Web Programming I CTEC 21042

Dr. Sandeli Kasthuri Arachchi

Dept. of Software Engineering

Faculty of Computing and Technology

University of Kelaniya.

HOW THE WEB WORKS

IN THIS CHAPTER

An explanation of the Web, as it relates to the Internet

The role of the server

The role of the browser

Introduction to URLs and their components

The anatomy of a web page

The Internet Versus the Web

- There are many ways information can be passed between computers, including email, file transfer (FTP), and many more specialized modes upon which the Internet is built.

- These standardized methods for transferring data or documents over a network are known as **protocols**.

The Internet: Connecting **Computers**


The Web: Connecting **People**

The Internet Versus the Web

- There are many ways information can be passed between computers, including email, file transfer (FTP), and many more specialized modes upon which the Internet is built.

- These standardized methods for transferring data or documents over a network are known as **protocols**.

The Internet: Connecting **Computers**

The Web: Connecting **People**
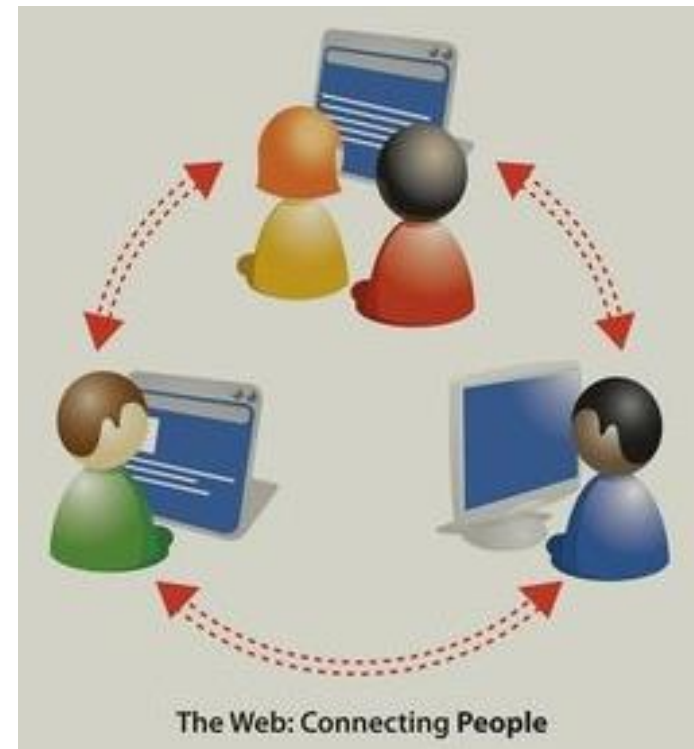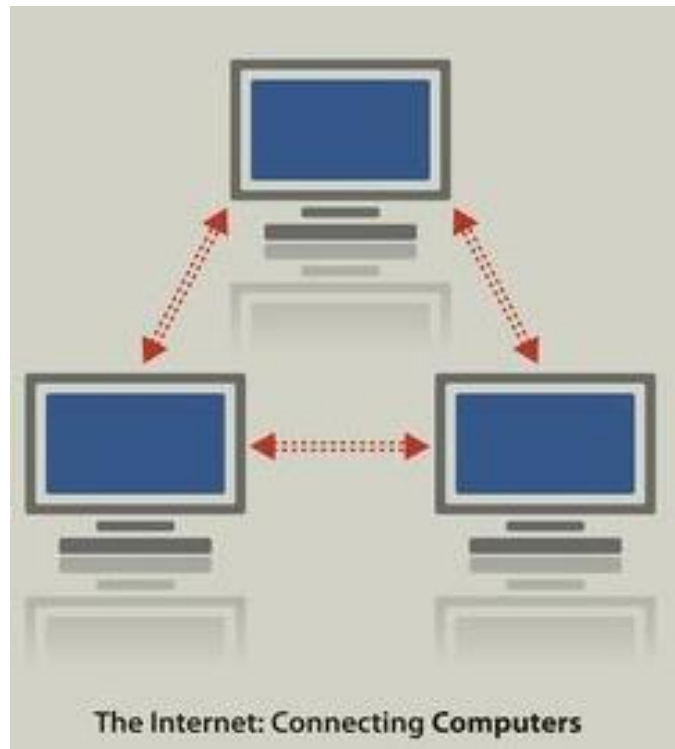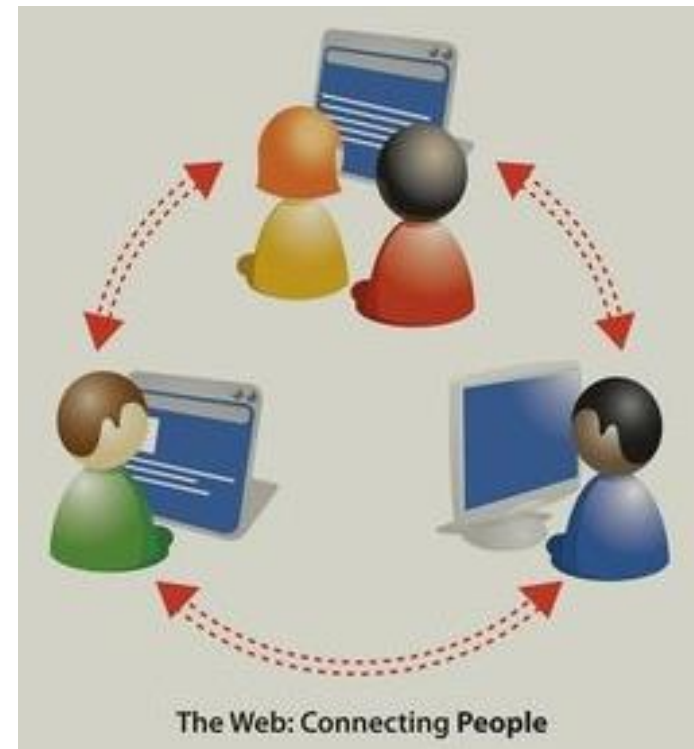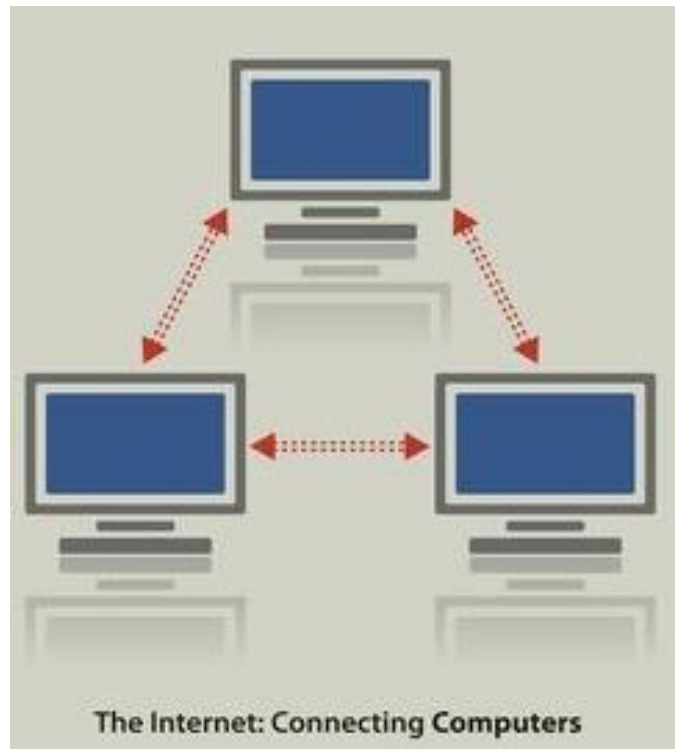
The Internet Versus the Web

- WWW - **W**orld **W**ide **W**eb

- It is unique in that it allows documents to be linked to one another using hypertext links—thus forming a huge "web" of connected information.

