Internet and New Media (INM)

Chapter 02 – XHTML

Victor M. Garcia-Barrios
IICM @ TUGRAZ

Agenda

Introduction

Editing XHTML

First XHTML Example

W3C XHTML Validation Service

Headings

Linking

Images

Special Characters and Horizontal Rules

Lists

Tables

Forms

Internal Linking

`meta` Elements

Wrap-Up

Web Resources

4.1 Introduction

XHTML (Extensible HyperText Markup Language)

- markup language for creating Web pages

Based on HTML (HyperText Markup Language)

- legacy technology of the World Wide Web Consortium (W3C)

XHTML 1.0

- Allows only a document's content and structure to appear in a valid XHTML document, and not its formatting
- Formatting is specified with Cascading Style Sheets

A machine that runs a specialized piece of software called a Web server stores XHTML documents.

4.2 Editing XHTML

Good Programming Practice

Assign filenames to documents that describe their functionality. This practice can help you identify documents faster. It also helps people who want to link to a page, by giving them an easy-to-remember name. For example, if you are writing an XHTML document that contains product information, you might want to call it `products.html`.

4.3 First XHTML Example

In XHTML, text is marked up with elements delimited by tags that are names contained in pairs of angle brackets

- Every XHTML document contains a start `<html>` tag and an end `</html>` tag

Some elements may contain attributes that provide additional information about the element

Comments in XHTML always begin with `<!--` and end with `-->`. The browser ignores all text inside a comment

4.3 First XHTML Example (Cont.)

Every XHTML document contains a `head` element which generally contains:

- A `title`
- A `body` element

`head` element

- generally is not rendered in the display window

4.3 First XHTML Example (Cont.)

The `title` element:

- Names a Web page
- Usually appears in the colored bar (called the title bar) at the top of the browser window
- Is the text identifying a page when users add your page to their list of Favorites or Bookmarks

The `body` element:

- Contains the document’s content, which may include text and tags

4.3 First XHTML Example (Cont.)

All text placed between the <p> and </p> tags forms one paragraph

4.3 First XHTML Example (Cont.)

XHTML documents delimit an element with start and end tags

- A start tag consists of the element name in angle brackets (e.g., `<html>`)
- An end tag consists of the element name preceded by a forward slash (`/`) in angle brackets (e.g., `</html>`)

Many start tags have attributes that provide additional information about an element

- Each attribute has a name and a value separated by an equals sign (`=`)

Fig. 4.1 | First XHTML example.

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.1: main.html -->
<!-- First XHTML example. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Welcome</title>
   </head>

   <body>
      <p>Welcome to XHTML!</p>
   </body>
</html>
```

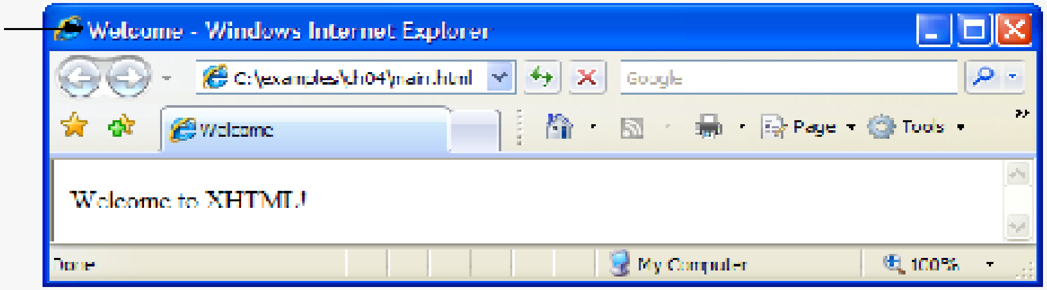
XHTML comments, not interpreted by the browser

Creates a head element

Creates a title element, which contains the text Welcome

Creates a p element within the body, which displays welcome text

Title bar shows contents of title element



Welcome - Windows Internet Explorer

C:\examples\ch04\main.html

Welcome

Welcome to XHTML!

Good Programming Practice 4.2

Place comments throughout your markup. Comments help other programmers understand the markup, assist in debugging and list useful information that you do not want the browser to render. Comments also help you understand your own markup when you revisit a document to modify or update it in the future.

Common Programming Error 4.1

Not enclosing attribute values in either single or double quotes is a syntax error. However, some Web browsers may still render the element correctly.

Common Programming Error 4.2

Using uppercase letters in an XHTML element or attribute name is a syntax error. However, some Web browsers may still render the element correctly.

Good Programming Practice 4.3

Indenting nested elements emphasizes a document’s structure and promotes readability.

Common Programming Error 4.3

XHTML does not permit tags to overlap—a nested element’s end tag must appear in the document before the enclosing element’s end tag.

For example, the “nested” (overlapping!) XHTML tags

`<head><title>hello</head></title>`

cause a syntax error, because the enclosing `head` element’s ending `</head>` tag appears before the nested title element’s ending `</title>` tag.

Good Programming Practice 4.4

Use a consistent `title`-naming convention for all pages on a site.

For example, if a site is named “Bailey’s Website,” then the `title` of the contact page might be “Bailey’s Website - Contact.” This practice can help users better understand the Website’s structure.

4.4 W3C XHTML Validation Service

XHTML documents that are syntactically correct are guaranteed to render properly

XHTML documents that contain syntax errors may not display properly

Validation services (e.g., `validator.w3.org`) ensure that an XHTML document is syntactically correct

Error-Prevention Tip 4.1

Most current browsers attempt to render XHTML documents even if they are invalid. This often leads to unexpected and possibly undesirable results. Use a validation service, such as the W3C MarkUp Validation Service, to confirm that an XHTML document is syntactically correct.

4.5 Headings

XHTML provides six headings (h1 through h6) for specifying the relative importance of information

- Heading element h1 is considered the most significant heading and is rendered in the largest font
- Each successive heading element (i.e., h2, h3, etc.) is rendered in a progressively smaller font

Portability Tip 4.1

The text size used to display each heading element can vary significantly between browsers. (In Chapter 5, we will discuss how to control the text size and other text properties.)

Look-and-Feel Observation 4.1

Placing a heading at the top of every XHTML page helps viewers understand the purpose of each page.

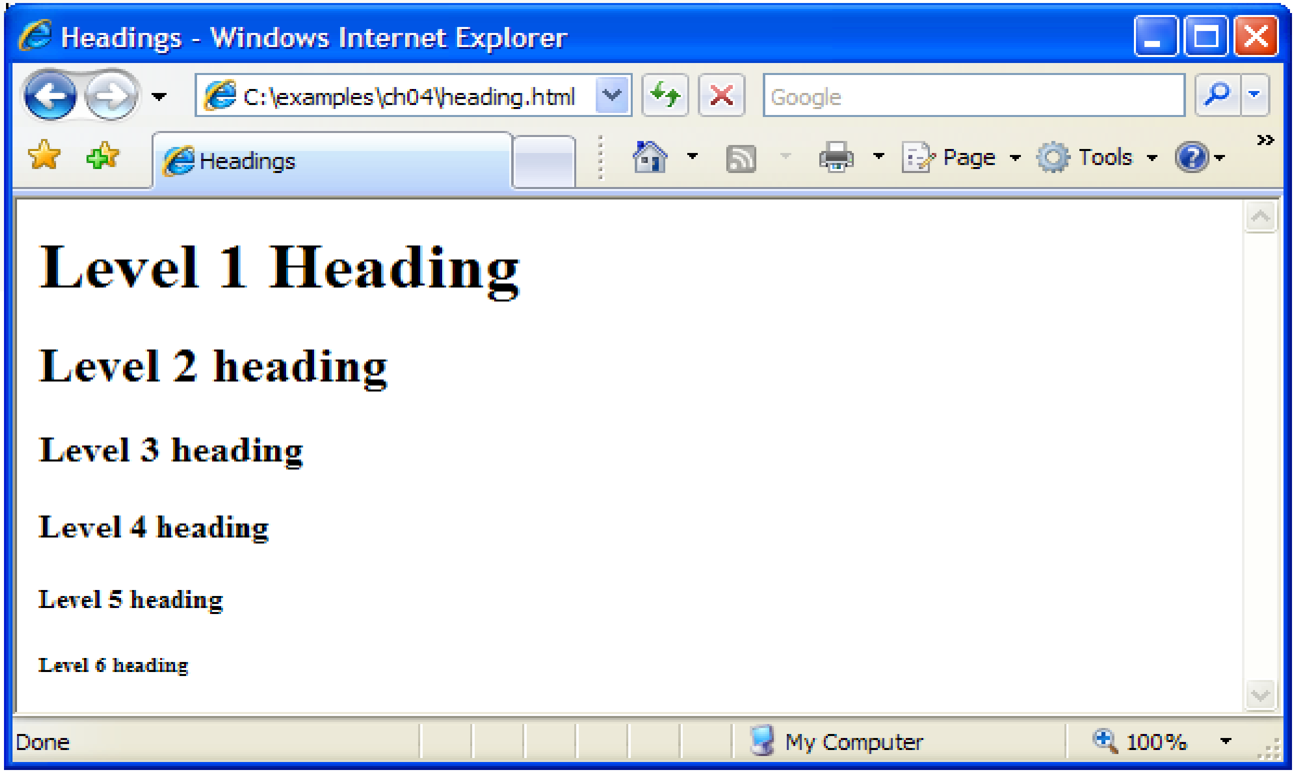
Look-and-Feel Observation 4.2

Use larger headings to emphasize more important sections of a Web page.

Fig. 4.2 | Heading elements `h1` through `h6`.

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.2: heading.html -->
<!-- Heading elements h1 through h6. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Headings</title>
   </head>

   <body>
      <h1>Level 1 Heading</h1>
      <h2>Level 2 heading</h2>
      <h3>Level 3 heading</h3>
      <h4>Level 4 heading</h4>
      <h5>Level 5 heading</h5>
      <h6>Level 6 heading</h6>
   </body>
</html>
```

Creates six headings, each with decreasing significance


Headings - Windows Internet Explorer
C:\examples\ch04\heading.html
Google
Headings
Page
Tools
Done
My Computer
100%

Level 1 Heading
Level 2 heading
Level 3 heading
Level 4 heading
Level 5 heading
Level 6 heading

4.6 Linking

A hyperlink references or links to other resources, such as XHTML documents and images

Web browsers typically underline text hyperlinks and color them blue by default

4.6 Linking (Cont.)

Users can insert links with the a (anchor) element.

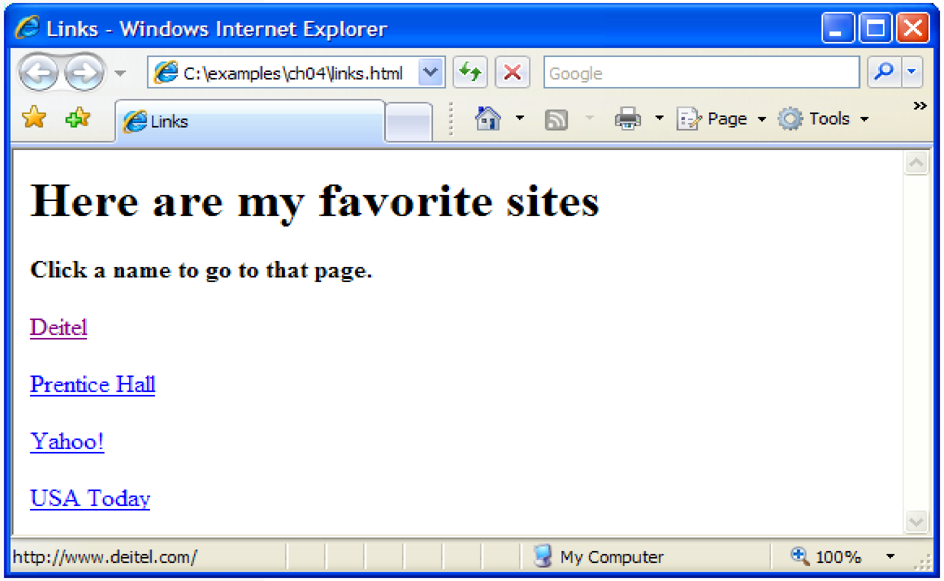
- The *href* attribute specifies the resource (e.g., page, file, e-mail address) being linked
- Anchors can link to an e-mail address using a *mailto:* URL
  - When a user clicks this type of anchored link, most browsers launch the default e-mail program (e.g., Outlook Express) to initiate an e-mail message addressed to the linked address

4.6 Linking (Cont.)

The `strong` element typically causes the browser to render text in a bold font

4.3 | Linking to other Web pages.

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.3: links.html -->
<!-- Linking to other web pages. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Links</title>
   </head>

   <body>
      <h1>Here are my favorite sites</h1>
      <p><strong>Click a name to go to that page.</strong></p>

      <!-- Create four text hyperlinks -->
      <p><a href = "http://www.deitel.com">Deitel</a></p>
      <p><a href = "http://www.prenhall.com">Prentice Hall</a></p>
      <p><a href = "http://www.yahoo.com">Yahoo!</a></p>
      <p><a href = "http://www.usatoday.com"> Today</a></p>
   </body>
</html>
```


Links - Windows Internet Explorer
C:\examples\ch04\links.html
Google
Links
Page
Tools
http://www.deitel.com/
My Computer
100%

Here are my favorite sites

Click a name to go to that page.

Deitel

Prentice Hall

Yahoo!

USA Today

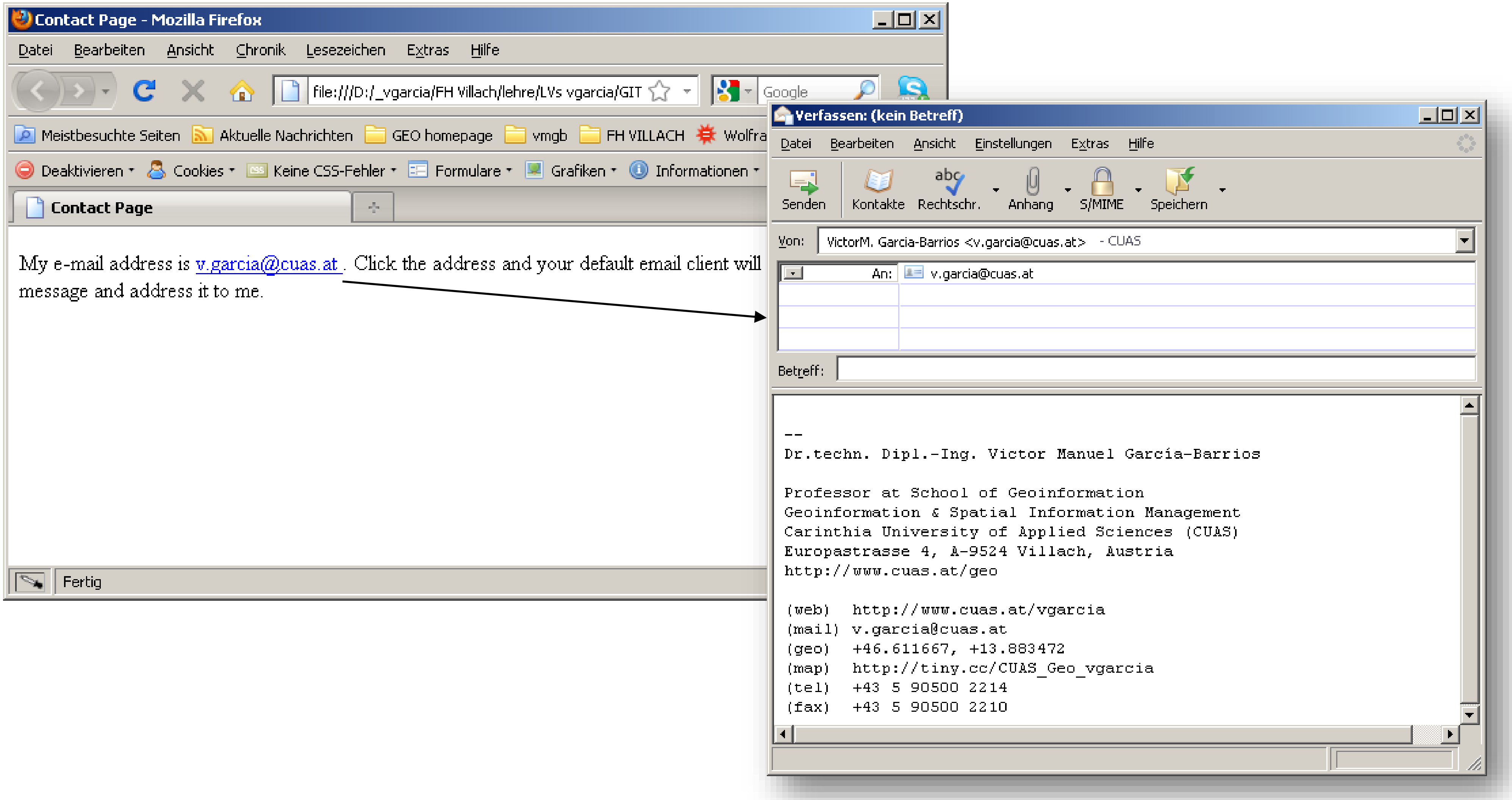
Creates anchor elements that link to the URL specified in the `href` attribute

Fig. 4.4 | Linking to an e-mail address (Part 1 of 2).

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.4: contact.html -->
<!-- Linking to an e-mail address. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Contact Page</title>
   </head>

   <body>
      <p>
         My email address is
         <a href = "mailto:">

         </a>
         . Click the address and your default email client
          will open an e-mail message and address it to me.
      </p>
   </body>
</html>
```

Hyperlink that creates a message to the address `vgarcia@iicm.edu` with the computer’s default e-mail program

Fig. 4.4 | Linking to an e-mail address (Part 2 of 2).


