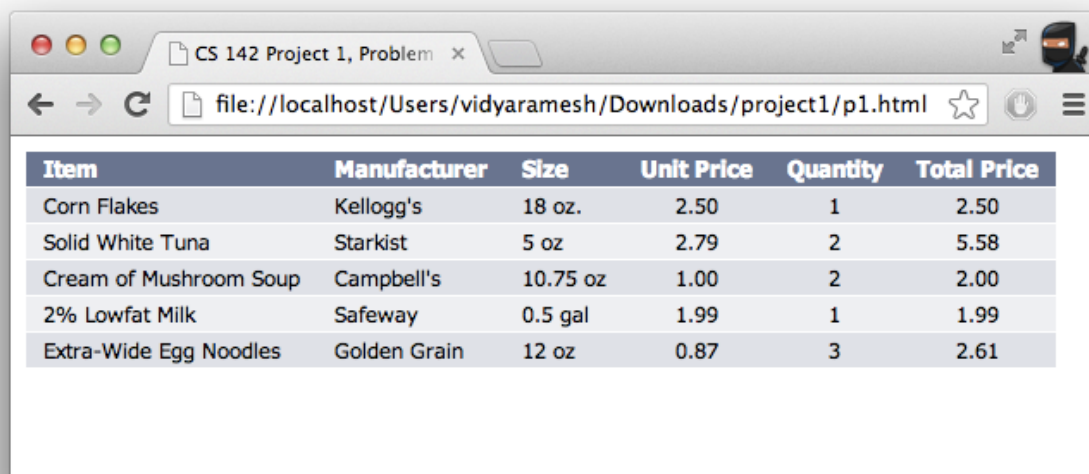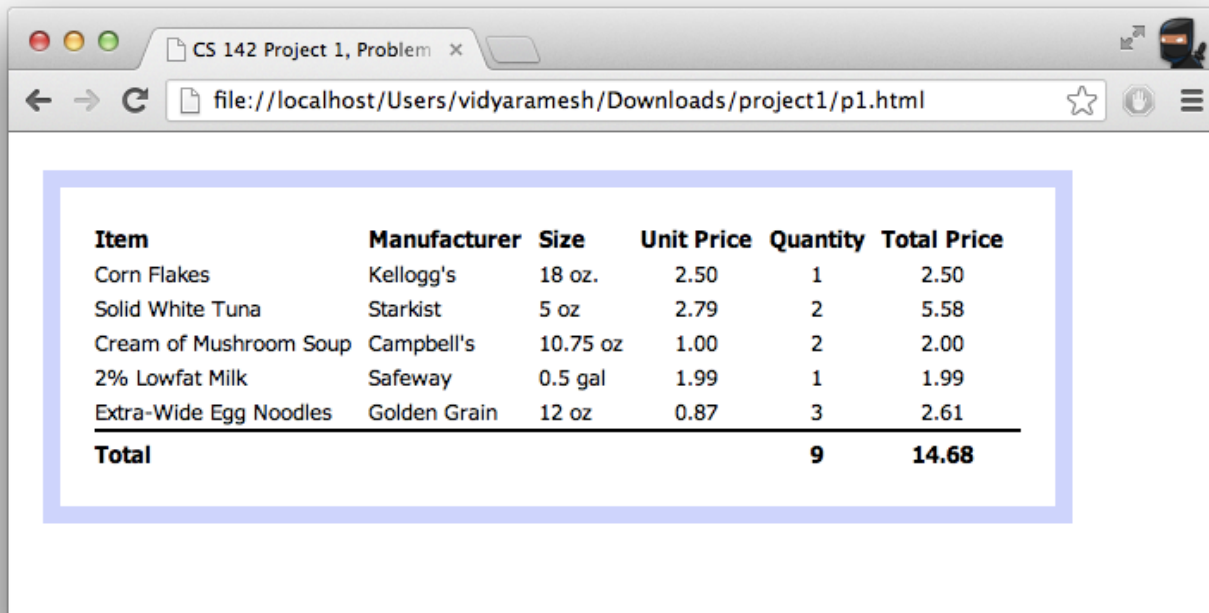# CS142 Project #1: HTML and CSS