- The Web uses a protocol called HTTP (HyperText Transfer Protocol).

The Web is a subset of the Internet. It is just one of many ways information can be transferred over networked computers.

The Internet: Connecting Computers


The Web: Connecting People
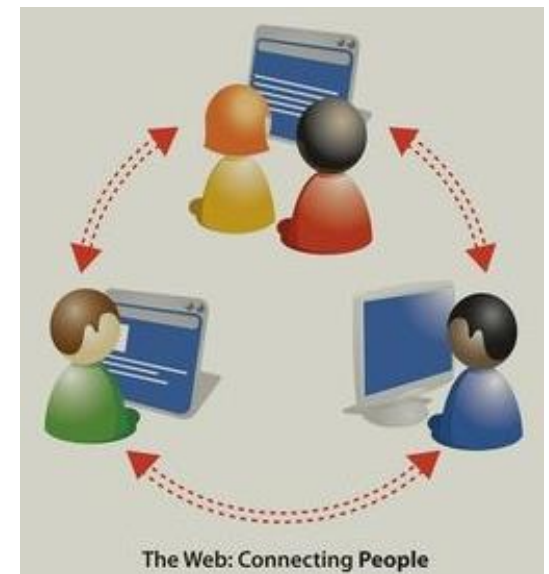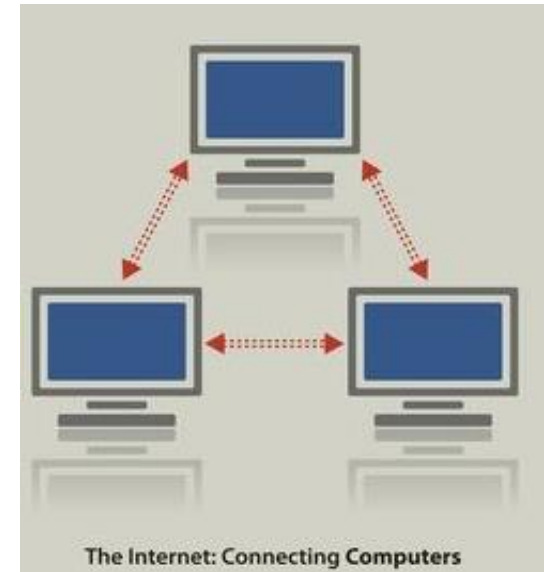
About Browsers

- The software that does the requesting is called the **client**.

- People use desktop browsers, mobile browsers, and other assistive technologies (such as screen readers) as clients to access documents on the Web.

- The server returns the documents for the browser (also referred to as the **user agent** in technical circles) to display

- The most popular desktop browsers as of this writing include Internet Explorer for Windows, Chrome, Firefox, and Safari, with Opera bringing up the rear.

- These days, however, more and more people are accessing the Web on the go using browsing clients built into mobile phones or tablets.

Web Page Addresses (URLs)

- Every page and resource on the Web has its own special address called a URL, which stands for **U**niform **R**esource **L**ocator.

The parts of a URL
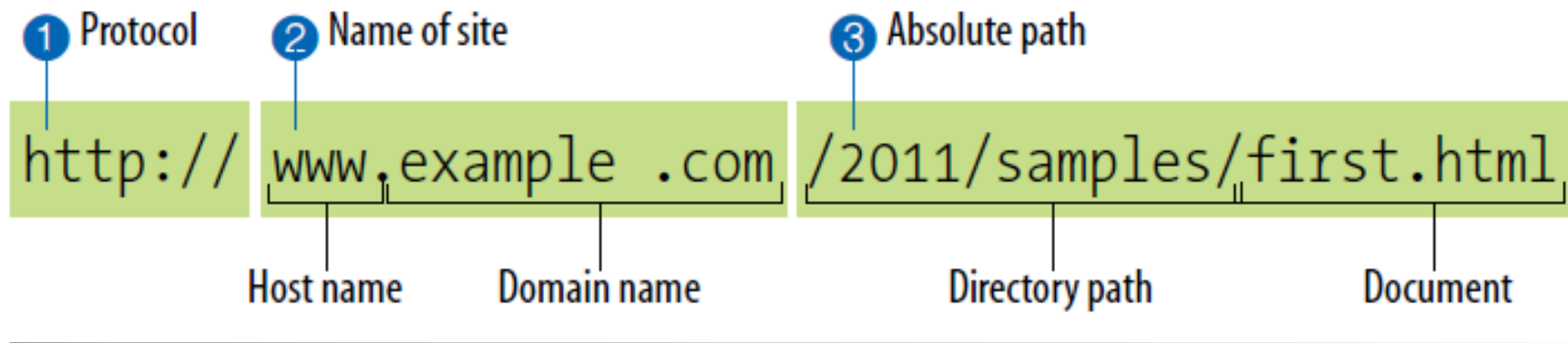
The parts of a URL.

Web Page Addresses (URLs)

- **http://**

The first thing the URL does is define the protocol that will be used for that particular transaction.

The letters HTTP let the server know to use Hypertext Transfer Protocol, or get into "web mode."

- Nearly all web pages use the Hypertext Transfer Protocol, the http:// part is often just implied.

- Browsers are programmed to add http:// automatically as a convenience to save you some keystrokes.

1 Protocol

http://

NOTE

Sometimes you'll see a URL that begins with https://. This is an indication that it is a secure server transaction. Secure servers have special encryption devices that hide delicate content, such as credit card numbers, while they are transferred to and from the browser. Look for it the next time you're shopping online.
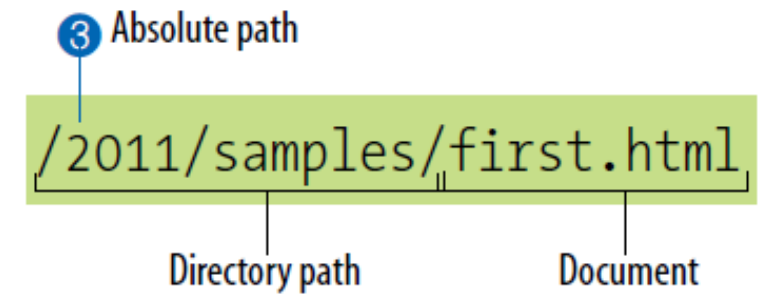
Web Page Addresses (URLs)

② Name of site

www.example .com

Host name   Domain name
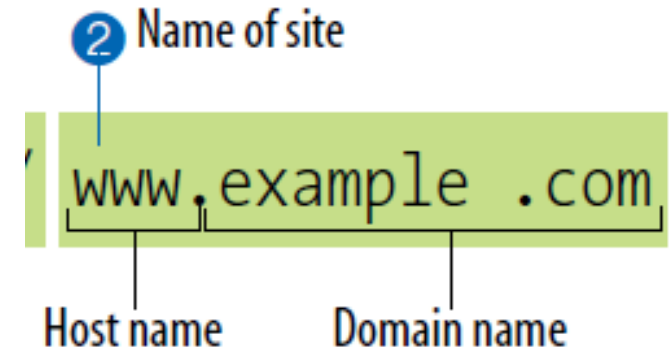
- ***www.example.com***
- In this example, the domain name is example.com.
- The "www." part at the beginning is the particular host name at that domain.
- The host name "www" has become a convention, but is not a rule.
- In fact, sometimes the host name may be omitted.
- There can be more than one website at a domain (sometimes called subdomains)
    - Example, there might also be *development.example.com*, *clients.example.com*, and so on

Web Page Addresses (URLs)

- **/2011/samples/first.html**

- The absolute path through directories on the server to the requested HTML document, first.html.

