

Short exercises for *Murach's HTML5 and CSS3* (4th edition)

Each of the short exercises requires you to use just one or two web development skills, and each is designed so it can be done in from 5 to 45 minutes.

Guidelines for doing the short exercises

- Do the exercise steps in sequence. That way, you will work from the most important tasks to the least important.
- Feel free to copy and paste code from the book applications, book examples, or exercises that you've already done.
- Use your book as a guide to coding.
- If you are doing an exercise in class with a time limit set by your instructor, do as much as you can in the time limit.
- Because each short exercise is independent of the others, many of the links in the pages that you will be given to start the exercises won't work. To provide for that, the href attribute for each of those links is coded as a hash character (#). That way, you won't get a "file not found" error if you click the link. Instead, you'll just stay on the same page.

Short 3-1 Create an HTML page for a speaker

In this exercise, you'll create a semantically correct, HTML page that looks like the one below. Estimated time: 20-30 minutes.

Andrew Ross Sorkin, author of *Too Big to Fail*

November 2015



New York Times columnist and author, Andrew Ross Sorkin, has been described as "the most famous financial journalist of his generation." His book, *Too Big to Fail*, was the first behind-the-scenes account of the financial crisis that led to our current recession.

The Economist, The Financial Times, and Business Week all named *Too Big To Fail* one of the best books of the year. The book was published by Viking October 20, 2009. The book was adapted as a movie by HBO Films and premiered on HBO on May 23, 2011.

The cast of the movie: *Too Big to Fail*

1. William Hurt as Hank Paulson
2. Paul Giamatti as Ben Bernanke
3. Billy Crudup as Timothy Geithner

Movie reviews

"Lots of heat but not very much light. It came across as a kind of daytime soap opera version showing the personalities of the players rather than dealing with the nuts and bolts of what really happened. Even that would not have been so bad but the movie tries to portray lots of these folks as sympathetic figures just trying to do the right thing."

"I am watching this for the 3rd time, trying to understand how this large group of capitalist pigs could royally screw their families, descendants, neighbors, countrymen and the rest of the whole world with absolutely no compassion."

Prepared by: Your Name

Date: Today's Date

1. Open this HTML file:

`short_exercises\town_hall\speakers\c03x_sorkin.html`

Note that it contains a head section and all the copy for the text in the body of the page, but the HTML tags haven't yet been applied to that text.

2. Apply the HTML tags to the text so the text will look like it does in the page above. Remember too that these tags should be semantically correct. The last two lines should be coded within a footer element, and everything else should be within a main element.
3. Add the image for the speaker: `sorkin_desk260.jpg`, which is 260 pixels wide.
4. Use character entities or inline formatting tags to add the quotation marks and italics that this page requires.

Short 5-1 Apply CSS to an HTML page

In this exercise, you'll apply the CSS skills that you learned in chapters 4 and 5 to an existing HTML document. The resulting page should look something like the screen shot that follows. Estimated time: 30 to 45 minutes.



1. Open the HTML and CSS files that follow, and note that the CSS file includes one rule set:


```
short_exercises\town_hall\speakers\c05x_sorkin.html
short_exercises\town_hall\styles\c05x_sorkin.css
```
2. Add a link element to the head section of the HTML file for the normalize.css style sheet.
3. If you want to use a reset selector, add that to the CSS file. But feel free to code the CSS in the way that you prefer.

4. Code a rule set for the `html` element that sets the background color to yellow.
5. Enhance the rule set for the `body` so the width is 650 pixels, the body is centered in the browser window, and the body has a double blue border around it like the one above. If you need to make any other changes to the body, do that too.
6. Code a rule set for the `main` element that puts padding around its contents. Then, code a rule set for the footer that puts a blue border above it. Note that this border doesn't touch the border for the body.
7. Code rule sets for the `h1`, `h2`, and `h3` elements. The `h1` font should be 150% of the default specified in the body, the `h2` font should be 125% of the default font, and the `h3` font should be 115% of the default font. The `h1` font should also be blue. Then, apply appropriate margins or padding to the `h1`, `h2`, and `h3` elements so the spacing before and after the headings is similar to what's shown above.
8. Code the rule sets for the `<p>`, `blockquote`, `ul`, and `li` elements so the spacing before and after the elements is similar to what's shown above.
9. Code a rule set for the `cite` element that changes its color to blue and removes the italics from the text.
10. Code a rule set for the paragraphs that contain `cite` elements. This rule set should right align the paragraphs and increase the spacing below to .75em. One way to do this is to add a class attribute to these paragraphs and use that class as the selector for the rule set.
11. Float the image to the left and apply appropriate margins or padding so the text flows to its right as shown above.
12. Apply rules to the footer or the paragraphs within the footer so the font size is 90% of the default, the font weight is bold, the paragraphs are centered, and the spacing above and below is similar to what's shown above.

