## 2.1 Origins and Evolution of HTML

- Derived from SGML

- Original intent: General layout of documents that could be displayed by a wide variety of computers

- Recent versions:
    - HTML 4.0 - 1997
        - Introduced many new features and deprecated many older features

    - HTML 4.01 - 1998 - A cleanup of 4.0

    - XHTML 1.0
        - Just 4.01 modified to fit the XML syntax
        - We’ll stick to nearly all of this syntax

## 2.2 Basic Syntax

- Elements are defined by tags (markers)

    - Tag format:
        - Opening tag: <name [ /]>
        - Closing tag: </name>

## 2.2 Basic Syntax (continued)

- The *content* of a tag appears between its opening tag and its closing tag

- Not all tags have content
  - If a tag has no content, its form is <name `/>`

- If a tag has attributes, they appear between its name and the right bracket (later)

- Tag names and attribute names must be written in lowercase letters

- Every tag that has content must have a closing tag

- Tags must be properly nested

- Comment form: `<!-- … -->`

- Browsers ignore comments, unrecognizable tags, line breaks, multiple spaces, and tabs

- Tags are suggestions to the browser, even if they are recognized by the browser

# 2.3 Standard HTML Document Structure

- `<html>`, `<head>`, `<title>`, `<body>` (or `<frameset>`) are required

  - All four can have attributes, but we'll ignore them for now

- The whole document must have `<html>` as its root

- A document consists of a head and a body or frameset

- The `<title>` tag is used to give the document a title, which is normally displayed in the browser's window title bar (at the top of the display)

- An SGML `DOCTYPE` command is required by XHTML 1.0, but there is not enough room on these slides to show it - see book for its exact form

  - DOCTYPE provides a DTD for the version of XHTML that is used for the document

  - There are three DTDs, or standards `Strict`, `Frameset`, and `Transitional`

## 2.4 Basic Text Formatting

- Text is normally placed in paragraph elements

- *Paragraph Elements*

  - The `<p>` tag breaks the current line and inserts a blank line - the new line gets the beginning of the content of the paragraph

  - The browser puts as many words of the paragraph’s content as will fit in each line

  - The closing paragraph tag is required by XHTML

```
<!-- greet.hmtl
     A trivial document
     -->
<html>
<head> <title> Our first document </title>
</head>
<body>
<p>
Greetings from your Webmaster!
</p>
</body>
</html>
```

# 2.4 Basic Text Formatting (continued)

- W3C HTML Validation Service
  `http://validator.w3.org/file-upload.html`

- Line breaks

  - The effect of the `<br />` tag is the same as that of `<p>`, except for the blank line
    - No closing tag!

- Example of paragraphs and line breaks

```
On the plains of hesitation <p> bleach the
bones of countless millions </p> <br />
who, at the dawn of victory <br /> sat down
to wait, and waiting, died.
```

- Typical display of this text:

```
On the plains of hesitation

bleach the bones of countless millions
who, at the dawn of victory
sat down to wait, and waiting, died.
```