Contact Page - Mozilla Firefox
Datei Bearbeiten Ansicht Chronik Lesezeichen Extras Hilfe
file:///D:/_vgarcia/FH Villach/lehre/LVs vgarcia/GIT
Google
Meistbesuchte Seiten Aktuelle Nachrichten GEO homepage vmgb FH VILLACH Wolfra
Deaktivieren Cookies Keine CSS-Fehler Formulare Grafiken Informationen
Contact Page
Fertig
Verfassen: (kein Betreff)
Datei Bearbeiten Ansicht Einstellungen Extras Hilfe
Senden Kontakte Rechtschr. Anhang S/MIME Speichern

My e-mail address is v.garcia@cuas.at . Click the address and your default email client will message and address it to me.

Von: VictorM. Garcia-Barrios <v.garcia@cuas.at> - CUAS

An: v.garcia@cuas.at

Betreff:

```
--
Dr.techn. Dipl.-Ing. Victor Manuel García-Barrios

Professor at School of Geoinformation
Geoinformation & Spatial Information Management
Carinthia University of Applied Sciences (CUAS)
Europastrasse 4, A-9524 Villach, Austria
http://www.cuas.at/geo

(web)  http://www.cuas.at/vgarcia
(mail) v.garcia@cuas.at
(geo)  +46.611667, +13.883472
(map)  http://tiny.cc/CUAS_Geo_vgarcia
(tel)  +43 5 90500 2214
(fax)  +43 5 90500 2210
```

4.7 Images

The `img` element’s `src` attribute specifies an image’s location

Every `img` element must have an `alt` attribute, which contains text that is displayed if the client cannot render the image

- The `alt` attribute makes Web pages more accessible to users with disabilities, especially vision impairments
- `width` and `height` are optional attributes
  - If omitted, the browser uses the image’s actual width and height
  - Images are measured in pixels

Good Programming Practice 4.5

Always include the *width* and the *height* of an image inside the *<img>* tag.

When the browser loads the XHTML file, it will know immediately from these attributes how much screen space to provide for the image and will lay out the page properly, even before it downloads the image.

Performance Tip 4.1

Including the *width* and *height* attributes in an *<img>* tag can result in the browser’ s loading and rendering pages faster.

Common Programming Error 4.4

Entering new dimensions for an image that change its inherent width-to-height ratio distorts the appearance of the image!

For example, if your image is 200 pixels wide and 100 pixels high, you should ensure that any new dimensions have a 2:1 width-to-height ratio.

4.7 Images (Cont.)

Some XHTML elements are empty elements that contain only attributes and do not mark up text

Empty elements (e.g., img) must be terminated, either by using the forward slash character (/) inside the closing right angle bracket or by explicitly writing an end tag

4.5 | Images in XHTML files.

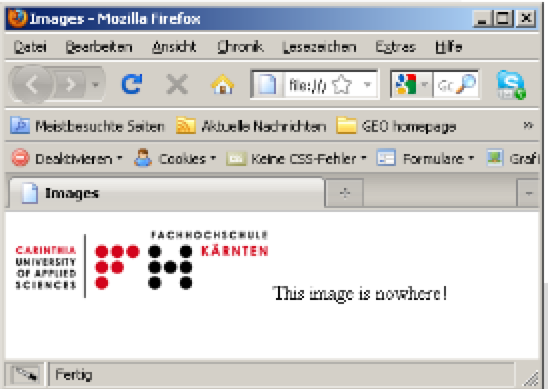
```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.5: picture.html -->
<!-- Images in XHTML files. -->
<html xmlns = "http://www.w3.org/1999/xht
   <head>
      <title>Images</title>
   </head>

   <body>
      <p>
         <img src = "cpphtp6.jpg" width = "92" height = "120"
          alt = "C++ How to Program book cover" />
         <img src = "jhtp.jpg" width = "92" height = "120"
            alt = "Java How to Program book cover" />
      </p>
   </body>
</html>
```

Specifies the image file's location

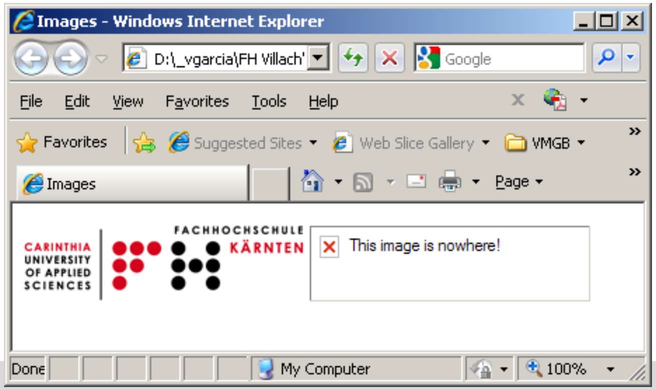
Specifies the image's width

Specifies the image's height

Specifies text to display if the image is unavailable



This image is nowhere!

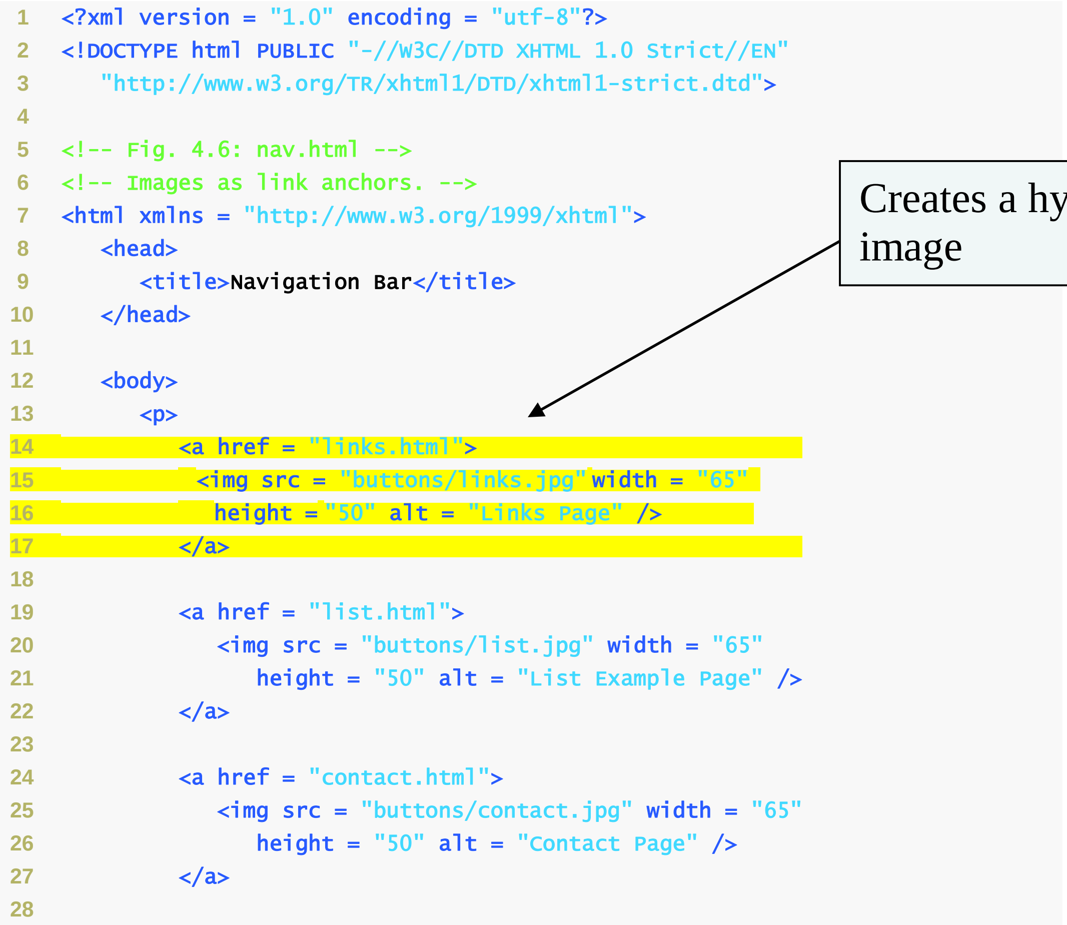


This image is nowhere!

4.7 Images (Cont.)

Create an image hyperlink by nesting an `img` element in an anchor element

Fig. 4.6 | Images as link anchors (Part 1 of 2).

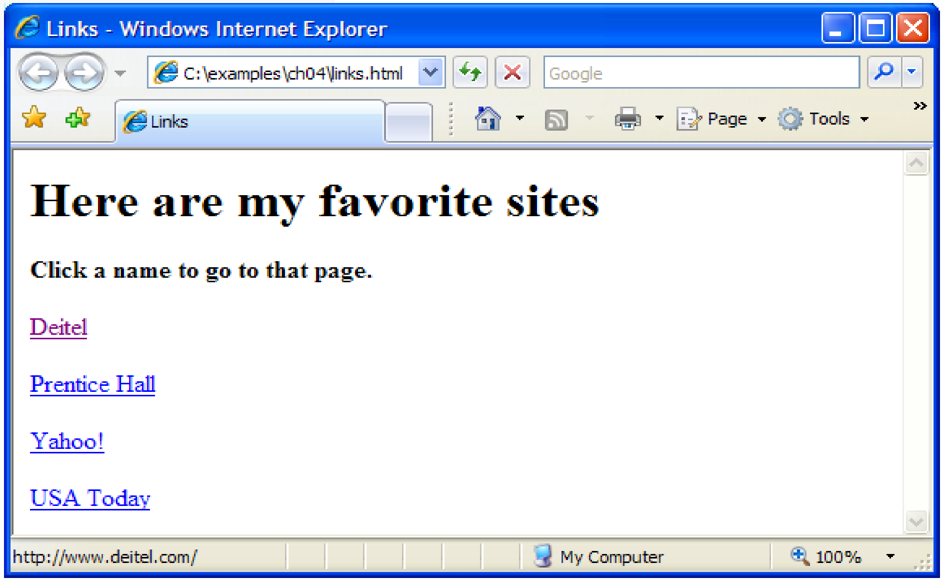


```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.6: nav.html -->
<!-- Images as link anchors. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Navigation Bar</title>
   </head>

   <body>
      <p>
         <a href = "links.html">
          <img src = "buttons/links.jpg"width = "65"
            height ="50" alt = "Links Page" />
         </a>

         <a href = "list.html">
            <img src = "buttons/list.jpg" width = "65"
               height = "50" alt = "List Example Page" />
         </a>

         <a href = "contact.html">
            <img src = "buttons/contact.jpg" width = "65"
               height = "50" alt = "Contact Page" />
         </a>

```

Creates a hyperlinked image

Fig. 4.6 | Images as link anchors (Part 1 of 2).

```
         <a href = "table1.html">
            <img src = "buttons/table.jpg" width = "65"
               height = "50" alt = "Table Page" />
         </a>

         <a href = "form.html">
            <img src = "buttons/form.jpg" width = "65"
               height = "50" alt = "Feedback Form" />
         </a>
      </p>
   </body>
</html>
```



Links - Windows Internet Explorer

C:\examples\ch04\links.html

Here are my favorite sites

Click a name to go to that page.

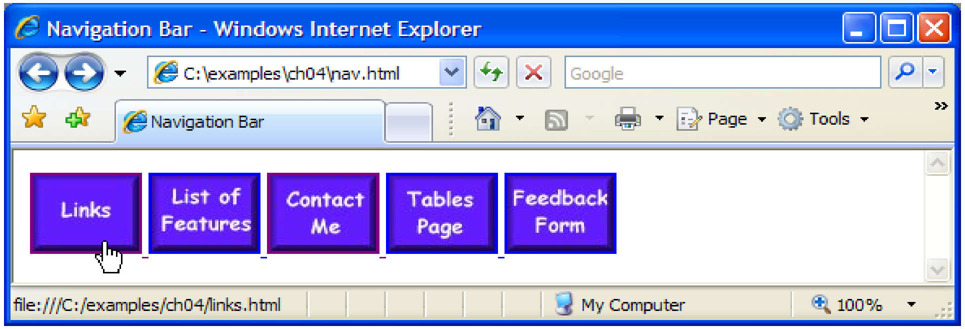
Deitel

Prentice Hall

Yahoo!

USA Today

http://www.deitel.com/



Navigation Bar - Windows Internet Explorer

C:\examples\ch04\nav.html

file:///C:/examples/ch04/links.html

4.8 Special Characters and Horizontal Rules

XHTML provides special characters or entity references (in the form &*code*`;`) for representing characters that cannot be rendered otherwise

The *code* can be:

- Word abbreviations
- Numbers (decimal or hexadecimal)

Example:

- & character represented by:
    - `&`*`amp`*`; or &#38;` (decimal) or `&#x26;` (hexadecimal)
- “Umlaute”
    - &*uuml*;

4.8 Special Characters and Horizontal Rules (Cont.)

Most browsers render a horizontal rule, indicated by the `<hr />` tag, as a horizontal line

The hr element also inserts a line break above and below the horizontal line

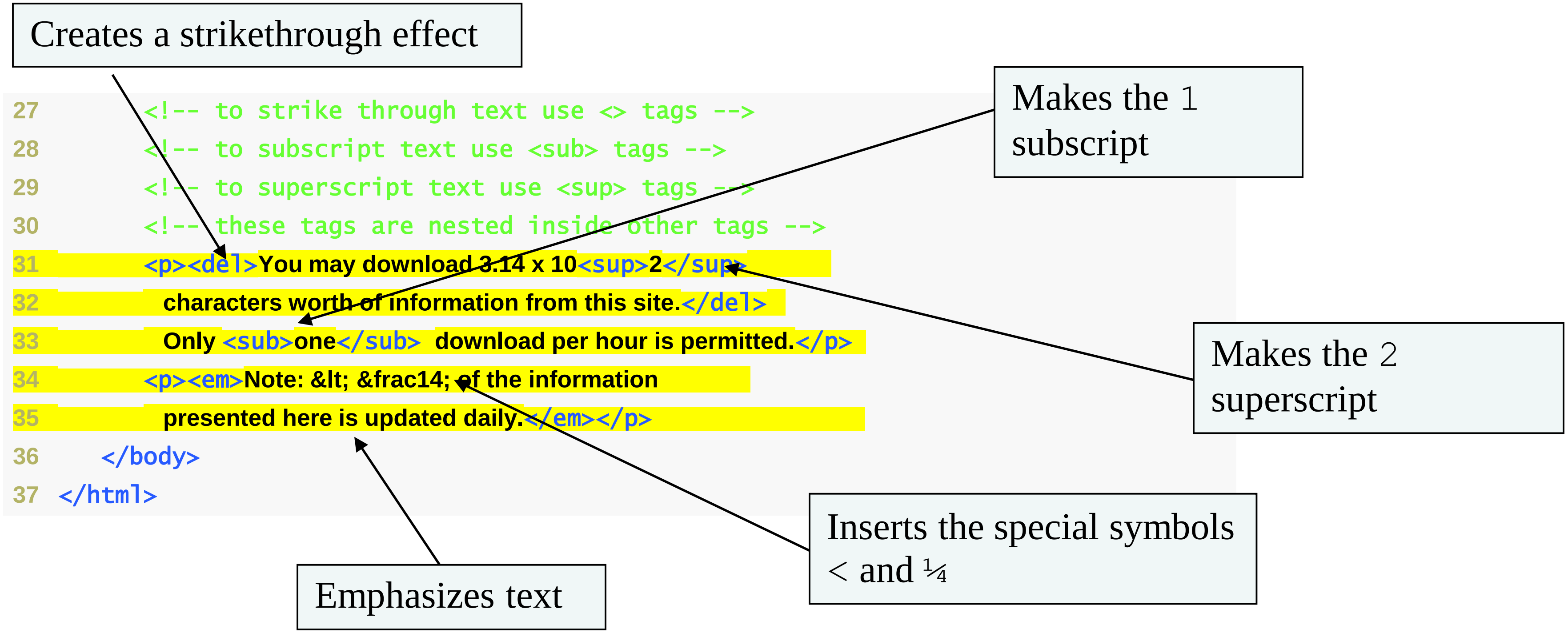
Fig. 4.7 | Inserting special characters (Part 1 of 2).

```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.7: contact2.html -->
<!-- Inserting special characters. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Contact Page</title>
   </head>

   <body>
      <p>
         Click
         <a href = "mailto:">here</a>
         to open an email message addressed to
         .
      </p>

      <hr /> <!-- inserts a horizontal rule -->

      <!-- special characters are entered -->
      <!-- using the form &code; -->
      <p>All information on this site is <strong>&copy;
       Victor &amp; Co., Inc. 1935.</strong></p>
```

Inserts a horizontal rule, with a line break before and after

Inserts the special characters © and &

Fig. 4.7 | Inserting special characters (Part 2 of 2).