- The words separated by slashes are the directory names, starting with the root directory of the host (as indicated by the initial /).

③ Absolute path

/2011/samples/first.html

Directory path   Document

Web Page Addresses (URLs)

- It would like to use the HTTP protocol to connect to a web server on the Internet called *www.example.com* and request the document *first.html* (located in the *samples* directory, which is in the *2011* directory).
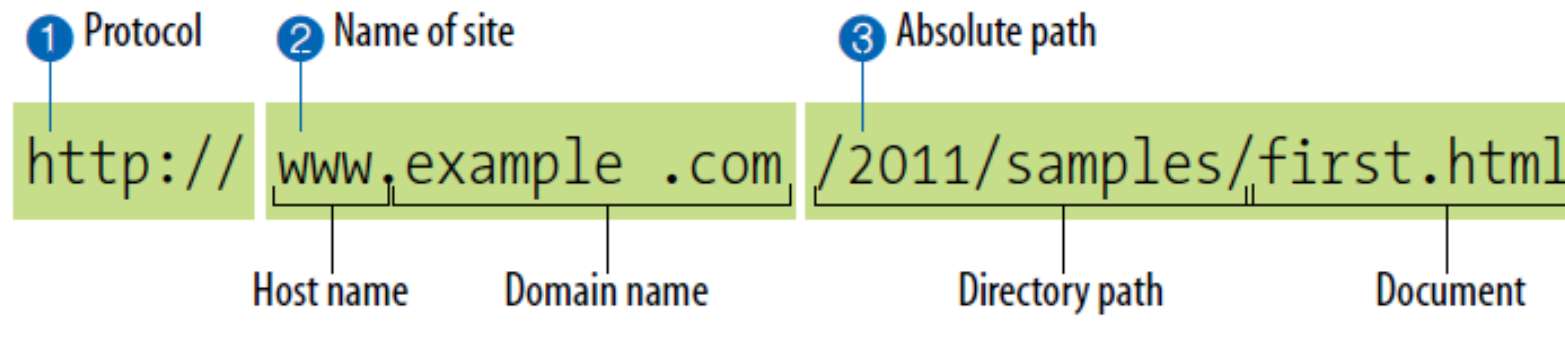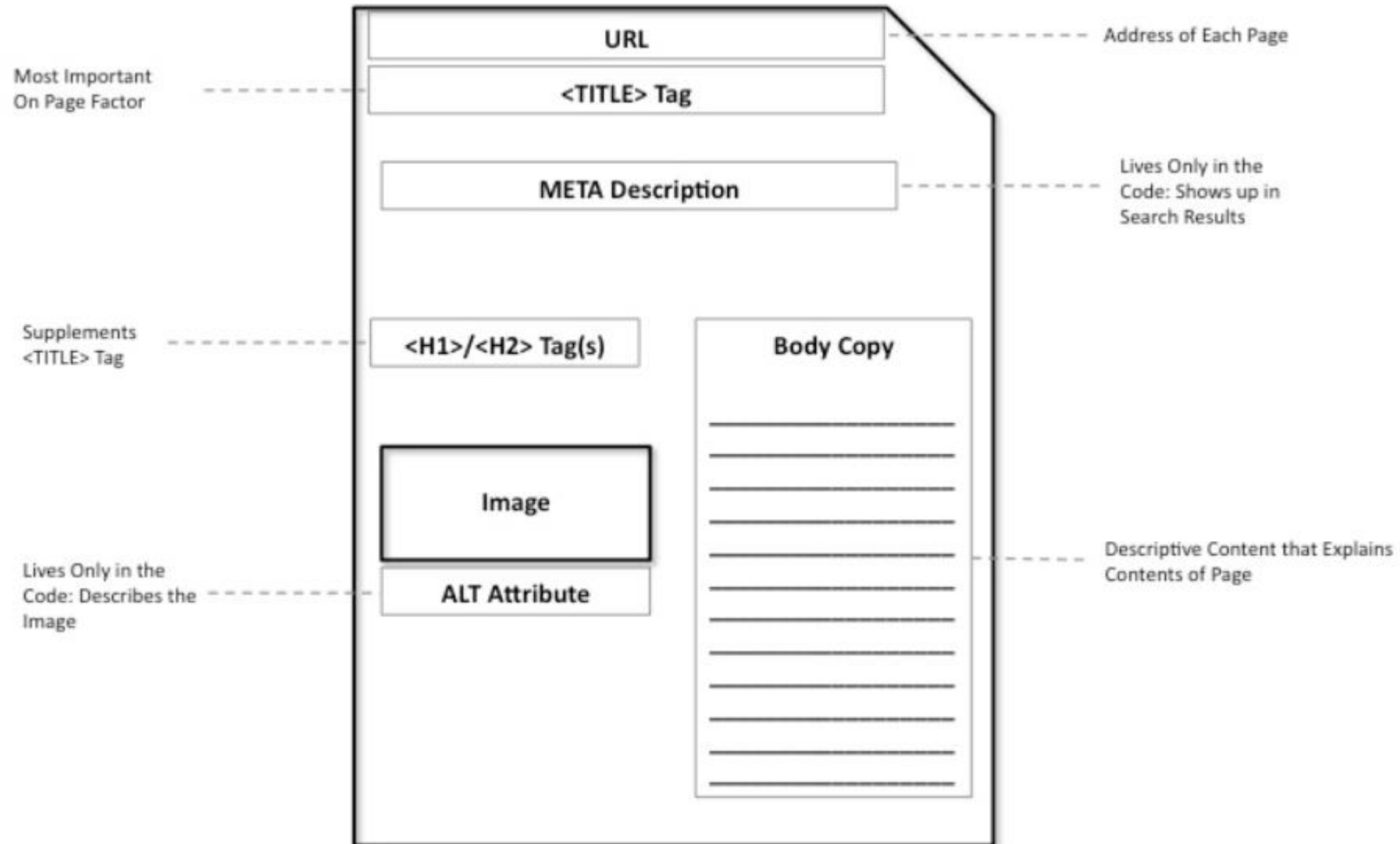

- Many addresses do not include a filename, but simply point to a directory, like these:

  http://www.oreilly.com

  http://www.jendesign.com/resume/

The Anatomy of a Web Page

URL — Address of Each Page

Most Important On Page Factor — <TITLE> Tag

META Description — Lives Only in the Code: Shows up in Search Results

Supplements <TITLE> Tag — <H1>/<H2> Tag(s)

Body Copy

Image

Lives Only in the Code: Describes the Image — ALT Attribute

Descriptive Content that Explains Contents of Page

```
<!DOCTYPE html>
<html>
<head>
<title>Jen's Kitchen</title>
<link rel="stylesheet" href="kitchen.css" type="text/css" >
</head>

<body>
<h1><img src="foods.gif" alt="food illustration"> Jen&rsquo;s Kitchen</h1>

<p>If you love to read about <strong>cooking and eating</strong>, would like to find out about
of some of the best restaurants in the world, or just want a few choice recipes to add to your
collection, <em>this is the site for you!</em></p>

<p><img src="spoon.gif" alt="spoon illustration"> Your pal, Jen at Jen's Kitchen</p>
<hr>
<p><small>Copyright 2011, Jennifer Robbins</small></p>
</body>
</html>
```

kitchen.css

```
body { font: normal 1em Verdana; margin: 1em 10%;}
h1 { font: italic 3em Georgia; color: rgb(23, 109, 109); margin: 1em 0 1em;}
img { margin: 0 20px 0 0; }
h1 img { margin-bottom: -20px; }
small { color: #666666; }
```

foods.gif

spoon.gif

. *The source file and images that make up a simple web page.*

Four separate files:
- HTML document (*index.html*),
- A style sheet (*kitchen.css*), and
- Two graphics (*foods.gif* and *spoon.gif*).