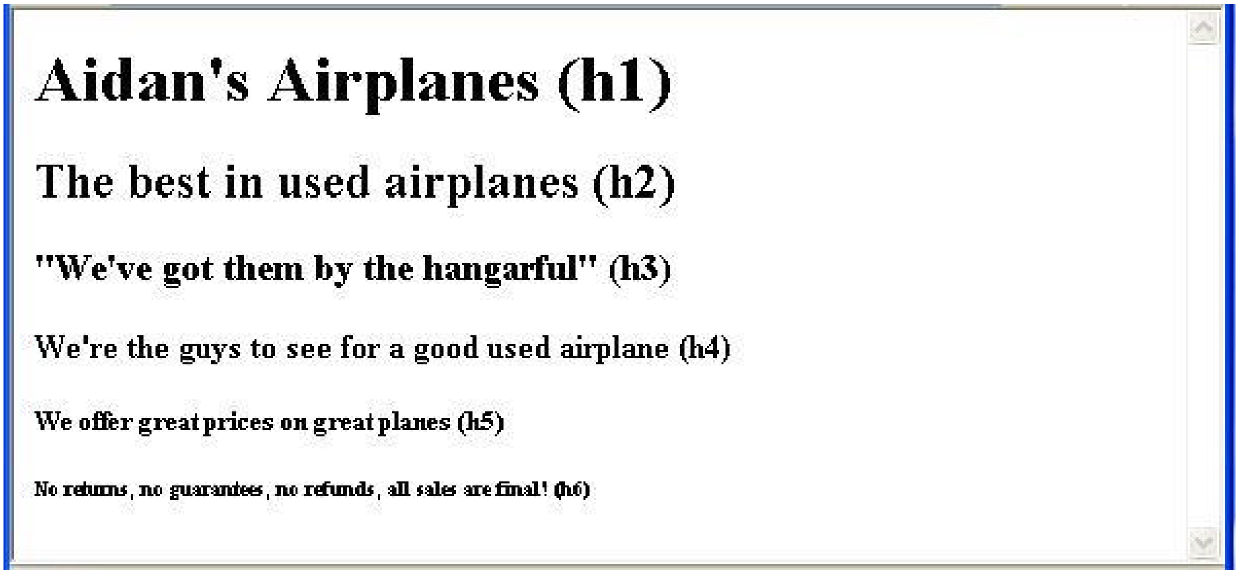
- Inline versus block elements
  - Block elements CANNOT be nested in inline elements

## 2.4 Basic Text Formatting (continued)

- ***Headings***
  - Six sizes, 1 - 6, specified with `<h1>` to `<h6>`
  - 1, 2, and 3 use font sizes that are larger than the default font size
  - 4 uses the default size
  - 5 and 6 use smaller font sizes

```
<!-- headings.html
     An example to illustrate headings
     -->
<html>
<head> <title> Headings </title>
</head>
<body>
<h1> Aidan’s Airplanes (h1) </h1>
<h2> The best in used airplanes (h2) </h2>
<h3> "We’ve got them by the hangarful" (h3)
</h3>
<h4> We’re the guys to see for a good used
     airplane (h4) </h4>
<h5> We offer great prices on great planes
     (h5) </h5>
<h6> No returns, no guarantees, no refunds,
     all sales are final (h6) </h6>
</body>
</html>
```

# 2.4 Basic Text Formatting (continued)



# Aidan's Airplanes (h1)

## The best in used airplanes (h2)

### "We've got them by the hangarful" (h3)

#### We're the guys to see for a good used airplane (h4)

##### We offer great prices on great planes (h5)

###### No returns, no guarantees, no refunds, all sales are final! (h6)

- Blockquotes
  - Content of `<blockquote>`
  - To set a block of text off from the normal flow and appearance of text
  - Browsers often indent, and sometimes italicize

- *Font Styles and Sizes (can be nested)*
  - Boldface - `<b>`
  - Italics - `<i>`
  - Larger - `<big>`
  - Smaller - `<small>`
  - Monospace - `<tt>`

## 2.4 Basic Text Formatting (continued)

```
The <big> sleet <big> in <big> <i> Crete
</i><br /> lies </big> completely </big>
 in </big> the street
```

The sleet in *Crete*
lies completely in the street

- These tags are not affected if they appear in the content of a `<blockquote>`, unless there is a conflict (e.g., italics)

- *Superscripts and subscripts*

  - Subscripts with `<sub>`
  - Superscripts with `<sup>`

Example: `x<sub>2</sub><sup>3</sup>`

Display: $x_2^3$

- All of this font size and font style stuff can be done with style sheets, but these tags are not yet deprecated

# 2.4 Basic Text Formatting (continued)

- Character Entities

  - There are some characters that HTML treats as special characters, so if you want one in a document, it must be coded

| *Char.* | *Entity* | *Meaning* |
|---|---|---|
| & | `&amp` | Ampersand |
| < | `&lt` | Less than |
| > | `&gt` | Greater than |
| " | `&quot` | Double quote |
| ' | `&apos` | Single quote |
| ¼ | `&frac14` | One quarter |
| ½ | `&frac12` | One half |
| ¾ | `&frac34` | Three quarters |
| ° | `&deg` | Degree |
| (space) | `&nbsp` | Non-breaking space |

