the code for the basic page elements, as well as the header, footer, and sidebar, from the meals page.

- Change the link element that refers to the meals.css style sheet so it refers to the main.css style sheet.

- Add the heading, paragraphs, and list shown above.

Shape Up! 14   Create a page that uses audio and video

In this exercise, you'll create a meditation page that contains audio and a YouTube video. When you're through, the page should look similar to this:
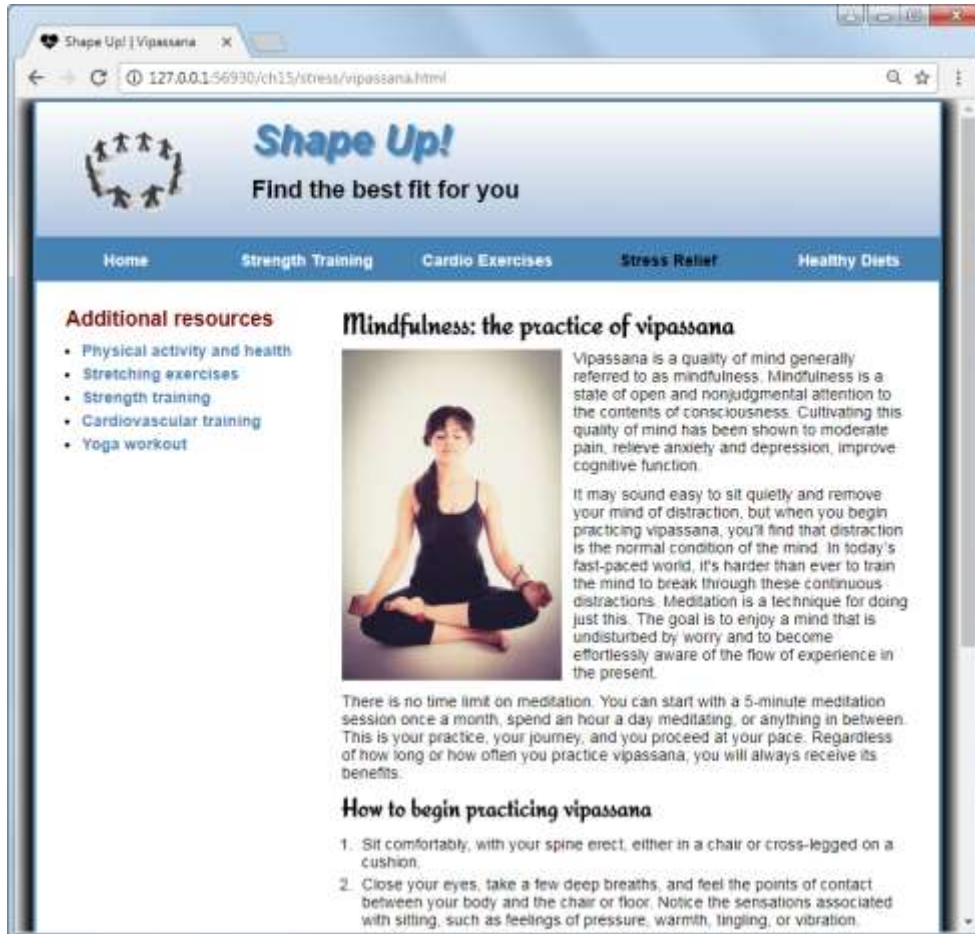

Specifications

- Open the meditate.html file in the stress folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the link elements for the favicon and style sheets, as well as the code for the header, footer, navigation menu,

and sidebar, from the index.html page for exercise 7, and add a section element before the sidebar.

- Modify the navigation menu so it indicates that one of the stress relief pages is the current page, and modify the links in the menu as necessary.

- Add the content of the meditate.txt file in the text folder to the sidebar and section as shown above.

- Add HTML below the first h3 heading in the section that plays an MP3 file of a Buddhist monk chanting. You'll find the MP3 file you need in the media folder. The control toolbar should be displayed and the audio should play automatically when the page is loaded.

- Add HTML below the second h3 heading that plays an MP3 file of a Buddhist monk reciting positive precepts. This MP3 file is also in the media folder. The control toolbar should be displayed for this audio file as well, but the audio should not start automatically.

- Add HTML below the third h3 heading for playing a YouTube video that presents different meditation poses. You choose the video to be played.