Short 6-1 Use the CSS3 columns feature

In this exercise, you'll float the speaker image to the left, adjust the formatting as needed, and apply two-column formatting to the article. Estimated time: 5-10 minutes.



1. Open these HTML and CSS files, and run the HTML file:
`short_exercises\town_hall\speakers\c06x_toobin.html`
`short_exercises\town_hall\styles\c06x_speaker.css`
2. Float the image to the left instead of the right and adjust the space around the image.
3. Apply two-column formatting to the article using the column-count property as in figure 6-10. If this creates any formatting problems, adjust the HTML or the CSS so the page looks like the one above. One hint: Use CSS to set the width of the image to 275 pixels so it fits in one column.

Short 6-2 Switch the columns of a page

In this exercise, you'll switch the section and aside of a speaker page so it looks like the one below. That will demonstrate your understanding of floating, margins, and padding. Estimated time: 5-10 minutes.



1. Open the HTML and CSS files for the page:
`short_exercises\town_hall\speakers\c06x_toobin.html`
`short_exercises\town_hall\styles\c06x_speaker.css`
2. Change the style rules for the section and aside so the columns are switched.
3. Adjust the margins and padding for the section and aside so the formatting is similar to the formatting shown above.

Short 6-3 Add a third column to a page

In this exercise, you'll add a third column to an index page. That will demonstrate your understanding of floating, margins, and padding. Estimated time: 10-15 minutes.



1. Open the HTML and CSS files for the page:
`short_exercises\town_hall\c06x_index.html`
`short_exercises\town_hall\styles\c06x_main.css`
2. Note that another aside has been added to the HTML page. That aside contains the heading and text shown in the third column above.
3. If necessary, rearrange the code in the HTML file so the page can be presented in three columns as shown above.
4. Modify the code in the CSS file so it provides for the three columns with the approximate spacing shown above.

Short 7-1 Start an email

In this exercise, you'll add a paragraph at the bottom of the speaker's page that will start an email. Estimated time: 5-10 minutes.



San Joaquin Valley Town Hall

Celebrating our 75th Year

Guest speakers

October
[Jeffrey Toobin](#)



November
[Andrew Ross Sorkin](#)



January
[Amy Chua](#)



February
[Scott Sampson](#)



[Return to Home page](#)

Fossil Threads in the Web of Life

February
Scott Sampson

What's 75 million years old and brand spanking new? A teenage Utahceratops! Come to the Saroyan, armed with your best dinosaur roar, when Scott Sampson, Research Curator at the Utah Museum of Natural History, steps to the podium. Sampson's research has focused on the ecology and evolution of late Cretaceous dinosaurs and he has conducted fieldwork in a number of countries in Africa.



Scott Sampson is a Canadian-born paleontologist who received his Ph.D. in zoology from the University of Toronto. His doctoral work focused on two new species of ceratopsids (horned dinosaurs) from the Late Cretaceous of Montana, as well as the growth and function of ceratopsid horns and frills.

Following graduation in 1993, Sampson spent a year working at the American Museum of Natural History in New York City, followed by five years as assistant professor of anatomy at the New York College of Osteopathic Medicine on Long Island. He arrived at the University of Utah accepting a dual position as assistant professor in the Department of Geology and Geophysics and curator of vertebrate paleontology at the Utah Museum of Natural History. His research interests largely revolve around the phylogenetics, functional morphology, and evolution of Late Cretaceous dinosaurs.

In addition to his museum and laboratory-based studies, Sampson has conducted paleontological work in Zimbabwe, South Africa, and Madagascar, as well as the United States and Canada. He was also the on-air host for the Discovery Channel's Dinosaur Planet and recently completed a book, *Dinosaur Odyssey: Fossil Threads in the Web of Life*, which is one of the most comprehensive surveys of dinosaurs and their worlds to date.