- Horizontal rules
  - `<hr />` draws a line across the display, after a line break

## 2.5 Images

- GIF (Graphic Interchange Format)
  - 8-bit color (256 different colors)

- JPEG (Joint Photographic Experts Group)
  - 24-bit color (16 million different colors)

- Both used compression, but JPEG compression is better

- Images are inserted into a document with the `<img />` tag with the `src` attribute

  - The `alt` attribute is required by XHTML

```
<img src = "comets.jpg"
     alt = "Picture of comets" />
```

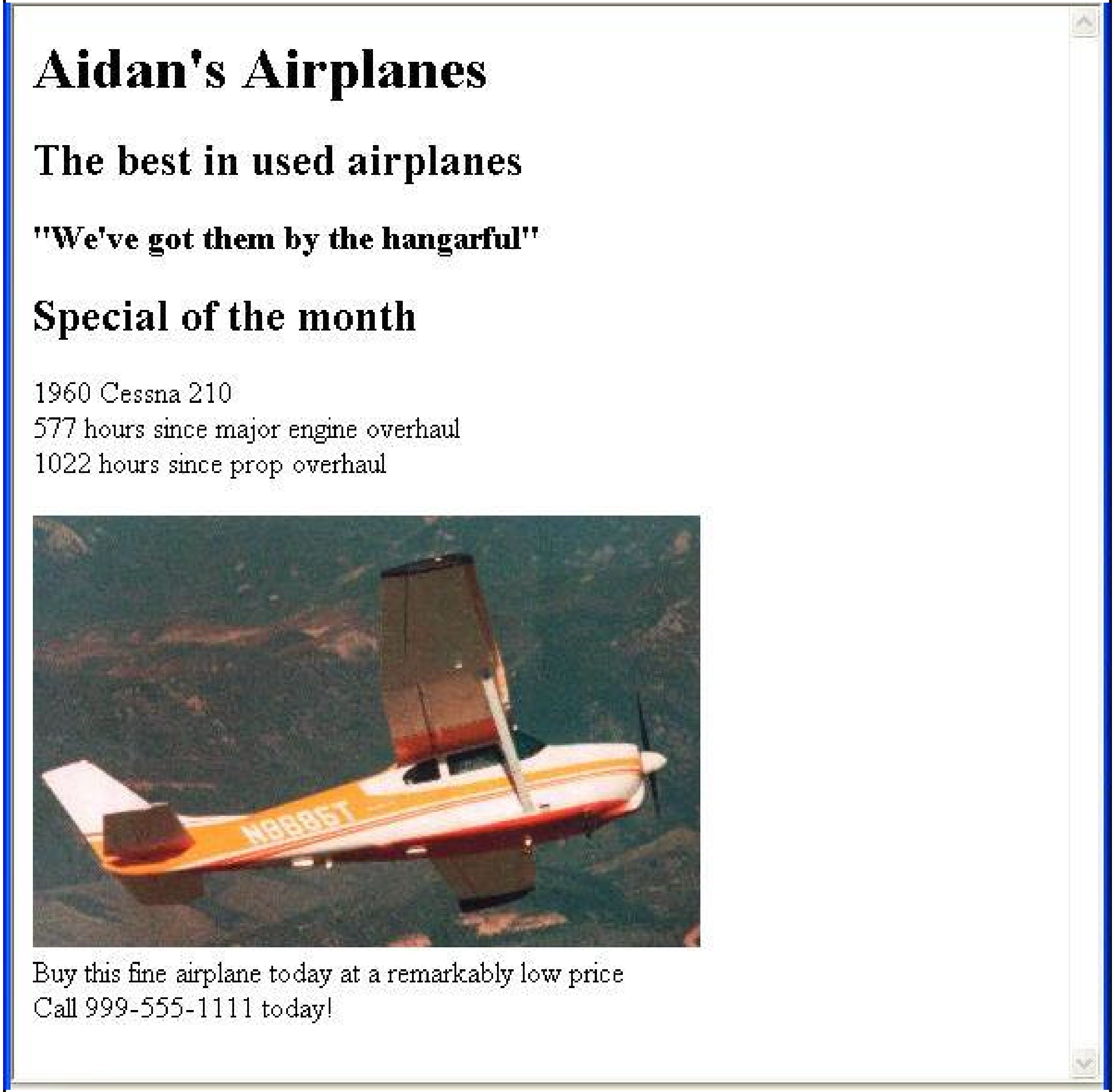
- Portable Network Graphics (PNG)
  - Relatively new
  - Not supported by IE5 or IE6