Creates a strikethrough effect

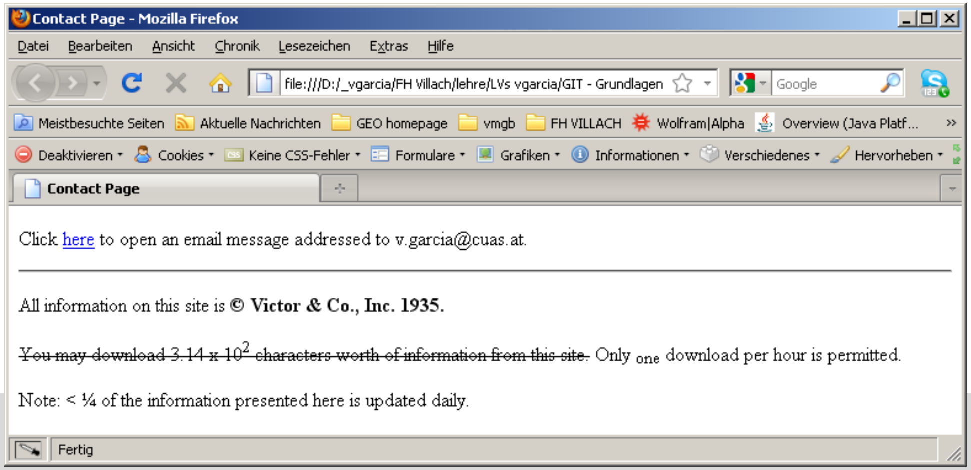
Makes the 1 subscript

Makes the 2 superscript

Inserts the special symbols < and ¼

Emphasizes text

```
      <!-- to strike through text use <> tags -->
      <!-- to subscript text use <sub> tags -->
      <!-- to superscript text use <sup> tags -->
      <!-- these tags are nested inside other tags -->
      <p><del>You may download 3.14 x 10<sup>2</sup>
       characters worth of information from this site.</del>
       Only <sub>one</sub>  download per hour is permitted.</p>
      <p><em>Note: &lt; &frac14; of the information
       presented here is updated daily.</em></p>
   </body>
</html>
```



Click here to open an email message addressed to v.garcia@cuas.at.

All information on this site is © **Victor & Co., Inc. 1935.**

~~You may download 3.14 x 10^2 characters worth of information from this site.~~ Only $_{\text{one}}$ download per hour is permitted.

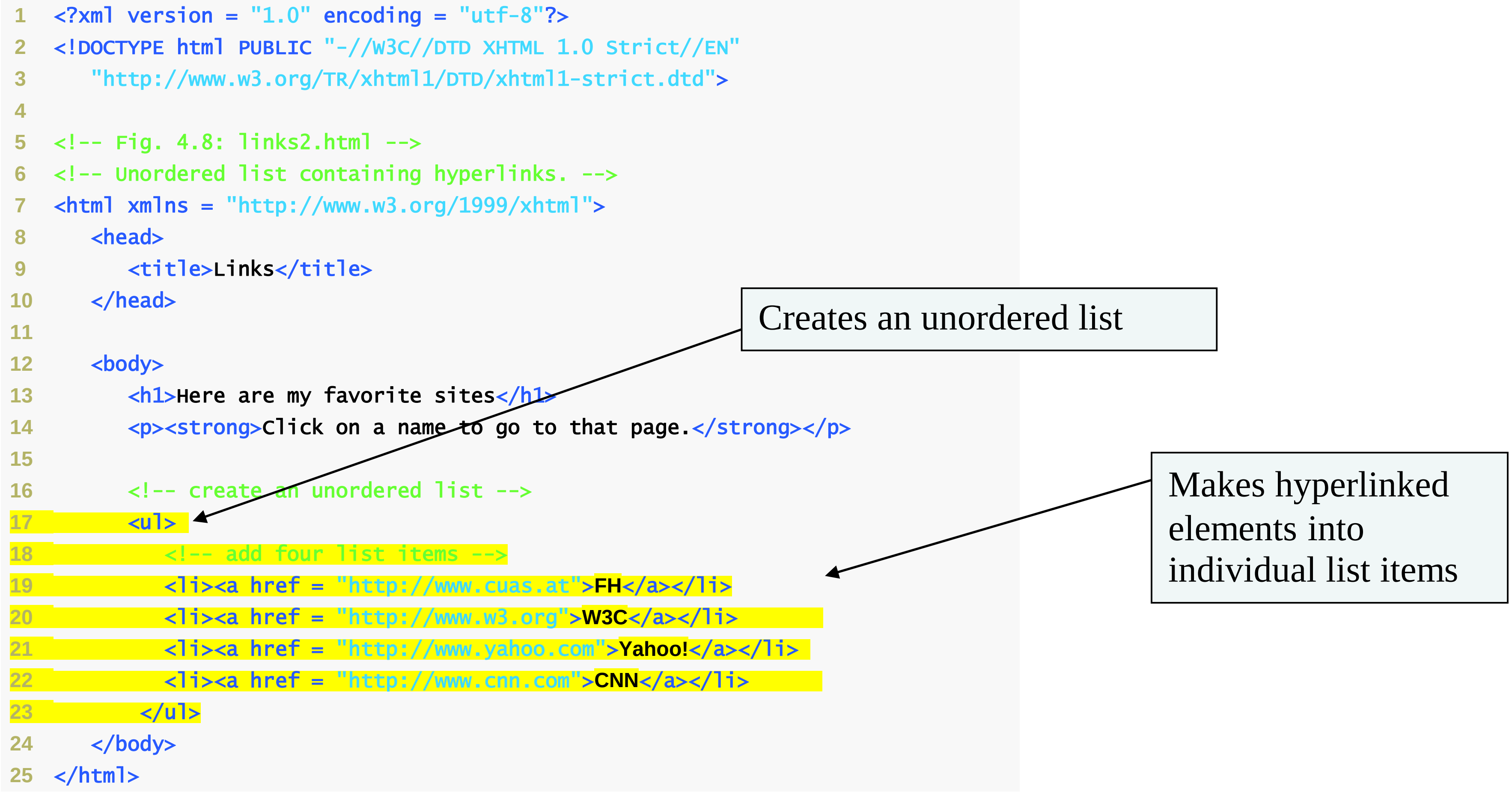
Note: < ¼ of the information presented here is updated daily.

4.9 Lists

Unordered list element `ul`

- creates a list in which each item in the list begins with a bullet symbol (called a disc)
- Each entry is an `li` (list item) element. Most Web browsers render these elements with a line break and a bullet symbol at the beginning of the line

Fig. 4.8 | Unordered list containing hyperlinks (Part 1 of 2).

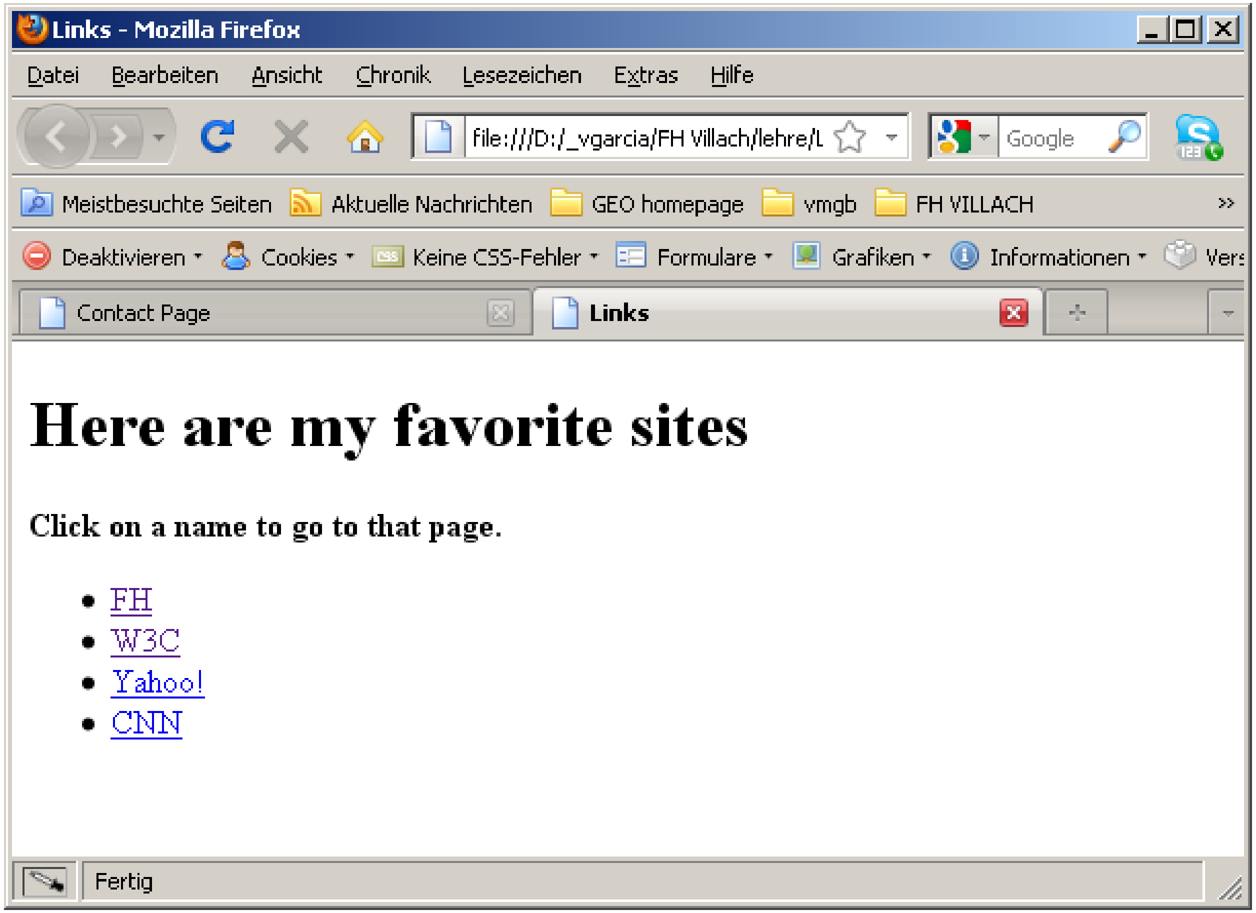


```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.8: links2.html -->
<!-- Unordered list containing hyperlinks. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Links</title>
   </head>

   <body>
      <h1>Here are my favorite sites</h1>
      <p><strong>Click on a name to go to that page.</strong></p>

      <!-- create an unordered list -->
      <ul>
         <!-- add four list items -->
         <li><a href = "http://www.cuas.at">FH</a></li>
         <li><a href = "http://www.w3.org">W3C</a></li>
         <li><a href = "http://www.yahoo.com">Yahoo!</a></li>
         <li><a href = "http://www.cnn.com">CNN</a></li>
       </ul>
   </body>
</html>
```

Creates an unordered list

Makes hyperlinked elements into individual list items

Fig. 4.8 | Unordered list containing hyperlinks (Part 2 of 2).



Here are my favorite sites

Click on a name to go to that page.

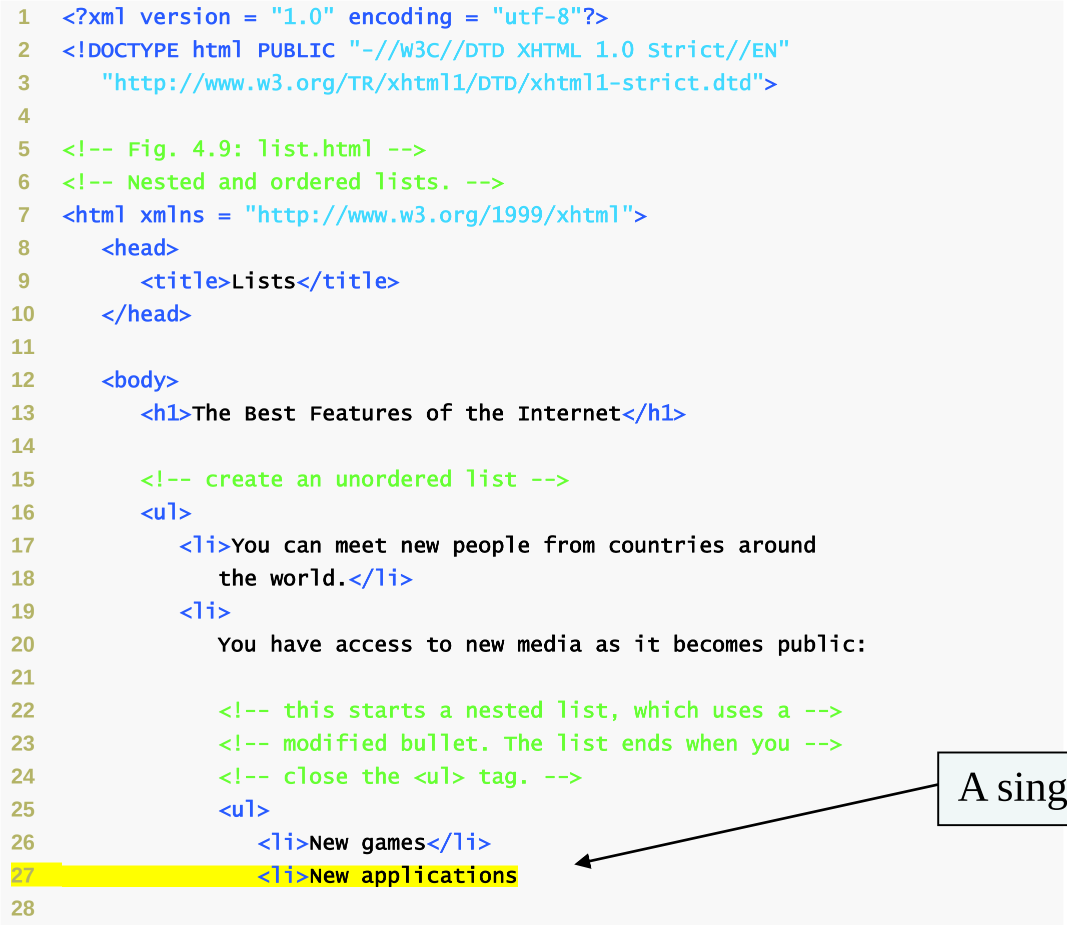
- FH
- W3C
- Yahoo!
- CNN

4.9 Lists (Cont.)

The ordered list element `ol` creates a list in which each item begins with a number

Lists may be nested to represent hierarchical data relationships

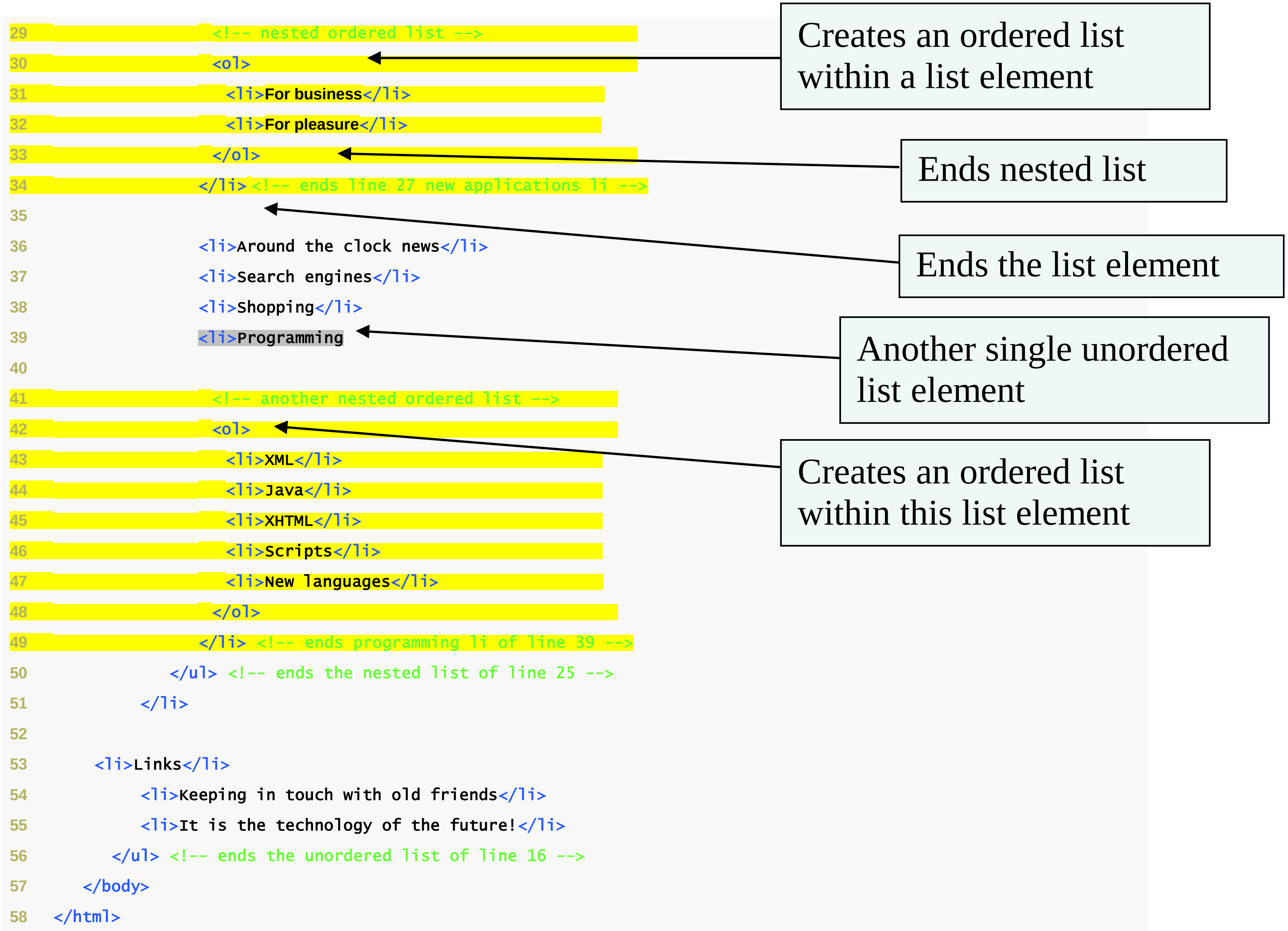
Fig. 4.9 | Nested and ordered lists (Part 1 of 3).



```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.9: list.html -->
<!-- Nested and ordered lists. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Lists</title>
   </head>

   <body>
      <h1>The Best Features of the Internet</h1>

      <!-- create an unordered list -->
      <ul>
         <li>You can meet new people from countries around
            the world.</li>
         <li>
            You have access to new media as it becomes public:

            <!-- this starts a nested list, which uses a -->
            <!-- modified bullet. The list ends when you -->
            <!-- close the <ul> tag. -->
            <ul>
               <li>New games</li>
               <li>New applications