For up-to-date information on the luncheon, please [email us](#).

© 2018, San Joaquin Valley Town Hall, Fresno, CA 93755

1. Open this HTML file and run it:

`short_exercises\town_hall\speakers\c07x_sampson.html`

2. In the HTML file, add the boldfaced paragraph at the bottom of the article (use a `` element to apply the boldfacing). When the user clicks on "email us", an email should be started that is populated with this data:

To: `eleanor@townhall.com`

CC: `georgia@townhall.com`

Subject: `Scott Sampson luncheon`

Reminder: When you populate two or more fields, you separate them with ampersands (&).

Short 7-2 Create a vertical navigation menu

In this exercise, you'll create a vertical navigation menu in the sidebar as shown below. Note that each item in the menu includes a right-arrow symbol, which is actually a background image. Estimated time: 20-30 minutes.



1. Open the HTML and CSS files that follow and display the page in a browser:
`short_exercises\town_hall\speakers\c07x_sampson.html`
`short_exercises\town_hall\styles\c07x_speaker.css`
2. Modify the HTML so the sidebar doesn't include the speaker images and so both the month and name for each speaker are in an `<a>` element. Then, enclose the `<a>` elements within the `li` elements of an unordered list as shown in figure 7-14, and enclose the `ul` element in a `nav` element that has "nav_list" as its id.
3. Add the CSS for formatting the navigation menu. To speed that up, you may want to copy the code from the example for figure 7-13 and then make the appropriate adjustments so the links are rectangular with a black border.
4. Add rounded corners to the links as well as box shadows with #800000 as the color. The current link should also have #800000 as its color. As you do this, you may want to refer to figure 5-10.
5. Add the symbols to the links. To do that, use the `background-image`, `background-repeat`, and `background-position` properties as shown in figure 5-11. The symbol is in the images file, and it is named `right.jpg`. In this case, you don't want to repeat the image, and you want to position it in the middle vertically and about 95% from the left.
6. If necessary, adjust the margins, spacing, and font sizes for any of the elements.

Short 8-1 Create a fluid design

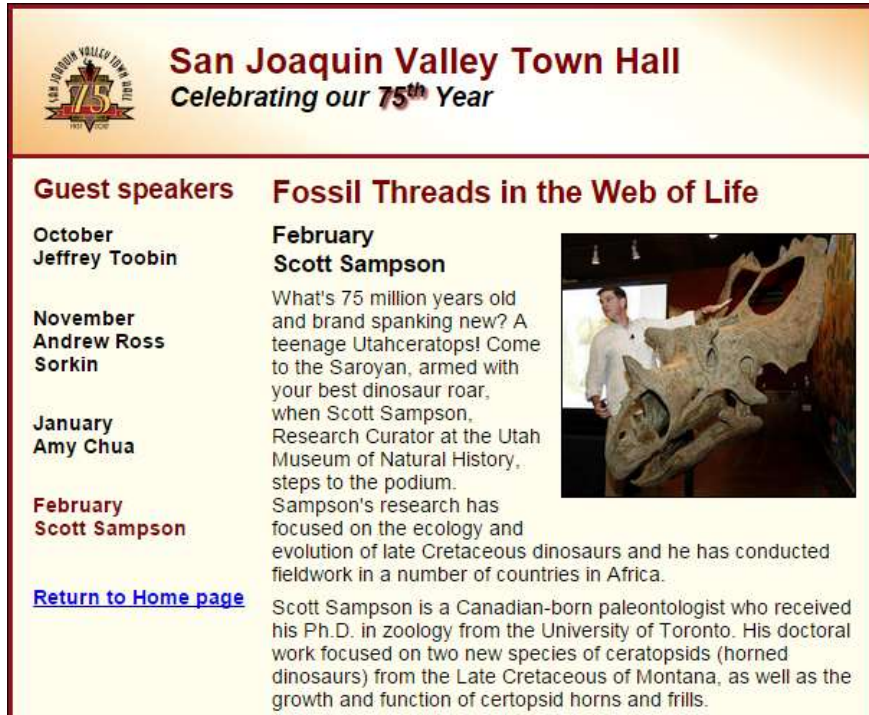
In this exercise, you'll convert a page so it uses a fluid layout and scalable images. When you're done, the page will look as shown below when the width of the browser window is reduced. Estimated time: 20 to 30 minutes.



