Introduction to CSS

CS5610: Web Development

Joydeep Mitra

Pre-Class Activity

- Fork the repo https://github.com/CSE-316-Software-Development/learn-css.git
  - Create a new branch with the name: *today's date(MMDDYY)*
  - Enter your full name in README.md
  - Push to the branch when done.
- Useful git commands:
  - Create new local branch
    ```
    $ git checkout -b <branch-name>
    ```
  - Push new branch to remote
    ```
    $ git push origin <branch-name>
    ```
  - Push modified/new files to remote branch
    ```
    $ git add <path/to/file>
    $ git commit -m "useful message"
    $ git push
    ```

What is CSS?

- CSS stands for cascading style sheets.
- It’s a language to stylize HTML elements in a web page.

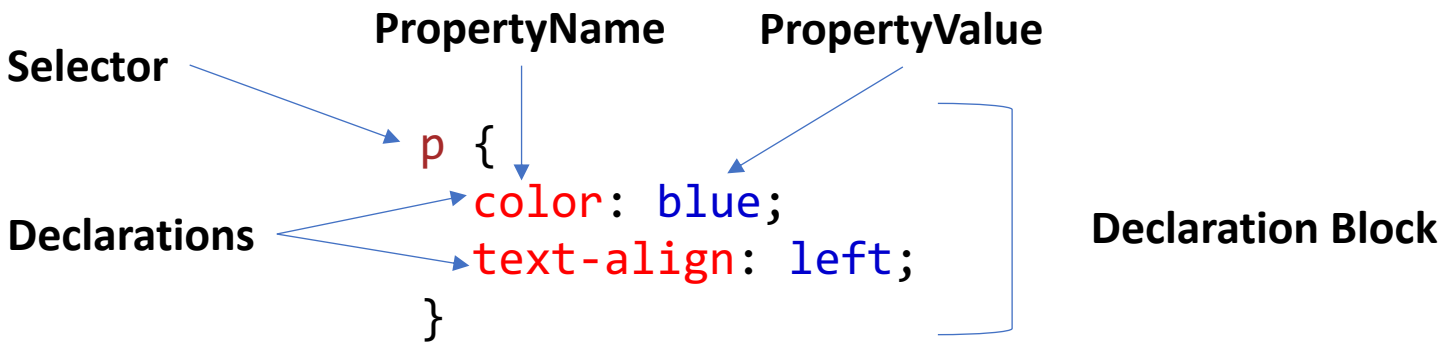
```
<style>
  body {
    background-color: lightblue;
  }

  h1 {
    color: white;
    text-align: center;
  }

  p {
    color: red;
    font-family: verdana;
    font-size: 20px;
    text-align: center;
  }
</style>
```

Why CSS?

- CSS allows us to have one style sheet for a set of related web pages.
- By having an external stylesheet, maintenance is easier, and modifications are simpler to make.

CSS Syntax

- A CSS stylesheet is a bunch of rules.
- Each rule is written as a selector (to which the rule will be applied) and a declaration block, which contains declarations for a style rule.
- A declaration is a colon separated property name, value pair.
- Declarations in a declaration block are separated by semicolons.



CSS Selectors

- A CSS selector is one of the following:
  - Tag name.

```
p {
  color: blue;
  text-align: left;
}
```

```
<p id="para" class="msg"> Hello World </p>
```

  - Tag Id.

```
#para {
  color: blue;
  text-align: left;
}
```

  - Tag Class name.

```
.msg {
  color: blue;
  text-align: left;
}
```

CSS Selectors

- Selectors can also be a combination of class name and tag name.
  - E.g., apply a style to *<p>* tags with class name *"msg"*.

```
p.msg {
  color: blue;
  text-align: left;
}
```

- Selectors can be grouped together so the same style applies to all the elements.
  - E.g., apply the same style to all <h1>, <h2>, and <p> tags.

```
h1, h2, p {
  font-size: 10px;
}
```

Where to Include CSS?

- There are three ways to specify CSS.
  - **External**. This implies that the CSS code is in a ".css" file and must be included in any HTML page that will be styled by the css file.

```
<head>
<link rel="stylesheet" href="/path/to/mystyle.css">
</head>
```

  - **Internal**. The CSS code is included within `<head></head>` of an HTML page. It applies to only that page.
  - **Inline**. The style is part of HTML tags. Generally, not encouraged.

```
<h1 style="color:red;text-align:center;">Inline CSS</h1>
```

For You to Do

- Add an external style *quiz/css/basic.css* to the HTML document *quiz/basic-external.html*.
- Modify the CSS in *quiz/css/combine-selector.css* such that <p> tag with class *main* has color red and the <p> tag with class *sub* has the color green in the HTML document *quiz/combine.html*.

