

# 19CSCI03P

# Web Programming

---

**Dr. Nivin Atef, Ph.D.**

Nivine.atef@bue.edu.eg

# Chapter 4

## Cascading Style Sheets Basics

### Key Concepts

---

Copyright © 2013  
Terry Ann Morris, Ed.D

# Learning Outcomes

---

- **Describe** the **purpose** of **Cascading Style Sheets**
- List **advantages** of using Cascading Style Sheets
- Configure **color** on web pages with Cascading Style Sheets
- Configure **inline styles**
- Configure **embedded** style sheets
- Configure **external** style sheets
- Configure web page **areas** with element **name**, **class**, **id**, and **descendant** selectors
- Test your CSS for valid syntax

# Overview of Cascading Style Sheets (CSS)

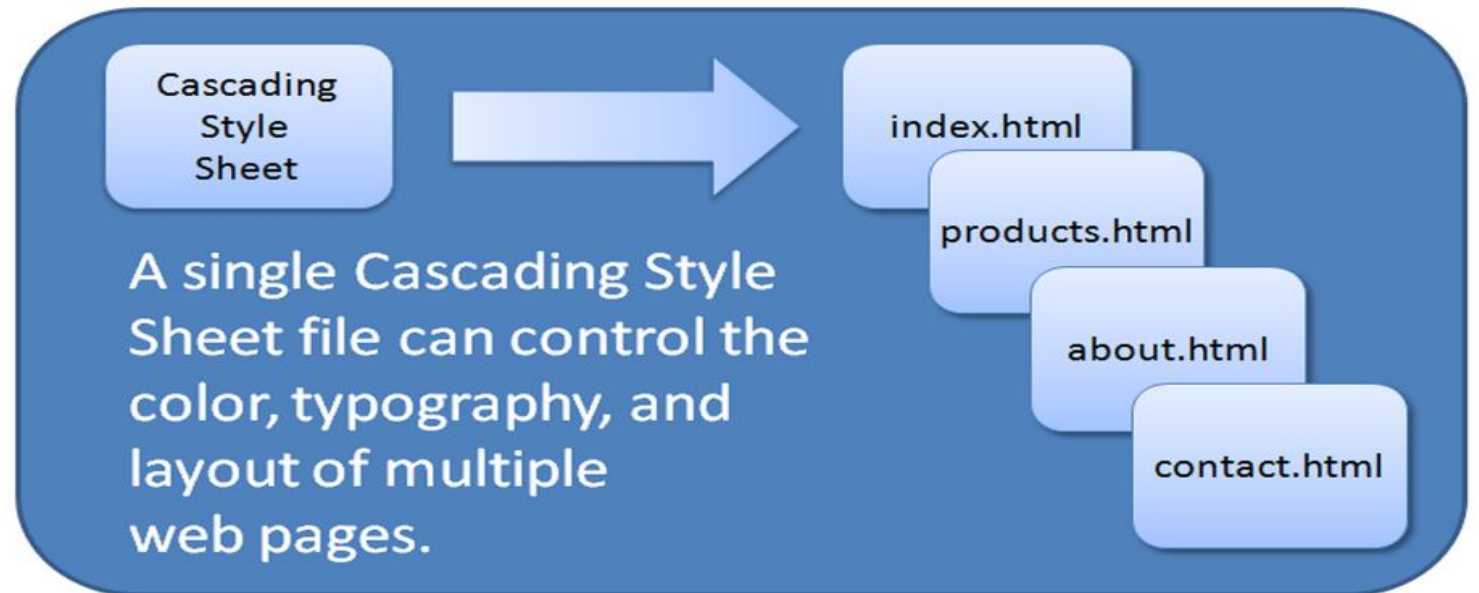
---

- See what is possible with CSS:
  - Visit <http://www.csszengarden.com>
- Style Sheets
  - used for years in Desktop Publishing
  - apply **typographical** styles and **spacing** to printed media
- CSS
  - provides the **functionality** of **style sheets** (*and much more*) for web developers
  - a **flexible, cross-platform, standards-based language** developed by the W3C.

# CSS Advantages

---

- Greater **typography** and **page layout control**
- **Style** is **separate** from **structure**
- Styles can be **stored** in a **separate** document and **linked** to from the web page
- Potentially **smaller documents**
- **Easier site maintenance**



# CSS Syntax

---

- Style sheets are composed of "**Rules**" that **describe** the **styling** to be applied.
- Each rule contains a **Selector** and a **Declaration**



# CSS Syntax Sample

---

Configure a web page to display blue text and yellow background.

```
body { color: blue; background-color: yellow; }
```

This could also be written using hexadecimal color values as shown below.

```
body { color: #0000FF; background-color: #FFFF00; }
```



# Types of Cascading Style Sheets

---

- Inline Styles
- Embedded Styles
- External Styles
- Imported Styles



# Description of the Types of Cascading Style Sheets

---

- **Inline Styles**

- Configured in the **body** of the web page
- Use the **style attribute** of an **HTML tag**
- Apply only to the **specific element**

- **Embedded Styles**

- Configured in the **head** section of a web page.
- Use the **HTML <style> element**
- Apply to the **entire web page document**

# Description of the Types of Cascading Style Sheets (Cont.)

---

- **External Styles**

- Configured in a **separate text file** with **.css** file extension
- The **HTML <link>** element in the **head section** of a web page associates it with the **.css** file

- **Imported Styles**

- Similar to External Styles

# Inline CSS with the Style Attribute

---

- **Inline CSS**
  - Configured in the **body** of the Web page
  - Use the **style attribute** of an **HTML tag**
  - Apply **only** to the **specific element**
- The Style Attribute
  - Value: **one** or **more** style declaration **property** and **value** pairs

# Inline CSS with the Style Attribute Example

---

```
<h1 style="color:#ff0000">Heading text is red</h1>
```

**Heading text is red**

```
<h1 style="color:#FF0000; background-color:#CCCCCC"> This is  
displayed as a red heading with gray background</h1>
```

