# Lab 3: Cascading Style Sheets

## Learning Objectives

After completion of this lab, **you should be able** to

- Work collaboratively as a **pair programming** team
- Use a externally-linked **cascading style sheet** to format web pages
- Use CSS rules to format ***site*** **colors**, **margins**, and **font** properties
- Add **borders** and **margins** to HTML elements
- Use CSS to selectively apply style rules
- Use CSS to modify the **display** property and to format a **list** to act as a navigation bar
- Screen **Snapshots**
- **To Receive Credit**

---

## Work collaboratively as a pair programming team

CS 250 in-class labs will be done using **pair programming**. Your partner for today's lab is listed in the table below:

**Hebeler 204**

Grader: John Wright II

| | | | | |
|---|---|---|---|---|
| **Team 1**<br>Abundiz, Sergio<br>Ahmady, Temourshah | **Team 2**<br>Bajwa, Deepinder<br>Belfiglio, Alexander | **Team 3**<br>Burley, Jonathan<br>Burton, Henry | **Team 4**<br>Byars, Frank<br>Strom, Brandt | **Team 5**<br>Carpenter, Daniel<br>Chandler, Alan |
| **Team 6**<br>Crockett, Jordan<br>Dickerson, Andrew | **Team 7**<br>Hansen, Christopher<br>Juarez, Adrian | **Team 8**<br>Kinkade, Kyle<br>Plitkins, Kristofer | **Team 9**<br>Porter Jr, Anthony<br>Prescott, Brandon | **Team 10**<br>Rozelle, William<br>Canada, Justin |
| **Fill in**: Taing, Pokuy | | | | |

You may wish to review basic pair programming guidelines before you begin.

- One team member (the **driver**) has control of the keyboard/mouse and actively implements the program
- The other team member (the **navigator**) continuously observes the work of the driver to identify tactical defects (such as syntactic and spelling errors, etc.) and also thinks strategically about the direction of the work

You should **change roles** about every ten minutes during lab.

# Use a externally-linked cascading style sheet to format web pages

(10 minutes)

1. Download the following two **web pages** and save the HTML code inside your `U:\htdocs\labs\lab3` folder
   - lab3.html - the home page for Ellensburg Food Coop
   - shoponline.html - an accessible simple data table
2. Fix the page information in all two web pages to add your CS 250 account number and names to the header comment and author meta tag
3. Both pages end with an `address` element identifying the web developers
   - Add both of your names to the `address` element on both web pages
4. Create the cascading style sheet named `lab3.css` to use to format both web pages
   - Add a **file header comment** to `lab3.css` identifying the account, file, honor code and your names
     - See CS 250 CSS Coding Standards
   - Add a `link` element to the head section on both web pages to reference your cascading style sheet
     - Both pages will use the same CSS file
5. Verify that both web pages **validate** to XHTML 1.0 strict standards and report **no accessibility issues** with the accessibility report - FAE Rule Set
   - Other than the changes noted above, you should **not** change any of the content on the two HTML web pages during the lab. All changes will be made with the CSS sheet

# Use CSS rules to format *site* colors, margins, and font properties

(10 minutes)

The pages use the following color scheme:

| | | | | |
|---|---|---|---|---|
| **#CCCC33**<br>`body` background color | **#800080**<br>`body` font color | **#E4E495**<br>`table` background color | **#202077**<br>`table` font color, `border` color | **#B9B931**<br>table heading `th` background color |

Both pages will use **two font families**

- The pages' `body` text is formatted using the serif fonts: Garamond, 'Times New Roman', serif
- The **headings** (h1, h2, table headings, and table caption) are formatted using the sans-serif fonts: Verdana, Arial, Helvetica, sans-serif

Use HTML-Kit's **bsStyle plugin** to specify the rules to provide consistent colors, fonts, and paddings to the web page as shown in the screen snapshots below.

1. Add CSS rules in `lab3.css` to apply the global site colors, fonts, and background colors
2. View your pages with the Firefox browser and visually confirm that your colors and fonts are correct
3. Use Firefox's Web Developer toolbar > tools > **Validate Local CSS** and fix any CSS errors or warnings identified

## Switch pair programming roles

Switch **driver** and **navigator** roles. You may be able to just slide the keyboard and mouse over, or it may be preferable to change seats. Do not change the computer in use however.

## Add borders and margins to HTML elements

(10 minutes)

The horizontal line above the `address` element at the bottom of both web pages (see screen snapshots) is the result of a 2 pixel border line applied to the element. The `address` element serves as a common *footer* on both pages.

1. Use CSS to add a **top border line** to the `address` element
   **Note**: to keep things simple and consistent, all borders used in this assignment have a width of 2 pixels, the solid style, and the color #202077

The **vertical space** separating the `address` element from it's immediately proceeding block element is the result of a top margin of 50 pixels.