CSS Backgrounds

- The *background-color* property specifies the background color of an HTML element.
- It takes a valid color name, RGB, or HEX as its value.

```
a {
  background-color: blue;
}

div {
  background-color: #ff0000;
}

p {
  background-color:  rgb(255,120,10);
}
```

CSS Backgrounds

- You can specify a background image for an element with *background-image* property.

```
body {
  background-image: url("images/paper.jpeg");
}
```

- If the image is too small, then the image repeats in across the x and y-axis of the element.
- The *background-repeat* property is used to control this effect. Valid values are:
  - `repeat-x`, `repeat-y`, `no-repeat`.

For You to Do

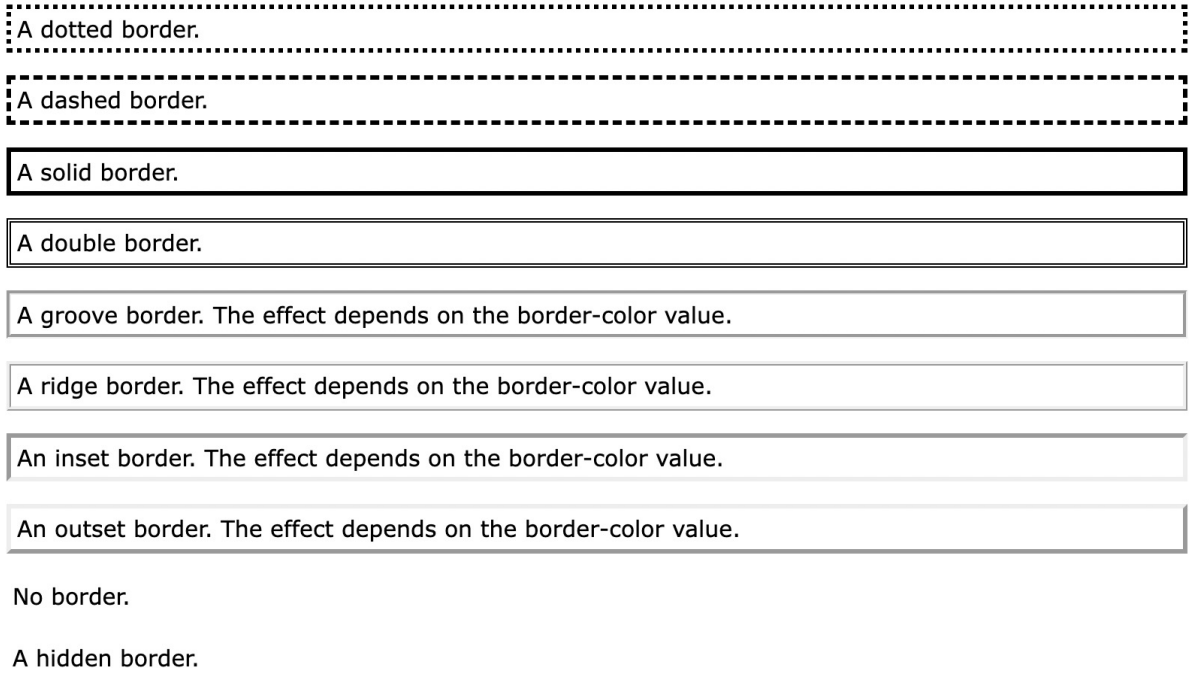
- Change *quiz/css/backgrnd.css* such that the background image is repeated in *quiz/backgrnd_img.html*

CSS Borders

- Border properties is used to add borders around an element.
- The *border-style* property is used to specify the style of the border.

```
div.dotted {border-style: dotted;}
div.dashed {border-style: dashed;}
div.solid {border-style: solid;}
div.double {border-style: double;}
div.groove {border-style: groove;}
div.ridge {border-style: ridge;}
div.inset {border-style: inset;}
div.outset {border-style: outset;}
div.none {border-style: none;}
div.hidden {border-style: hidden;}
```