**This is displayed as a red heading with gray background**

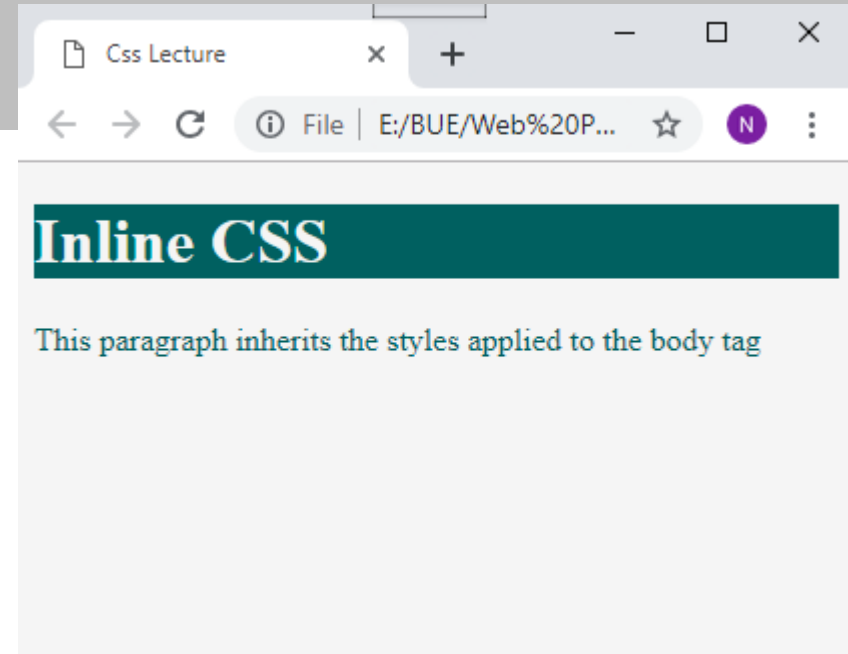
# Inline CSS with the Style Attribute Example

```
<body style="background-color: #F5F5F5; color: #006060;">
```

```
<h1 style="background-color: #006060; color: #F5F5F5;"> Inline CSS</h1>
```

```
<p>This paragraph inherits the styles applied to the body tag </p>
```

```
</body>
```



# Inline CSS with the Style Attribute Example

```
<body style="background-color: #F5F5F5; color: #006060;">
```

```
<h1 style="background-color: #006060; color: #F5F5F5;"> Inline CSS</h1>
```

```
<p>This paragraph inherits the styles applied to the body tag </p>
```

```
<p style="color:#333333;">This paragraph Overrides the styles applied to the  
body tag </p>
```

```
</body>
```



# Configure Embedded CSS with the Style Element

- Configured in the **head** section of a web page.
- Use the HTML **<style>** element
- Apply to the **entire web page** document
- **Style declarations** are contained between the opening and closing **<style>** tags
- The optional type attribute indicates the MIME type of text/css
- Example:

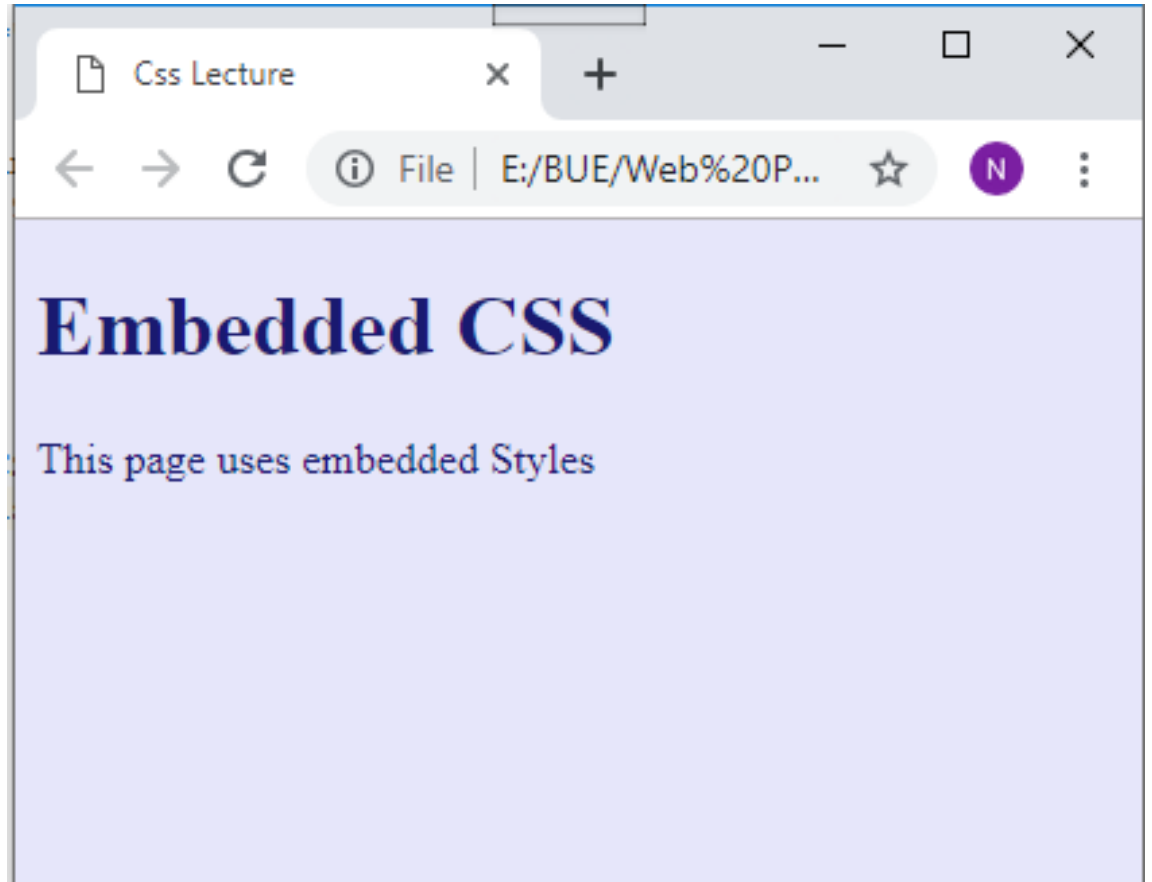
```
<style  
body { background-color: #000000;  
        color: #FFFFFF;  
}  
</style>
```

# Embedded Styles Example

```
<head>
<title>Css Lecture</title>
<meta charset="utf-8">

<style>
body {background-color:#E6E6FA;
      color:#191970;
}
</style>
</head>

<body>
<h1>Embedded CSS</h1>
<p>This page uses embedded Styles</p>
</body>
```