- Create a new style sheet named meditate.css for the meditation page, and copy the styles you need from the main.css file. Then, modify the link element for the style sheet in the meditation page so it points to the correct style sheet.

- Add any formatting necessary for the page to look as shown above.

Shape Up! 15   Add an embedded font and provide for printing textual information

In this exercise, you'll modify a vipassana page so it uses an Adobe Edge web font. You'll also add information to this page about the meditation technique of vipassana, and you'll provide for printing that information. When you're done, the page should look similar to this:

Specifications

- Open the vipassana.html page in the stress folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the link elements for the favicon and style sheets, as well as the code for the header, footer, navigation menu, and sidebar, from the index.html page for exercise 7, and add a section element before the sidebar.

- Modify the navigation menu so it indicates that one of the stress relief pages is the current page, and modify the links in the menu as necessary.

- Add the content of the vipassana.txt file in the text folder to the section.

- Create a new style sheet named vipassana.css, and copy the styles you need from the main.css file. Then, modify the link element for the style sheet in the vipassana page so it points to the correct style sheet.

- Add the image named vipassana.jpg after the first heading in the section. You'll find this file in the images folder.

- Format the content in the section as shown above, using existing style rules for the headings.

- Use the Adobe Edge Web Fonts page (edgewebfonts.adobe.com) to get the script element for using the Redressed font, or another font of your choosing. Copy this script element into the head section of the vipassana page.

- Apply the Adobe Edge web font to both headings in the section and size the headings appropriately. (Note that if the page takes a few seconds to load after you add this font, it's because it's being downloaded from the Internet.)

- Add the required code to the vipassana.css style sheet to provide for printing the information in the section. Print the body of the article in a serif font, but print the headings in a sans-serif font. In addition, add a 3-pixel black border below the first heading and a 2-pixel black border above the footer.