1. Open the HTML and CSS files that follow and display the page in a browser:
`short_exercises\town_hall\speakers\c08x_sampson.html`
`short_exercises\town_hall\styles\c08x_speaker.css`
2. Reduce the width of the browser window so it's less than the width of the page. That way, you'll be able to see how the page changes as you convert it to a fluid design.
3. Modify the CSS for the body so it takes up 98% of the browser window and so its maximum width is the current width.
4. Convert the widths of the section, the aside, and the li elements within the navigation menu to percentages.
5. Convert any left or right margins or padding to percentages. (Hint: If any measurements are specified in ems, you can convert them to pixels by multiplying by 16 since that's the base font size.)
6. Make the image in the article scalable so it's always 50 percent of the width of the article.

Short 8-2 Add a media query

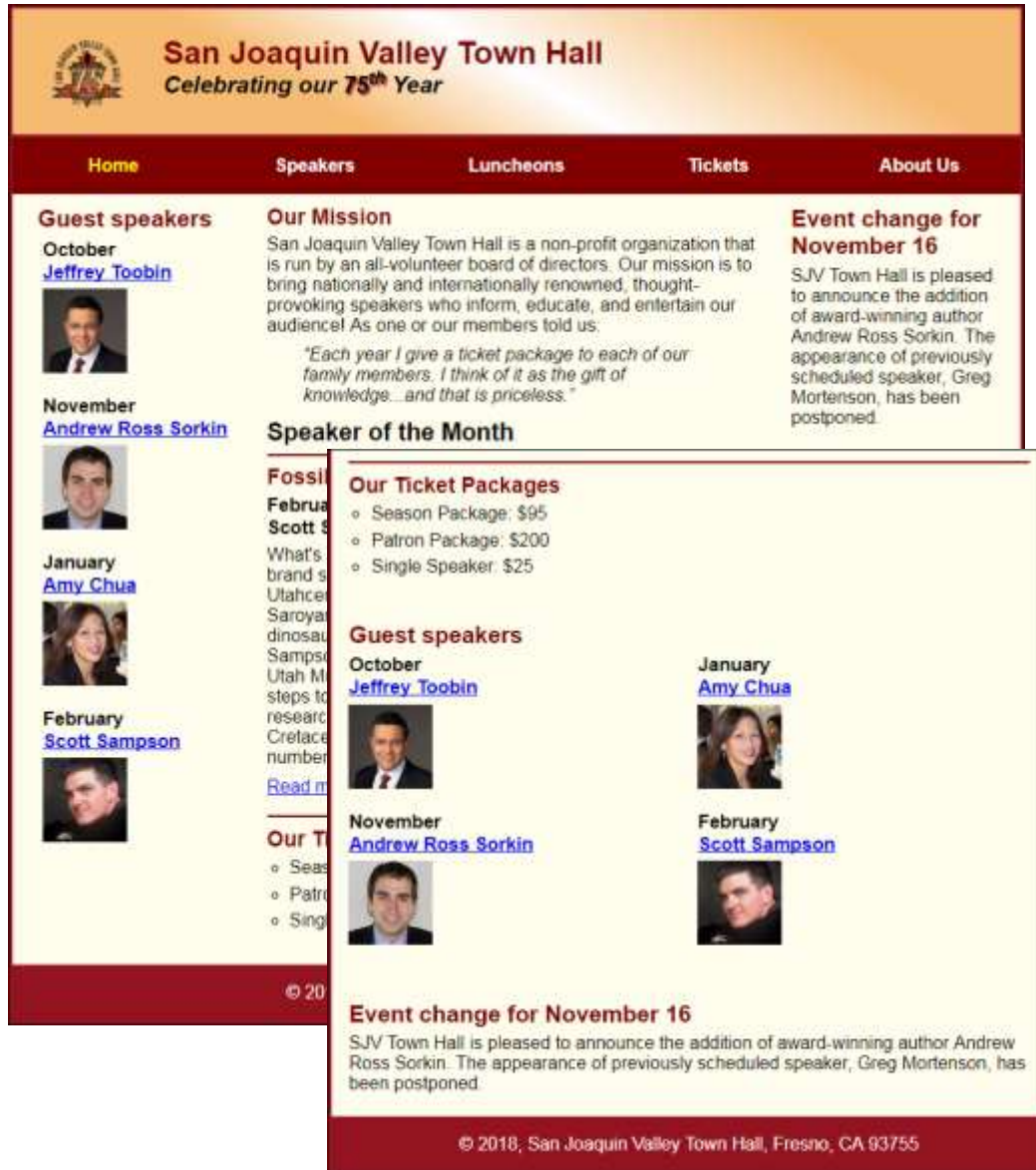
In this exercise, you'll add a media query to a page so it provides for tablets in portrait orientation. When you're done, the page will look as shown below when the width of the browser window is reduced. Estimated time: 10 to 15 minutes.