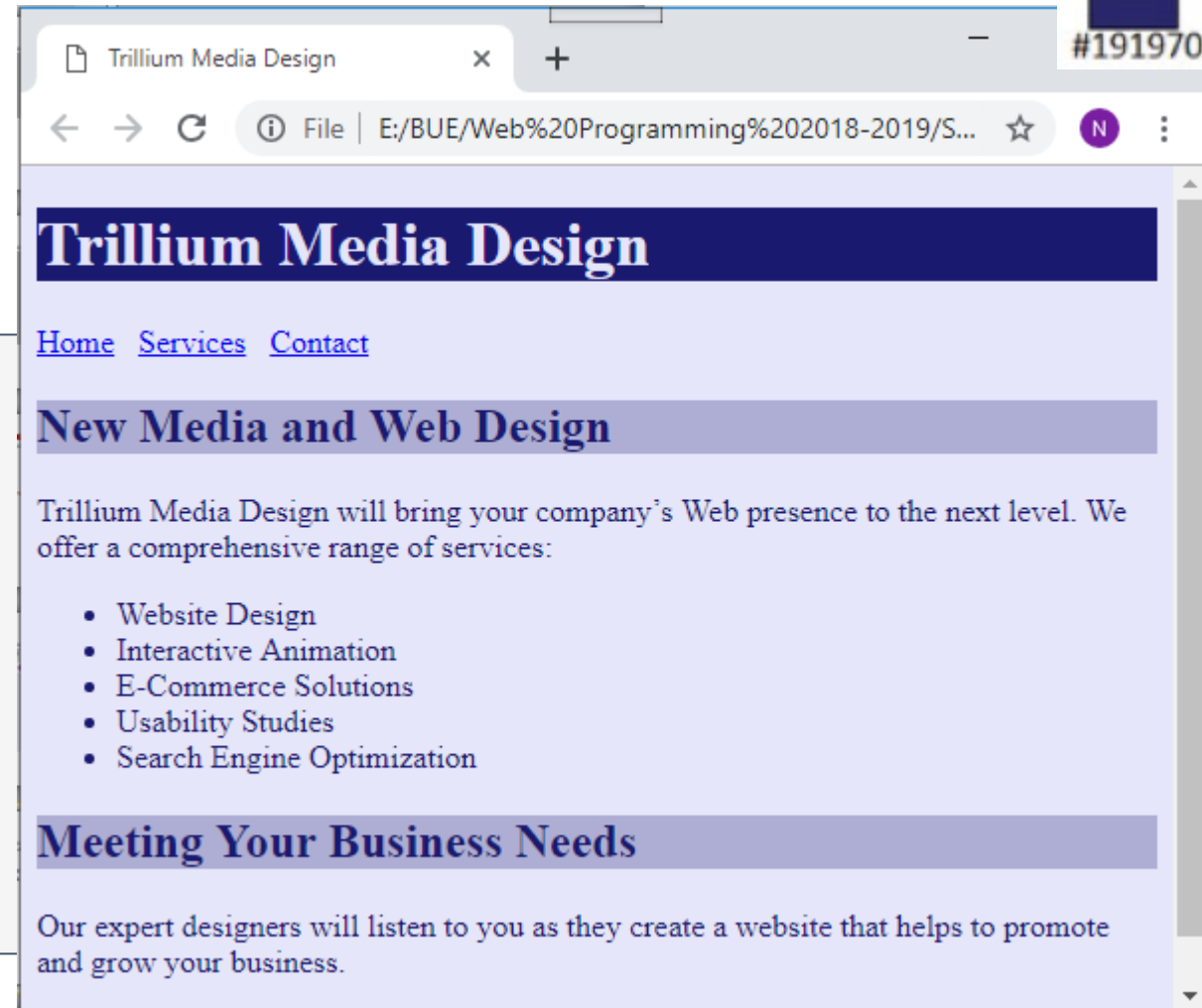




# CSS Embedded Styles Example

- The body selector sets the global style rules for the entire page.
- These global rules are **overridden** for <h1> and <h2> elements by the h1 and h2 style rules.

```
<style >
body { background-color: #E6E6FA;
      color: #191970;}
h1 { background-color: #191970;
     color: #E6E6FA;}
h2 { background-color: #AEAED4;
     color: #191970;}
</style>
```



# External Style Sheets

---

- CSS style rules are contained in a **text file separate** from the **HTML documents**.
- The External Style Sheet text file:
  - extension ".css"
  - contains **only style rules**
  - **does not contain** any **HTML** tags

# External Style Sheets (Cont.)

Multiple web pages can associate with the same external style sheet file.

site.css

```
body { background-color: #E6E6FA;  
        color: #000000;  
}  
h2 { color: #003366; }
```

index.html

clients.html

about.html

Etc...

# The <link> Element

---

- A **self-contained** tag
- Placed in the **header** section
- Purpose: **associates** the **external style sheet** file with the **web page**.
- Example:

```
<link rel="stylesheet" href="color.css">
```

# Using an External Style Sheet

---

External Style Sheet color.css

```
body { background-color: #E6E6FA;  
        color: #191970;}  
h1 { background-color: #191970;  
      color: #E6E6FA;}  
h2 { background-color: #AEAED4;  
      color: #191970;}
```

To associate with the external style sheet called color.css, place the following code in the head section:

```
<link rel="stylesheet" href="color.css">
```

# CSS Selectors

---

## Common Types of Selectors:

- HTML element name selector
- class selector
- id selector
- descendant selector

# Using CSS with “class”

---

- Class Selector
  - **Apply** a CSS rule to **ONE OR MORE** elements on a web page
  - Does not associate the style to a particular HTML element
- Configure with .classname
- The sample creates a class called “new” with red italic text.
- To use the class, code the following HTML:

`<p class=“new”>This is text is red and in italics</p>`

*This is text is red and in italics*

```
<style>
.new { color: #FF0000;
      font-style: italic;
      }
</style>
```

# Using a CSS id Selector

- id Selector

**Apply** a CSS rule to **ONLY ONE** element on a web page.

- Configure with #idname

- The sample creates an id called “new” with red, large, italic text.

```
<style>
#new { color: #FF0000;
        font-size:2em;
        font-style: italic;
      }
</style>
```

- To use the id, code the following HTML:

```
<p id=“new”>This is text is red, large, and in italics</p>
```

*This is text is red, large, and in italics*



# Using a CSS Descendant Selector

---

- Descendant Selector
  - Apply a CSS rule within the **context** of the **container** (parent) element.
  - Sometimes called a **contextual selector**.
- Configure by listing the **container selector** followed by the **selector you are styling**.
- The sample specifies a green text color for **only** the paragraph elements located *within* the footer element.

```
<style>  
footer p {color: #00ff00; }  
</style>
```

