



Prifysgol Cymru  
Y Drindod Dewi Sant  
University of Wales  
Trinity Saint David

200  
1822 - 2022

# CREATING A SIMPLE PAGE

(HTML OVERVIEW)

*Man Geni Addysg Uwch yng Nghymru*  
*Where Higher Education Began in Wales*

[www.uwtsd.ac.uk](http://www.uwtsd.ac.uk)

# OVERVIEW

- How markup works: Elements and attributes
- Minimal HTML document structure
- Identifying text elements: Block and inline
- Troubleshooting and validating HTML

# Content Without Markup

Without HTML markup to describe content structure, text runs together; line breaks are ignored:

Black Goose Bistro

The Restaurant

The Black Goose Bistro offers casual lunch and dinner fare in a relaxed atmosphere. The menu changes regularly to highlight the freshest local ingredients.

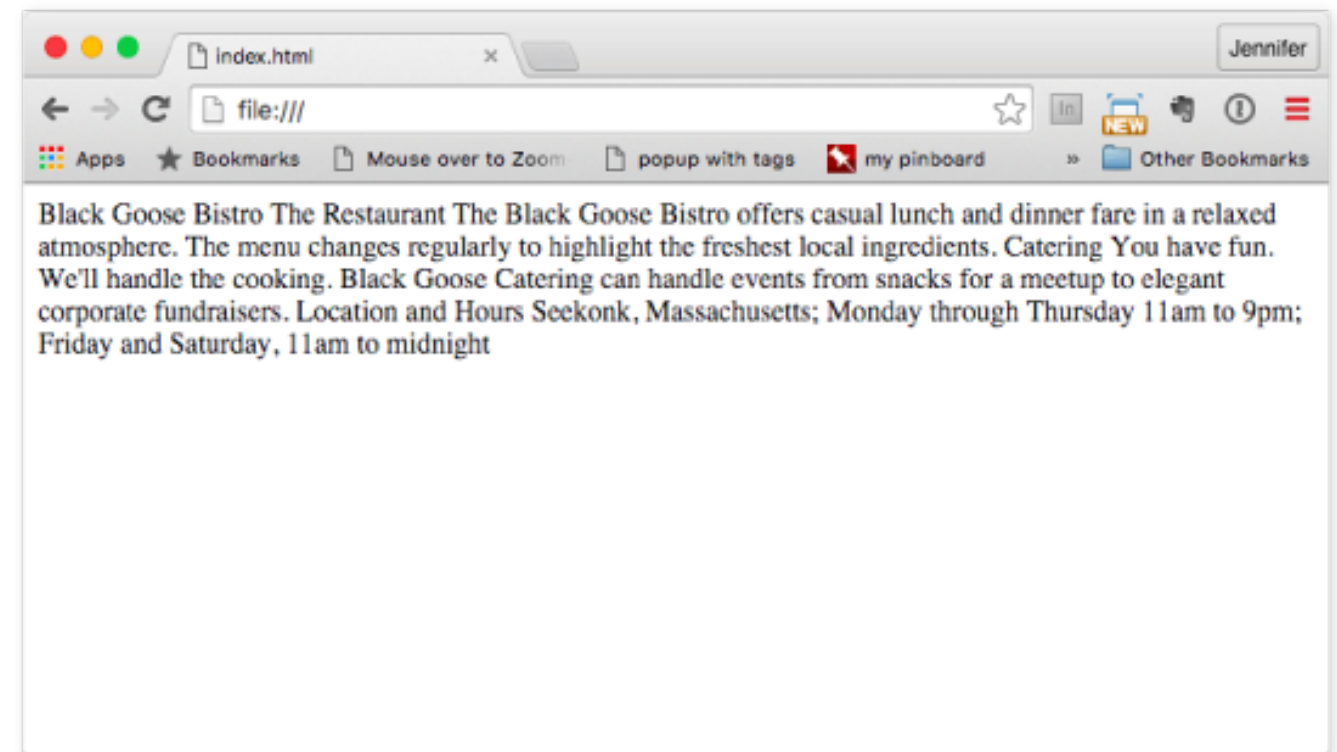
Catering

You have fun. We'll handle the cooking. Black Goose Catering can handle events from snacks for a meetup to elegant corporate fundraisers.

Location and Hours

Seekonk, Massachusetts;

Monday through Thursday 11am to 9pm; Friday and Saturday, 11am to midnight



# What Browsers Ignore

- Multiple character spaces (white space)
- Line breaks (carriage returns)
- Tabs
- Unrecognized markup

# Markup Basics

Text must be marked up meaningfully and accurately (semantically)

with HTML tags:

- Browsers need markup to display content correctly.
- Markup makes content elements available to scripts and style rules.
- Markup aids search engines.