13

HTML Documents

- Adding descriptive tags to a text document is known as "**marking up**" the document.

- Web pages use a markup language called *HyperText Markup Language*, or HTML for short, which was created especially for documents with hypertext links.

- HTML defines dozens of text elements that make up documents such as headings, paragraphs, emphasized text, and of course, links.

index.html

```
<!DOCTYPE html>
<html>
<head>
<title>Jen's Kitchen</title>
<link rel="stylesheet" href="kitchen.css" type="text/css" >
</head>

<body>
<h1><img src="foods.gif" alt="food illustration"> Jen&rsquo;s Kitchen</h1>

<p>If you love to read about <strong>cooking and eating</strong>, would like to find out about of some of the best restaurants in the world, or just want a few choice recipes to add to your collection, <em>this is the site for you!</em></p>

<p><img src="spoon.gif" alt="spoon illustration"> Your pal, Jen at Jen's Kitchen</p>
<hr>
<p><small>Copyright 2011, Jennifer Robbins</small></p>
</body>
</html>
```

A quick introduction to HTML markup

Jen's Kitchen

If you love to read about **cooking and eating**, would like to find out about of some of the best restaurants in the world, or just want a few choice recipes to add to your collection, *this is the site for you!*

Your pal, Jen at Jen's Kitchen

Copyright 2011, Jennifer Robbins

But where are the pictures?

- The images are placed in the flow of the text with the HTML image element (img) that tells the browser where to find the graphic (its URL).

- When the browser sees the *img* element, it makes another request to the server for the image file, and then places it in the content flow.

- Videos and other embedded media files are added in much the same way.

index.html

```
<!DOCTYPE html>
<html>
<head>
<title>Jen's Kitchen</title>
<link rel="stylesheet" href="kitchen.css" type="text/css" >
</head>

<body>
<h1><img src="foods.gif" alt="food illustration"> Jen&rsquo;s Kitchen</h1>

<p>If you love to read about <strong>cooking and eating</strong>, would like to find out about
of some of the best restaurants in the world, or just want a few choice recipes to add to your
collection, <em>this is the site for you!</em></p>

<p><img src="spoon.gif" alt="spoon illustration"> Your pal, Jen at Jen's Kitchen</p>
<hr>
<p><small>Copyright 2011, Jennifer Robbins</small></p>
</body>
</html>
```
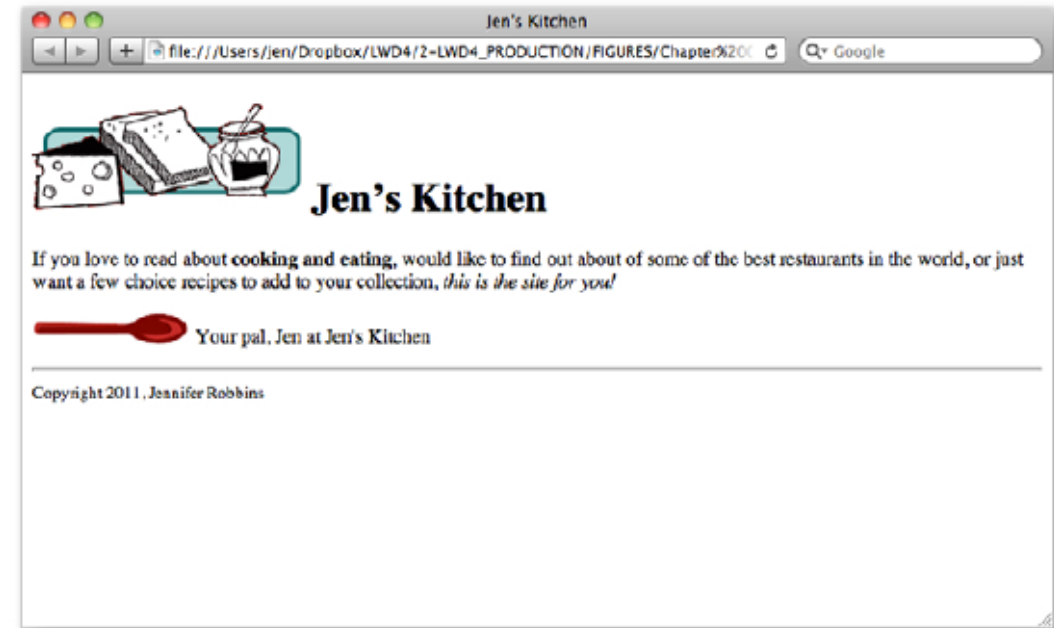
foods.gif

spoon.gif

Adding a little style

- That style sheet includes a few lines of instructions for how the page should look in the browser.

- These are style instructions written according to the rules of Cascading Style Sheets (CSS).

- CSS allows designers to add visual style instructions (known as the document's presentation) to the marked-up text

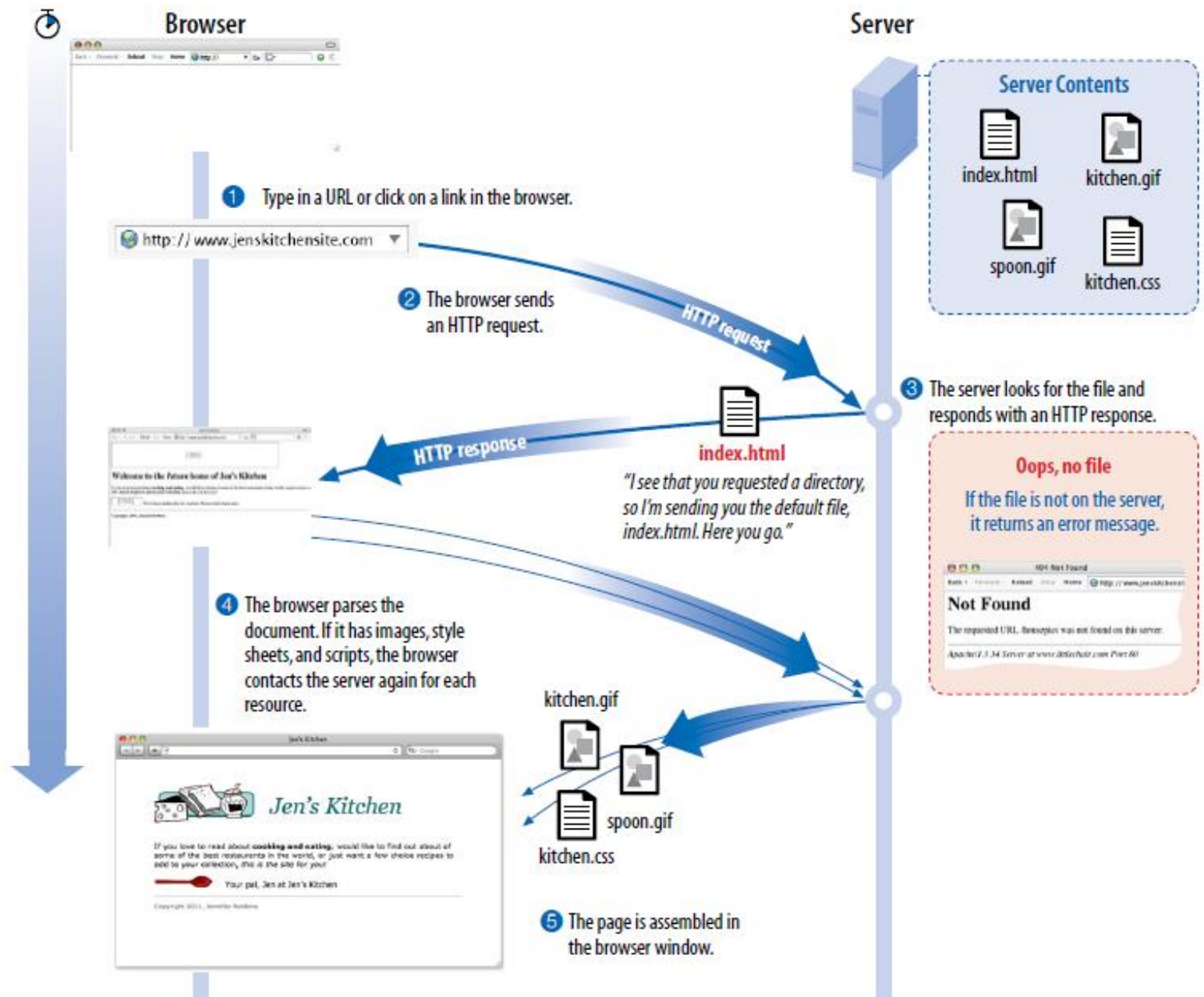*The Jen's Kitchen page before (top) and after (bottom) style rules.*