- Remove all other elements from the page, and remove the box shadow from around the page. In addition, increase the left and right margins for the page to make it more readable. When you're done, the printed output should look like this:
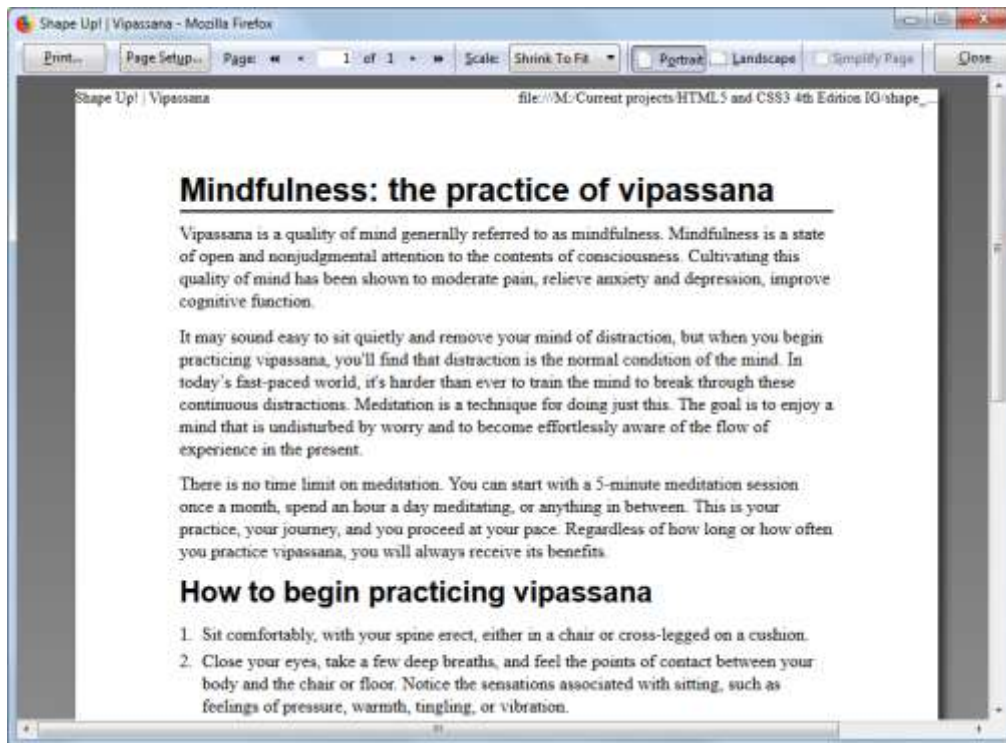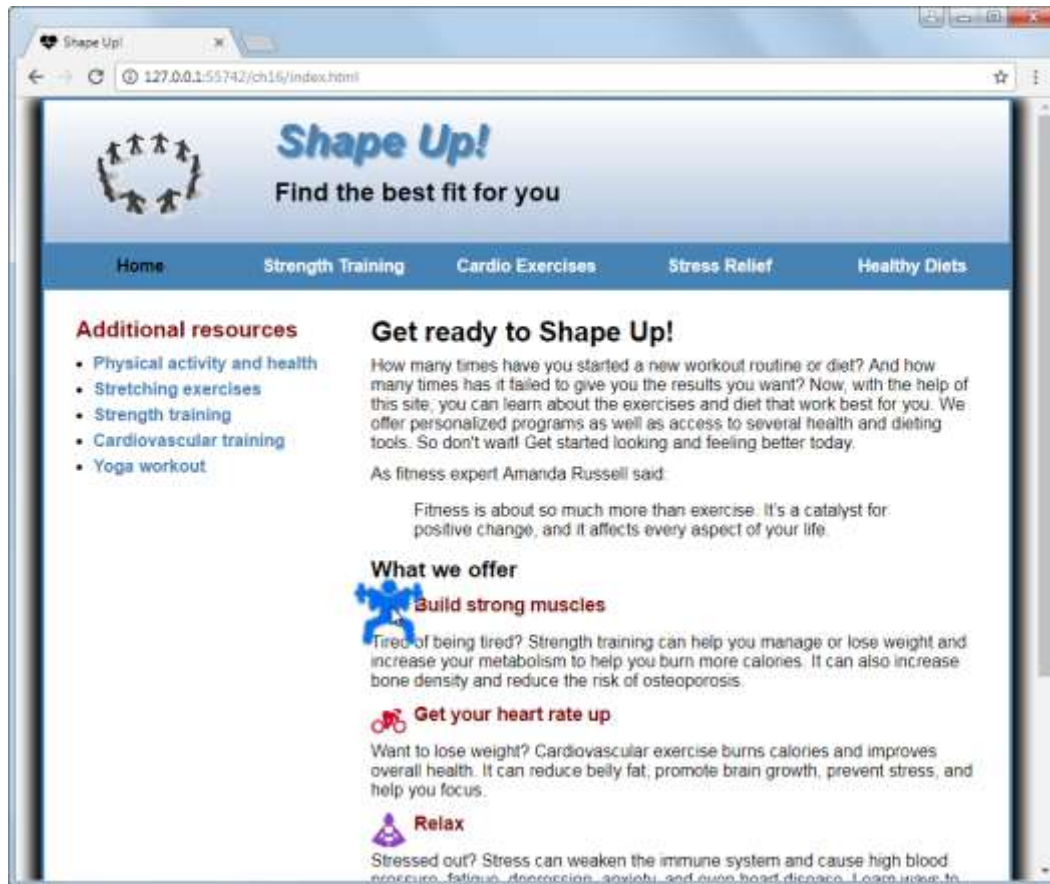
## Shape Up! 16   Add a transform, a transition, and an animation

In this exercise, you'll add a transform, a transition, and an animation to the home page that you worked on in exercise 11.

Specifications for the transform

- Open the index.html page in the root folder. Then, add a transform that will cause the icons in the section to be displayed at two times their initial sizes when the mouse hovers over them.

- Add a transition that will cause the transform to last for two seconds. Be sure to provide for all browsers. When the transform is performed, note that the icon doesn't affect the content around it, as shown above.