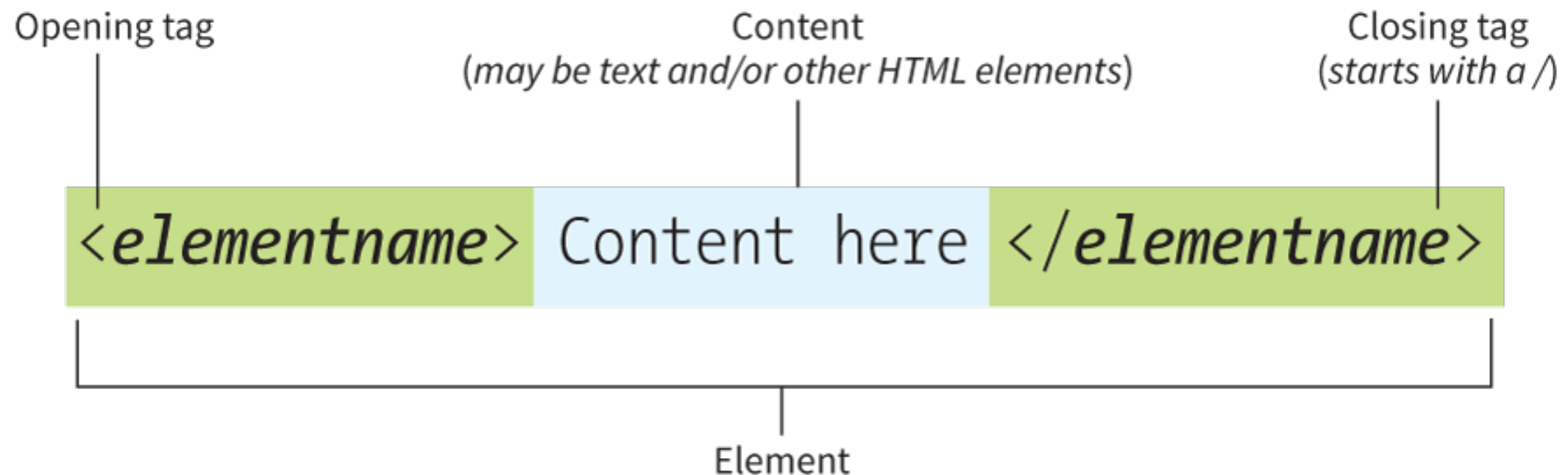
# Anatomy of an HTML Element

## tag

The element name in angle brackets

## element

The content and its markup (start and end tags)



Example:

`<h1>` Black Goose Bistro `</h1>`

# Some Elements Are Empty

Some elements have no content and provide a simple directive. They are called empty elements:

```
<element-name>
```

Example: The **br** element inserts a line break.

```
<p>1005 Gravenstein Highway North<br>Sebastopol, CA 95472</p>
```

# Attributes

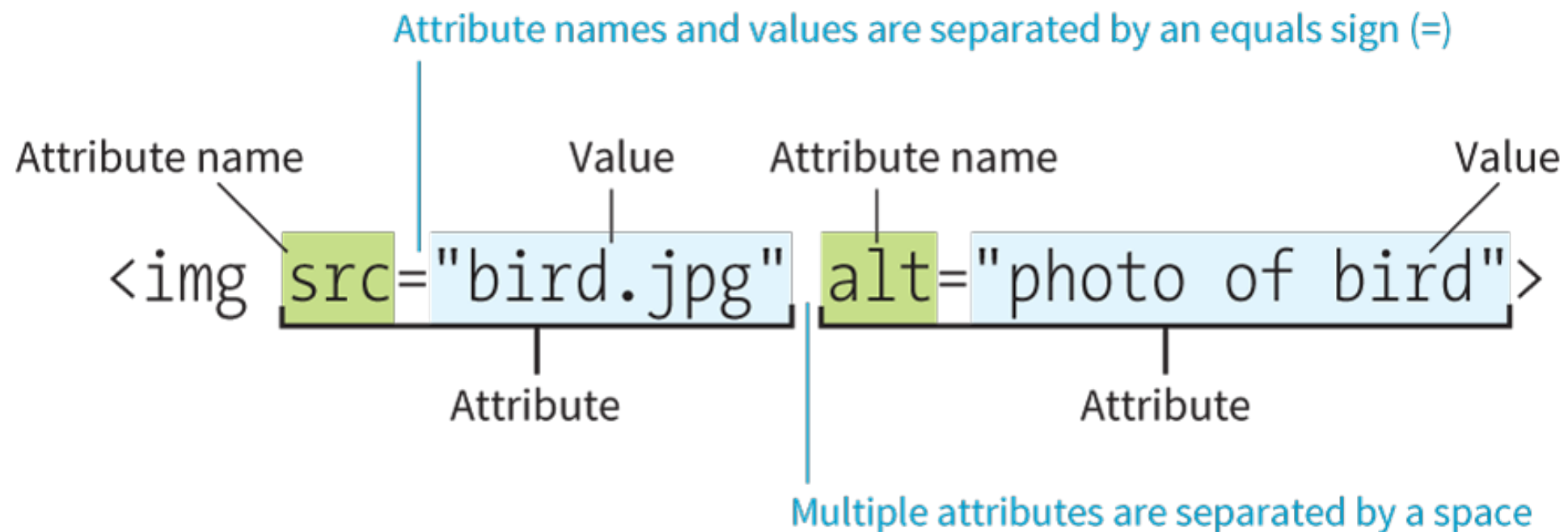
Attributes are instructions that clarify or modify an element. They appear in the opening tag after the element name:

```
<element attribute="value">Content</element>
```

```
<a href="http://oreilly.com">O'Reilly site</a>
```

## Attributes (cont'd)

There can be more than one attribute in a tag:



They are separated by spaces and can go in any order.

## Attributes (cont'd)

- Most attributes take values, which follow an = sign; some are single descriptive words.
- A value might be a number, word, string of text, URL, or measurement.
- Quotation marks aren't strictly required but are recommended for consistency.
- Single or double quotation marks are okay.
- Attribute names and values are defined in the HTML specification.
- Some attributes are required.

# Nesting Elements

Putting elements inside other elements is called nesting. Make sure closing tags don't overlap:

```
<div>
  <h1>Headline</h1>
  <p>This is <em>emphasized</em> text.</p>
</div>
```

# HTML Comments

To leave a note in an HTML document, mark it up as a comment:

```
<!-- start global navigation -->  
<ul>  
...  
</ul>  
<!-- end global navigation  
and begin the main header  
for content page -->
```

Text between `<!--` and `-->` won't display in the browser. Keep in mind that they are still visible in the source.