Problem 1 (10 points)

Create a single HTML document that presents two different appearances, determined by the document's CSS stylesheet. Your HTML file should be called p1.html and the two stylesheets should be called p1a.css and p1b.css. If the HTML file links to p1a.css then it should appear like this ("Version A"), assuming you are running Chrome 26.0.1410.43 on a MacOS machine:

Item	Manufacturer	Size	Unit Price	Quantity	Total Price
Corn Flakes	Kellogg's	18 oz.	2.50	1	2.50
Solid White Tuna	Starkist	5 oz	2.79	2	5.58
Cream of Mushroom Soup	Campbell's	10.75 oz	1.00	2	2.00
2% Lowfat Milk	Safeway	0.5 gal	1.99	1	1.99
Extra-Wide Egg Noodles	Golden Grain	12 oz	0.87	3	2.61

If the HTML file links to p1b.css then it should appear like this in Chrome 26.0.1410.43 on MacOS ("Version B"):

Item	Manufacturer	Size	Unit Price	Quantity	Total Price
Corn Flakes	Kellogg's	18 oz.	2.50	1	2.50
Solid White Tuna	Starkist	5 oz	2.79	2	5.58
Cream of Mushroom Soup	Campbell's	10.75 oz	1.00	2	2.00
2% Lowfat Milk	Safeway	0.5 gal	1.99	1	1.99
Extra-Wide Egg Noodles	Golden Grain	12 oz	0.87	3	2.61
Total				**9**	**14.68**

On other platforms you may see a slightly different appearance, due to font differences. Here are the corresponding screen dumps for a few other platforms:

- Windows 7, Chrome 26.0.1410.43, Version A
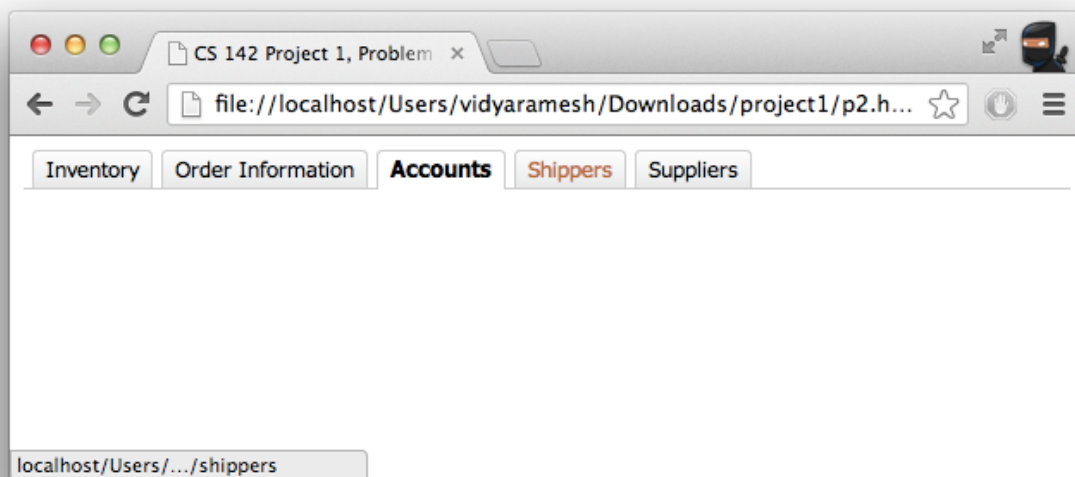- Windows 7, Chrome 26.0.1410.43, Version B

- Macintosh OSX, Firefox 11.0, Version A
- Macintosh OSX, Firefox 11.0, Version B
- Windows XP, Firefox, Version A
- Windows XP, Firefox, Version B