```

A single list element

Fig. 4.9 | Nested and ordered lists (Part 2 of 3).



```
29                  <!-- nested ordered list -->
30                  <ol>
31                     <li>For business</li>
32                     <li>For pleasure</li>
33                  </ol>
34               </li> <!-- ends line 27 new applications li -->
35
36               <li>Around the clock news</li>
37               <li>Search engines</li>
38               <li>Shopping</li>
39               <li>Programming
40
41                  <!-- another nested ordered list -->
42                  <ol>
43                     <li>XML</li>
44                     <li>Java</li>
45                     <li>XHTML</li>
46                     <li>Scripts</li>
47                     <li>New languages</li>
48                  </ol>
49               </li> <!-- ends programming li of line 39 -->
50            </ul> <!-- ends the nested list of line 25 -->
51         </li>
52
53    <li>Links</li>
54         <li>Keeping in touch with old friends</li>
55         <li>It is the technology of the future!</li>
56      </ul> <!-- ends the unordered list of line 16 -->
57   </body>
58 </html>
```

Creates an ordered list within a list element

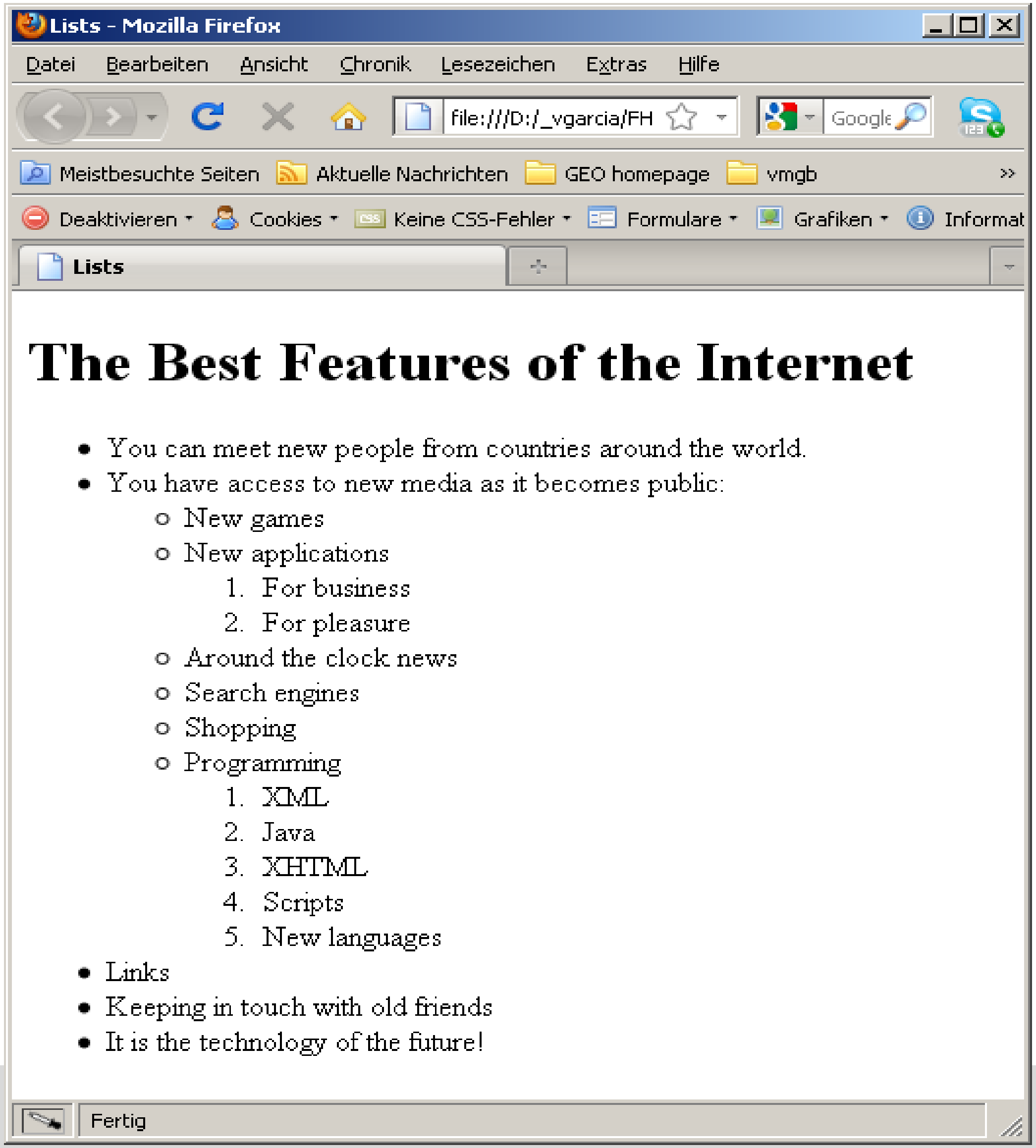
Ends nested list

Ends the list element

Another single unordered list element

Creates an ordered list within this list element

Fig. 4.9 | Nested and ordered lists (Part 3 of 3).



The Best Features of the Internet

- You can meet new people from countries around the world.
- You have access to new media as it becomes public:
  - New games
  - New applications
    1. For business
    2. For pleasure
  - Around the clock news
  - Search engines
  - Shopping
  - Programming
    1. XML
    2. Java
    3. XHTML
    4. Scripts
    5. New languages
- Links
- Keeping in touch with old friends
- It is the technology of the future!

4.10 Tables

table element

- defines an XHTML table
- Attribute *summary* summarizes the table's contents and is used by speech devices to make the table more accessible to users with visual impairments
- Element *caption* describes the table's content
    - The text inside the *<caption>* tag is rendered above the table in most browsers

4.10 Tables (Cont.)

A table can be split into three distinct sections:

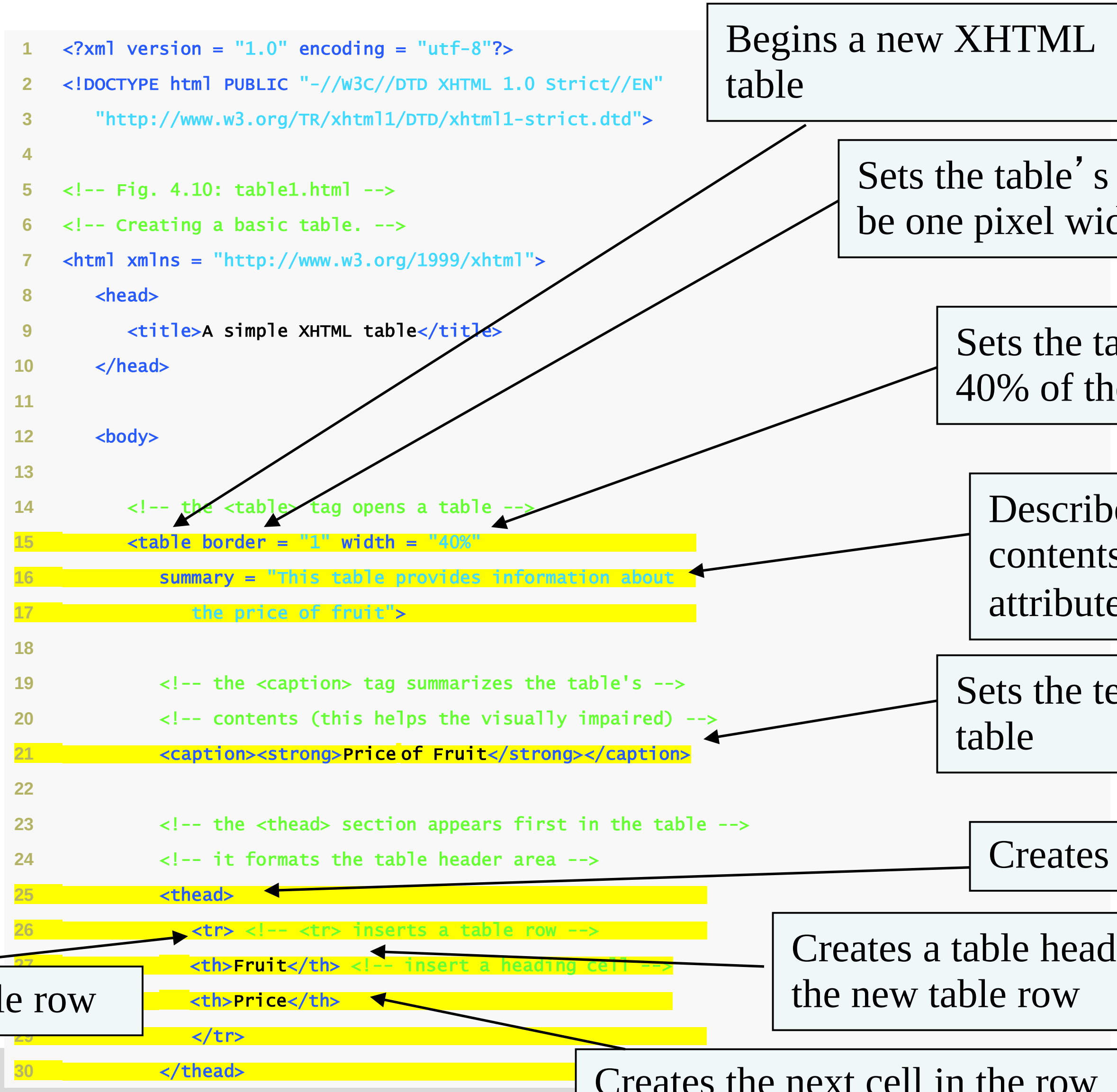
- Head (*thead* element)
  - Table titles
  - Column headers
- Body (*tbody* element)
  - Primary table data
- Foot (*tfoot* element)
  - Calculation results
  - Footnotes
  - Above body section in the code, but displays at the bottom in the page

4.10 Tables (Cont.)

Element *tr*

- Defines individual table rows
- Element *th*
  - Defines a header cell
- Element *td*
  - Contains table data elements

4.10 | Creating a basic table (Part 1 of 3).



```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.10: table1.html -->
<!-- Creating a basic table. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>A simple XHTML table</title>
   </head>

   <body>

      <!-- the <table> tag opens a table -->
      <table border = "1" width = "40%"
         summary = "This table provides information about
            the price of fruit">

         <!-- the <caption> tag summarizes the table's -->
         <!-- contents (this helps the visually impaired) -->
         <caption><strong>Price of Fruit</strong></caption>

         <!-- the <thead> section appears first in the table -->
         <!-- it formats the table header area -->
         <thead>
            <tr> <!-- <tr> inserts a table row -->
            <th>Fruit</th> <!-- insert a heading cell -->
            <th>Price</th>
            </tr>
         </thead>
```

Begins a new XHTML table

Sets the table's border to be one pixel wide

Sets the table's width to 40% of the screen

Describes the table's contents in the `summary` attribute

Sets the text above the table

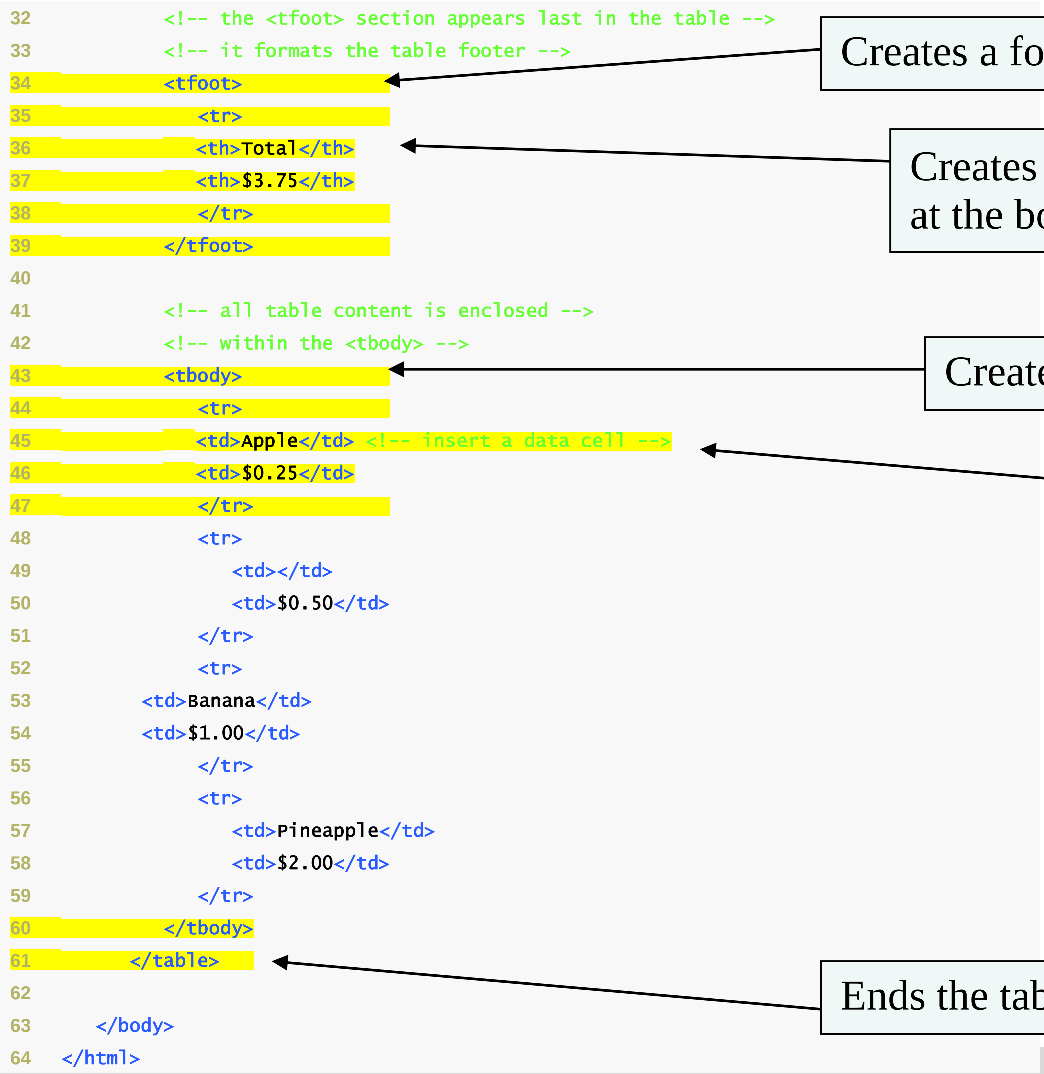
Creates a head element

Makes a new table row

Creates a table heading cell in the new table row

Creates the next cell in the row

4.10 | Creating a basic table (Part 2 of 3).



```
         <!-- the <tfoot> section appears last in the table -->
         <!-- it formats the table footer -->
         <tfoot>
            <tr>
            <th>Total</th>
            <th>$3.75</th>
            </tr>
         </tfoot>

         <!-- all table content is enclosed -->
         <!-- within the <tbody> -->
         <tbody>
            <tr>
            <td>Apple</td> <!-- insert a data cell -->
            <td>$0.25</td>
            </tr>
            <tr>
               <td></td>
               <td>$0.50</td>
            </tr>
            <tr>
       <td>Banana</td>
       <td>$1.00</td>
            </tr>
            <tr>
               <td>Pineapple</td>
               <td>$2.00</td>
            </tr>
         </tbody>
      </table>

   </body>
</html>
```