Specifications for the transition
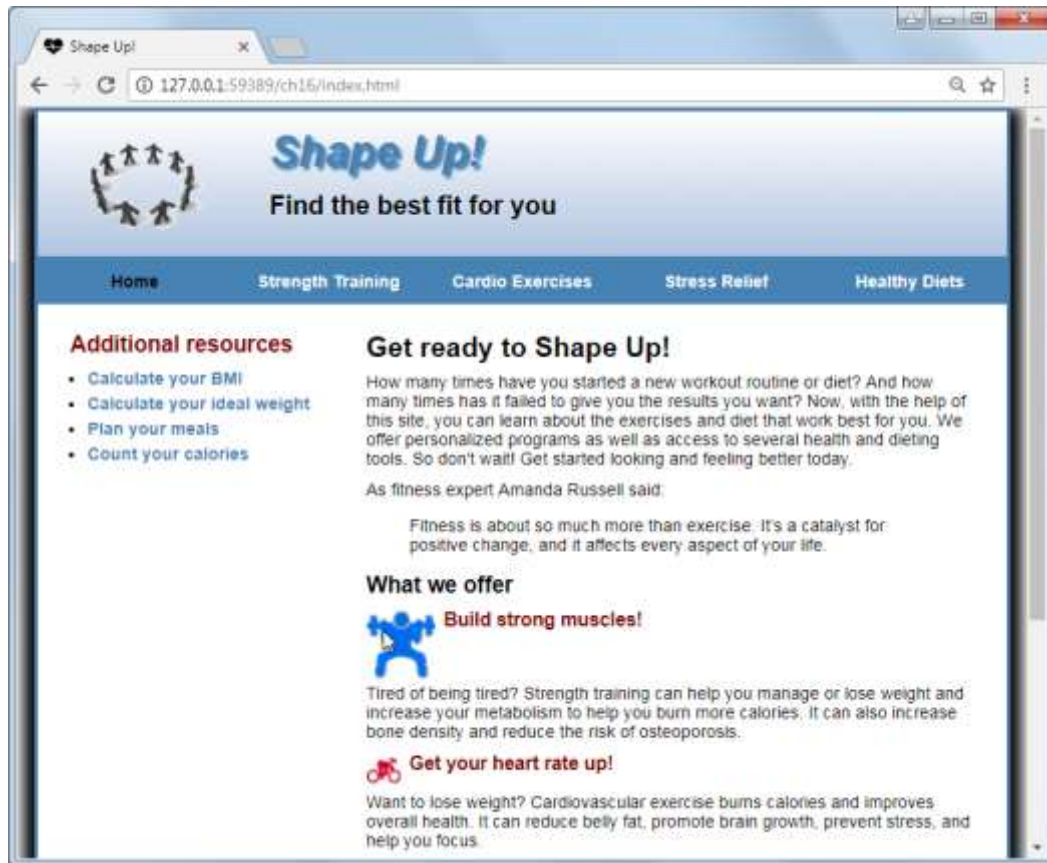
- Comment out the code you just added for the transform and transition.

- Add a transition for the widths of the images that will last for two seconds and use the ease speed curve. The transition should increase the width of an image to 64px when the mouse hovers over the image. (For this to work, you'll need to set the initial widths of the icons to their native size of 32px.) Be sure to provide for all browsers. When the transition is performed, note that, unlike the transform, the transition does affect the content around it, as shown above.

**Specifications for the animation**

- Add an animation for the h2 heading in the header that will cause the heading to move to the right side of the page and then back to the left when the page is loaded. The animation should last for eight seconds, use the ease-in-out speed curve, and occur just one time.

Shape Up! 19   Use JavaScript and jQuery to calculate BMI and add image swaps

In this exercise, you'll create a page that uses JavaScript to calculate BMI. In addition, you'll create a diet page that uses jQuery for image swaps with ideas for replacing unhealthy foods with healthy foods.