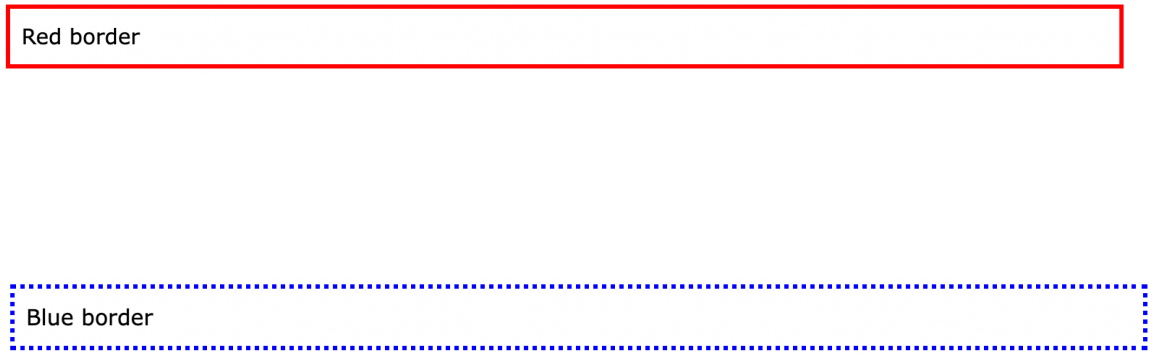
- *Note the border-style property must be mentioned for other border properties to take effect.*



CSS Borders

- The *border-width* property specifies how wide the border will be. It takes either predefined values (*thick, thin, or medium*) or fixed values in **px**.
- The *border-color* property specifies a color. Takes RGB or HEX or color names.

```
p {
  border-style: solid;
  border-width: 5px;
  border-color: red;
}

div {
  border-style: dotted;
  border-width: thick;
  border-color: blue;
}
```



For You to Do

- Add a thick purple border to the image in *quiz/border.html.* Also, the caption should be italics and in red color.

CSS Margins

- The *margin* properties are used to add space around elements, *outside of defined borders*. Space can be added to all 4 sides.
- Margin properties can have the following values:
  - *auto*: adds equal space around the element.
  - *length*: fixed space in px.
  - *%*: relative to the width of the containing element.
  - *inherit*: margins inherited from parent element.

```
p {
  margin-top: 100px;
  margin-bottom: 100px;
  margin-right: 150px;
  margin-left: 80px;
}
```

```
div {
  width: 300px;
  margin: auto;
  border: 1px solid red;
}
```

For You to Do

- Modify *quiz/margin.html* such that the image inherits the margin properties of body.
- Observe what happens when the *margin* of *body* is changed.

CSS Padding

- The padding property is used to add space around in an element, *inside its defined border*.
- Padding can be added to each side of the element using the values – *length (in px), %, and inherit* and the properties – `padding-top, padding-right, padding-bottom, padding-left`.

```
div {
  padding-top: 50px;
  padding-right: 30px;
  padding-bottom: 50px;
  padding-left: 80px;
}
```

CSS Padding

- Shorthand padding: *<top><right><bottom><left>*

```
div {
  padding: 50px 30px 50px 80px;
}
```

- Padding with 3 values -- <top>, <right>, <bottom>

```
div {
  padding: 50px 30px 50px;
}
```

- Padding with 2 values -- <top-bottom>, <right-left>

```
div {
  padding: 50px 30px;
}
```

CSS Height and Width

- The `height` and `width` properties set the height and width of an element; does not include padding, borders, and margins.
- Possible values:
  - *auto*: default.
  - *length*: in px, cm, etc.
  - *%* - in percent of containing block.
  - *initial* – sets to default.
  - *inherit* – same as parent values.

```
div {
  height: 200px;
  width: 50%;
}
```

For You to Do

- In *quizzes/padding.html* change the file such that inner div has the same padding as the outer div.
- Add a negative padding value to *quizzes/padding.html* and see how the display changes.

CSS Text Formatting

- The two most common text-related properties are `color` and `text-align`.

```
h1 {
  text-align: center;
  color: blue;
}

h2 {
  text-align: left;
  color: green;
}

h3 {
  text-align: right;
}
```

CSS Fonts

- The following properties are used to specify fonts.

```
.p1 {
  font-family: "Times New Roman", Times, serif;
  font-size: 40px;
}
p.normal {
  font-style: normal;
}
p.italic {
  font-style: italic;
}
p.thick {
  font-weight: bold;
}
```

CSS Display

- The **display** property is used to control the layout of HTML elements.
  - `none`. Hide the element.
  - `block`. Show element as a block.
  - `inline`. Show elements in one line.
  - `Inline-block`. Show blocks in one line.