Adding a little style

- CSS allows designers to add visual style instructions (known as the document's presentation) to the marked-up text
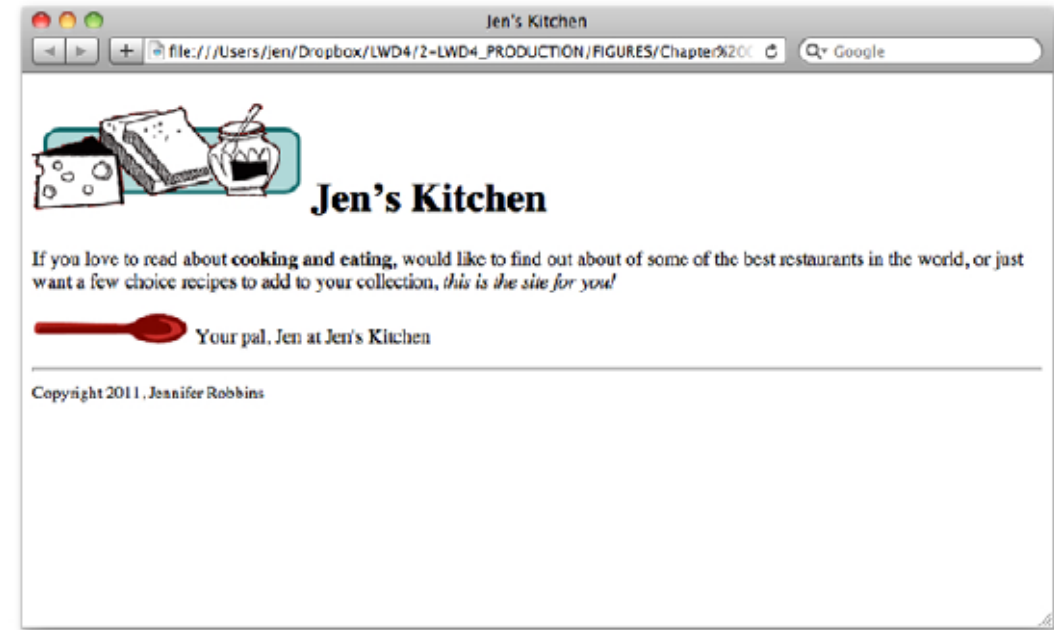
The Jen's Kitchen page before (top) and after (bottom) style rules.

Putting It All Together

Putting It All Together

❶ You request a web page by either typing its URL (for example, *http://jenskitchensite.com*) directly in the browser or by clicking on a link on a page. The URL contains all the information needed to target a specific document on a specific web server on the Internet.

❷ Your browser sends an HTTP Request to the server named in the URL and asks for the specific file. If the URL specifies a directory (not a file), it is the same as requesting the default file in that directory.

❸ The server looks for the requested file and issues an HTTP response.

   a. If the page cannot be found, the server returns an error message. The message typically says "404 Not Found," although more hospitable error messages may be provided.

   b. If the document *is* found, the server retrieves the requested file and returns it to the browser.

❹ The browser parses the HTML document. If the page contains images (indicated by the HTML **img** element) or other external resources like scripts, the browser contacts the server again to request each resource specified in the markup.

❺ The browser inserts each image in the document flow where indicated by the **img** element. And *voila!* The assembled web page is displayed for your viewing pleasure.

Browser

Server

Server Contents

index.html   kitchen.gif

spoon.gif   kitchen.css

① Type in a URL or click on a link in the browser.

http://www.jenskitchensite.com

② The browser sends an HTTP request.

HTTP request

③ The server looks for the file and responds with an HTTP response.

HTTP response

index.html

"I see that you requested a directory, so I'm sending you the default file, index.html. Here you go."

Oops, no file

If the file is not on the server, it returns an error message.

Not Found

④ The browser parses the document. If it has images, style sheets, and scripts, the browser contacts the server again for each resource.

kitchen.gif

Jen's Kitchen

spoon.gif

kitchen.css
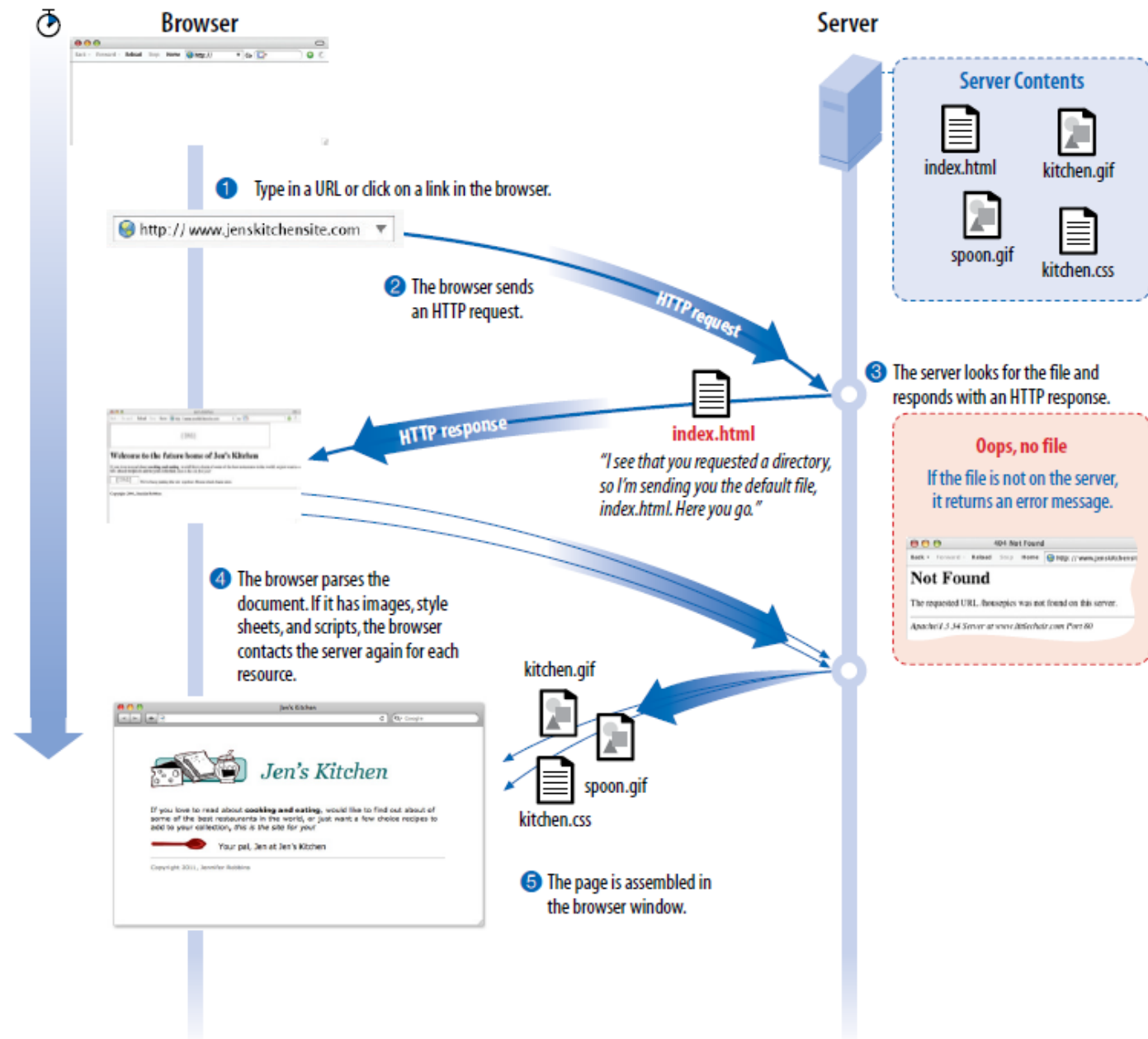
⑤ The page is assembled in the browser window.