1. Open the HTML and CSS files that follow and display the page in a browser:
`short_exercises\town_hall\speakers\c08x_sampson.html`
`short_exercises\town_hall\styles\c08x_speaker.css`
2. Add a meta element to the HTML that sets the width property for the viewport to the device width and the initial zoom factor to 1.
3. Code a media query for the screen media type that has a maximum width of 850 pixels.
4. Within the media query, change the base font size for the page so it's 90% of the default font size.
5. Remove the border, box shadow, and background image from the list items in the sidebar.
6. Remove the left padding from the links in the sidebar.

Short 9-1 Use Flexible Box Layout to add a third column to the Town Hall home page

In this exercise, you'll add a third column to a Town Hall home page that uses flexbox so it works in both desktop and mobile layouts. Estimated time: 20 to 30 minutes.



1. Open the HTML and CSS files that follow:

`short_exercises\town_hall\c09x_index.html`

`short_exercises\town_hall\styles\c09x_main.css`

2. Note that a second aside has been added to the HTML after the first aside and that both asides have been given an id. The new aside contains the heading and text shown in the third column above.
3. Test the index page to see that the second aside is displayed between the first aside and the section.
4. In the CSS file, note that the main element is implemented as a flexbox and that the left aside and the section elements contain flex properties that specify their initial width and order.
5. Modify the CSS so the flex items in the flexbox for the main element are laid out as shown above in both desktop and mobile layouts.

Short 10-1 Use a 12-column grid for the Town Hall home page

In this exercise, you'll modify the Town Hall home page so it uses a 12-column grid instead of template areas for the body of the page in desktop layout. However, the elements within the body like the navigation menu and the main element will continue to use template areas for their components, both in desktop and mobile layout. Estimated time: 20 to 30 minutes.



1. Open the HTML and CSS files that follow:
`short_exercises\town_hall\c10x_index.html`
`short_exercises\town_hall\styles\c10x_main.css`
2. Test the index page to see that it works correctly.
3. In the CSS file, note that the code for the body area sets up a grid with template areas that are used for the header, navigation menu, main element, and footer. In addition, grids with template areas are used to lay out the navigation menu and main element in desktop layout and the header in mobile layout.
4. Modify the CSS so it uses the 12-column grid concept for the body element in desktop layout, but retains the grids that use template areas for the other elements.

Short 11-1 Do an image rollover with CSS

In this exercise, you'll do an image rollover by using background images. When you're done, the page will look as shown below before and after the rollover. Estimated time: 5-10 minutes.



1. Open these HTML and CSS files, and run the HTML file:
`short_exercises\town_hall\speakers\c11x_sampson.html`
`short_exercises\town_hall\styles\c11x_rollovers.css`
 Note that it displays the image for Scott Sampson (sampson_dinosaur.jpg), as shown above.
2. Modify the HTML and CSS as shown in figure 11-7 so the image rolls over to the image for Andrew Ross Sorkin (sorkin_desk260.jpg) when the user hovers the mouse over the first image. To make this work, set the height and width properties for the `<p>` element to 260px. Because the images aren't quite that size, you'll also need to set the background-repeat property to no-repeat.

Short 12-1 Enhance a table

In this exercise, you'll start from a table that looks like this:

Name	E-mail	Years of Service
Joel Murach	joelmurach@yahoo.com	22
Anne Boehm	anne@murach.com	34
Zak Ruvalcaba	zak@modulemedia.com	4
Judy Taylor	judy@murach.com	39
Cyndi Vasquez	cyndi@murach.com	10
Kelly Slivkoff	kelly@murach.com	25
Juliette Baylon	juliette@murach.com	1

Then, you'll expand and format the table so it looks like the one that follows. Estimated time: 20-30 minutes.