For You to Do

- Change *quiz/css/nav-hor.css* such that the menu is displayed as follows:

CSS Table Borders

- Table borders are specified using the `border` property.

```
table, th, td {
  border: 1px solid black;
}
```

Firstname	Lastname
Peter	Griffin
Lois	Griffin

```
table {
  width: 100%;
  border-collapse: collapse;
}
```

Firstname	Lastname
Peter	Griffin
Lois	Griffin

For You to Do

- Modify *quiz/table-border.html* such that the images are displayed as follows:

CSS Overflow

- The `overflow` property is used to fit content too big for its area.
- Valid values are:
  - `visible`. Extra content is rendered outside the element's box (default).
  - `hidden`. Extra content will be clipped and made invisible.
  - `scroll`. A scroll bar is added to make extra content visible in the box.
  - `auto`. Same as scroll; adds scroll bar only when necessary.
- The `overflow-x` and `overflow-y` properties are used to specify if there will be a horizontal or vertical scroll.

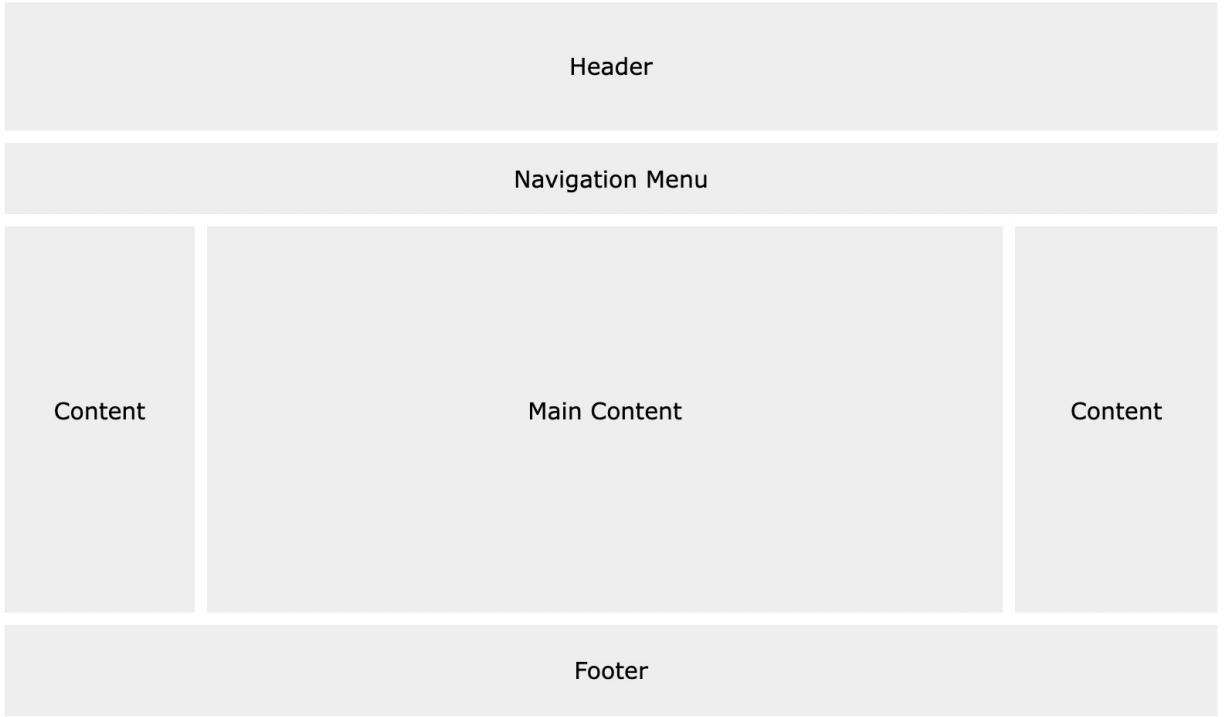
CSS Links

- CSS Links can be styled depending on which state they are in:
  - a:link - unvisited link.
  - a:visited - visited link.
  - a:hover - mouse over link.
  - a:active - on clicking the link.

```
a:link, a:visited {
  background-color: #f44336;
  color: white;
  padding: 14px 25px;
  text-align: center;
  text-decoration: none;
  display: inline-block;
  border-radius: 5px;
}

a:hover, a:active {
  background-color: blue;
}
```

A Standard Web Page Layout

- A standard web page layout has the following UI elements:
  - A header.
  - A navigation menu.
  - Main content.
  - Footer.



For You to Do

- In css/*layout.css*, the responsive design rule does not work as expected. Fix it.

Additional References

- https://developer.mozilla.org/en-US/docs/Web/CSS