- The `<img>` tag has 30 different attributes!

# 2.5 Images (continued)

```
<!-- image.html
     An example to illustrate an image
     -->
<html>
<head> <title> Images </title>
</head>
<body>
<h1> Aidan's Airplanes </h1>
<h2> The best in used airplanes </h2>
<h3> "We've got them by the hangarful" </h3>
<h2> Special of the month </h2>
<p>
1960 Cessna 210 <br />
577 hours since major engine overhaul<br />
1022 hours since prop overhaul <br /><br />
 <img src = "c210new.jpg"
      alt = "Picture of a Cessna 210"/> <br />
Buy this fine airplane today at a remarkably
  low price <br />
Call 999-555-1111 today!
</p>
</body>
</html>
```

# 2.5 Images (continued)



# Aidan's Airplanes

## The best in used airplanes

### "We've got them by the hangarful"

## Special of the month

1960 Cessna 210
577 hours since major engine overhaul
1022 hours since prop overhaul

Buy this fine airplane today at a remarkably low price
Call 999-555-1111 today!

# 2.6 Hypertext Links

- Hypertext is the essence of the Web!

- Both links (sometimes called sources) and targets (or anchors) of links are specified with the anchor tag (<a>)

- A link is specified with the `href` (*h*ypertext *ref*erence) attribute of <a>

  - The content of <a> is the visual link in the document

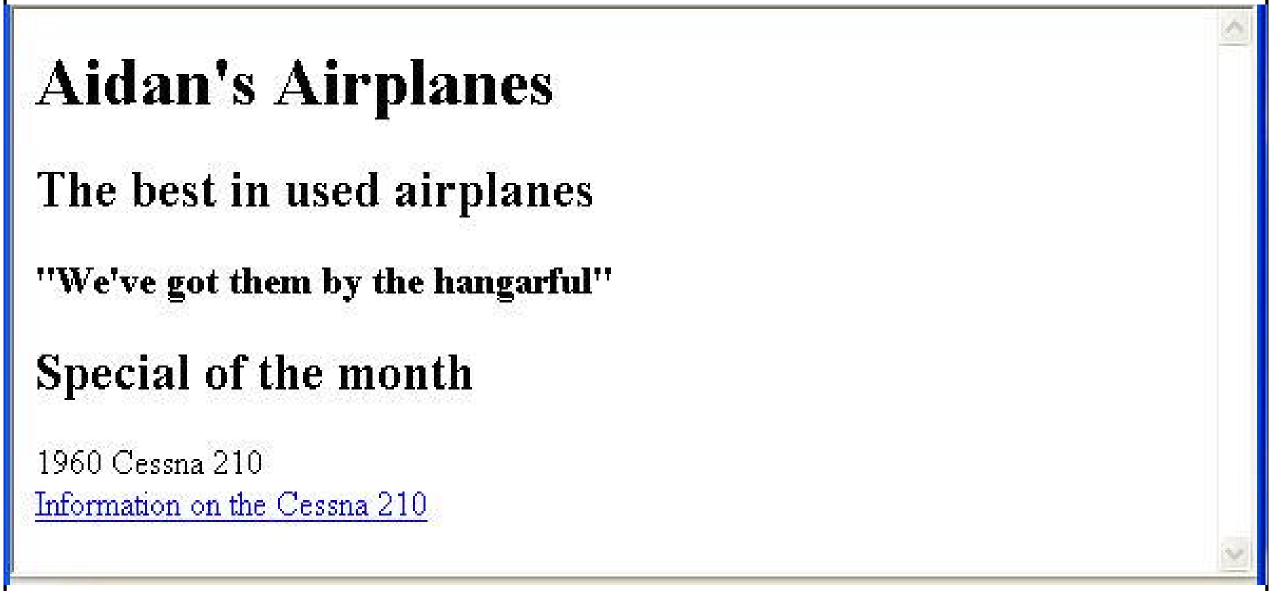
  - If the target is a whole document (not the one in which the link appears), the target need not be specified in the target document as being the target