# Minimal Document Structure

It is recommended that HTML documents have the following minimal structure:

```
<!DOCTYPE html>
<html>

  <head>
    <meta charset="utf-8">
    <title>Title here</title>
  </head>

  <body>
    <!-- Page content goes here.-->
  </body>

</html>
```

# Minimal Document Structure (cont'd)

The DOCTYPE declaration lets browsers know that the document is written according to the HTML5 specification:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Title here</title>
  </head>
  <body>
    [Page content goes here.]
  </body>
</html>
```

# Minimal Document Structure (cont'd)

The **html** element is the root element that contains all the elements in the document:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Title here</title>
  </head>
  <body>
    [Page content goes here.]
  </body>
</html>
```

# Minimal Document Structure (cont'd)

The **head** element contains elements that pertain to the document and are not rendered as content, such as title, metadata, style sheets, and scripts:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Title here</title>
  </head>
  <body>
    [Page content goes here.]
  </body>
</html>
```

# Minimal Document Structure (cont'd)

**meta** elements provide document metadata. In this case, it says that the document uses an expanded character set (UTF-8). It could also provide keywords, author information, publishing status, and more:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Title here</title>
  </head>
  <body>
    [Page content goes here.]
  </body>
</html>
```

# Minimal Document Structure (cont'd)

The **title** element is required. Titles display in the browser tab and bookmark lists, and are used by search engines:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Title here</title>
  </head>
  <body>
    [Page content goes here.]
  </body>
</html>
```

# Minimal Document Structure (cont'd.)

The **body** element contains the content of the document. The content is everything you want to show up in the browser window:

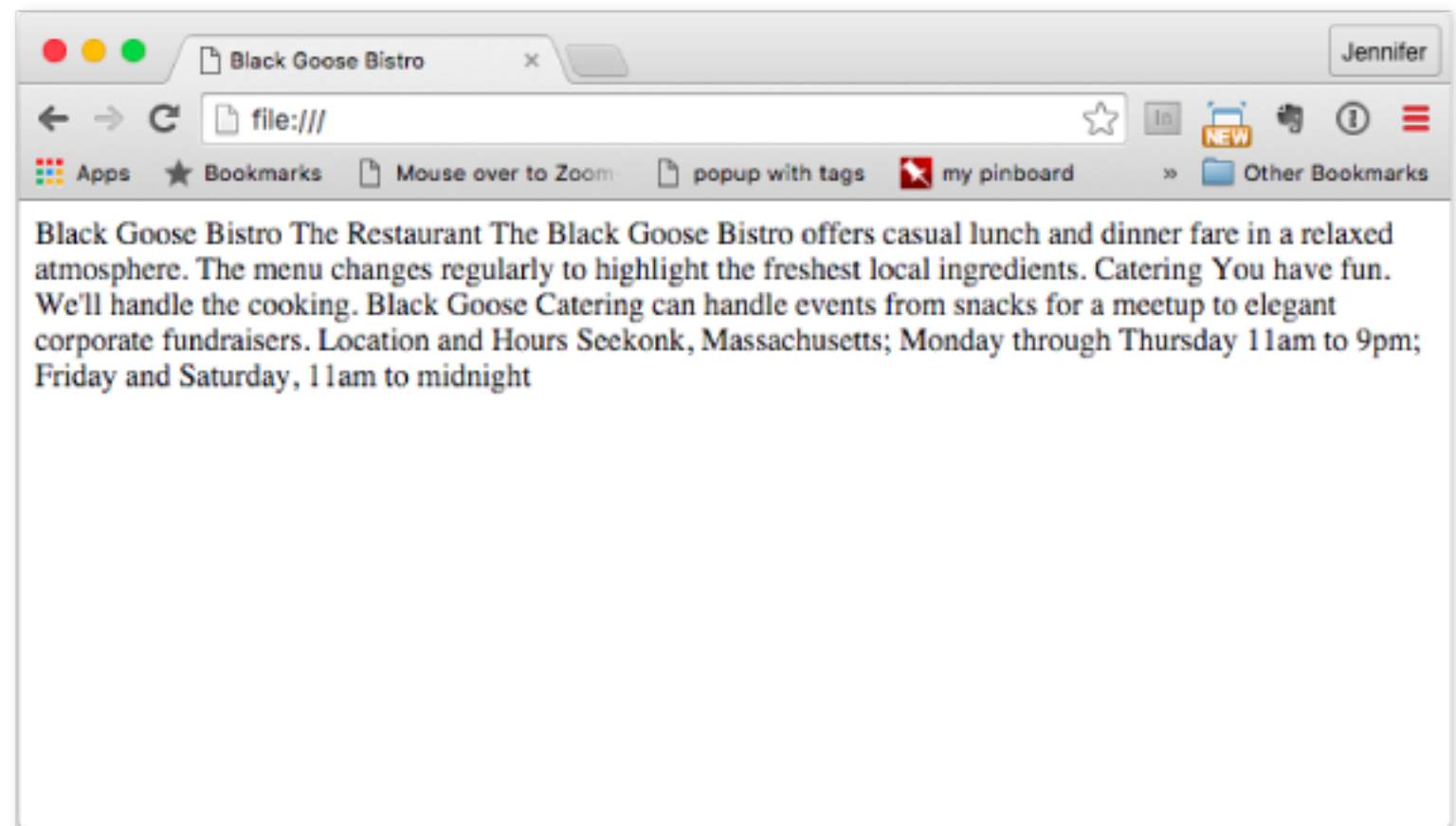
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Title here</title>
  </head>
  <body>
    [Page content goes here.]
  </body>
</html>
```

# A Structured Document

With the document structure in place, there is a title in the browser tab, but the content is still unstructured.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Black Goose Bistro</title>
</head>
<body>
Black Goose Bistro