Creates a foot section

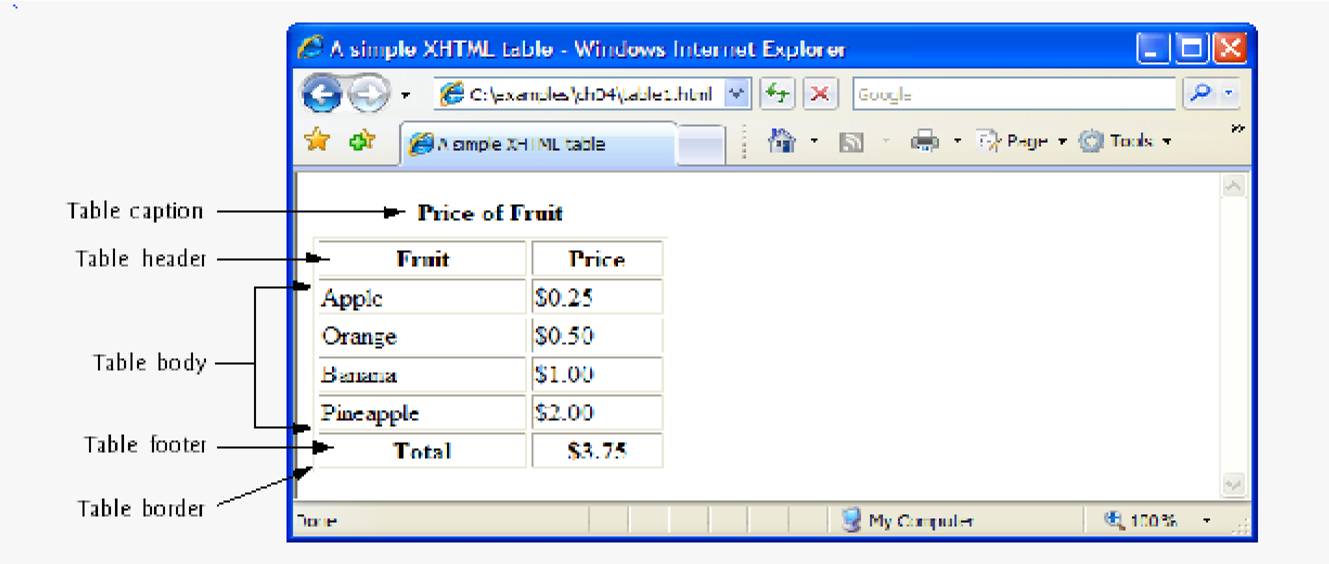
Creates table header cells at the bottom of the table

Creates a body section

Inserts a cell in the body of the table

Ends the table

4.10 | Creating a basic table (Part 3 of 3).



Price of Fruit

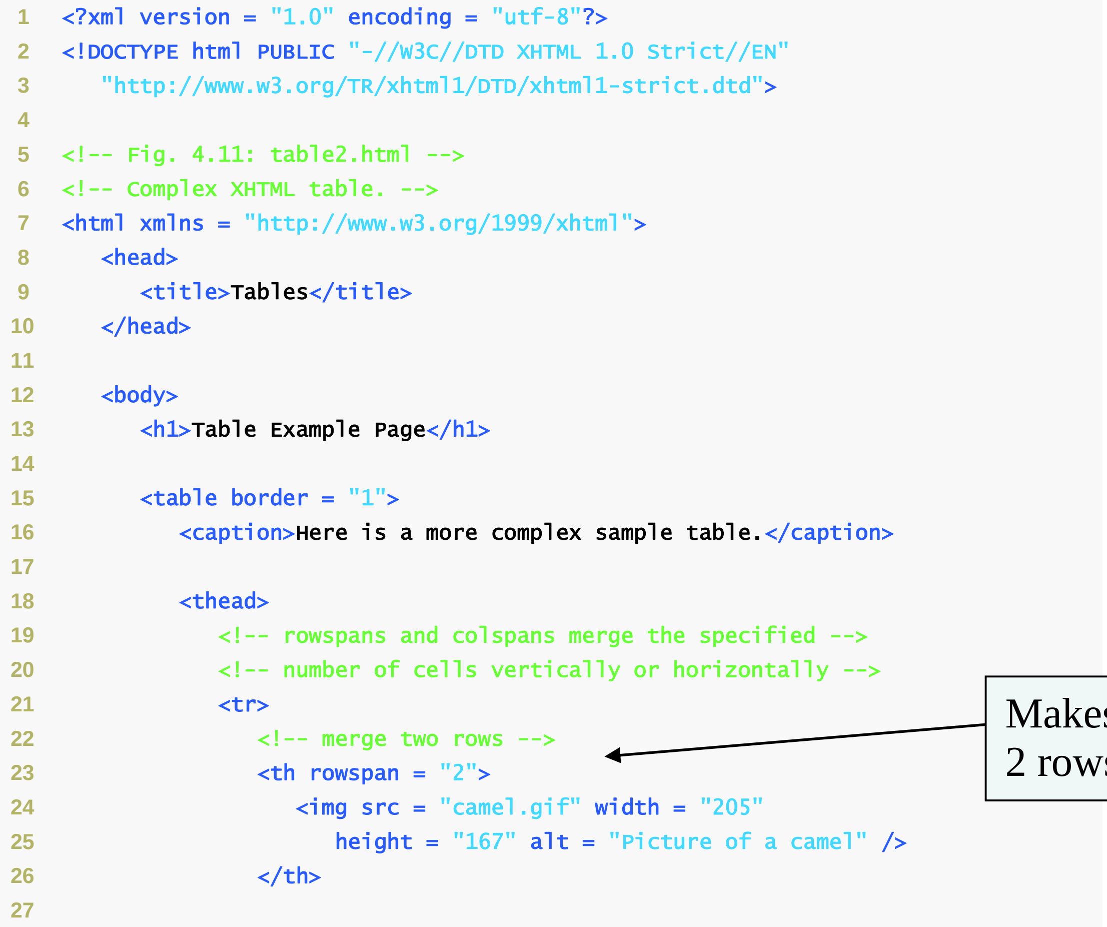
Fruit	Price
Apple	$0.25
Orange	$0.50
Banana	$1.00
Pineapple	$2.00
Total	$3.75

4.10 Tables (Cont.)

You can merge data cells with the *rowspan* and *colspan* attributes

- The values of these attributes specify the number of rows or columns occupied by the cell
- Can be placed inside any data cell or table header cell

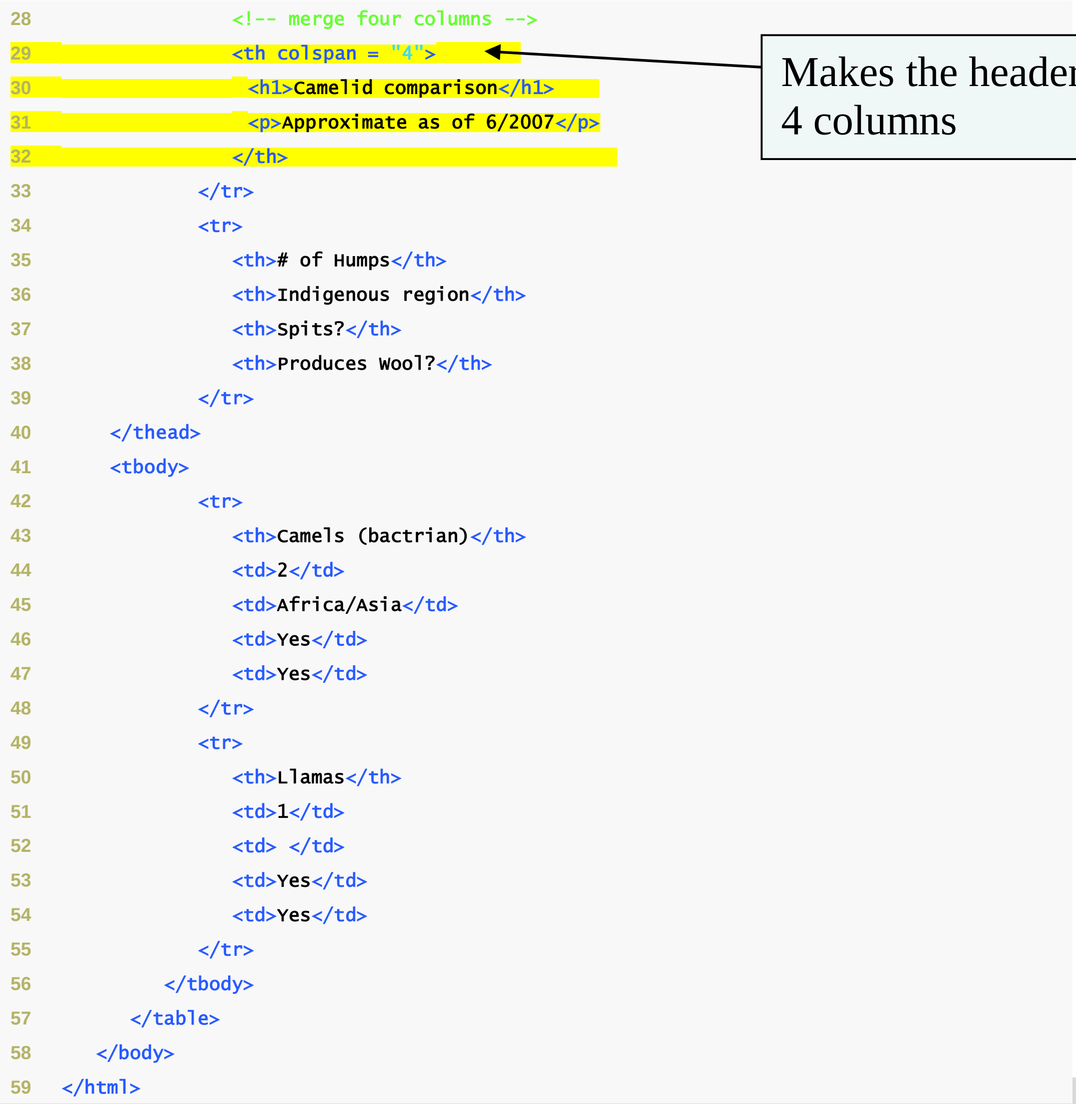
4.11 | Complex XHTML table (Part 1 of 3).



```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.11: table2.html -->
<!-- Complex XHTML table. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Tables</title>
   </head>

   <body>
      <h1>Table Example Page</h1>

      <table border = "1">
         <caption>Here is a more complex sample table.</caption>

         <thead>
            <!-- rowspans and colspans merge the specified -->
            <!-- number of cells vertically or horizontally -->
            <tr>
               <!-- merge two rows -->
               <th rowspan = "2">
                  <img src = "camel.gif" width = "205"
                     height = "167" alt = "Picture of a camel" />
               </th>

```

Makes the header cell span 2 rows

4.11 | Complex XHTML table (Part 2 of 3).



```
               <!-- merge four columns -->
               <th colspan = "4">
                 <h1>Camelid comparison</h1>
                 <p>Approximate as of 6/2007</p>
               </th>
            </tr>
            <tr>
               <th># of Humps</th>
               <th>Indigenous region</th>
               <th>Spits?</th>
               <th>Produces Wool?</th>
            </tr>
    </thead>
    <tbody>
            <tr>
               <th>Camels (bactrian)</th>
               <td>2</td>
               <td>Africa/Asia</td>
               <td>Yes</td>
               <td>Yes</td>
            </tr>
            <tr>
               <th>Llamas</th>
               <td>1</td>
               <td> </td>
               <td>Yes</td>
               <td>Yes</td>
            </tr>
         </tbody>
      </table>
   </body>
</html>
```

Makes the header cell span 4 columns

4.11 | Complex XHTML table (Part 3 of 3).

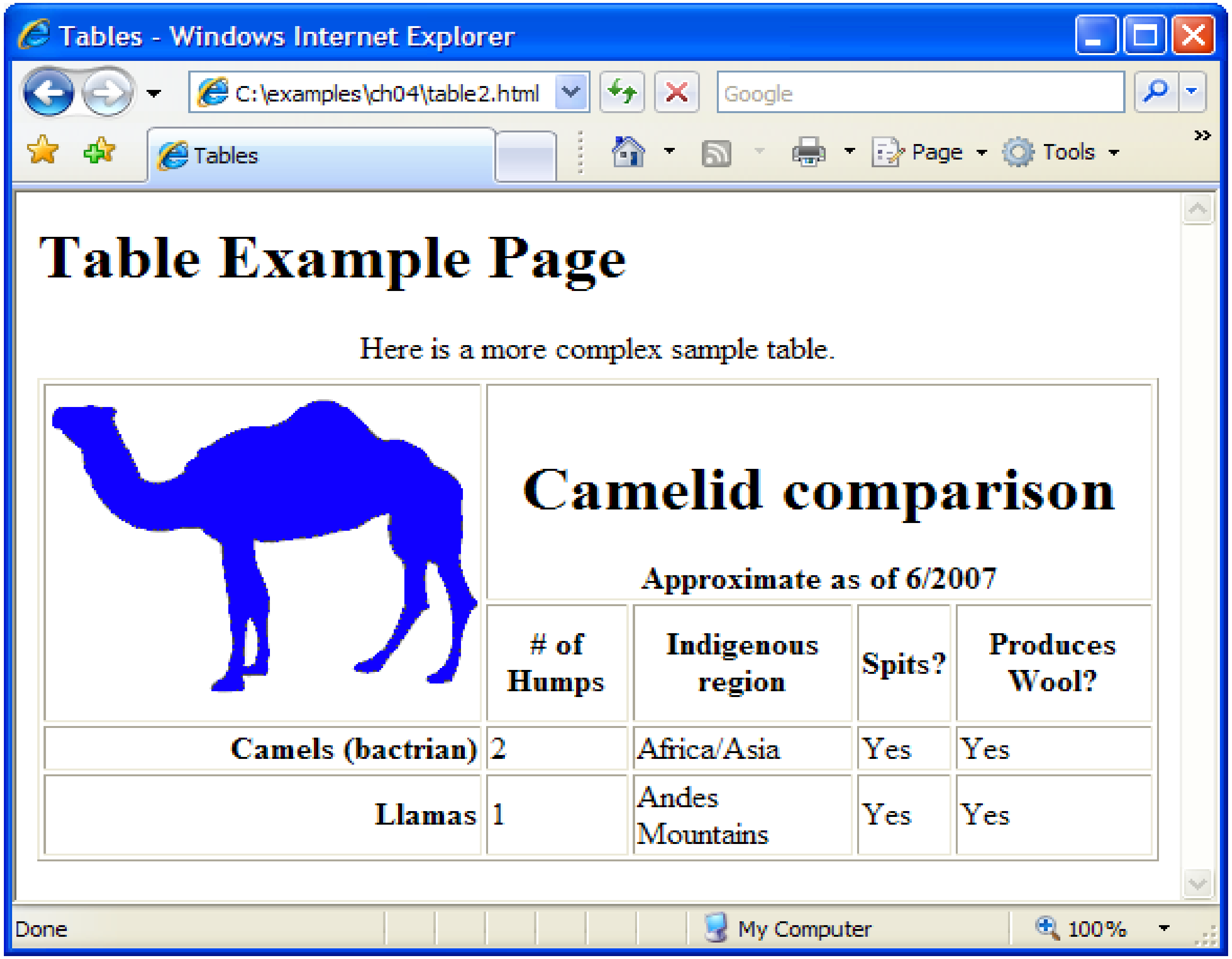

Tables - Windows Internet Explorer
C:\examples\ch04\table2.html
Google
Tables
Page
Tools
Done
My Computer
100%

Table Example Page

Here is a more complex sample table.