- Note: Relative addressing of targets is easier to maintain and more portable than absolute addressing

## 2.6 Hypertext Links (continued)

```
<!-- link.html
     An example to illustrate a link
     -->
<html>
<head> <title> Links </title>
</head>
<body>
<h1> Aidan's Airplanes </h1>
<h2> The best in used airplanes </h2>
<h3> "We've got them by the hangarful" </h3>
<h2> Special of the month </h2>
<p>
1960 Cessna 210 <br />
<a href = "C210data.html">
  Information on the Cessna 210 </a>
</p>
</body>
</html>
```

# 2.6 Hypertext Links (continued)



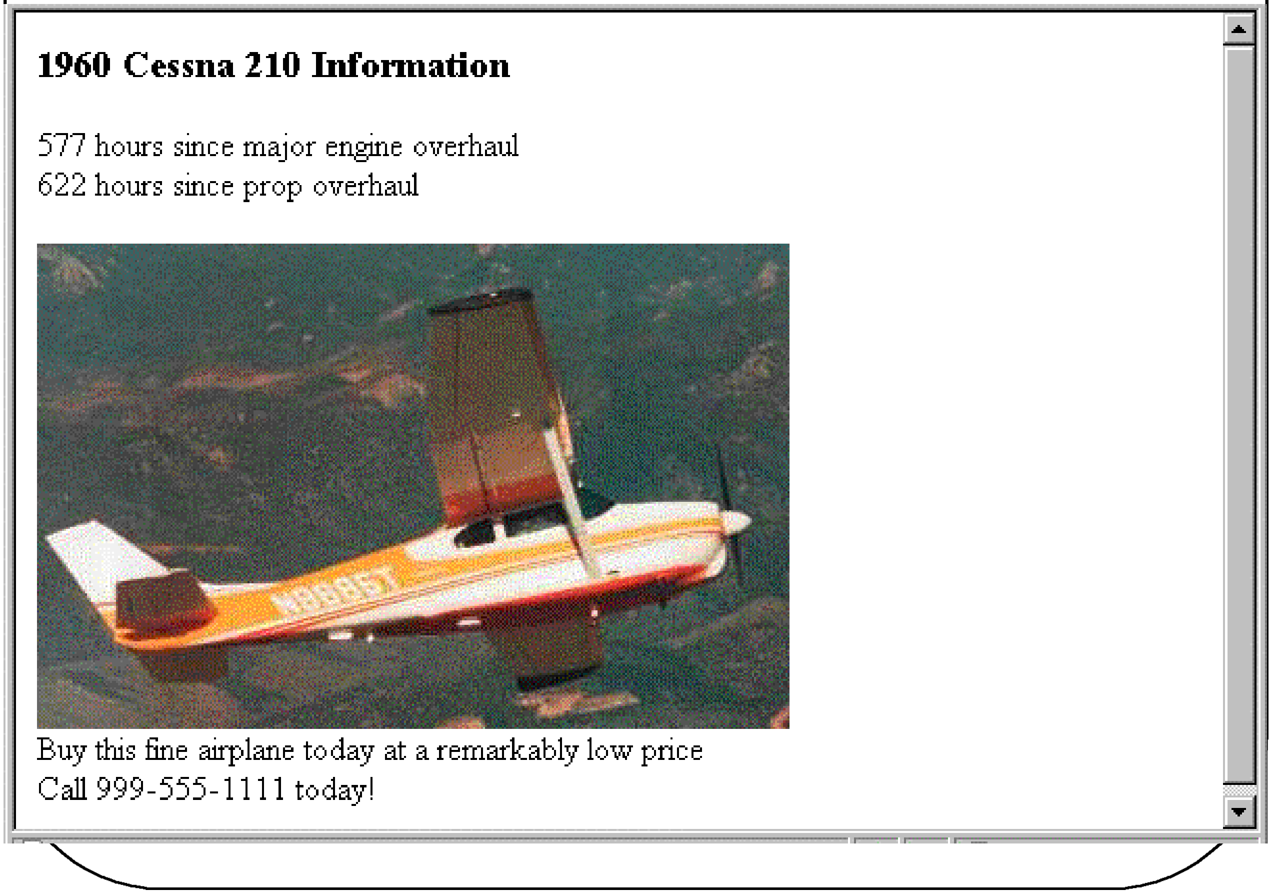
# Aidan's Airplanes

## The best in used airplanes

#### "We've got them by the hangarful"

## Special of the month

1960 Cessna 210
Information on the Cessna 210



#### 1960 Cessna 210 Information

577 hours since major engine overhaul
622 hours since prop overhaul

Buy this fine airplane today at a remarkably low price
Call 999-555-1111 today!

## 2.6 Hypertext Links (continued)

- If the target is a document fragment, the target spot must be marked somehow

- Target labels can be defined in many different tags with the `id` attribute, as in