The Restaurant
The Black Goose Bistro offers casual
lunch and dinner fare in a relaxed
atmosphere. The menu changes regularly
to highlight the freshest local
ingredients.
...
Location and Hours
Seekonk, Massachusetts;
Monday through Thursday 11am to 9pm;
Friday and Saturday, 11am to midnight
</body>
</html>
```



# Marking Up Content

- The purpose of HTML is to add meaning and structure to the content. This is called semantic markup.
- It is not intended to describe how the content should look (its presentation).
- The way elements relate to one another forms an outline-like structure called the DOM (Document Object Model).
- The DOM is the foundation on which you apply styles and scripts.

# Basic Text Elements

With headings (<h1>) and paragraphs (<p>) identified, the browser can display the content appropriately:

...

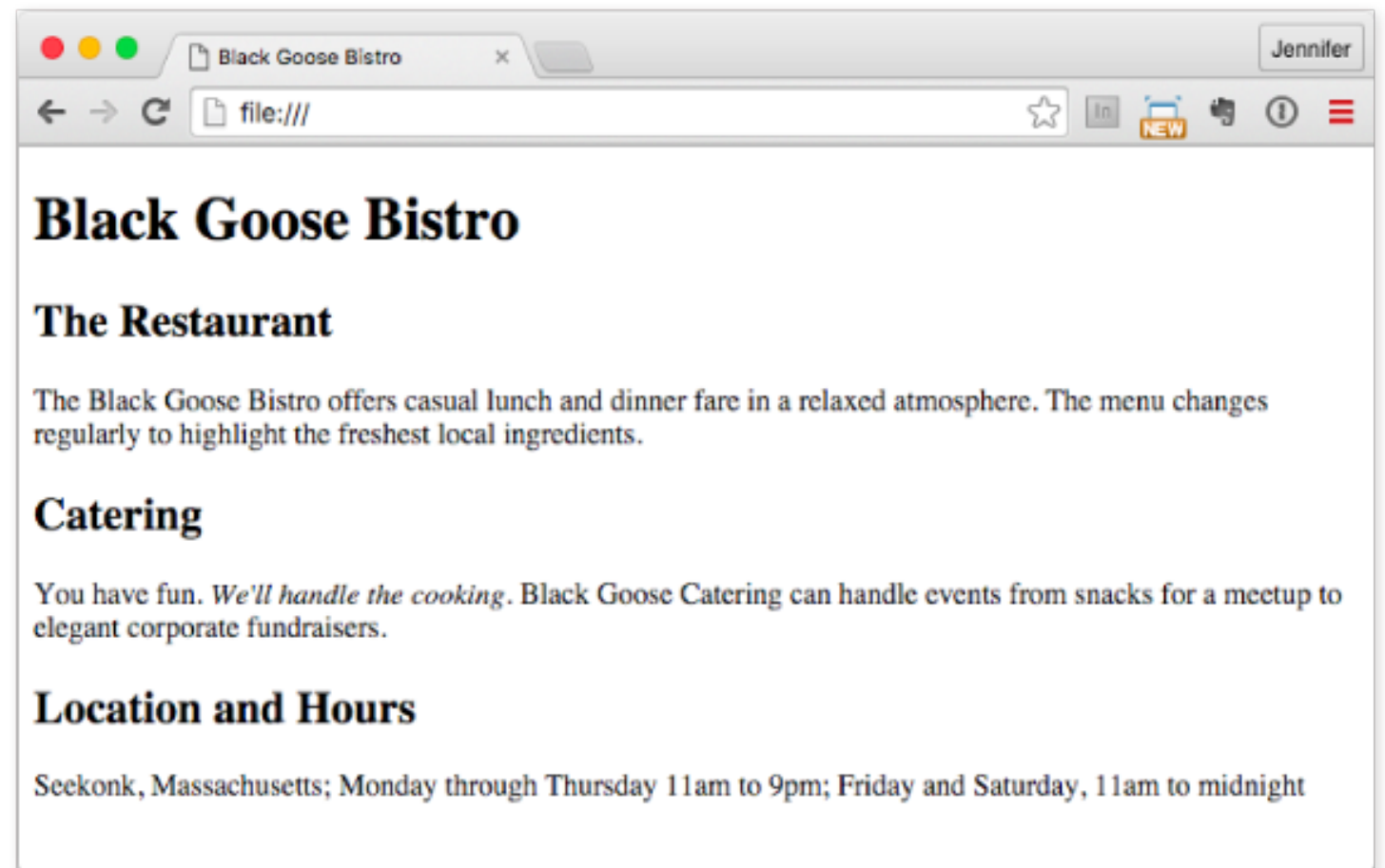
```
<body>
```

```
<h1>Black Goose Bistro</h1>
```

```
<h2>The Restaurant</h2>
```

```
<p>The Black Goose Bistro  
offers casual lunch and  
dinner fare in a relaxed  
atmosphere. The menu changes  
regularly to highlight the  
freshest local  
ingredients.</p>
```