	Camelid comparison Approximate as of 6/2007			
	# of Humps	**Indigenous region**	**Spits?**	**Produces Wool?**
Camels (bactrian)	2	Africa/Asia	Yes	Yes
Llamas	1	Andes Mountains	Yes	Yes

4.11 Forms

XHTML provides *forms* for collecting information from users

Forms contain visual components, such as buttons, that users interact with

Forms may also contain non-visual components, called hidden inputs, which are used to store any data that needs to be sent to the server, but is not entered by the user

4.11 Forms (Cont.)

A form begins with the `form` element

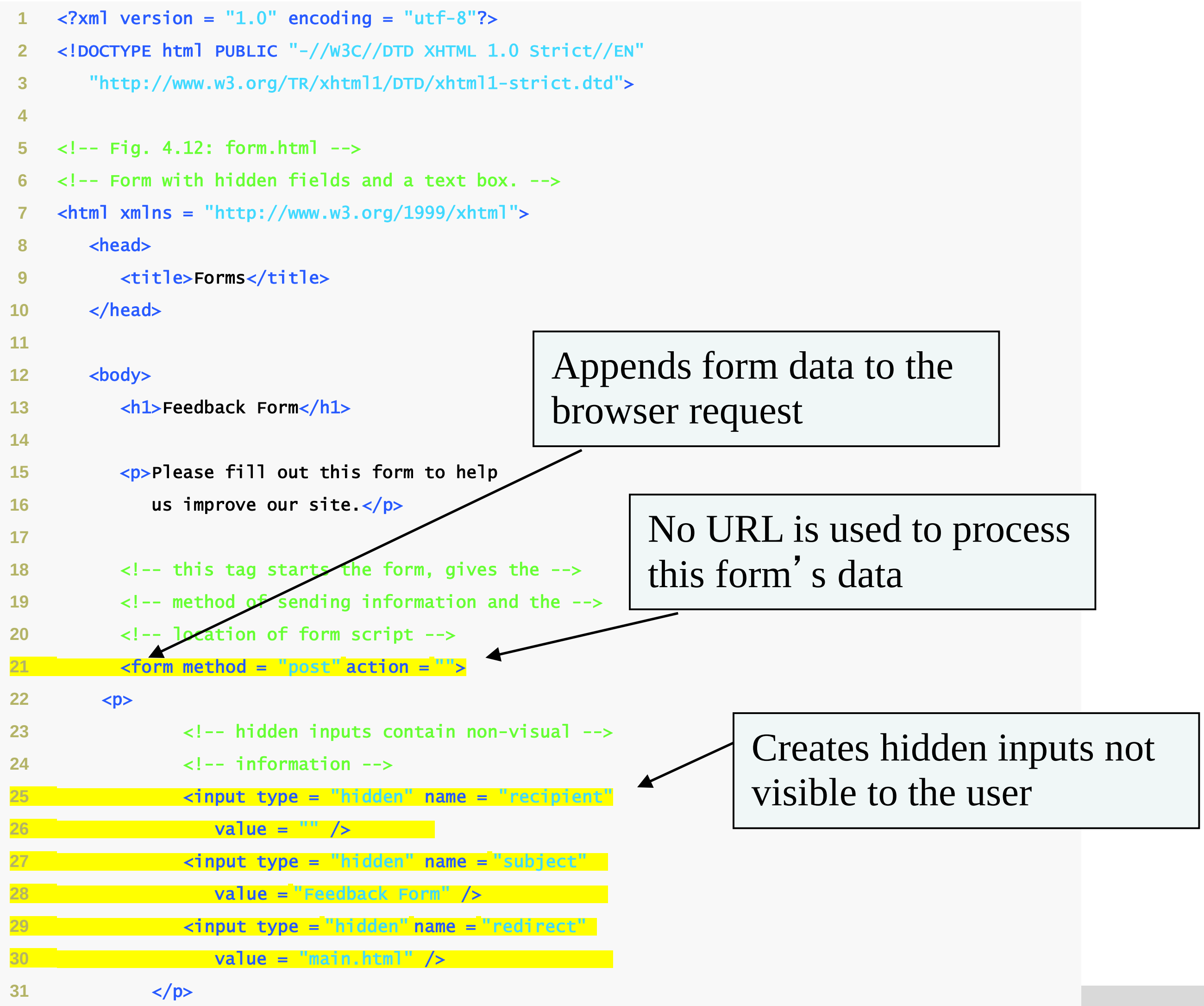
- Attribute *`method`* specifies how the form’s data is sent to the Web server
- The *`action`* attribute of the form element specifies the script to which the form data will be sent

4.11 Forms (Cont.)

The input element provides data to the script that processes the form

- The *text* input inserts a text box into the form, which allows the user to input data
- The *label* element provides users with information about the input element’s purpose
- The *size* attribute specifies the number of characters visible in the input element
- Optional attribute *maxlength* limits the number of characters input into a text box
- The *submit* input submits the data entered in the form to the Web server for processing
  - Most Web browsers create a button that submits the form data when clicked
- The *reset* input allows a user to reset all form elements to their default values

Fig. 4.12 | Form with hidden fields and a text box (Part 1 of 2).



```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.12: form.html -->
<!-- Form with hidden fields and a text box. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Forms</title>
   </head>

   <body>
      <h1>Feedback Form</h1>

      <p>Please fill out this form to help
         us improve our site.</p>

      <!-- this tag starts the form, gives the -->
      <!-- method of sending information and the -->
      <!-- location of form script -->
      <form method = "post" action = "">
    <p>
            <!-- hidden inputs contain non-visual -->
            <!-- information -->
            <input type = "hidden" name = "recipient"
               value = "" />
            <input type = "hidden" name = "subject"
               value = "Feedback Form" />
            <input type = "hidden" name = "redirect"
               value = "main.html" />
         </p>
```

Appends form data to the browser request

No URL is used to process this form’s data

Creates hidden inputs not visible to the user

Fig. 4.12 | Form with hidden fields and a text box (Part 2 of 2).

```

         <!-- <input type = "text"> inserts a text box -->
         <p><label>Name:
               <input name = "name" type = "text" size = "25"
                maxlength = "30" />
            </label></p>

         <p>
            <!-- input types "submit" and "reset" insert -->
            <!-- buttons for submitting and clearing the -->
            <!-- form's contents -->
            <input type = "submit" value = "Submit" />
            <input type = "reset" value = "Clear" />
         </p>
      </form>
   </body>
</html>
```

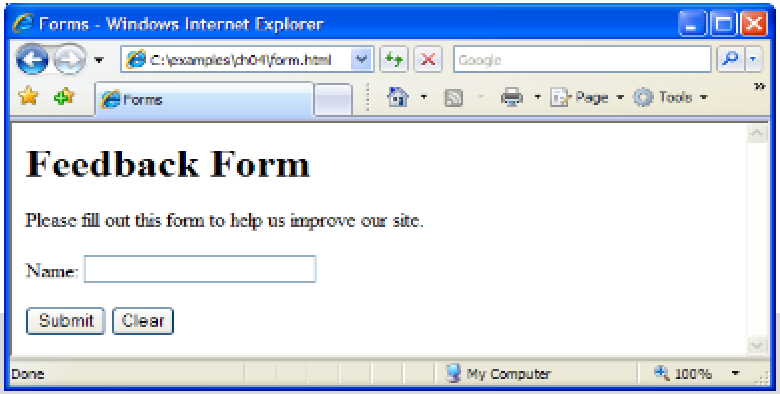
Creates a label for the text field

Inserts a text box called “name” with 25 characters visible and a 30 character limit

Inserts a submit button with “Submit” written on it

Inserts a reset button with “Clear” written on it

Ends the XHTML form



Feedback Form

Please fill out this form to help us improve our site.

Name:

Submit Clear

Look-and-Feel Observation 4.3

Include a *label* element for each form element to help users determine the purpose of each form element.

Good Programming Practice 4.6

Place hidden `input` elements at the beginning of a form, immediately after the opening `<form>` tag. This placement allows document authors to locate hidden `input` elements quickly.

4.11 Forms (Cont.)

The *`textarea`* element inserts a multiline text box, called a text area, into a form

- The number of rows in the text area is specified with the `rows` attribute
- The number of columns (i.e., characters per line) is specified with the `cols` attribute

The *`password`* input inserts a password box into a form

- Allows users to enter sensitive information, such as credit card numbers and passwords, by “masking” the information input with another character, usually asterisks
- The actual value input is sent to the Web server, not the asterisks that mask the input

4.11 Forms (Cont.)

The *`checkbox`* input allows the user to make a selection

- When the checkbox is selected, a check mark appears in the checkbox. Otherwise, the checkbox is empty
- Checkboxes can be used individually and in groups. Checkboxes that are part of the same group have the same name

A *radio* button is similar in function and use to a checkbox, except that only one radio button in a group can be selected at any time

- All radio buttons in a group have the same name attribute but different value attributes.

The *`select`* input provides a drop-down list of items

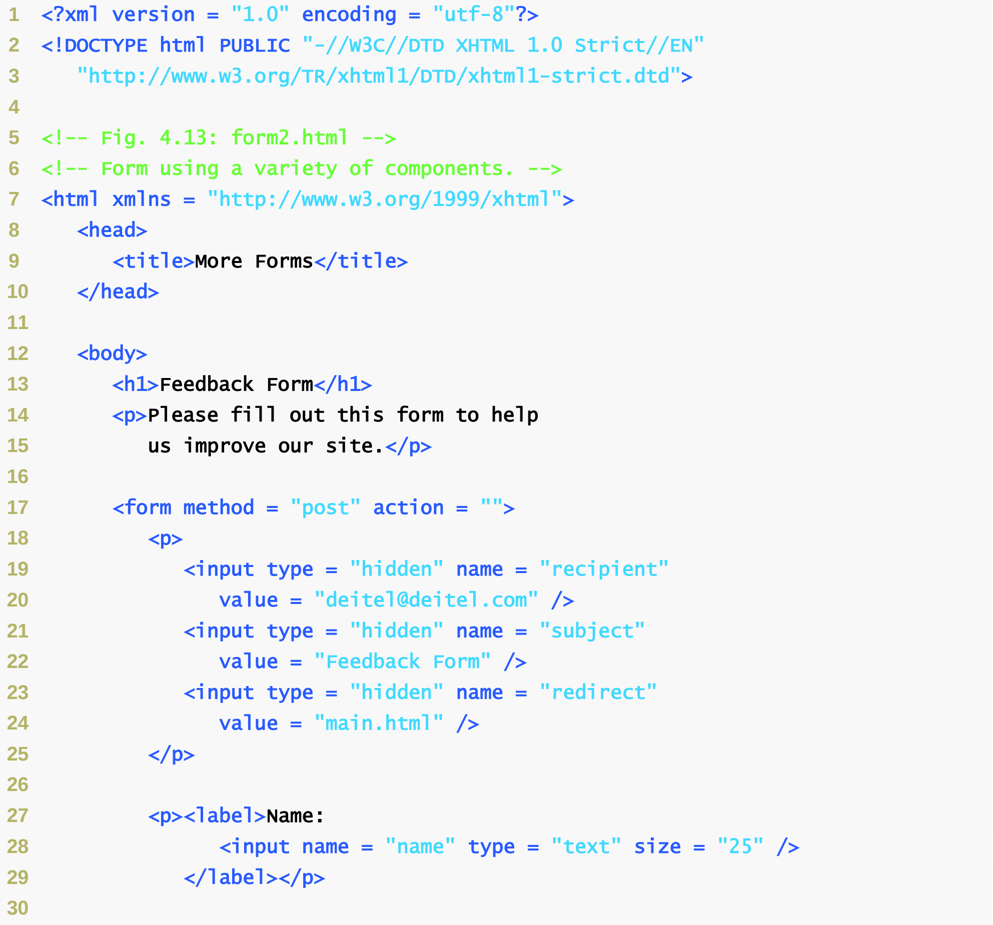
- The `name` attribute identifies the drop-down list
- The `option` element adds items to the drop-down list

4.11 Forms (Cont.)

The *`br`* element causes most browsers to render a line break

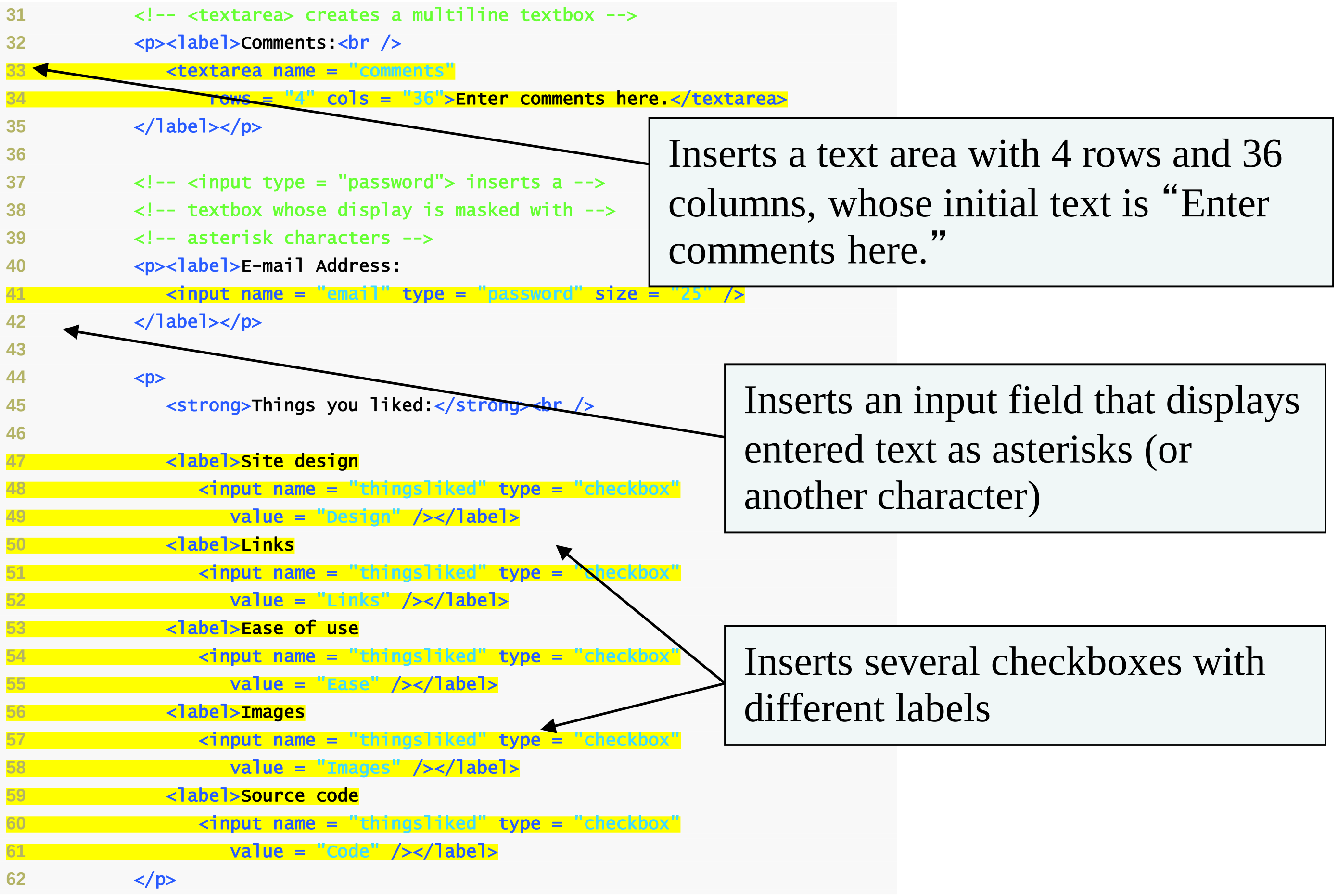
Any markup or text following a `br` element is rendered on the next line

Fig. 4.13 | Form using a variety of components (Part 1 of 5).



```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.13: form2.html -->
<!-- Form using a variety of components. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>More Forms</title>
   </head>

   <body>
      <h1>Feedback Form</h1>
      <p>Please fill out this form to help
         us improve our site.</p>

      <form method = "post" action = "">
         <p>
            <input type = "hidden" name = "recipient"
               value = "deitel@deitel.com" />
            <input type = "hidden" name = "subject"
               value = "Feedback Form" />
            <input type = "hidden" name = "redirect"
               value = "main.html" />
         </p>

         <p><label>Name:
               <input name = "name" type = "text" size = "25" />
            </label></p>

```

Fig. 4.13 | Form using a variety of components (Part 2 of 5).