# Practice

```
<style>
```

```
.feature{color:#FF0000}
```

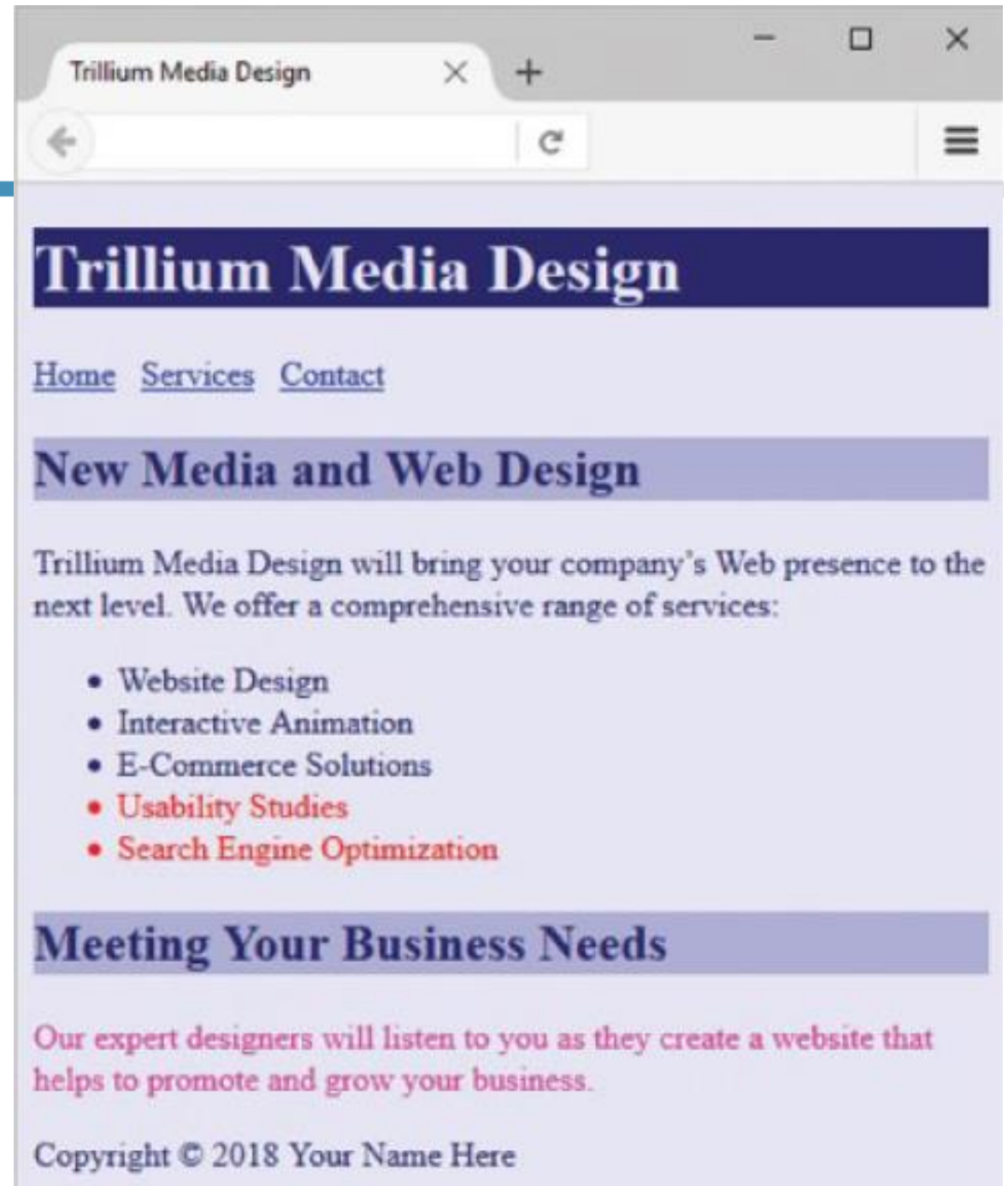
```
#new {color:#FF00CC}
```

```
</style>
```

```
<li class="feature">Usability  
Studies</li>
```

```
<li class="feature">Search Engine  
Optimization</li>
```

```
<p id="new">Our expert designers  
will listen to you as they create a  
website that helps to promote and  
grow your business.</p>
```