```
<!--more content-->  
</body>
```



# Block and Inline Elements

## Block elements

Major components of the page that display like rectangular blocks stacking up on the page

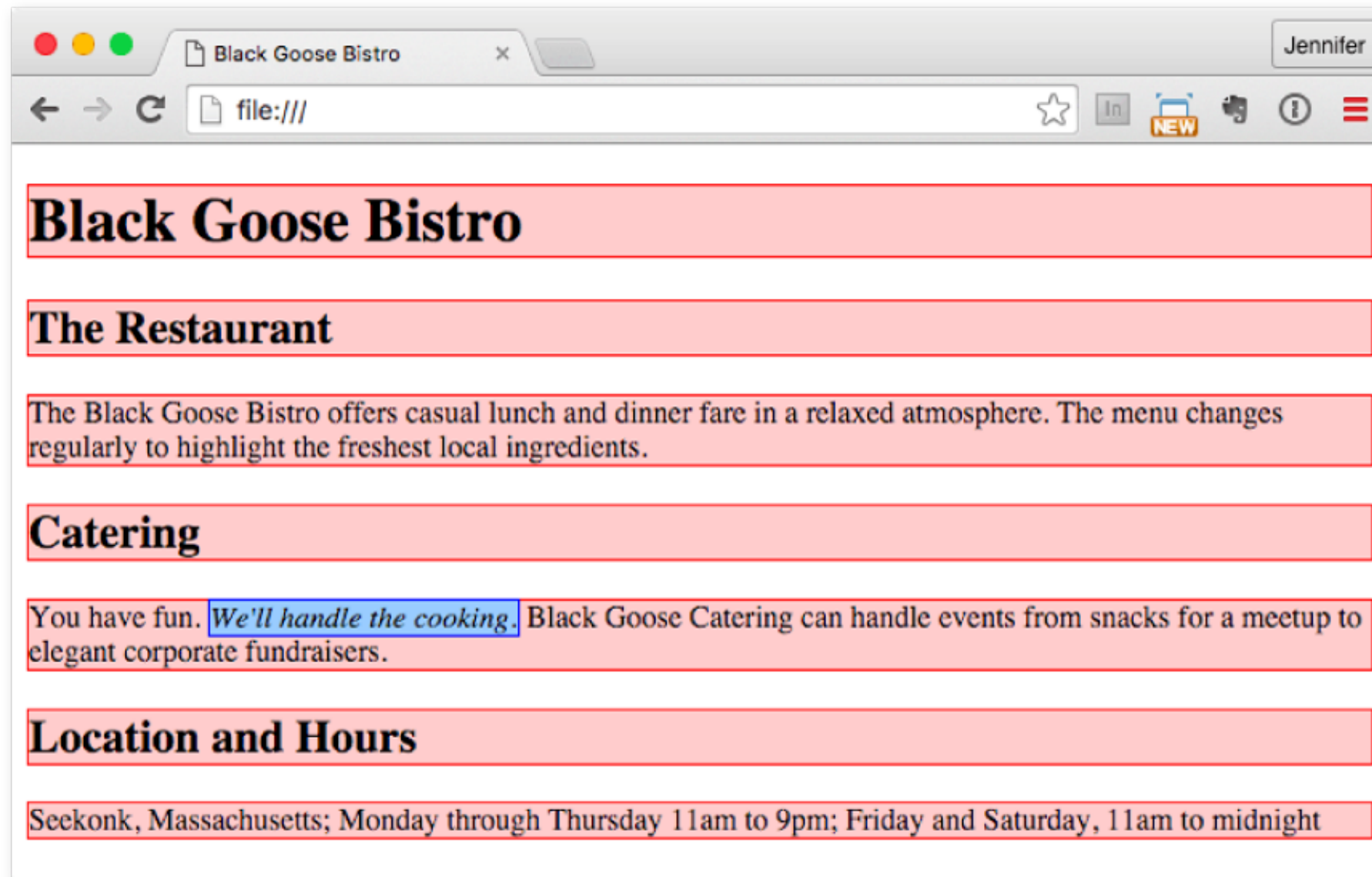
Example: headings, paragraphs, long quotations

## Inline elements

Appear within the text flow of a block element

Example: emphasized text, links, abbreviations

# Block and Inline Elements (cont'd)



Block elements (headings, paragraphs) outlined in RED.

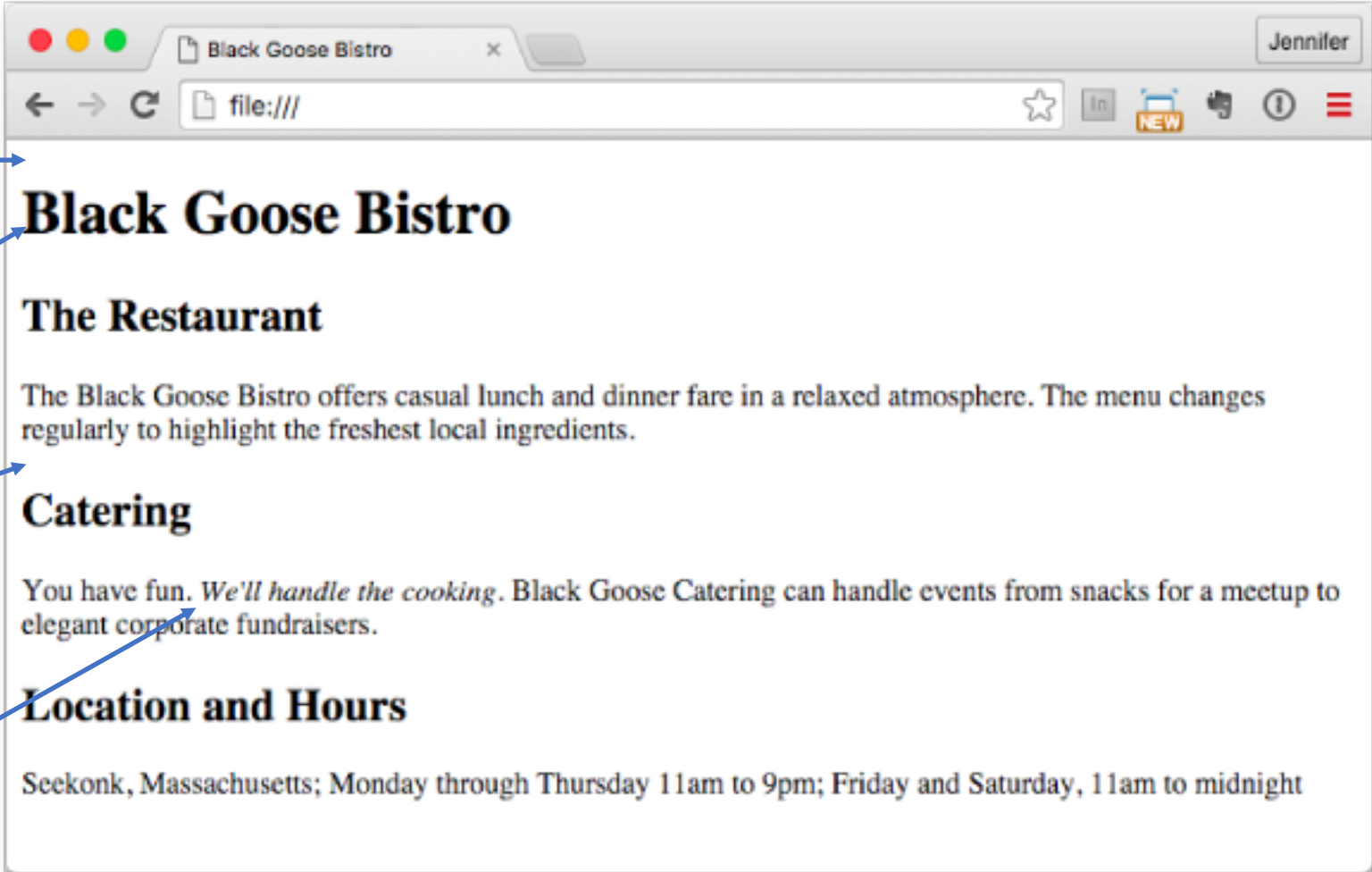
Inline element (emphasized text) outlined in BLUE.

# Style Sheets

- HTML only describes structure, not presentation
- Presentation is controlled by style sheets (CSS)
- Browsers have their own style sheets (user agent style sheets) with default styles for HTML elements
- You can write your own styles to override the default styles

# Style Sheets (cont'd)

The browser parsed the markup and used its built-in style sheet to format the text elements in the example:



The screenshot shows a web browser window with the title 'Black Goose Bistro' and a user profile 'Jennifer'. The address bar shows 'file:///'. The page content includes a main heading 'Black Goose Bistro', a subheading 'The Restaurant', a paragraph of text, another subheading 'Catering', a paragraph of text with some italicized words, and a final subheading 'Location and Hours' followed by a paragraph of text. Blue arrows point from text labels on the left to specific elements on the page:

- Margin around page
- Headings bold and large
- Space added above block elements
- Emphasized text in italic font

# Troubleshooting HTML

Small mistakes and missing characters can cause HTML documents to “break.”

Common mishaps:

- Missing closing tag (or / in closing tag)
- Missing closing bracket in a tag
- Missing quotation mark around an attribute value
- Not saving your document before viewing changes in the browser

# Troubleshooting (cont'd)

When a slash is omitted, the browser doesn't know when the element ends:

```
<h2>Catering</h2>  
<p>You have fun. <em>We'll handle the cooking.<em> Black Goose  
Catering can handle events from snacks for a meetup to elegant  
corporate fundraisers.</p>
```

g. <em>

## Catering

*You have fun. We'll handle the cooking. Black Goose Catering can handle events from snacks for a meetup to elegant corporate fundraisers.*

## Location and Hours

*Seekonk, Massachusetts;  
Monday through Thursday 11am to 9pm;  
Friday and Saturday, 11am to midnight*

# Troubleshooting (cont'd)

A missing end bracket makes the browser interpret all the following characters as part of the tag:

```
<h2The Restaurant</h2>
```

```
<p>The Black Goose Bistro offers casual lunch and dinner fare  
in a relaxed atmosphere. The menu changes regularly to highlight  
the freshest local ingredients.</p>
```

```
<h2The
```

Missing subhead

Without the bracket, all the following characters are interpreted as part of the tag, and “The Restaurant” disappears from the page.

## BLACK GOOSE

The Black Goose Bistro offers casual lunch and dinner fare in a relaxed atmosphere. The menu changes regularly to highlight the freshest local ingredients.

**Catering**

You have fun. *We'll handle the cooking.* Black Goose offers a meetup to elegant corporate fundraisers.

# Validating Your Documents

Validate a document to make sure that you have abided by the HTML rules and that there are no errors:

- Include the DOCTYPE declaration
- Indicate the character encoding
- Include required attributes
- Don't use non-standard elements
- Don't mismatch tags
- Nest elements correctly (no overlaps)
- Check for typos and other minor errors

# Validating Your Documents

- Use a validator tool like the one at <https://validator.w3.org/>
- Upload a document or provide a link, and the validator returns an error report.

(X)HTML5 validation results for ex4-1 index.html

Validator Input

File Upload  no file selected

☐ Show Image Report

☐ Show Source

Group Messages

1. Error: The character encoding was not declared. Proceeding using windows-1252.
2. Error: Non-space characters found without seeing a doctype first. Expected <!DOCTYPE html>.  
From line 1, column 1; to line 11, column 75  
Black Goose Bistro<^>The Restaurant<^>The Black Goose Bistro offers casual lunch and dinner fare in a relaxed atmosphere. The menu changes regularly to highlight the freshest local ingredients.<^>Catering<^>You have fun. We'll handle the cooking. Black Goose Catering can handle events from snacks for a meetup to elegant corporate fundraisers.<^>Location and Hours<^>Seekonk, Massachusetts;<^>Monday through Thursday 11am to 9pm; Friday and Saturday, 11am to midnight<^>
3. Error: Element head is missing a required instance of child element title.  
From line 1, column 1; to line 11, column 75  
Black Goose Bistro<^>The Restaurant<^>The Black Goose Bistro offers casual lunch and dinner fare in a relaxed atmosphere. The menu changes regularly to highlight the freshest local ingredients.<^>Catering<^>You have fun. We'll handle the cooking. Black Goose Catering can handle events from snacks for a meetup to elegant corporate fundraisers.<^>Location and Hours<^>Seekonk, Massachusetts;<^>Monday through Thursday 11am to 9pm; Friday and Saturday, 11am to midnight<^>

Content model for element head:

If the document is an [iframe srcdoc document](#) or if title information is available from a higher-level protocol: Zero or more elements of [metadata content](#), of which no more than one is a [title](#) element and no more than one is a [base](#) element.

Otherwise: One or more elements of [metadata content](#), of which exactly one is a [title](#) element and no more than one is a [base](#) element.

A [head](#) element's [start tag](#) can be omitted if the element is empty, or if the first thing inside the [head](#) element is an element.

A [head](#) element's [end tag](#) can be omitted if the [head](#) element is not immediately followed by a [space character](#) or a [comment](#).

There were errors. (Tried in the text/html mode.)

The Content-Type was text/html. Used the HTML parser.

Total execution time 2 milliseconds.

[About this Service](#) • [More options](#)

# MARKING UP TEXT

- It is important to mark up content semantically, in a way that accurately describes the content's meaning or function.
- This ensures your content is accessible in the widest range of viewing environments:
  - Desktop and mobile browsers
  - Assistive reading devices
  - Search engine indexers

# Paragraphs

`<p> </p>`

Paragraphs are the most rudimentary elements in a text document.

`<p>` Serif typefaces have small slabs at the ends of letter strokes. In general, serif fonts can make large amounts of text easier to read.`</p>`

# Headings

`<h#> </h#>`

There are six levels of headings (**h1** to **h6**).

`<h1>`Top-Level Heading`</h1>`

`<p>`This is a regular paragraph that will display at the browser's default font size and weight for comparison.`</p>`

`<h2>`Second-Level Heading`</h2>`

`<h3>`Third-Level heading`</h3>`

`<p>`This is another paragraph for comparison. Of course, you can change the presentation of all of these elements with your own style sheets.`</p>`

# Headings (cont'd)

## **h1 Top-Level Heading**

This is a regular paragraph that will display at the browser's default font size and weight for comparison.

## **h2 Second-Level Heading**

## **h3 Third-Level heading**

This is another paragraph for comparison. Of course, you can change the presentation of all of these elements with your own style sheets.

## **h4 Fourth Level Heading**

## **h5 Fifth Level Heading**

## **h6 Sixth-Level Heading**

This is another paragraph to show the default relationship of headings to body paragraphs. Of course, you can change the presentation of all of these elements with your own style sheets.

# Long Quotations (blockquote)

`<blockquote> </blockquote>`

`<p>Renowned type designer, Matthew Carter, has this to say about his profession:</p>`

`<blockquote>`

`<p>Our alphabet hasn't changed in eons; there isn't much latitude in what a designer can do with the individual letters.</p>`

`<p>Much like a piece of classical music, the score is written down. It's not something that is tampered with, and yet, each conductor interprets that score differently. There is tension in the interpretation.</p>`

`</blockquote>`

Renowned type designer, Matthew Carter, has this to say about his profession:

Our alphabet hasn't changed in eons; there isn't much latitude in what a designer can do with the individual letters.

Much like a piece of classical music, the score is written down. It's not something that is tampered with, and yet, each conductor interprets that score differently. There is tension in the interpretation.

# Preformatted Text

<pre> </pre>

Preformatted text preserves white space when it is important for conveying meaning. By default, **pre** text displays in a constant-width font, such as Courier.

<pre>