Figure 2-5. *How browsers display web pages.*

Learn HTML Using Notepad or TextEdit

Web pages can be created and modified by using professional HTML editors.

However, for learning HTML we recommend a simple text editor like Notepad (PC) or TextEdit (Mac).

We believe that using a simple text editor is a good way to learn HTML.

Follow the steps below to create your first web page with Notepad or TextEdit.

Step 1: Open Notepad (PC)

Windows 8 or later:

Open the **Start Screen** (the window symbol at the bottom left on your screen). Type **Notepad**.

Windows 7 or earlier:

Open **Start** > **Programs** > **Accessories** > **Notepad**

Step 2: Write Some HTML

Write or copy the following HTML code into Notepad:

```html
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>
</html>
```

Untitled - Notepad

File   Edit   Format   View   Help

```html
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

</body>
</html>
```

22
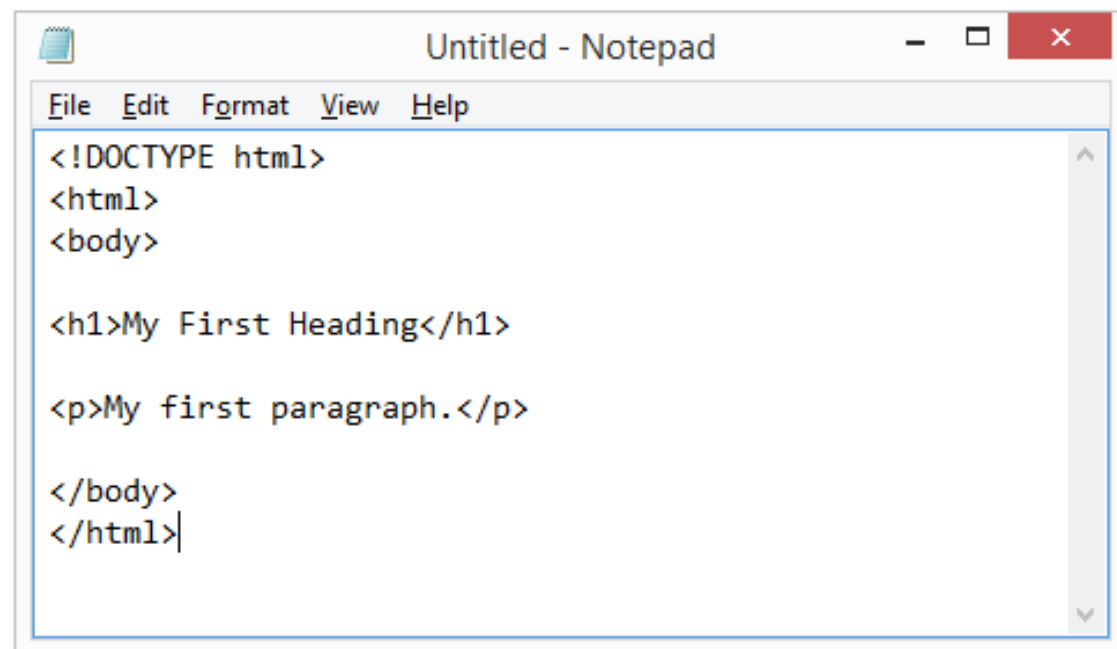
Step 3: Save the HTML Page

Save the file on your computer. Select **File > Save as** in the Notepad menu.

Name the file **"index.htm"** and set the encoding to **UTF-8** (which is the preferred encoding for HTML files).
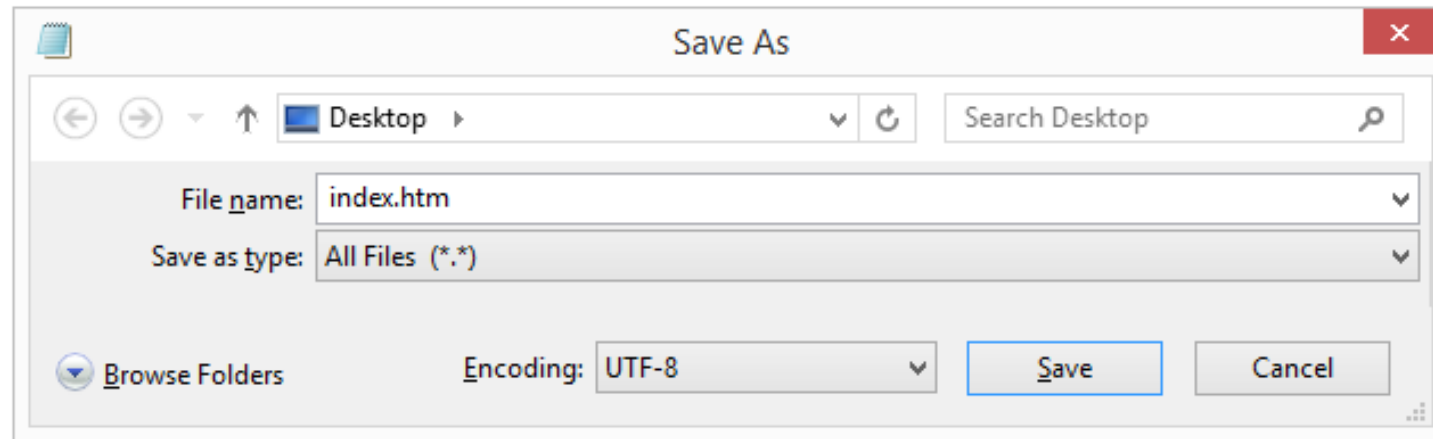
How to Name HTML Files

- Don't Use Special Characters.
- Don't Use Spaces.
- Start the File Name With a Letter.
- Use All Lowercase Letters.
- Keep Your Filenames Short.
- Remember the File Extension.
- Good HTML File-Naming Practices.

Tip: You can use either .htm or .html as file extension. There is no difference; it is up to you.

23

Step 4: View the HTML Page in Your Browser

Open the saved HTML file in your favorite browser (double click on the file, or right-click - and choose "Open with").