# The Span Element <span>

---

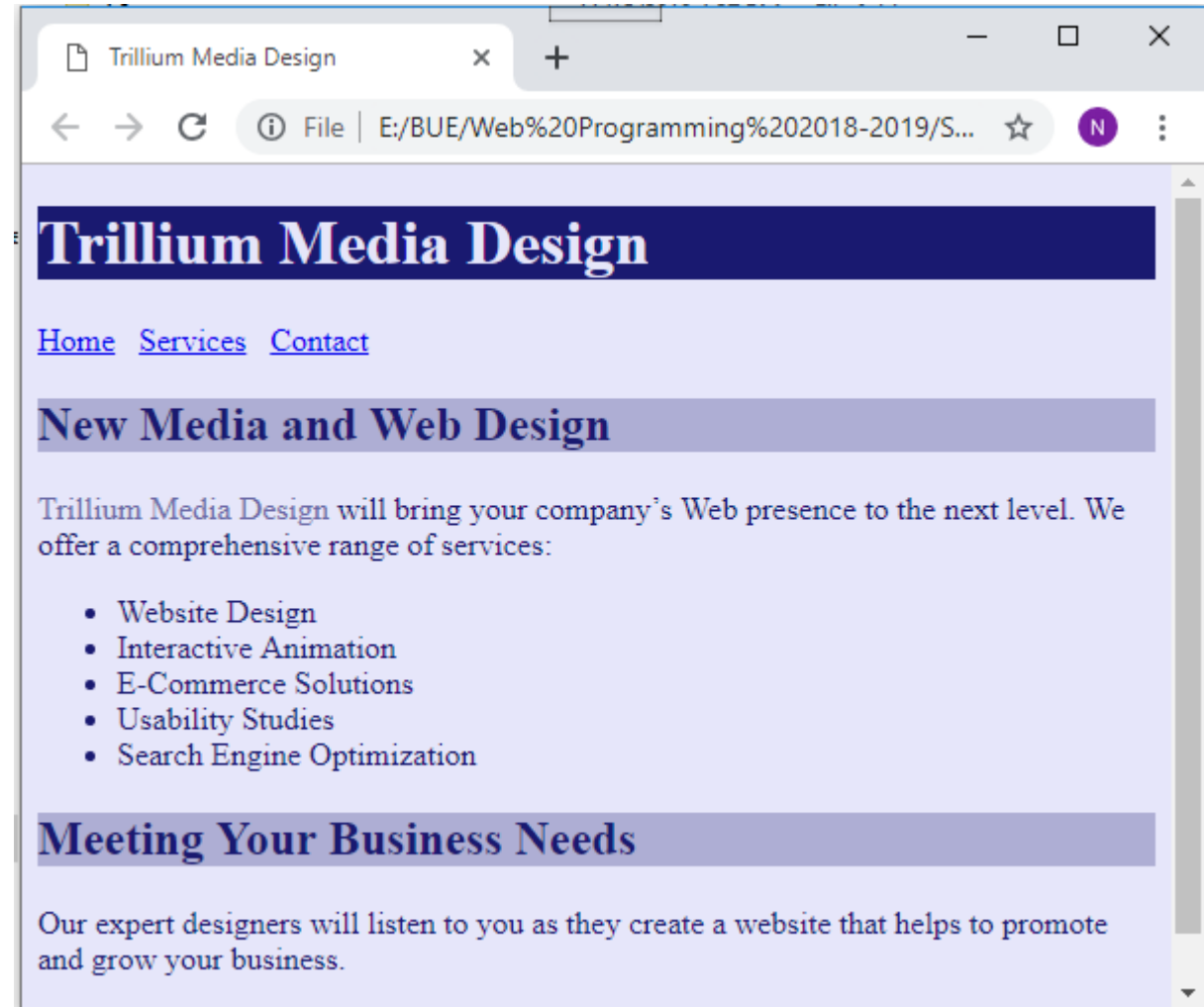
- An **inline-level** element
- Purpose:
  - **Configure a specially formatted** area displayed **in-line** with other elements, such as within a paragraph.
- There is **no line break** before and after the span.

# <span> Example

- Embedded CSS:

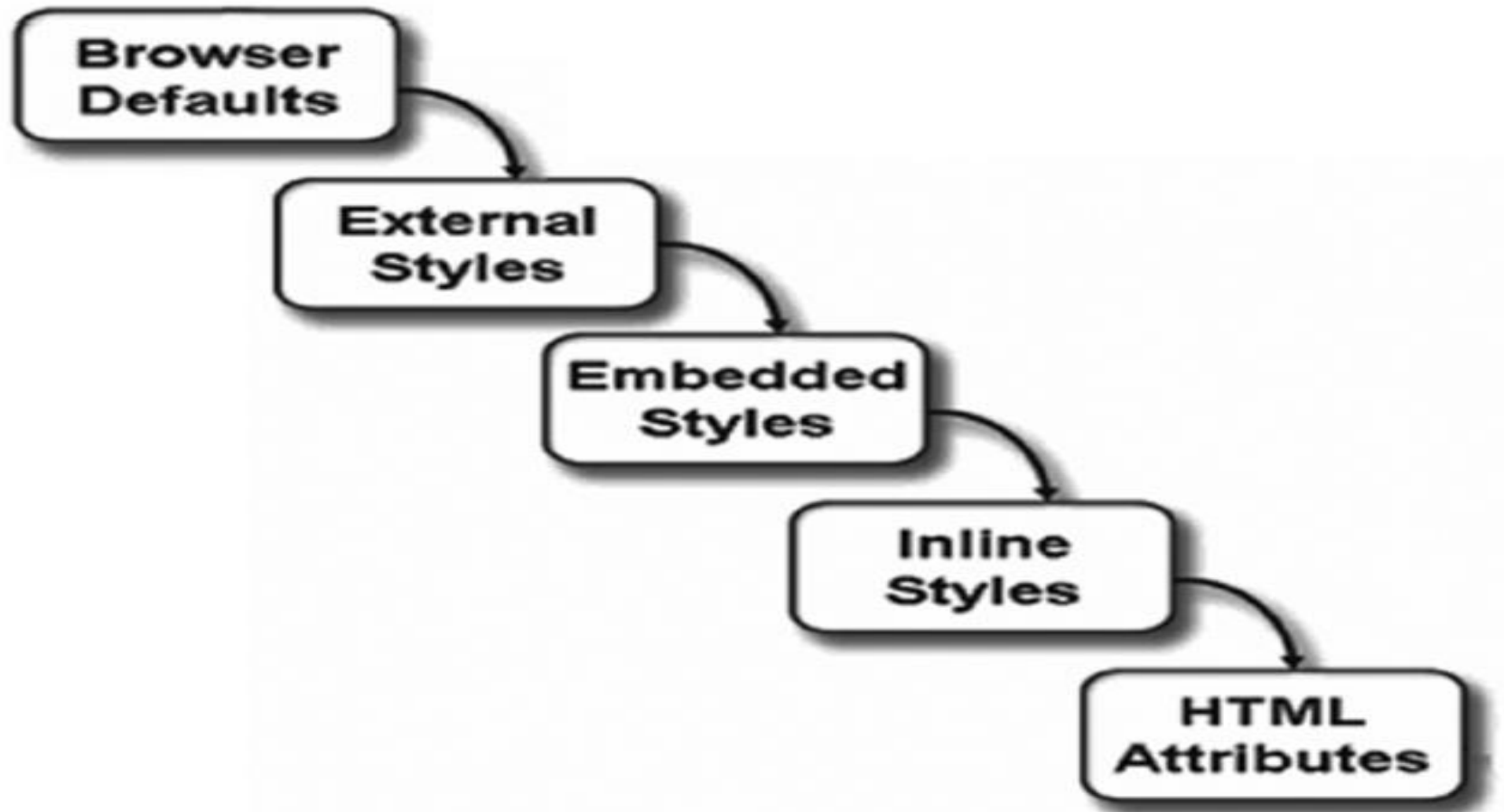
```
<style>
.companyname {color:#6A6AA7; }
</style>
```

```
<p><span class="companyname">Trillium
Media Design </span> will bring your
company's Web presence to the next
level.
```



# The “Cascade”

---



# The Cascade Example

```
<Head>
```

```
<style>
```

```
    .special { color: red; }
```

```
    p { color: blue; }
```

```
</style>
```

```
</head>
```

```
<body>
```