Employee Table			
Department	Name	E-mail	Years of Service
Editorial	Joel Murach	joelmurach@yahoo.com	22
	Anne Boehm	anne@murach.com	34
	Zak Ruvalcaba	zak@modulemedia.com	4
Marketing	Judy Taylor	judy@murach.com	39
	Cyndi Vasquez	cyndi@murach.com	10
Customer Service	Kelly Slivkoff	kelly@murach.com	25
	Juliette Baylon	juliette@murach.com	1
Total Years of Service			105

1. Open this HTML file and run it:
`short_exercises\town_hall\c12x_employee_table.html`
 Note that this HTML file uses embedded CSS in the head section to do the formatting.
2. Expand the HTML so the table has the caption and data shown in the second table above.
3. Expand the CSS so it does the formatting shown in the second table above.
4. If you used classes to do the alignment and apply the colors to the cells of the table, try using pseudo-classes to get the same results.

Short 13-1 Create a form

In this exercise, you'll complete a form so it looks as shown below. Estimated time: 20-30 minutes.

1. Open this HTML file and run it to see that it already provides the first two headings, the first three fields, the last heading, and the two buttons:
`short_exercises\town_hall\c13x_forms.html`
2. Modify the attributes for the form so it uses the “get” method and submits the form to the file named `survey_data.html`. Then, test the submission of the completed form. This should display a page that shows the submitted data.
3. Add the autofocus attribute to the first field and the required attribute to all three of the fields. Then, test to make sure the data validation works.
4. Add the “Geographic information” heading and fields. These fields should also be required and they should have the placeholders that are shown above. To validate the state and zip code fields with regular expressions, you can use these patterns:
`[A-Za-z]{2}` and `\d{5}`
5. Add the “How did you hear about us” heading and checkbox fields. Use labels to provide for user accessibility as shown in figure 13-8.
6. Enhance the CSS in the HTML file so the form looks like the one above. In particular, you'll need to change the width of the checkboxes so they will align right, and you should remove the box shadow from required and invalid fields so they won't have red shadows when displayed in Firefox.
7. Do a final test to make sure everything works as it should.

Short 14-1 Embed audio in a page

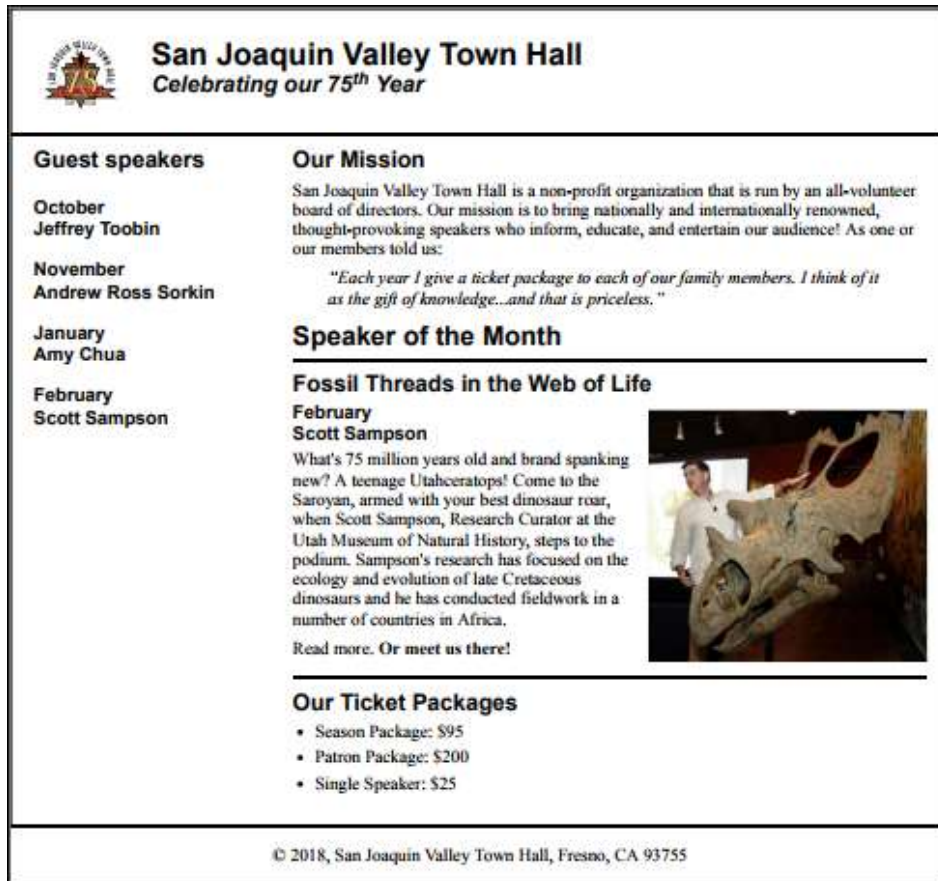
In this exercise, you'll add audio to the top of the section in a speaker's page, but the audio controls won't show so the page will look the same. However, the audio will play automatically after the page is loaded. Estimated time: 5-10 minutes.