Specifications for the BMI calculation
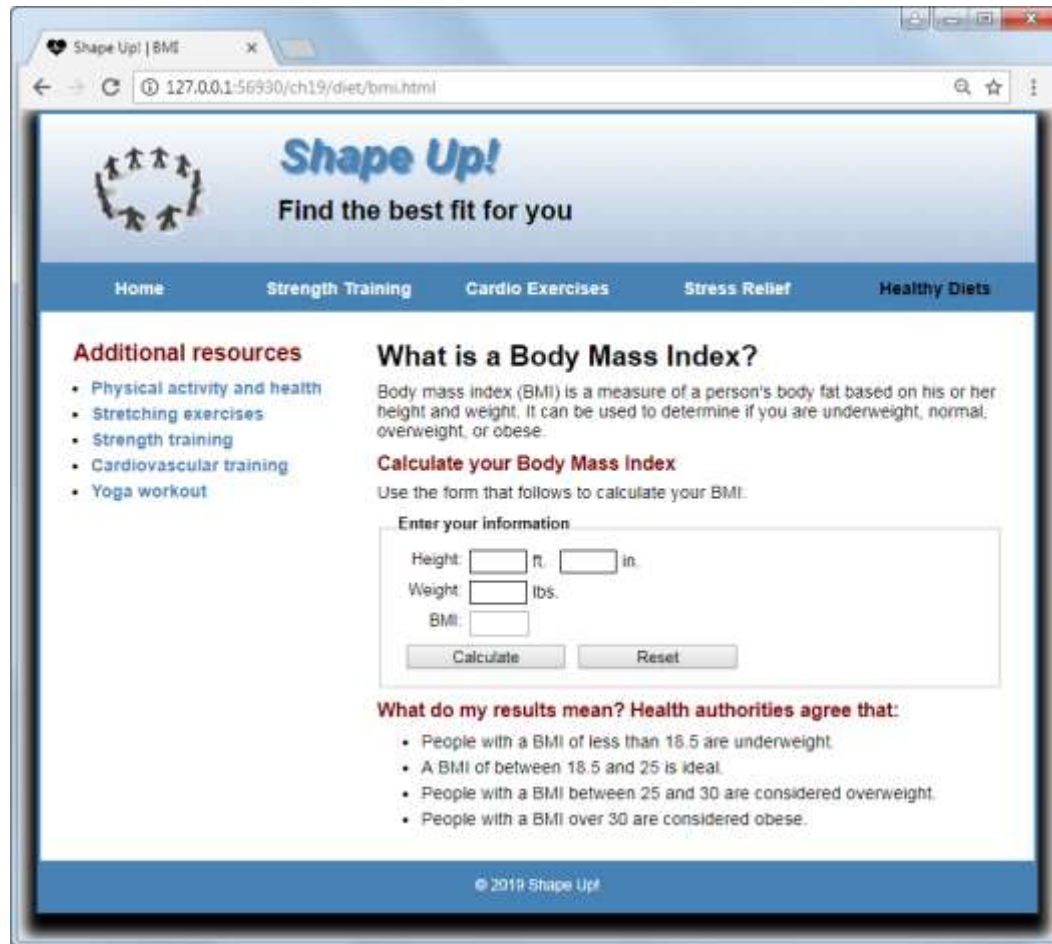
- Open the bmi.html file in the diet folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the link elements for the favicon and style sheets, as well as the code for the header, footer, navigation menu, and sidebar, from the index.html page for exercise 7, and add a section element before the sidebar.

- Modify the navigation menu so it indicates that one of the healthy diets pages is the current page, and modify the links in the menu as necessary.

- Create a new style sheet named bmi.css, and copy the styles you need from the main.css file to this style sheet. Then, modify the link element for the style sheet in the bmi.html file so it points to the correct style sheet.

- Add the content of the bmi.txt file in the text folder to the section. Then, add the form and controls shown above. Note that because JavaScript will be used to process the information that's entered, it's not necessary to include action or method attributes. Also note that the value in the BMI text box shouldn't be able to be changed.

- Add a script element to the bmi page that includes the bmi.js file in the js folder.

- Open the bmi.js file and review its code. Use this code to determine the required ids of the four text boxes and the Calculate button in the form on the bmi page. Then, set the ids of these controls. In addition, set the for attributes of the labels associated with the text boxes.

Specifications for the diet page

- Open the index.html page in the diet folder. This file contains the basic content for an "under construction" page.

- Delete the two headings in the main element. Then, copy the code for the header, footer, navigation menu, and sidebar from the index.html page for exercise 7, and add a section element before the sidebar.

- Modify the navigation menu so it indicates that one of the healthy diets pages is the current page, and modify the links in the menu as necessary.

- Create a new style sheet named diet.css, and copy the styles you need from the main.css file to this style sheet. Then, modify the link element for the style sheet in the index.html file so it points to the correct style sheet.

- Add the content of the diet.txt file in the text folder to the section. Then, add the script elements that are needed to create the image swaps. You'll find the image_swap.js file in the js folder, and you can get the jQuery library from the jQuery Content Delivery Network.

- Add the heading and text, followed by the image swaps, to the bottom of the section. You'll find the image files you need in the images/foods folder.

- Add any other formatting as necessary so the page looks as shown above.