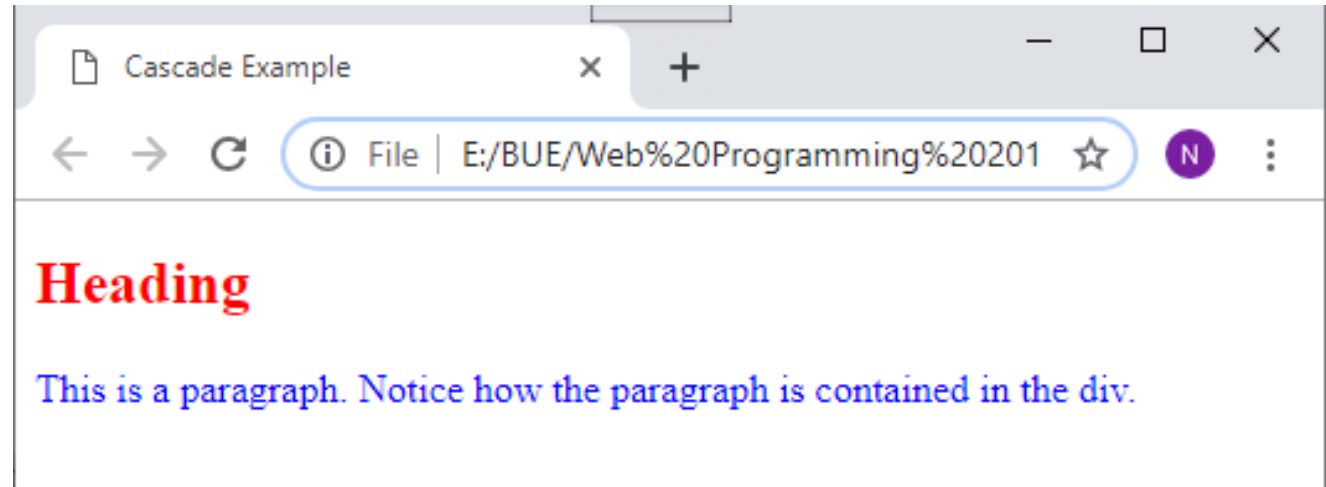
```
<div class="special">
```

```
    <h2>Heading</h2>
```

```
    <p>This is a paragraph. Notice how the paragraph is contained in the div.</p>
```

```
</div>
```

```
</body>
```



# Summary

---

- This chapter introduced you to Cascading Style Sheet Rules associated with color on web pages.
- You configured inline styles, embedded styles, and external styles.
- You applied CSS style rules to HTML, class, and id selectors.
- You are able to submit your CSS to the W3C CSS Validation test.

# Chapter 5

## Web Graphics Styling Basics

### Key Concepts

---

Copyright © 2013  
Terry Ann Morris, Ed.D



# Learning Outcomes

---

- Describe types of graphics used on the Web
- Apply the ***image element*** to add graphics to web pages
- Configure ***images*** as ***backgrounds*** on web pages
- Configure ***images*** as ***hyperlinks***

# Types of Graphics

---

- Graphic types commonly used on web pages:
  - GIF
  - JPG
  - PNG

# GIF

---

- **Graphics Interchange Format**
- Best used for **line art** and **logos**
- Maximum of **256** colors
- One color can be configured as transparent
- Can be **animated**
- Uses **lossless** compression



The image shows two side-by-side rectangular areas. The left area has a dark blue, textured background with the text 'GIF saved with Transparency' in white. A light blue speech bubble points to this area. The right area has a solid teal background with the text 'GIF saved without Transparency' in white. A light blue speech bubble points to this area.

GIF saved with  
Transparency

Background color  
configured to be  
transparent

GIF saved without  
Transparency

Background color –  
no transparency

# JPEG

---

- **Joint Photographic Experts Group**
- Best used for photographs
- Up to 16.7 million colors
- Use lossy compression
- Cannot be animated
- Cannot be made transparent



# PNG

---

- **Portable Network Graphic**
- Support **millions** of colors
- Support multiple levels of transparency  
*(but browsers do not -- so limit to one transparent color for Web display)*
- Use lossless compression
- Combines the best of GIF & JPEG
- Browser support is growing

# Web Graphics

Image Type	File Extension	Compression	Transparency	Animation	Colors
Graphic Interchange Format (GIF)	.gif	Lossless	Yes	Yes	256
Joint Photographic Experts Group (JPEG)	.jpg or .jpeg	Lossy	No	No	Millions
Portable Network Graphic (PNG)	.png	Lossless	Yes (multiple levels)	No	Millions

# Image Optimization

---

- The process of creating an image with the lowest file size that still renders a good quality image—**balancing image quality and file size.**
- Photographs taken with digital cameras are not usually optimized for the Web
- Use a graphics application to:
  - Reduce image dimensions
  - Reduce size of the image file



# Optimize An Image for the Web

---

- Image Editing Tools:
  - GIMP (free!)
  - Adobe Fireworks
  - Adobe Photoshop
  - <http://pixlr.com/editor> (free!)



# Choosing Names for Image Files

---

- Use all **lowercase** letters
- **Do not** use punctuation **symbols** and **spaces**
- **Do not change** the file extensions  
(*should be .gif, .jpg, .jpeg, or .png*)
- Keep your file **names short but descriptive**
  - i1.gif is probably too short
  - myimagewithmydogonmybirthday.gif is too long
  - dogbday.gif may be just about right

# The Image Element <img>

---

- Configures graphics on a web page

```

```

- **src** Attribute: File name of the graphic
- **alt** Attribute: Configures alternate text content (description)
- **height** Attribute: Height of the graphic in pixels
- **width** Attribute: Width of the graphic in pixels

# KayakDoorCounty.net

[Home](#) [Tours](#) [Reservations](#) [Contact](#)

## Your next adventure is only a paddle away ....

Take a guided kayak tour while you explore the shoreline of scenic Door County.

### Featured tours this week:

- Cana Island
- Mink River
- Europe Lake

*Copyright © 2018 KayakDoorCounty.net*

# KayakDoorCounty.net

[Home](#) [Tours](#) [Reservations](#) [Contact](#)

## Your next adventure is only a paddle away ....



Take a guided kayak tour while you explore the shoreline of scenic Door County.