The result will look much like this:
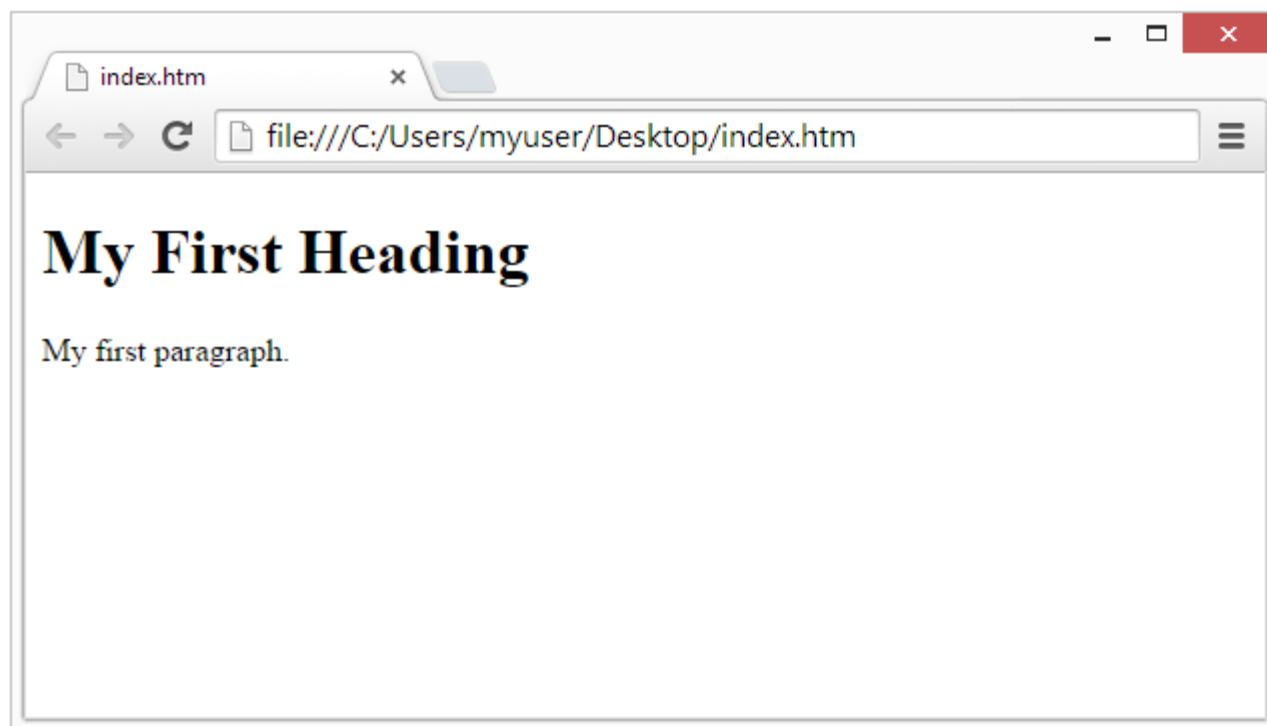
HTML Documents

All HTML documents must start with a document type declaration: `<!DOCTYPE html>`.

The HTML document itself begins with `<html>` and ends with `</html>`.

The visible part of the HTML document is between `<body>` and `</body>`.

Example

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

My First Heading

My first paragraph.

A quick introduction to HTML markup

The <!DOCTYPE> Declaration

The `<!DOCTYPE>` declaration represents the document type, and helps browsers to display web pages correctly.

It must only appear once, at the top of the page (before any HTML tags).

The `<!DOCTYPE>` declaration is not case sensitive.

The `<!DOCTYPE>` declaration for HTML5 is:

```
<!DOCTYPE html>
```

HTML Headings

HTML headings are defined with the `<h1>` to `<h6>` tags.

`<h1>` defines the most important heading. `<h6>` defines the least important heading:

Example

```
<!DOCTYPE html>
<html>
<body>

<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>

</body>
</html>
```

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

27

HTML Paragraphs

HTML paragraphs are defined with the `<p>` tag:

Example

```
<!DOCTYPE html>
<html>
<body>

<p>This is a paragraph.</p>
<p>This is another paragraph.</p>

</body>
</html>
```

This is a paragraph.

This is another paragraph.

HTML Links

HTML links are defined with the `<a>` tag:

Example

```
<!DOCTYPE html>
<html>
<body>

<h2>HTML Links</h2>
<p>HTML links are defined with the a tag:</p>

<a href="https://www.w3schools.com">This is a link</a>

</body>
</html>
```

HTML Links

HTML links are defined with the a tag:

This is a link

Learn to Code

With the world's largest web developer site.

29

HTML Images

HTML images are defined with the `<img>` tag.

The source file (`src`), alternative text (`alt`), `width` , and `height` are provided as attributes:

Example

```
<!DOCTYPE html>
<html>
<body>

<h2>HTML Images</h2>
<p>HTML images are defined with the img tag:</p>

<img src="w3schools.jpg" alt="W3Schools.com" width="104" height="142">

</body>
</html>
```

HTML Images

HTML images are defined with the img tag:

30

HTML <area> alt Attribute

‹ HTML <area> tag

```
<!DOCTYPE html>
<html>
<body>

<h1>The area alt attribute</h1>

<p>Click on the sun or on one of the planets to watch it closer:</p>

<img src="planets.gif" width="145" height="126" alt="Planets"
usemap="#planetmap">

<map name="planetmap">
  <area shape="rect" coords="0,0,82,126" alt="Sun" href="sun.htm">
  <area shape="circle" coords="90,58,3" alt="Mercury" href="mercur.htm">
  <area shape="circle" coords="124,58,8" alt="Venus" href="venus.htm">
</map>

</body>
</html>
```

HTML <area> alt Attribute

- **Definition and Usage**

- The alt attribute provides alternative information for an image if a user for some reason cannot view it (because of slow connection, an error in the *src* attribute, or if the user uses a screen reader).

- The *alt* attribute is required if the *href* attribute is present.

The area alt attribute

Click on the sun or on one of the planets to watch it closer:
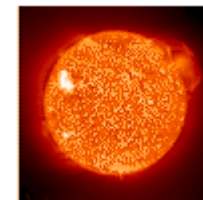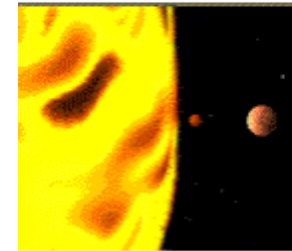
How to View HTML Source

- **View HTML Source Code**:
  - Right-click in an HTML page and select "View Page Source" (in Chrome) or "View Source" (in Edge), or similar in other browsers. This will open a window containing the HTML source code of the page.

- **Inspect an HTML Element**:
  - Right-click on an element (or a blank area), and choose "Inspect" or "Inspect Element" to see what elements are made up of (you will see both the HTML and the CSS). You can also edit the HTML or CSS on-the-fly in the Elements or Styles panel that opens.

HTML Elements

The HTML **element** is everything from the start tag to the end tag:

<tagname>Content goes here...</tagname>

Examples of some HTML elements:

<h1>My First Heading</h1>

Start tag	Element content	End tag
<h1>	My First Heading	</h1>
<p>	My first paragraph.	</p>
 	none	*none*

<p>My first paragraph.</p>

-
 element: line break in text

These elements are called **empty elements**. Empty elements **do not have** an **end tag**!