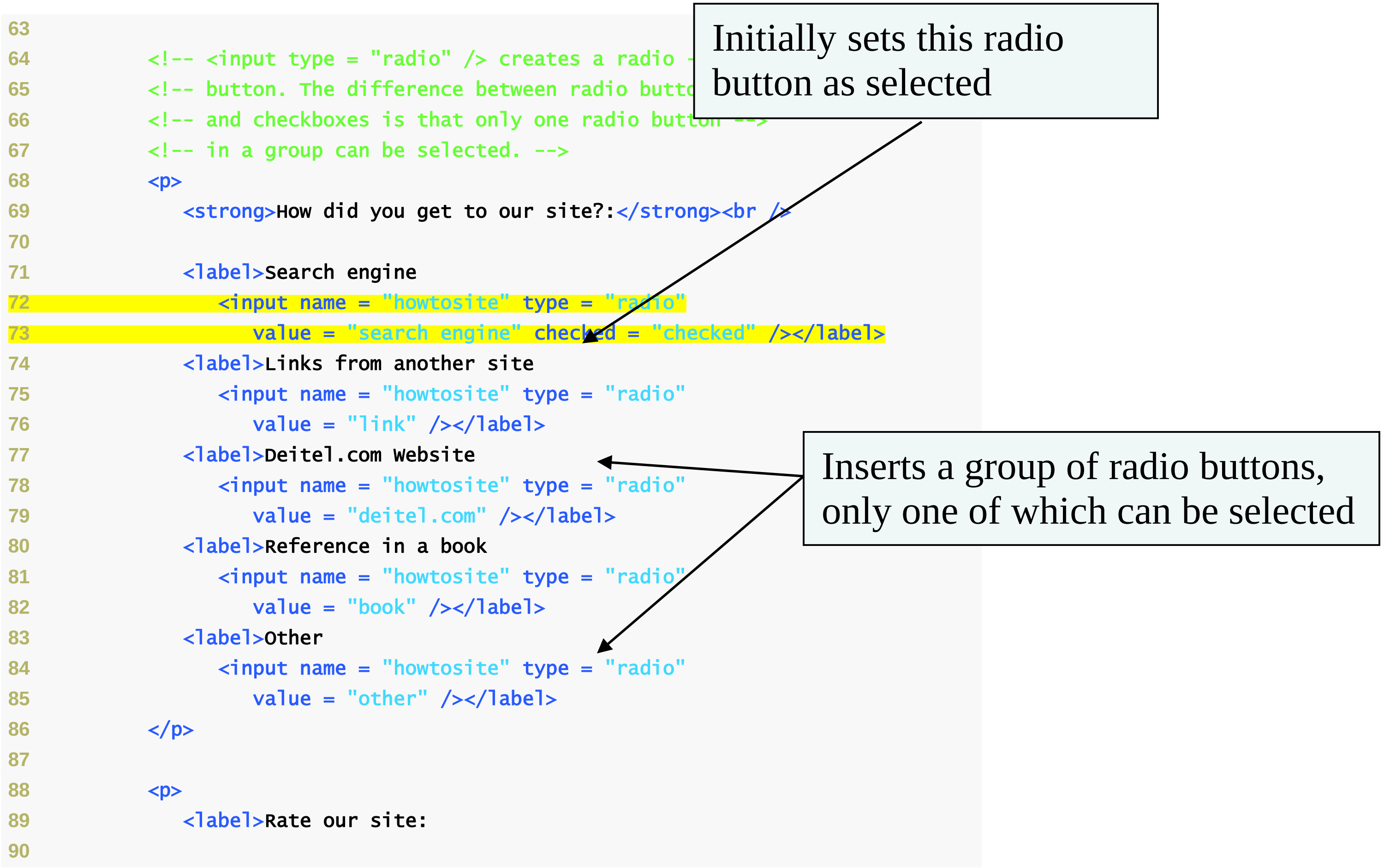
```
         <!-- <textarea> creates a multiline textbox -->
         <p><label>Comments:<br />
            <textarea name = "comments"
                rows = "4" cols = "36">Enter comments here.</textarea>
         </label></p>

         <!-- <input type = "password"> inserts a -->
         <!-- textbox whose display is masked with -->
         <!-- asterisk characters -->
         <p><label>E-mail Address:
            <input name = "email" type = "password" size = "25" />
         </label></p>

         <p>
            <strong>Things you liked:</strong><br />

            <label>Site design
               <input name = "thingsliked" type = "checkbox"
                  value = "Design" /></label>
            <label>Links
               <input name = "thingsliked" type = "checkbox"
                  value = "Links" /></label>
            <label>Ease of use
               <input name = "thingsliked" type = "checkbox"
                  value = "Ease" /></label>
            <label>Images
               <input name = "thingsliked" type = "checkbox"
                  value = "Images" /></label>
            <label>Source code
               <input name = "thingsliked" type = "checkbox"
                  value = "Code" /></label>
         </p>
```

Inserts a text area with 4 rows and 36 columns, whose initial text is “Enter comments here.”

Inserts an input field that displays entered text as asterisks (or another character)

Inserts several checkboxes with different labels

Fig. 4.13 | Form using a variety of components (Part 3 of 5).

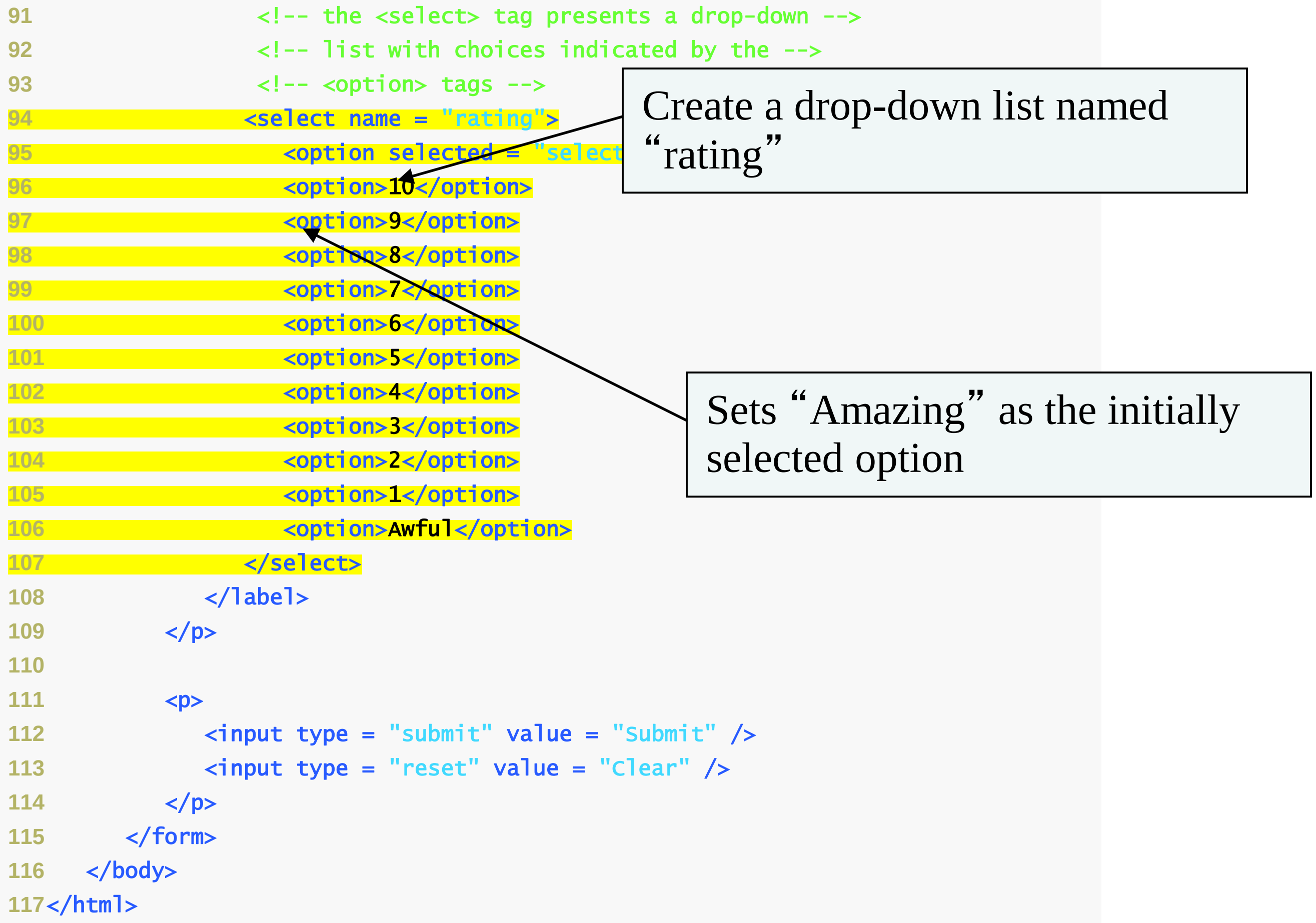


```

            <!-- <input type = "radio" /> creates a radio -
            <!-- button. The difference between radio butto
            <!-- and checkboxes is that only one radio button -->
            <!-- in a group can be selected. -->
            <p>
               <strong>How did you get to our site?:</strong><br />

               <label>Search engine
                  <input name = "howtosite" type = "radio"
                     value = "search engine" checked = "checked" /></label>
               <label>Links from another site
                  <input name = "howtosite" type = "radio"
                     value = "link" /></label>
               <label>Deitel.com Website
                  <input name = "howtosite" type = "radio"
                     value = "deitel.com" /></label>
               <label>Reference in a book
                  <input name = "howtosite" type = "radio"
                     value = "book" /></label>
               <label>Other
                  <input name = "howtosite" type = "radio"
                     value = "other" /></label>
            </p>

            <p>
               <label>Rate our site:

```

Initially sets this radio button as selected

Inserts a group of radio buttons, only one of which can be selected

Fig. 4.13 | Form using a variety of components (Part 4 of 5).

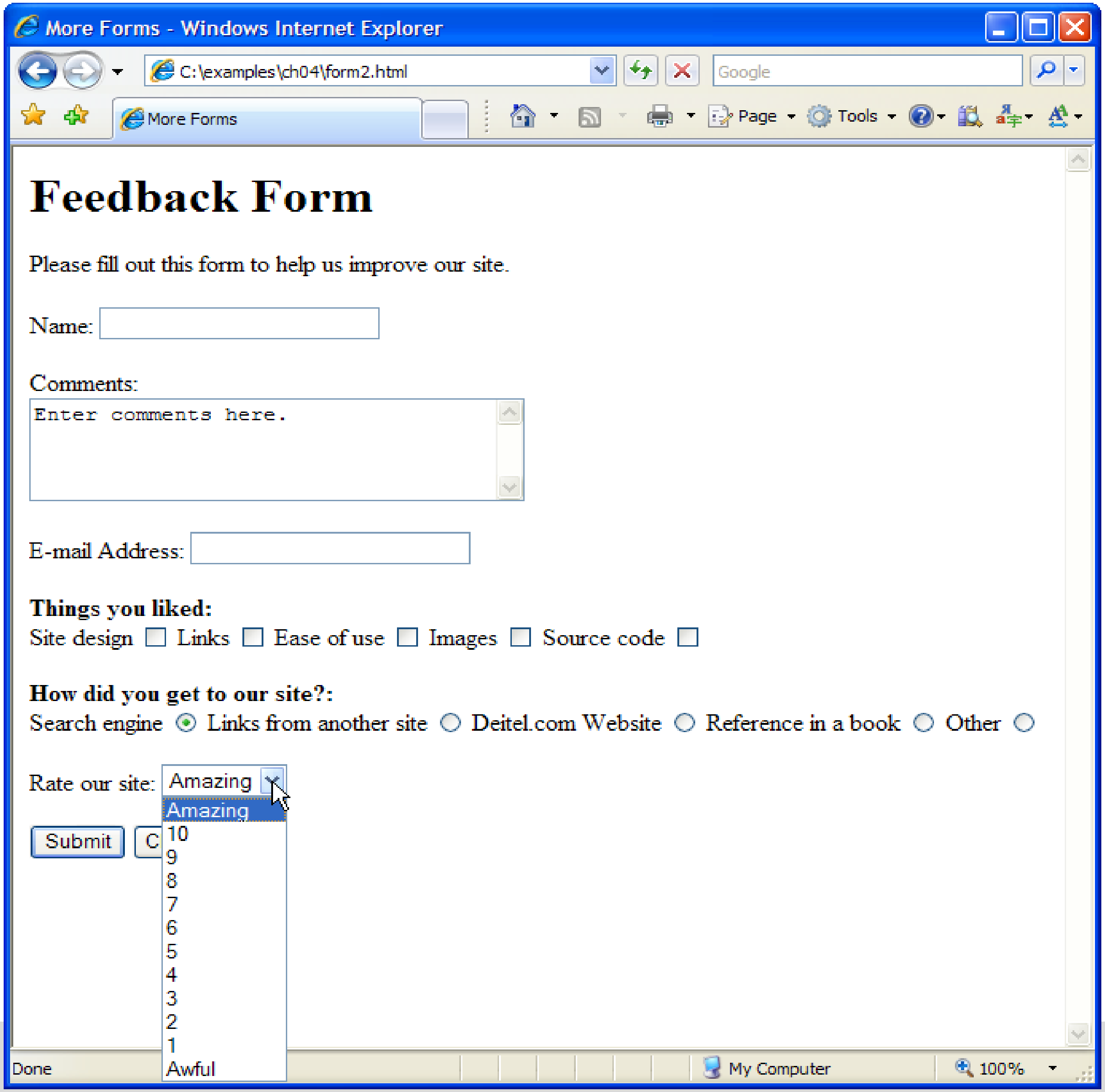


```
                <!-- the <select> tag presents a drop-down -->
                <!-- list with choices indicated by the -->
                <!-- <option> tags -->
               <select name = "rating">
                  <option selected = "select
                  <option>10</option>
                  <option>9</option>
                  <option>8</option>
                  <option>7</option>
                  <option>6</option>
                  <option>5</option>
                  <option>4</option>
                  <option>3</option>
                  <option>2</option>
                  <option>1</option>
                  <option>Awful</option>
               </select>
            </label>
         </p>

         <p>
            <input type = "submit" value = "Submit" />
            <input type = "reset" value = "Clear" />
         </p>
      </form>
   </body>
</html>
```

Create a drop-down list named “rating”

Sets “Amazing” as the initially selected option

Fig. 4.13 | Form using a variety of components (Part 5 of 5).



More Forms - Windows Internet Explorer
C:\examples\ch04\form2.html
Google
More Forms
Page
Tools
Done
My Computer
100%

Feedback Form

Please fill out this form to help us improve our site.

Name:

Comments:

```
Enter comments here.
```

E-mail Address:

Things you liked:

Site design □ Links □ Ease of use □ Images □ Source code □

How did you get to our site?:

Search engine ● Links from another site ○ Deitel.com Website ○ Reference in a book ○ Other ○

Rate our site: Amazing

- Amazing
- 10
- 9
- 8
- 7
- 6
- 5
- 4
- 3
- 2
- 1
- Awful

Submit | C

Common Programming Error 4.5

When your `form` has several checkboxes with the same `name`, you must make sure that they have different `values`, or the scripts running on the Web server will not be able to distinguish them.

Common Programming Error 4.6

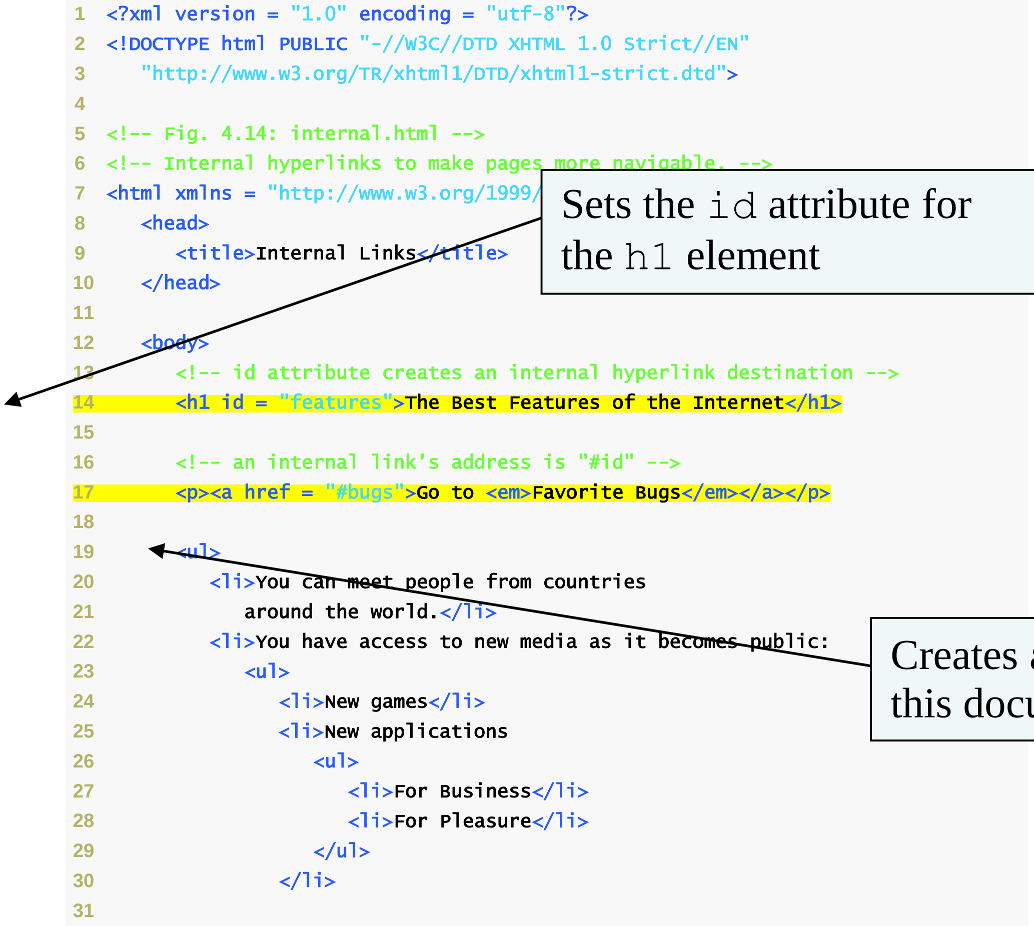
Not setting the `name` attributes of the radio buttons in a form to the same name is a logic error because it lets the user select all of them at the same time.

4.12 Internal Linking

The *a* tag can be used to link to another section of the same document by specifying the element’s *id* as the link’s *href*.

To link internally to an element with its *id* attribute set, use the syntax #*id*.

