## Case Problem 1

**Data Files needed for this Case Problem: sp_home_txt.html, sp_layout_txt.css, 2 CSS files, 11 PNG files**

**Slate & Pencil Tutoring**     Karen Cooke manages the website for *Slate & Pencil Tutoring*, an online tutoring service for high school and college students. Karen is overseeing the redesign of the website and has hired you to work on the layout of the site's home page. Figure 3–84 shows a preview of the page you'll create for Karen.

**Figure 3–84**     **Slate & Pencil Tutoring home page**



© Monkey Business Images/Shutterstock.com;
© Courtesy Patrick Carey

Karen has supplied you with the HTML file and the graphic files. She has also given you a base style sheet to initiate your web design and a style sheet containing several typographic styles. Your job will be to write up a layout style sheet according to Karen's specifications.

Complete the following:

1. Using your editor, open the **sp_home_txt.html** and **sp_layout_txt.css** files from the html03 ▶ case1 folder. Enter *your name* and *the date* in the comment section of each file, and save them as **sp_home.html** and **sp_layout.css** respectively.

2. Go to the **sp_home.html** file in your editor. Within the document head, create links to the **sp_base.css**, **sp_styles.css**, and **sp_layout.css** style sheet files. Study the content and structure of the file and then save your changes to the document.

3. Go to the **sp_layout.css** file in your editor. Go to the Window and Body Styles section. Create a style rule for the `html` element that sets the height of the browser window at 100%.

4. Create a style rule for the page body that sets the width to 95% of the browser window ranging from 640 pixels up to 960 pixels. Horizontally center the page body within the browser window. Finally, Karen wants to ensure that the height of the page body is always at least as high as the browser window itself. Set the minimum height of the browser window to 100%.

5. Add a style rule to display all inline images as blocks.

6. Within the CSS Grid Styles section create a style rule that displays the `body` element as a grid with four columns of length 1fr.

7. Create a style rule for the `img` element with id "logo" so that the logo image spans three columns and has a width of 100%.

8. For the horizontal navigation list and the `footer` element create a style rule so that those elements span four columns. Create a style for the `aside` element to span two columns.

9. Within the Horizontal Navigation List Styles section create a style rule for `li` elements nested within the horizontal navigation list that display each element as a block with a width of 12.5% and floated on the left margin.

10. Within the Section Styles section create a style rule for inline images within the `section` element that sets the width of the image to 50% and centers the image using a top/bottom margin of 0 and a left/right margin of `auto`.

11. Create a style rule for paragraphs within the `section` element that sets the width of the paragraph to 70% and centers the paragraph using a top/bottom margin of 0 and a left/right margin of `auto`.

12. Go to the Customer Comment Styles section and create a style rule for the `aside` element setting the width to 75% and the bottom padding to 30 pixels.

13. The six `aside` elements will be displayed in two columns. For odd-numbered `aside` elements, use the `justify-self` grid property to place the element on the end (right) margin. (Hint: Use the `nth-of-type(odd)` pseudo-class to select the odd-numbered `aside` elements.)

14. Float inline images nested within the `aside` element on the left with a width of 20%.

15. Float paragraphs nested within the `aside` element on the left with a width of 75% and a left margin of 5%.

16. Save your changes to the file and then open the **sp_home.html** file in your browser. Verify that the layout and appearance of the page elements resemble that shown in Figure 3–84.

## Case Problem 2

**Data Files needed for this Case Problem: ss_dday_txt.html, ss_layout_txt.css, 1 CSS file, 3 PNG files**

*A Soldier's Scrapbook*   Jakob Bauer is a curator at the Veteran's Museum in Raleigh, North Carolina. Currently he is working on an exhibit called *A Soldier's Scrapbook* containing mementos, artifacts, journals, and other historic items from the Second World War. You've been asked to work on a page for an interactive kiosk used by visitors to the exhibit. Jakob has already supplied much of the text and graphics for the kiosk pages but he wants you to complete the job by working on the page layout.

The page you will work on provides an overview of the Normandy beach landings on June 6th, 1944. Since this page will be displayed only on the kiosk monitor, whose screen dimensions are known, you'll employ a fixed layout based on a screen width of 1152 pixels.

Jakob also wants you to include an interactive map of the Normandy coast where the user can hover a mouse pointer over location markers to view information associated with each map point. To create this effect, you'll mark each map point as a hypertext link so that you can apply the `hover` pseudo-class to the location. In addition to the interactive map, Jakob wants you to create a drop cap for the first letter of the first paragraph in the article describing the Normandy invasion. Figure 3–85 shows a preview of the page you'll create.