

# CSCI 3308 – Software Development Methods and Tools



University of Colorado  
Boulder



# Reminders



# Topics

- Cascading Style Sheets
- Bootstrap



# Text Editors

- Sublime: <https://www.sublimetext.com/>
- Brackets: <http://brackets.io/>
- Atom: <https://atom.io/>
- Komodo IDE: <https://www.activestate.com/products/komodo-edit/>
- Or any normal text editor.

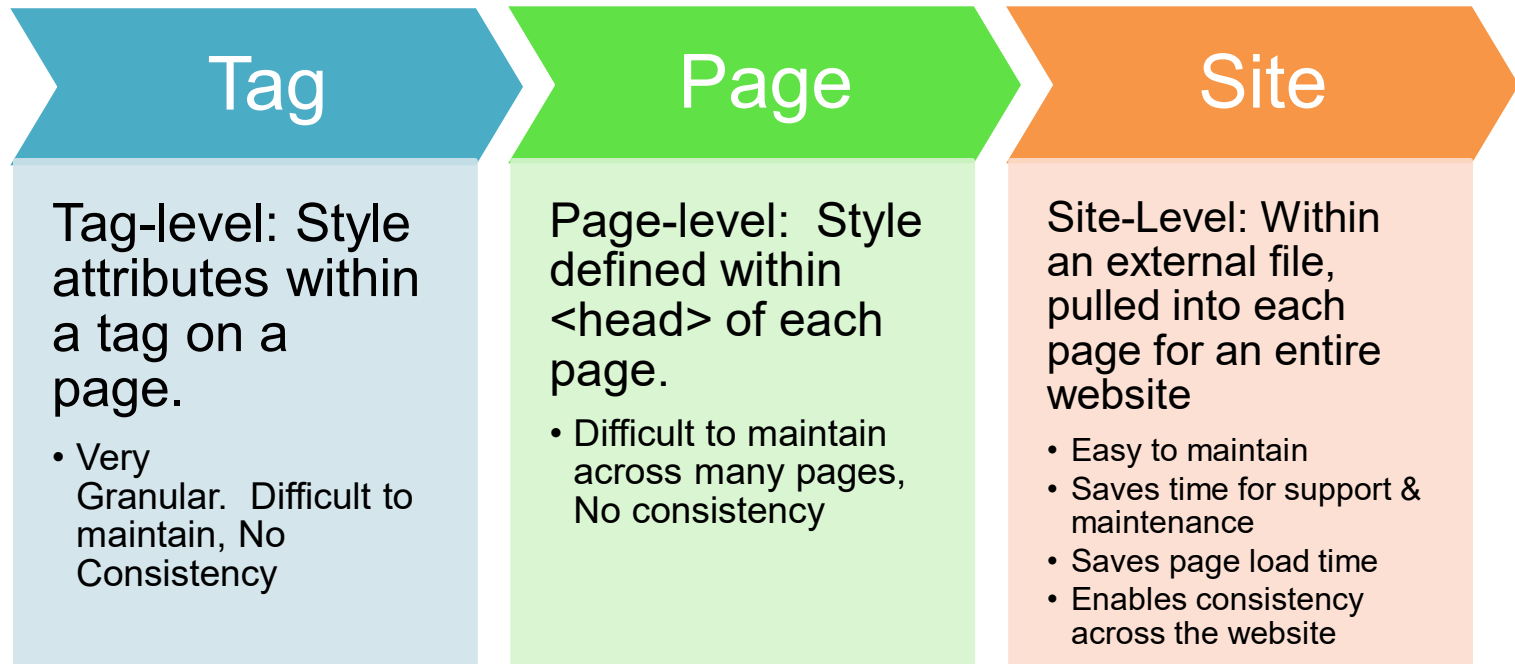


# What is a Style?

- **Objects in an HTML document have “style” attributes**
  - Font, font size, font color assigned to headings, paragraphs
  - Background color
  - Size and shape of images
  - Hyperlinks, colors, behaviors
  - Placement of objects on the page



# Applying Style



# Cascading Style Sheets

- What is a style sheet?
  - The style sheet is an EXTERNAL document (.css) containing the rules of style to be applied to your document. You “link” to it to bring it into one or more pages.
- Why do we use them?
  - Consistency from page to page
  - Faster, easier page construction