1. Open this HTML file:
`short_exercises\town_hall\speakers\c14x_sampson.html`
2. Add an audio element like the one in figure 14-8 that plays the welcome.mp3 file in the media folder. Then, using that figure as a guide, modify the code so the audio plays once after starting automatically.

Short 15-1 Format the home page for printing

In this exercise, you'll format the home page of the Town Hall website for printing. Estimated time: 15-20 minutes.



1. Open the HTML file for this page and the CSS file for printing it:
`short_exercises\town_hall\c15_index.html`
`short_exercises\town_hall\styles\home_print.css`
 Note that the CSS file is a copy of the main.css file that is used to format the index page for screens.
2. In the HTML file, add a link element for the home_print.css file and be sure to specify that the medium is "print".
3. In the CSS file, delete all of the rule sets and rules that you don't need to override, and code the rule sets or rules that will stop the display of the navigation menu and the images in the sidebar.
4. Increase the font size in the body element to 100% so the page will be easier to read when it's printed.
5. Modify or add styles to the home_print.css file until the printed page looks like the one above.

Short 16-1 Use a CSS3 transition and transform

In this exercise, you'll use a transition and a transform to rotate elements when the mouse hovers over them as shown below. Estimated time: 10 to 15 minutes.



1. Open these HTML and CSS files, and run the HTML file:
`short_exercises\town_hall\speakers\c16x_sampson.html`
`short_exercises\town_hall\styles\c16x_speaker.css`
2. Add CSS for a transform that will rotate the list items in the sidebar horizontally 360 degrees when the mouse hovers over them.
3. Add CSS for a transition that will cause the transform to last for 3 seconds.

Short 16-2 Use CSS3 animation

In this exercise, you'll animate the image in the page header so it appears to bounce from left to right. Estimated time: 10 to 15 minutes.



1. Open these HTML and CSS files, and run the HTML file:
`short_exercises\town_hall\speakers\c16x_sampson.html`
`short_exercises\town_hall\styles\c16x_speaker.css`
2. Add CSS to the image in the header that will use an animation named “bounce” with a duration of half a second that will repeat eight times, use the ease-in timing function, and alternate direction.
3. Add an @keyframes selector rule that defines the keyframes for the animation. In the from group, set the left and right padding so the animation starts at the image's current position on the page. In the to group, set the left and right padding so the image moves to the right 20 pixels.

Short 19-1 Do image rollovers using jQuery

In this exercise, you'll use jQuery to do image rollovers without knowing how the JavaScript and jQuery work. In the aside for the web page, each speaker image should roll over when the mouse pointer hovers over it. Estimated time: 15-20 minutes.



1. Open these HTML and CSS files, and run the HTML file:
`short_exercises\town_hall\speakers\c19x_sampson.html`
`short_exercises\town_hall\styles\c19x_speaker.css`
2. Add a script element like the one in figure 19-12 to the head section of the HTML document that loads the jQuery library from a Content Delivery Network.
3. Add another script element to the head section for the rollover.js file that's in the javascript folder.
4. To set up the HTML so the rollovers work, you first need to set the id attribute of the ul element in the side bar that contains the images to "image_rollovers".
5. Then, you need set the id attribute of each image to the rollover image. The rollover images are named:

toobin_court.jpg
 sorkin_desk260.jpg
 chua_220.jpg
 sampson_dinosaur.jpg

Because these images are larger than the initial images, you'll need to set the image heights to 75 pixels in the CSS.