Here are some additional details and requirements for this problem:

- The content should be described in a single HTML file, using a `<table>` element to display the main table.
- There may not be any formatting information in the HTML file other than `class` and `id` attributes, and `cellspacing="0"` attributes on tables (see Hints below for more details on this).
- The appearance must be generated entirely with CSS style information; you may not use images or Javascript for this problem. Do not use CSS3 features in this project; CSS2 has everything you need, and is more standardized.
- The only change that should be required to switch from Version A to Version B is to change the `<link>` element in the header to refer to a different CSS file.
- Try to duplicate the appearance in the images above. Your solution doesn't need to be a "pixel perfect" duplicate, but it should be very close. For example:
  - Some of the columns should be centered whereas others are left-justified.
  - The "Total" line appears only in Version B (hint: you may find the `display` attribute useful in producing this effect).
  - The title in the browser title bar should read "CS 142 Project 1, Problem 1".
  - Both versions use the Tahoma font in a 13-pixel size.
  - The background color for the header row in Version A is #687291.
  - The background colors for the body rows in Version A are #eeeff2 and #dfe1e7.
  - The white lines between rows in Version A are 1 pixel wide.
  - The color for the frame around Version B is #d0d0ff.
  - The frame in Version B is 10 pixels wide; there are 20 pixels of empty space on either side of the frame.
  - The horizontal rule above the "Total" line in Version B is 2 pixels wide.
  - Match the paddings and spacings as closely as possible.
- Your HTML file must be a valid XHTML document that passes validation at http://validator.w3.org, and your CSS files must pass validation at http://jigsaw.w3.org/css-validator/
- Note: the `border` and `margin` styles are not supported for `<tr>` elements; `<td>` elements support `border` but not `margin`.

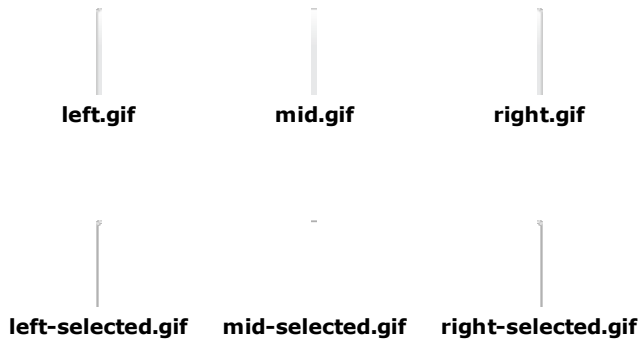## Problem 2 (10 points)

Create an HTML file p2.html and its corresponding CSS file p2.css, which will create a set of tabs that appear like this on MacOS with Chrome 26.0.1410.43:

For the corresponding appearance on Windows 7 with Chrome 26.0.1410.43, click here

In your solution for this problem you must use images in conjunction with the `background-image` CSS attribute to create the appearance of the tabs. We have generated 6 images for you to use. For tabs that are not selected there are three images, one for the left edge of the tab, one for the middle, and one for the right edge. There are three different images to use for the selected tab. Here are the images:

left.gif   **mid.gif**   **right.gif**

left-selected.gif   **mid-selected.gif**   **right-selected.gif**

To make a copy of any of these images for use in your solution, right-click over the image in Chrome and select "Save Image As...".