```
This is          an          example of
      text with a      lot of
                           curious
                           whitespace.
```

</pre>

```
This is          an          example of
      text with a      lot of
                           curious
                           whitespace.
```

# Line Breaks

`<br>`

The empty br element inserts a line break.

`<p>So much depends <br>upon <br><br>a red wheel <br>barrow</p>`

So much depends  
upon

a red wheel  
barrow

# Thematic Breaks (Horizontal Rules)

`<hr>`

Indicates one topic has completed and another one is beginning. Browsers display a horizontal rule (line) in its place:

```
<h3>Times</h3>
<p>Description and history of the Times typeface.</p>
<hr>
<h3>Georgia</h3>
<p>Description and history of the Georgia typeface.</p>
```

## **Times**

Description and history of the Times typeface.

---

## **Georgia**

Description and history of the Georgia typeface.

# Lists

There are three types of lists in HTML:

- Unordered lists
- Ordered lists
- Description lists

# Unordered Lists

In unordered lists items may appear in any order (examples, names, options, etc.). Most lists fall into this category.

`<ul>` `</ul>` Defines the whole list  
`<li>` `</li>` Defines each list item

# Unordered Lists (cont'd)

```
<ul>
  <li>Serif</li>
  <li>Sans-serif</li>
  <li>Script</li>
  <li>Display</li>
  <li>Dingbats</li>
</ul>
```

You can change the appearance of the list dramatically with style sheet rules.

- Serif
- Sans-serif
- Script
- Display
- Dingbats

Serif  
Sans-serif  
Script  
Display  
Dingbats

SERIF  
SANS-SERIF  
SCRIPT  
DISPLAY  
DINGBATS

Serif Sans-serif Script Display Dingbats

SERIF SANS-SERIF SCRIPT DISPLAY DINGBATS

# Ordered Lists

In ordered lists items occur in a particular order, such as step-by-step instructions or driving directions.

|                         |                          |                        |
|-------------------------|--------------------------|------------------------|
| <code>&lt;ol&gt;</code> | <code>&lt;/ol&gt;</code> | Defines the whole list |
| <code>&lt;li&gt;</code> | <code>&lt;/li&gt;</code> | Defines each list item |

# Ordered Lists (cont'd.)

```
<ol>  
  <li>Gutenberg develops moveable type (1450s)</li>  
  <li>Linotype is introduced (1890s)</li>  
  <li>Photocomposition catches on (1950s)</li>  
  <li>Type goes digital (1980s)</li>  
</ol>
```

1. Gutenberg develops moveable type (1450s)
2. Linotype is introduced (1890s)
3. Photocomposition catches on (1950s)
4. Type goes digital (1980s)

# Description Lists

Description lists are used for any type of name/value pairs, such as terms/definitions, questions/answers, etc.

- |                         |                          |                                       |
|-------------------------|--------------------------|---------------------------------------|
| <code>&lt;dl&gt;</code> | <code>&lt;/dl&gt;</code> | Defines the whole list                |
| <code>&lt;dt&gt;</code> | <code>&lt;/dt&gt;</code> | Defines a name, such as a term        |
| <code>&lt;dd&gt;</code> | <code>&lt;/dd&gt;</code> | Defines a value, such as a definition |

# Description Lists (cont'd)

`<d1>`

`<dt>Linotype</dt>`

`<dd>Line-casting allowed type to be selected, used, then recirculated into the machine automatically. This advance increased the speed of typesetting and printing dramatically.</dd>`

`<dt>Photocomposition</dt>`

`<dd>Typefaces are stored on film then projected onto photo-sensitive paper. Lenses adjust the size of the type.</dd>`

`</d1>`

## Linotype

Line-casting allowed type to be selected, used, then recirculated into the machine automatically. This advance increased the speed of typesetting and printing dramatically.

## Photocomposition

Typefaces are stored on film then projected onto photo-sensitive paper. Lenses adjust the size of the type.

# Page Organizing Elements

HTML5 introduced elements that give meaning to the typical sections of a web page:

- `main`
- `header`
- `footer`
- `section`
- `article`
- `aside`
- `nav`

# Main Content

`<main> </main>`

- Identifies the primary content of a page or application
- Helps users with screen readers get to the main content of the page