# Guidelines for a Style Sheet

- Should not contain any HTML tags
- Saved as file with a type/extension of “.css”
- Style Sheet –
  - Reference it and bring it into your document:

**<head>**

**< link rel="stylesheet" type="text/css"  
href="mystyle.css">**

**< /head>**





# Cascading Style Sheets

- Using the <style> tag
  - Entered in <head> section
  - Applies to entire document
- Style rules are not HTML
- Rule Syntax
  - selector – the tag that the rule applies to
  - { } Braces to contain the rule
  - Property:value pair
  - Ends with semicolon



# Examples

- Let's Look at Some Examples
- [www.w3schools.com/css](http://www.w3schools.com/css)



# Examples

**<style type="text/css">**

```
body {  
  font-size: 16pt;  
  color: blue;  
  background-color: pink;  
}  
h1{  
  font-size: 24pt;  
  color: black;  
}  
p {  
  margin-left: 10%;  
  margin-right: 10%;  
}
```

**</style>**



# Types of Style Rules

- element (tag)
- # id
- .class
- Element.class
- Group (multiple selectors)
- Nested (selector within a selector)



# Types of Style Rules

- A class allows you to set different styles for the same tag
- Example:

```
<style>  
    h1.majorheading {  
        font-family: arial;  
        color: blue;  
    }  
</style>
```

then you refer to the **class** in the tag:

```
<h1 class="majorheading">
```





# Style Sheets

- **Multiple selectors are OK**

```
h1, h2 {  
  color: purple;  
  text-align: center;  
}
```

- **Nested selectors are OK**

```
div > h1 {  
  color: red;  
}
```

- **ID References are OK**

```
<h1 id="beginning">  
<style>  
  #beginning {color: red;}  
</style>
```



# Overriding Styles

- “!important” override indicator
  - For a rule that must NOT be overridden in cascade
  - “!important” must follow the rule
- `<p style="color: green !important;">`



# Some Style Properties

Font Size

Font Family

Font-weight

Text-  
transform

Word-  
spacing

Letter-  
spacing

Text-align

Text-indent

Color

Background-  
color

Background-  
image

Box  
properties



# Bootstrap

- A style library
- Easy formatting using templates
- Easy transition from computer-based viewing to phone-based viewing i.e. responsive, mobile-first websites
- <https://getbootstrap.com/docs/4.3/getting-started/introduction/>



# Using Bootstrap



INCLUDE AN EXTERNAL CSS  
SHEET FROM A CDN  
(CONTENT DELIVERY  
NETWORK)



INCLUDE A JQUERY (JAVA  
SCRIPT) LIBRARY (ONLY  
NEEDED FOR CERTAIN JS  
PLUG-INS)



INCLUDE A CLIENT JAVA  
SCRIPT ENGINE FROM A CDN





# Using Bootstrap

## **<!-- Latest compiled and minified CSS -->**

```
<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
```

## **<!-- Javascript & jQuery library -->**

```
<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-
q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"
crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"
integrity="sha384-
UO2eT0CpHqdSjQ6hJty5KVphtPhzWj9WO1clHTMGa3JDZwrnQq4sF86dIHNDz0W1"
crossorigin="anonymous"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"
integrity="sha384-JjSmVgyd0p3pXB1rRibZUAYoIlly6OrQ6VrjIEaFf/nJGzlxFDsf4x0xIM+B07jRM"
crossorigin="anonymous"></script>
```



# Bootstrap Grid System

## Bootstrap Grid System

Bootstrap's grid system allows up to 12 columns across the page.

If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:

span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1	span 1
span 4				span 4				span 4			
span 4				span 8							
span 6						span 6					
span 12											

Bootstrap's grid system is responsive, and the columns will re-arrange automatically depending on the screen size.



# Bootstrap Grid System

## Grid Classes

The Bootstrap grid system has four classes:

- **xs** (for phones - screens less than 768px wide)
- **sm** (for tablets - screens equal to or greater than 768px wide)
- **md** (for small laptops - screens equal to or greater than 992px wide)
- **lg** (for laptops and desktops - screens equal to or greater than 1200px wide)

The classes above can be combined to create more dynamic and flexible layouts.



# Questions?