Here are some additional details and requirements for this problem:

- Use the Tahoma font in a 13-pixel size.
- The text in the selected tab should be displayed in bold.
- When the mouse moves over a tab the text color should change to #cc6600. In addition, the mouse cursor should appear as a pointing hand (cursor type `pointer`) when the mouse moves over a tab. In the screen dump above the mouse is over the "Shippers" tab; unfortunately the screen dump does not display the mouse cursor.
- The text in each tab should be a link that will switch to a different page when clicked. The URLS for the links should be, from left to right, "inventories", "orders", "accounts", "shippers", and "suppliers". You do not have to implement the Web pages corresponding to these links. Thus, when you click on a tab you will get a "missing page" error; in a real application each of these pages would display the same set of tabs, but with a different tab selected.
- The color of the horizontal line separating the tabs from the rest of the page is #cccccc. The line should extend from the rightmost tab to the right edge of the window, except for a small margin at the right side; the line length should adjust as the window is resized.
- The tabs should automatically resize both horizontally and vertically. If you replace the text in a tab with longer or shorter text, the tab should automatically get wider or narrower. If you change the font size for the tab text, the tabs should automatically get taller or shorter. Note that the images are taller than you will need in most cases; you should only display as much of each image as needed for the actual font size. You do not need to handle fonts that are too tall for the images we have provided.
- There may not be any formatting information in the HTML file other than `class` and `id` attributes, and `cellspacing="0"` attributes on tables (see Hints below for more details on this).
- You may not use Javascript for this assignment; you can solve this problem entirely with HTML and CSS.
- As with Problem 1, try to imitate the screen shots closely (but "pixel perfect" is not required).
- Note: browsers sometimes behave badly if a `<td>` contains nothing; for example, they may not display a background or border in this case. You can use an empty `<div>` inside the `<td>` to get around this problem.
- Your HTML file must be a valid XHTML document that passes validation at http://validator.w3.org, and your CSS files must pass validation at http://jigsaw.w3.org/css-validator/

Problem 3 (5 points)

Create a simple mashup page that displays both the Yahoo! news page (http://news.yahoo.com) and the San Jose Mercury News home page (http://www.mercurynews.com). Your solution should be in a file p3.html. Note: your solution for this problem does not need to validate (there are validation errors in the Yahoo! and Mercury News pages). For this problem you need not create a separate CSS file: you can put the style information directly in p3.html.

Style Points (5 points)

These points will be awarded if your XHMTL is clean and readable, Problems 1 and 2 pass validation (both for XHTML and CSS), and your solutions for Problems 1 and 2 are reasonably close matches to the samples.

Resources

- Use the Chrome browser for this project, and for all projects in this class. This will eliminate browser compatibility issues.
- Download the Firebug add-on if you haven't already: https://addons.mozilla.org/en-US/firefox/addon/firebug/
- Download the Firefox HTML Validator add-on if you haven't already. This add-on will automatically validate every page loaded into your browser and display information about errors in the lower-left corner of the browser window. If you click on the error message, a larger window will open with details about any problem(s). Built-in validators are available for Chrome also.
- Another tool you may find useful is the "Virtual Magnifying Glass", which allows you to examine individual pixels on the screen. You can download the Virtual Magnifying Glass from http://sourceforge.net/projects/magnifier.
- Yet another useful tool is the PerfectPixel Chrome extension, which allows you to overlay an image transparently on top of a Web page to compare them.

Hints

- There is one exception to the course rule that prohibits the use of formatting information in HTML elements: it is OK (and encouraged) to specify `cellspacing="0"` in each `<table>` element. This eliminates spacing between cells that the browser provides by default and which is usually a nuisance. You have to specify this attribute on the `<table>` element because there are no CSS style attributes that correspond exactly to this feature (`border-spacing` and `border-collapse` come close, but in my experience they have unpleasant side effects and are not portable).

- You may find some of the following CSS style attributes useful for this project (check w3schools for details):

```
width: 100%;
white-space: nowrap;
text-decoration: none;
```

- The `margin` style attribute is ignored for `td` elements (a strange exception to the uniformity of CSS). If you need a margin around the contents of a table cell, place a `div` inside the `td` and specify `margin` for the `div`.

Deliverables

Use the standard class submission mechanism to submit all of the files needed to display the pages for Problems 1-3, including `p1.html`, `p1a.css`, `p1b.css`, `p2.html`, `p2.css`, all of the images for the tabs, and `p3.html`.