Fig. 4.14 | Internal hyperlinks to make pages more navigable (Part 1 of 3).

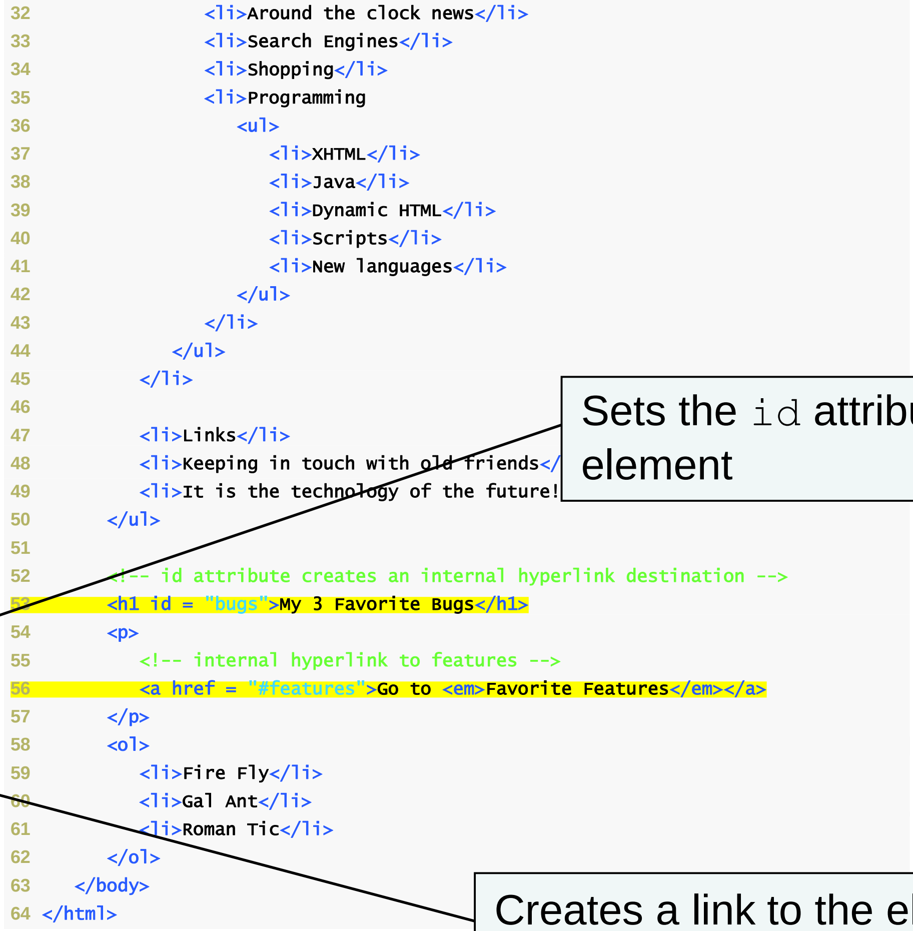


```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.14: internal.html -->
<!-- Internal hyperlinks to make pages more navigable. -->
<html xmlns = "http://www.w3.org/1999/
   <head>
      <title>Internal Links</title>
   </head>

   <body>
      <!-- id attribute creates an internal hyperlink destination -->
      <h1 id = "features">The Best Features of the Internet</h1>

      <!-- an internal link's address is "#id" -->
      <p><a href = "#bugs">Go to <em>Favorite Bugs</em></a></p>

      <ul>
         <li>You can meet people from countries
            around the world.</li>
         <li>You have access to new media as it becomes public:
            <ul>
               <li>New games</li>
               <li>New applications
                  <ul>
                     <li>For Business</li>
                     <li>For Pleasure</li>
                  </ul>
               </li>

```

Sets the id attribute for the h1 element

Creates a link to the element in this document with id = bugs

Fig. 4.14 | Internal hyperlinks to make pages more navigable (Part 2 of 3).

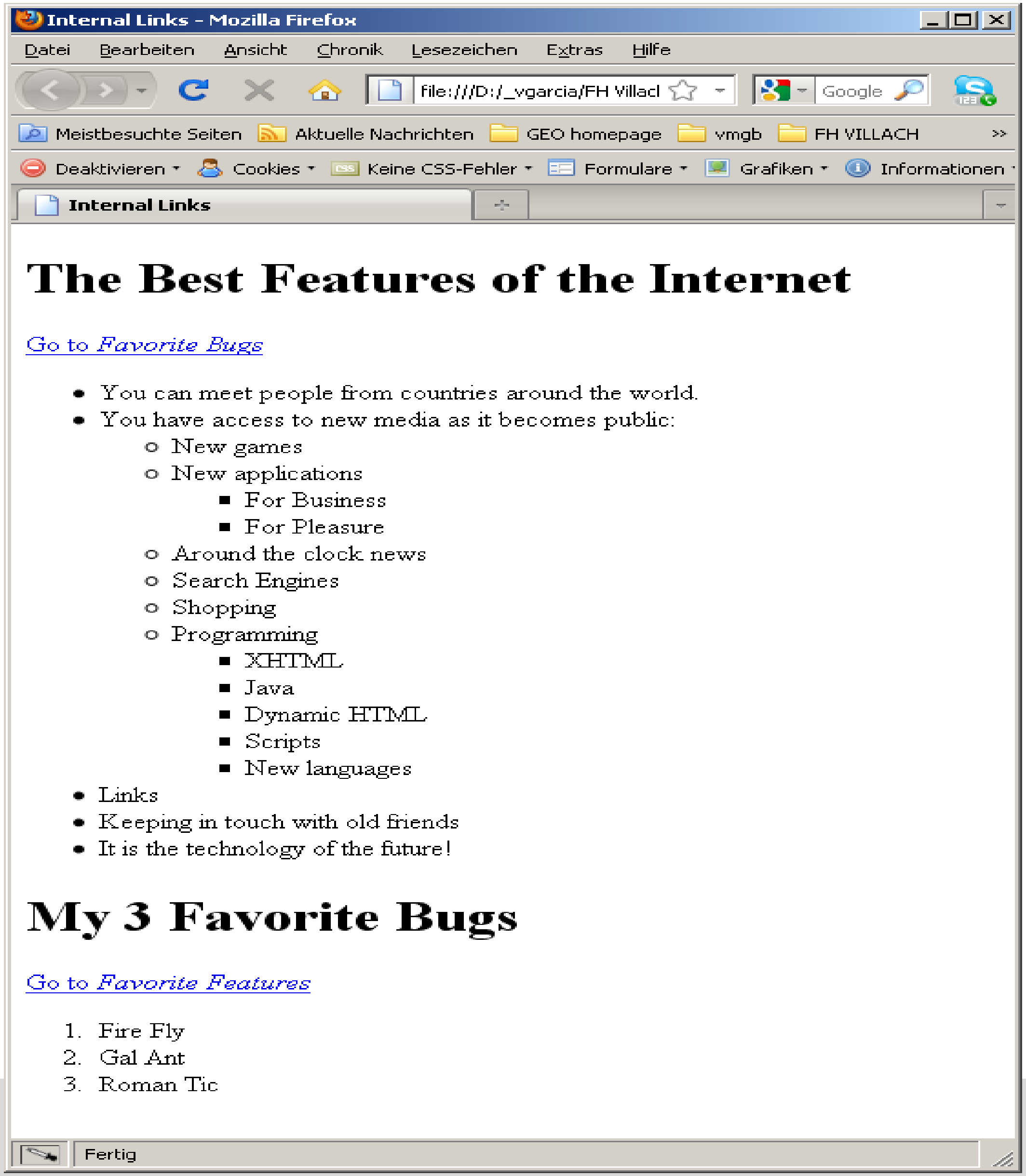


```
               <li>Around the clock news</li>
               <li>Search Engines</li>
               <li>Shopping</li>
               <li>Programming
                  <ul>
                     <li>XHTML</li>
                     <li>Java</li>
                     <li>Dynamic HTML</li>
                     <li>Scripts</li>
                     <li>New languages</li>
                  </ul>
               </li>
            </ul>
         </li>

         <li>Links</li>
         <li>Keeping in touch with old friends</
         <li>It is the technology of the future!
      </ul>

      <!-- id attribute creates an internal hyperlink destination -->
      <h1 id = "bugs">My 3 Favorite Bugs</h1>
      <p>
         <!-- internal hyperlink to features -->
         <a href = "#features">Go to <em>Favorite Features</em></a>
      </p>
      <ol>
         <li>Fire Fly</li>
         <li>Gal Ant</li>
         <li>Roman Tic</li>
      </ol>
   </body>
</html>
```

Sets the `id` attribute for this `h1` element

Creates a link to the element in this document with `id` = `features`

Fig. 4.14 | Internal hyperlinks to make pages more navigable (Part 3 of 3).



The Best Features of the Internet

Go to *Favorite Bugs*

- You can meet people from countries around the world.
- You have access to new media as it becomes public:
    - New games
    - New applications
        - For Business
        - For Pleasure
    - Around the clock news
    - Search Engines
    - Shopping
    - Programming
        - XHTML
        - Java
        - Dynamic HTML
        - Scripts
        - New languages
- Links
- Keeping in touch with old friends
- It is the technology of the future!

My 3 Favorite Bugs

Go to *Favorite Features*

1. Fire Fly
2. Gal Ant
3. Roman Tic

Look-and-Feel Observation 4.4

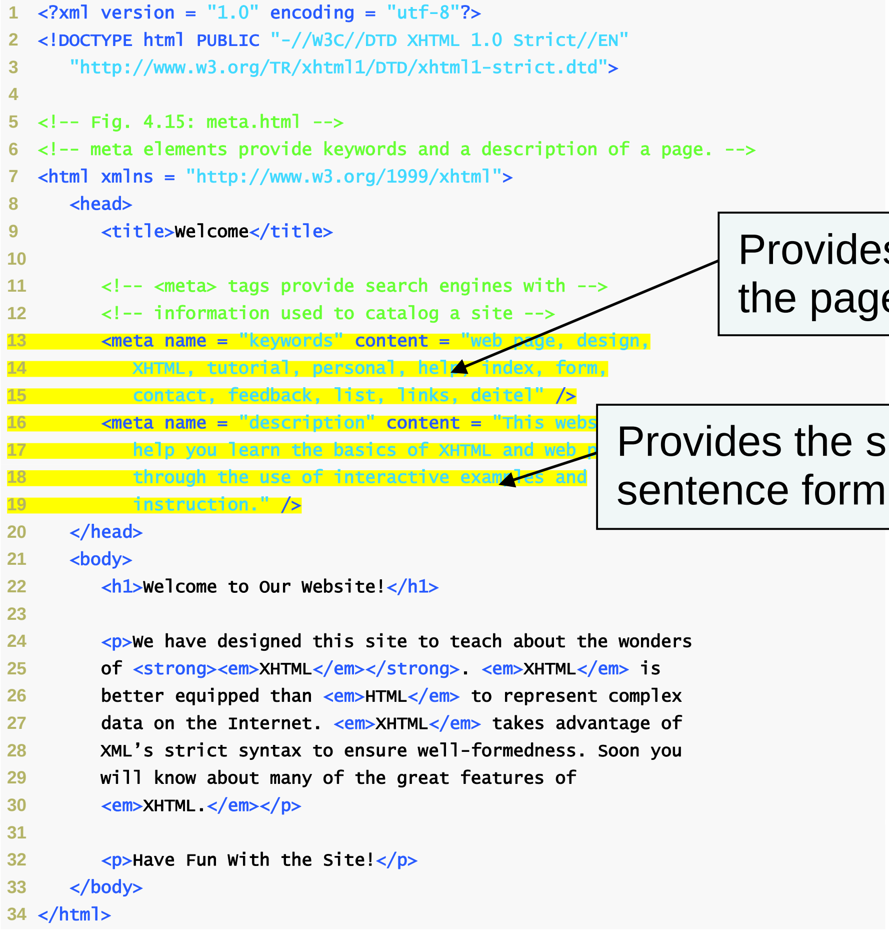
Internal hyperlinks are useful in XHTML documents that contain large amounts of information. Internal links to different parts of the page make it easier for users to navigate the page - they do not have to scroll to find the section they want.

4.13 `meta` Elements

One way that search engines catalog pages is by reading the *`meta`* element's contents.

- The *`name`* attribute identifies the type of `meta` element
- The *`content`* attribute
  - Of a *`keywords`* `meta` element: provides search engines with a list of words that describe a page, which are compared with words in search requests
  - Of a *`description`* `meta` element: provides a three- to four-line description of a site in sentence form, used by search engines to catalog your site. This text is sometimes displayed as part of the search result

Fig. 4.15 | `meta` elements provide keywords and a description of a page (Part 1 of 2).

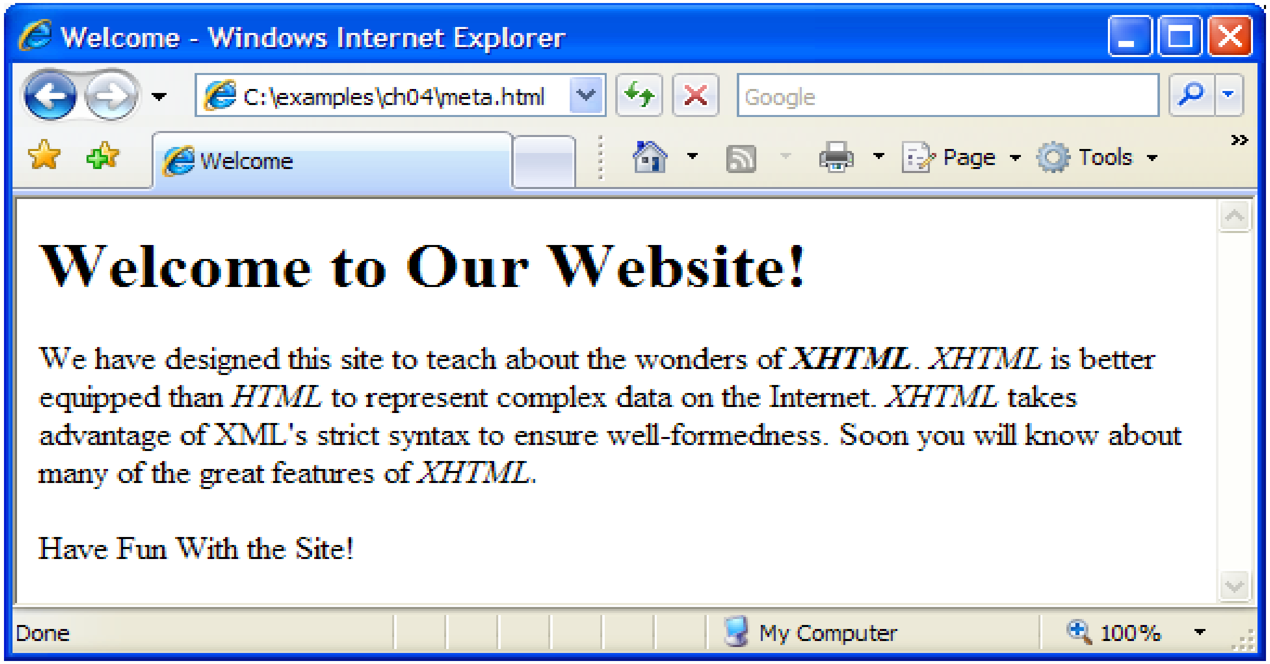


```
<?xml version = "1.0" encoding = "utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
   "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- Fig. 4.15: meta.html -->
<!-- meta elements provide keywords and a description of a page. -->
<html xmlns = "http://www.w3.org/1999/xhtml">
   <head>
      <title>Welcome</title>

      <!-- <meta> tags provide search engines with -->
      <!-- information used to catalog a site -->
      <meta name = "keywords" content = "web page, design,
         XHTML, tutorial, personal, help, index, form,
         contact, feedback, list, links, deitel" />
      <meta name = "description" content = "This webs
         help you learn the basics of XHTML and web p
         through the use of interactive examples and
         instruction." />
   </head>
   <body>
      <h1>Welcome to Our Website!</h1>

      <p>We have designed this site to teach about the wonders
      of <strong><em>XHTML</em></strong>. <em>XHTML</em> is
      better equipped than <em>HTML</em> to represent complex
      data on the Internet. <em>XHTML</em> takes advantage of
      XML’s strict syntax to ensure well-formedness. Soon you
      will know about many of the great features of
      <em>XHTML.</em></p>

      <p>Have Fun With the Site!</p>
   </body>
</html>
```

Provides keywords describing the page for a search engine

Provides the site’s description in sentence form for a search engine

Fig. 4.15 | `meta` elements provide keywords and a description of a page (Part 2 of 2).


Welcome - Windows Internet Explorer
C:\examples\ch04\meta.html
Google
Welcome
Page
Tools
Done
My Computer
100%

Welcome to Our Website!

We have designed this site to teach about the wonders of ***XHTML***. *XHTML* is better equipped than *HTML* to represent complex data on the Internet. *XHTML* takes advantage of XML's strict syntax to ensure well-formedness. Soon you will know about many of the great features of *XHTML*.

Have Fun With the Site!