HTML Elements

Opening tag

Content
(may be text and/or other HTML elements)

Closing tag
(starts with a /)

`<element name>` Content here `</element name>`

Element

`<h1>`Black Goose Bistro`</h1>`
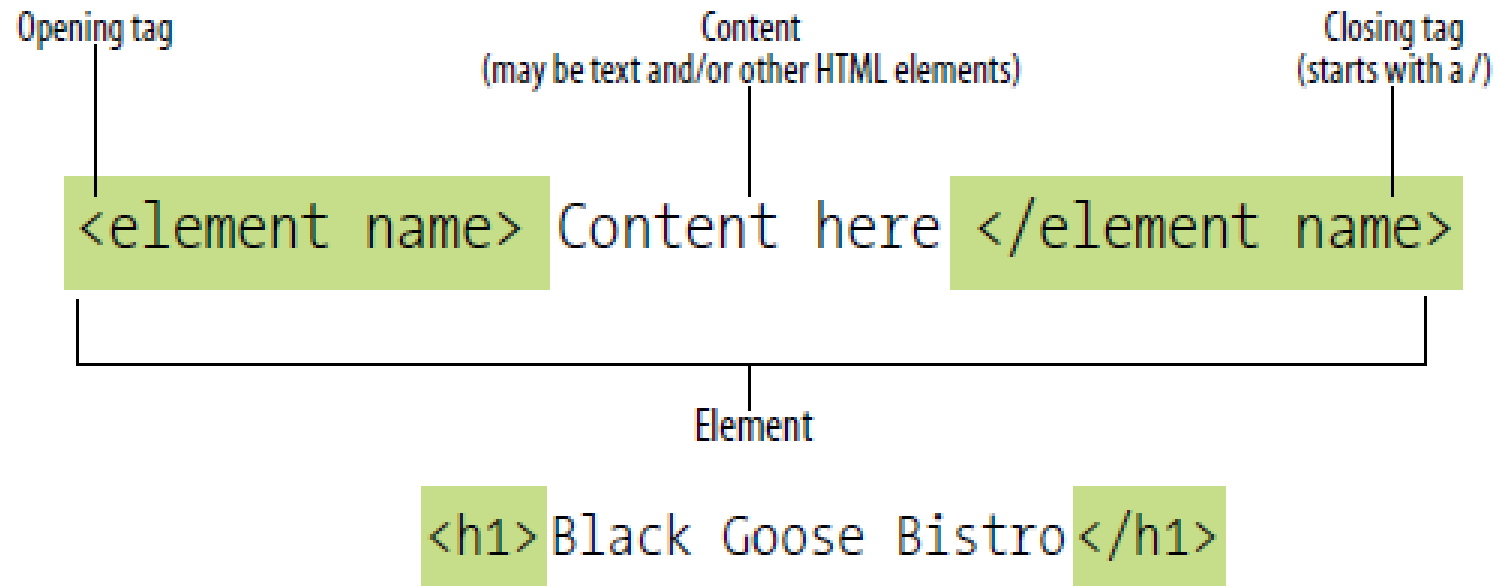
Nested HTML Elements

HTML elements can be nested (this means that elements can contain other elements).

All HTML documents consist of nested HTML elements.

The following example contains four HTML elements (`<html>`, `<body>`, `<h1>` and `<p>`):

Example

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

Example Explained

The `<html>` element is the root element and it defines the whole HTML document.

It has a start tag `<html>` and an end tag `</html>`.

Then, inside the `<html>` element there is a `<body>` element:

```
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
```

The `<body>` element defines the document's body.

It has a start tag `<body>` and an end tag `</body>`.

36

Example

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

Example Explained

Then, inside the `<body>` element there are two other elements: `<h1>` and `<p>` :t.

```
<h1>My First Heading</h1>
<p>My first paragraph.</p>
```

The `<h1>` element defines a heading.

It has a start tag `<h1>` and an end tag `</h1>` :

```
<h1>My First Heading</h1>
```

The `<p>` element defines a paragraph.

It has a start tag `<p>` and an end tag `</p>` :

```
<p>My first paragraph.</p>
```

Never Skip the End Tag

Some HTML elements will display correctly, even if you forget the end tag:

Never rely on this!
Unexpected results and errors may occur if you forget the end tag!

HTML is Not Case Sensitive

HTML tags are not case sensitive: `<P>` means the same as `<p>`.

The HTML standard does not require lowercase tags, but normally it recommends lowercase in HTML, and demands lowercase for stricter document types like XHTML.

Case-Sensitive Search

C ≠ c

Case sensitive means that it matters if characters are in lower or uppercase, while case insensitive applications do not care about that.

HTML Attributes

HTML Attributes

- All HTML elements can have **attributes**
- Attributes provide **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

The href Attribute

The `<a>` tag defines a hyperlink. The `href` attribute specifies the URL of the page the link goes to:

Example

```
<!DOCTYPE html>
<html>
<body>

<h2>The href Attribute</h2>

<p>HTML links are defined with the a tag. The link address is specified in the href attribute:</p>

<a href="https://www.w3schools.com">Visit W3Schools</a>

</body>
</html>
```

The href Attribute

HTML links are defined with the a tag. The link address is specified in the href attribute:

Visit W3Schools

40

The src Attribute

- The <img> tag is used to embed an image in an HTML page.
- The src attribute specifies the path to the image to be displayed:

```
<!DOCTYPE html>
<html>
<body>

<h2>The src Attribute</h2>
<p>HTML images are defined with the img tag, and the filename of the image source is speci
attribute:</p>

<img src="img_girl.jpg" width="500" height="600">

</body>
</html>
```


The src Attribute

HTML images are defined with the img tag, and the filename of the image source is specified in the src attribute:

The src Attribute

- There are two ways to specify the URL in the src attribute:

1. **Absolute URL** - Links to an external image that is hosted on another website. **Example: src="https://www.kln.ac.lk/images/img_girl.jpg"**.

- Notes: External images might be under copyright.

2. **Relative URL** - Links to an image that is hosted within the website.

Here, the URL does not include the domain name.

If the URL begins without a slash, it will be relative to the current page.

Example: src="img_girl.jpg".

If the URL begins with a slash, it will be relative to the domain.

Example: src="/images/img_girl.jpg".

The width and height Attributes

The `<img>` tag should also contain the `width` and `height` attributes, which specify the width and height of the image (in pixels):

Example

```html
<!DOCTYPE html>
<html>
<body>

<h2>Width and Height Attributes</h2>

<p>The width and height attributes of the img tag, defines the width and height of the image:</p>

<img src="img_girl.jpg" width="500" height="600">

</body>
</html>
```
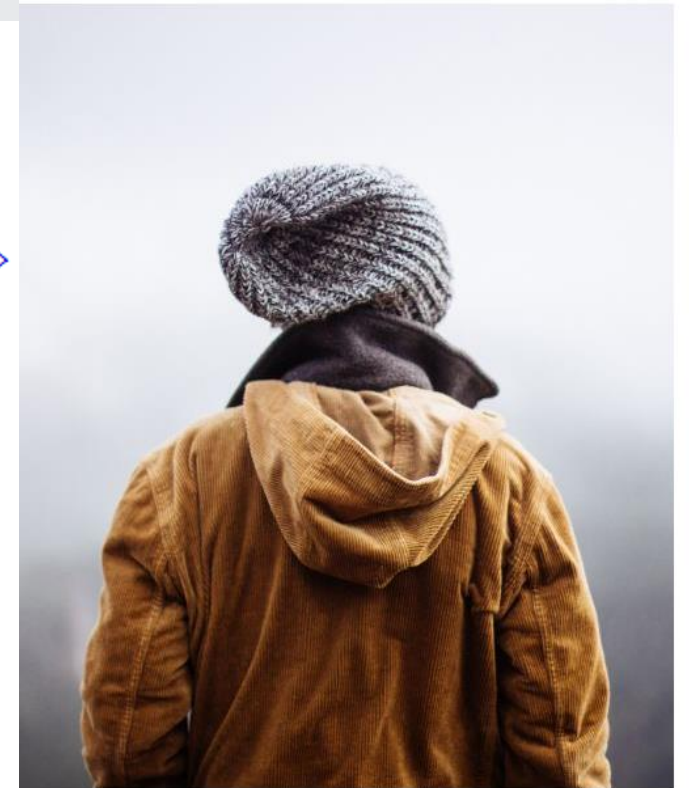
Width and Height Attributes

The width and height attributes of the img tag, defines the width and height of the image:

Let's learn more HTML tags

Next Lesson….