```
<body>
<header>...</header>
<main>
  <h1>Humanist Sans Serif</h1>
  ...content continues...
</main>
</body>
```

# Headers and Footers

`<header> </header>`

`<footer> </footer>`

**header** identifies the introductory material that comes at the beginning of a page, section, or article (logo, title, navigation, etc.).

**footer** indicates the type of information that comes at the end of a page, section, or article (author, copyright, etc.)

# Headers and Footers (cont'd)

```
<article>
```

```
  <header>
```

```
    <h1>More about WOFF</h1>
```

```
    <p>by Jennifer Robbins, <timedatettime="2017-11-11">  
      November 11, 2017</time></p>
```

```
  </header>
```

```
  <!-- ARTICLE CONTENT HERE -->
```

```
  <footer>
```

```
    <p><small>Copyright &copy;2017 Jennifer  
Robbins.</small></p>
```

```
    <nav>
```

```
      <ul>
```

```
        <li><a href="/">Previous</a></li>
```

```
        <li><a href="/">Next</a></li>
```

```
      </ul>
```

```
    </nav>
```

```
  </footer>
```

```
</article>
```

# Footer Element in HTML

```
<footer>
```

```
<p>&copy; 2026 My Cooking Hobby. All rights reserved.</p>
```

```
<p>Contact: myemail@example.com</p>
```

```
<p>
```

Follow us on:

```
<a href="https://twitter.com" target="_blank">Twitter</a> |
```

```
<a href="https://facebook.com" target="_blank">Facebook</a> |
```

```
<a href="https://instagram.com" target="_blank">Instagram</a>
```

```
</p>
```

```
</footer>
```

# Inserting special characters

| Symbol                   | Entity Name | Entity Number |
|--------------------------|-------------|---------------|
| & (ampersand)            | &amp;       | &#38;         |
| < (less than)            | &lt;        | &#60;         |
| > (greater than)         | &gt;        | &#62;         |
| (nonbreaking space)      | &nbsp;      | &#160;        |
| ¢ (cent)                 | &cent;      | &#162;        |
| £ (pound)                | &pound;     | &#163;        |
| ¥ (yen)                  | &yen;       | &#165;        |
| © (copyright)            | &copy;      | &#169;        |
| ® (registered trademark) | &reg;       | &#174;        |
| ° (degree)               | &deg;       | &#176;        |
| ± (plus or minus)        | &plusmn;    | &#177;        |
| † (dagger)               | &dagger;    | &#8224;       |
| ™ (trademark)            | &trade;     | &#8482;       |

# Sections

`<section> </section>`

**section** identifies thematic section of a page or an article. It can be used to divide up a whole page or a single article:

`<section>`

```
<h2>Typography Books</h2>
```

```
<ul>
```

```
<li>...</li>
```

```
</ul>
```

`</section>`

`<section>`

```
<h2>Online Tutorials</h2>
```

```
<p>These are the best tutorials on the Web.</p>
```

```
<ul>
```

```
<li>...</li>
```

```
</ul>
```

`</section>`

# Navigation

`<nav> </nav>`

**nav** identifies the primary navigation for a site or lengthy section or article. It provides more semantic meaning than a simple unordered list.

```
<nav>
<ul>
  <li><a href="/">Serif</a></li>
  <li><a href="/">Sans-serif</a></li>
  <li><a href="/">Script</a></li>
  <li><a href="/">Display</a></li>
  <li><a href="/">Dingbats</a></li>
</ul>
</nav>
```

# Inline Elements

- Called text-level semantic elements in the spec.
- Describe the types of elements that appear in the flow of text.

`a`

`em`

`strong`

`q`

`abbr`

`cite`

`dfn`

`code`

`var`

`samp`

`kbd`

`sub/sup`

`mark`

`time`

`data`

`ins/del`

`b`

`i`

`s`

`u`

`small`

`bdi/bdo`

`data`

`span`

# Emphasis

`<em> </em>`

Text that should be emphasized. Usually displayed in italics.

```
<p><em>Arlo</em> is very smart.</p>
```

```
<p>Arlo is <em>very</em> smart.</p>
```

`<strong></strong>`

Text that is important, serious, or urgent. Usually displayed in bold.

```
<p>When returning the car, <strong>drop the keys in the  
red box by the front desk</strong>.</p>
```

**TIP:** Use these elements semantically, not to achieve font styles.  
Think of how it would be read with a screen reader.

# Short Quotations

<q> </q>

For quoted phrases in the flow of text. Browsers add appropriate quotation marks automatically.

<p>Matthew Carter says, <q>Our alphabet hasn't changed in eons.</q></p>

Matthew Carter says, "Our alphabet hasn't changed in eons."

# Abbreviations and Acronyms

`<abbr> </abbr>`

The **title** attribute provides the long version of a shortened term, which is helpful for search engines and assistive devices.

```
<abbr title="Points">pts.</abbr>
```

```
<abbr title="American Type Founders">ATF</abbr>
```

# Superscript and Subscript

`<sup> </sup>`

`<sub> </sub>`

Causes the selected text to display in a smaller size and slightly above (**sup**) or below (**sub**) the baseline.

`<p>H<sub>2</sub>O</p>`

`<p>E=MC<sup>2</sup></p>`

H<sub>2</sub>O

E=MC<sup>2</sup>

# Citations

`<cite> </cite>`

Identifies a reference to another document.

`<p>Passages of this article were inspired by <cite>The Complete Manual of Typography</cite> by James Felici.</p>`

# Further Reading :

Part I: Getting Started

Part II: HTML Markup for Structure

Robbins, J., 2018, Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics, O'Reilly Media

# Tutorial 2

Create a webpage (HomePage.html) for an upcoming event of your choice.

- Identify target audience (e.g., students, families, professionals)
- Decide key message (why should people attend?)
- Use persuasive content in the website