### Featured tours this week:

- Cana Island
- Mink River
- Europe Lake

*Copyright © 2018 KayakDoorCounty.net*

```


```

# Accessibility & Images

---

- Required:
  - Configure the alt attribute
    - Alternate text content to convey the meaning/intent of the image
    - If the image contains a text message, then the text should typically be the value of the alt attribute
    - NOT the file name of the image
    - Use alt="" for purely decorative images
- Recommended:
  - If your site navigation uses image links for the main navigation, provide simple text links at the bottom of the page.

# Image Links

---

- To create an image link use an anchor element to contain an image element

```
<a href="index.html"></a>
```



**Home**

# Image Link Example

```
<style>
#bar { background-color: #152420; }
</style>

<nav id="bar">
<a href="home.html"></a>
<a href="tours.html"></a>
<a href="reservations.html"></a>
<a href="contact.html"></a>
</nav>
```

## KayakDoorCounty.net

**Home Tours Reservations Contact**

Your next adventure is only a paddle away ....





# Thumbnail Image

A small image configured to link to a larger version of that image.

```
<a href="big.jpg"></a>
```



# CSS background-image Property

---

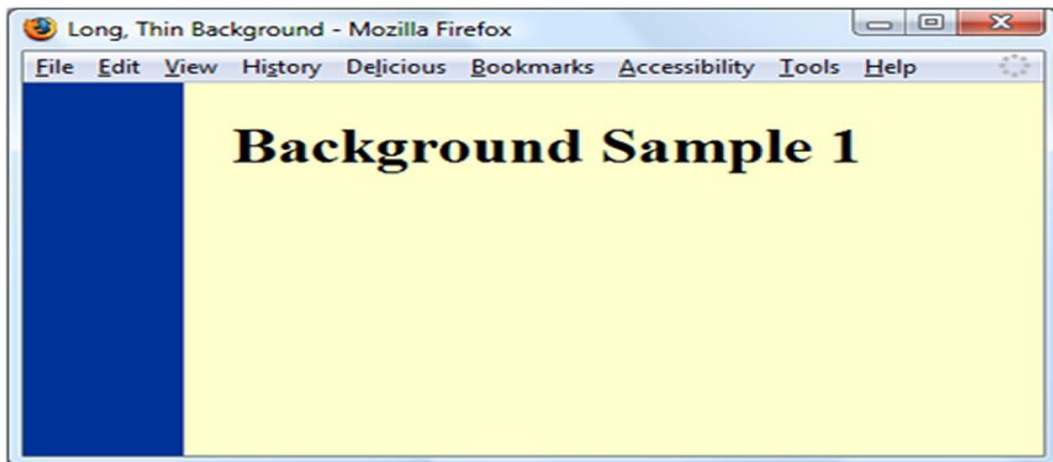
- Configures a background-image
- By default, background images tile (repeat)

```
body { background-image: url(background1.gif); }
```

## Background Image

---

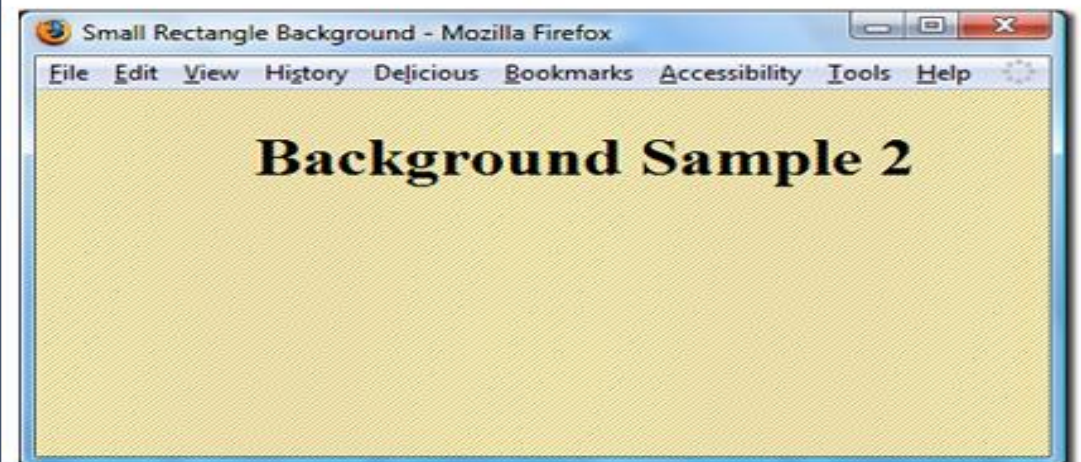
### Web Page with Background Image



## Background Image

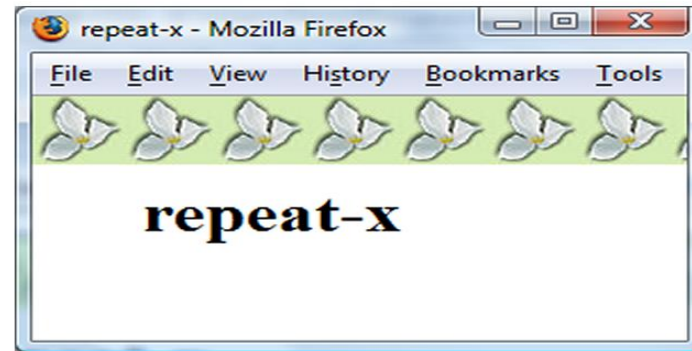
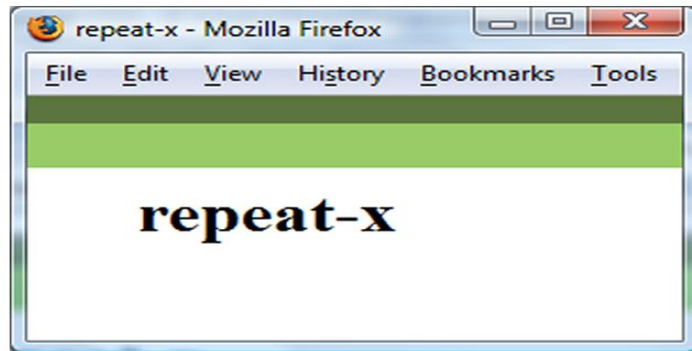
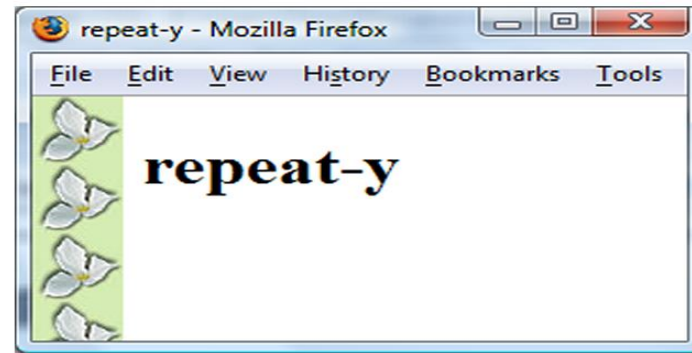
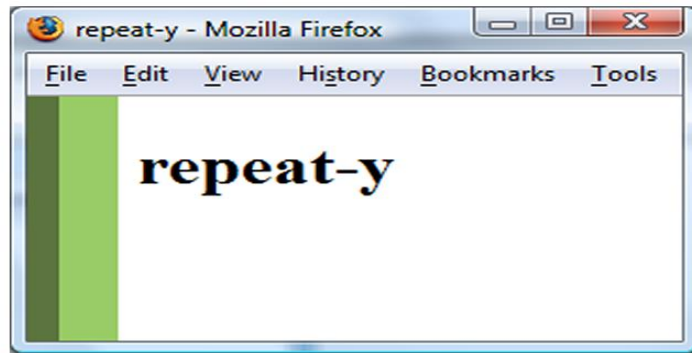
---