2. Use CSS to set the `address` element's top margin to 50 pixels
3. View your pages with the Firefox browser and visually confirm that all your page *footers* are correct
4. Use Firefox's Web Developer toolbar > tools > Validate Local CSS and fix any errors or warnings

identified

All tables used in this assignment have 2 pixel solid borders on all sides.

5. Use CSS to add a 2 pixel solid **border** (color #202077) to all tables
6. Note: to avoid the small gap around table cells, use CSS to set the table's **border-collapse** to collapse

---

## Use CSS to selectively apply style rules

(10 minutes)

The table headings used on `shoponline.html` have a different background color to make them stand out.

1. Use CSS rules with the `th` element to
    - Set the `th` background color to `#B9B931`
    - Set the `th` padding top and bottom to 10 pixels and right and left to 20 pixels
    - Give the heading a bottom border line

Examine the shoponline.html screen snapshot and add the CSS rules to adjust paddings, alignments, and borders to the table's elements to duplicate the style shown in the screen snapshot.

---

## Switch pair programming roles

Switch **driver** and **navigator** roles. You may be able to just slide the keyboard and mouse over, or it may be preferable to change seats. Do not change the computer in use however.

---

## Use CSS to modify the display property and to format a list to act as a navigation bar

(10 minutes)

The CSS **display** property may be set to modify the way an element is displayed on the screen. Elements can be displayed as block, inline, list-item, or none. None causes an element **not** to be displayed on the screen.

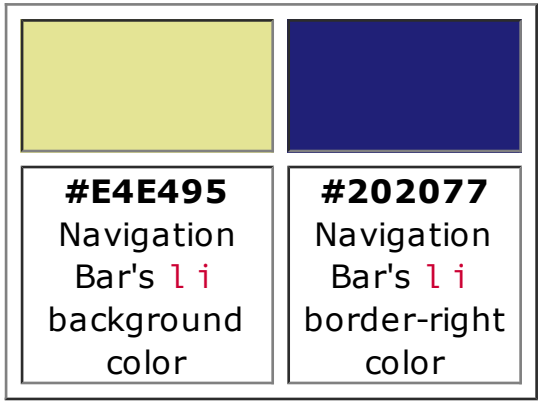
1. Add the CSS rule to set the display property for the **navbar** `h2` element **not** to display on the screen

```
/* Do not display navigation bar */
#navbar {
    display:none;
}
```

1. Add the CSS rule to set the display property for the **navbar** `li` elements to display as inline elements

Add CSS style rules to make the navbar's `li` elements look more like a navigation menu

3. Set their background color to #E4E495;
4. Give them a padding of 5 pixels top & bottom and 15 pixels right & left
5. Give them a 15 pixel solid #202077 right border




**Finishing touches**

6. Set the navbar's ul left padding to 0
7. Set the anchor element's (`a`) text-decoration to none (no underline)

---

## Screen Snapshots

`lab3.html` **screen snapshot**


shoponline.html **screen snapshot**

# Ellensburg Food Coop

Home | Shop

## Shop Online

Current Inventory

| | Member | Non-Member |
|---|---|---|
| **Apples** | $1.00 | $1.50 |
| **Oranges** | $2.00 | $2.50 |
| **Rice** | $0.99 | $1.99 |
| **Beans** | $2.75 | $3.75 |

*Web design by Full_Name_1 and Full_Name_2*

## To Receive Credit

Pair programming teams will receive the scoring rubric sheet at the start of lab. Write both names on the sheet to turn in the sheet in when you finish. Name 1 should be the student who saved the pair work in their CS 250 account.

- If you **finish during lab**, have your work checked for completeness, and turn in to your instructor the **lab's scoring rubric** [PDF].
  - If you have **not** yet satisfied all criteria to the level of 4, you need to continue working on the lab outside of class time
  - Your saved solution should be stored in the CS 250 account listed under Name 1
- If you are **unable** to complete the solution during the lab period
  - Leave lab with both students having a copy of the partially completed solution
  - Agree to finish the lab together (establish a time) or independently
  - Turn in the **lab's scoring rubric** [PDF] at the **start of next Lab**
    - Use one rubric for teams finishing together or two rubrics for students finishing independently
    - Keep track of and include the **total completion time** (rounded to closest half-hour) it took to complete the lab assignment and include the time on the rubric
    - Write a score `0..4` in the rubric's **self assessment column** representing your completion status
  - Make sure your work is stored in your CS 250 account in folder `U:\htdocs\labs\lab3\`

Lab 3 is due at the start of lab next Monday. **No** late lab assignments will be accepted without prior approval. Your lowest lab score will be dropped.