### Web Page with Background Image





# CSS background-repeat Property



# Background Image Example

```
<style>
```

```
#bar { background-color: #152420; }
```

```
main {background-image: url(heroback.jpg);  
      background-repeat: no-repeat;}
```

```
</style>
```

## KayakDoorCounty.net

**Home Tours Reservations Contact**

**Your next adventure is only a paddle away ....**

Take a guided kayak tour while you  
explore the shoreline of scenic Door County.

**Featured tours this week:**

- Cana Island
- Mink River
- Europe Lake

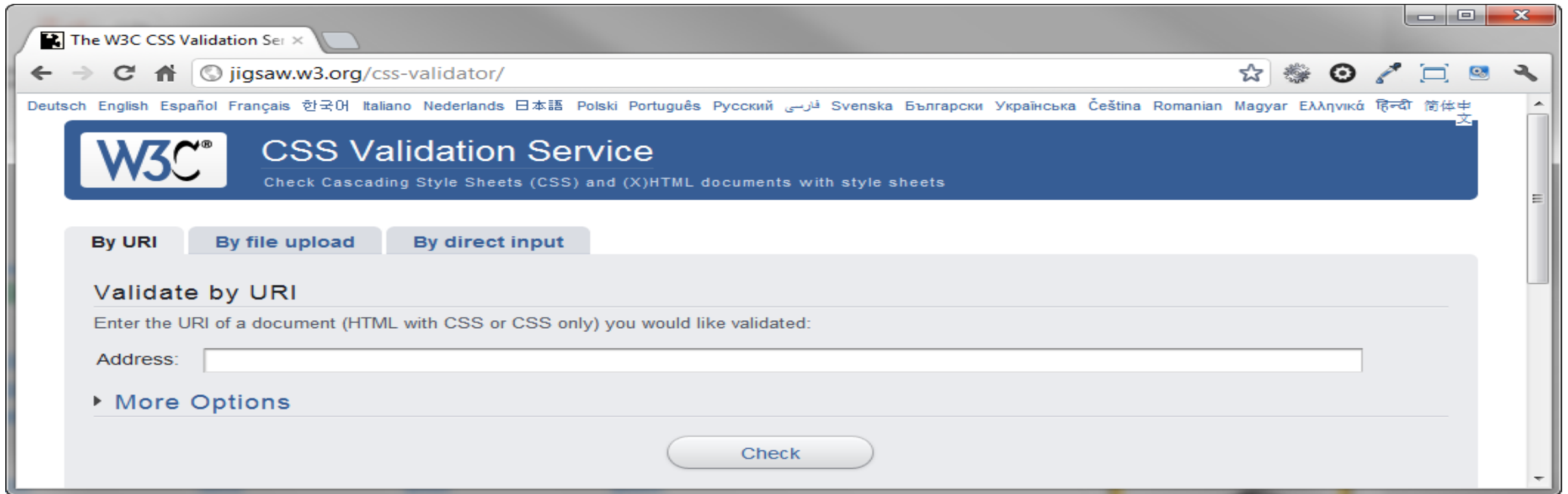
[Home](#) [Tours](#) [Reservations](#) [Contact](#)

Copyright © 2018 KayakDoorCounty.net



# W3C CSS Validation

- <http://jigsaw.w3.org/css-validator/>



The screenshot shows a web browser window with the address bar displaying `jigsaw.w3.org/css-validator/`. The page features a blue header with the W3C logo and the text "CSS Validation Service" and "Check Cascading Style Sheets (CSS) and (X)HTML documents with style sheets". Below the header, there are three tabs: "By URI", "By file upload", and "By direct input". The "By URI" tab is selected, and the section is titled "Validate by URI". It contains the instruction "Enter the URI of a document (HTML with CSS or CSS only) you would like validated:" followed by a text input field labeled "Address:". Below the input field is a link "More Options". At the bottom of the form is a "Check" button.

The W3C CSS Validation Service interface includes the following elements:

- Language selection bar: Deutsch, English, Español, Français, 한국어, Italiano, Nederlands, 日本語, Polski, Português, Русский, فارسی, Svenska, Български, Українська, Čeština, Romanian, Magyar, Ελληνικά, हिन्दी, 简体中文.
- Navigation tabs: By URI (selected), By file upload, By direct input.
- Section title: Validate by URI.
- Instruction: Enter the URI of a document (HTML with CSS or CSS only) you would like validated:
- Input field: Address: [text input box]
- Link: More Options
- Action button: Check

# CSS Troubleshooting Tips

---

- Verify you are using the : and ; symbols in the right spots—they are easy to confuse.
- Check that you are *not* using = signs instead of : between each property and its value.
- Verify that the { and } symbols are properly placed
- Check the syntax of your selectors, their properties, and property values for correct usage.
- If part of your CSS works, and part doesn't:
  - Review your CSS
  - Determine the first rule that is not applied.  
Often the error is in the rule above the rule that is not applied.
- Validate your CSS at <http://jigsaw.w3.org/css-validator>

# Summary

---

- This chapter introduced the HTML techniques and technologies used to place images on web pages.
- Issues related to accessibility and copyright were also discussed.
- The number one reason for visitors to leave web pages is too long of a download time. When using images, be careful to minimize this issue.