```
<h1 id = "baskets"> Baskets </h1>
```

- The link to an `id` must be preceded by a pound sign (#); If the `id` is in the same document, this target could be

```
<a href = "#baskets">
                What about baskets? </a>
```

- If the link is in a different document, the document reference must be included

```
<a href = "myAd.html#baskets”> Baskets </a>
```

- Style note: links should blend in with the surrounding text, so reading it without taking the link should not be made less pleasant

- Links can have images:

```
<a href = "c210data.html"
  <img src = "smallplane.jpg"
    alt = "Small picture of an airplane " />>
      Info on C210 </a>
```

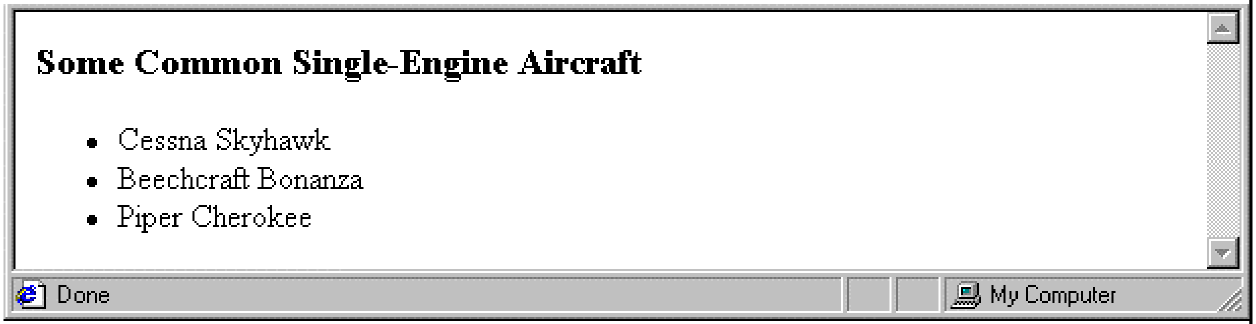
# 2.7 Lists

- *Unordered lists*

  - The list is the content of the `<ul>` tag

  - List elements are the content of the `<li>` tag

```
<h3> Some Common Single-Engine Aircraft </h3>
   <ul>
      <li> Cessna Skyhawk </li>
      <li> Beechcraft Bonanza </li>
      <li> Piper Cherokee </li>
   </ul>
```



### Some Common Single-Engine Aircraft

- Cessna Skyhawk
- Beechcraft Bonanza
- Piper Cherokee

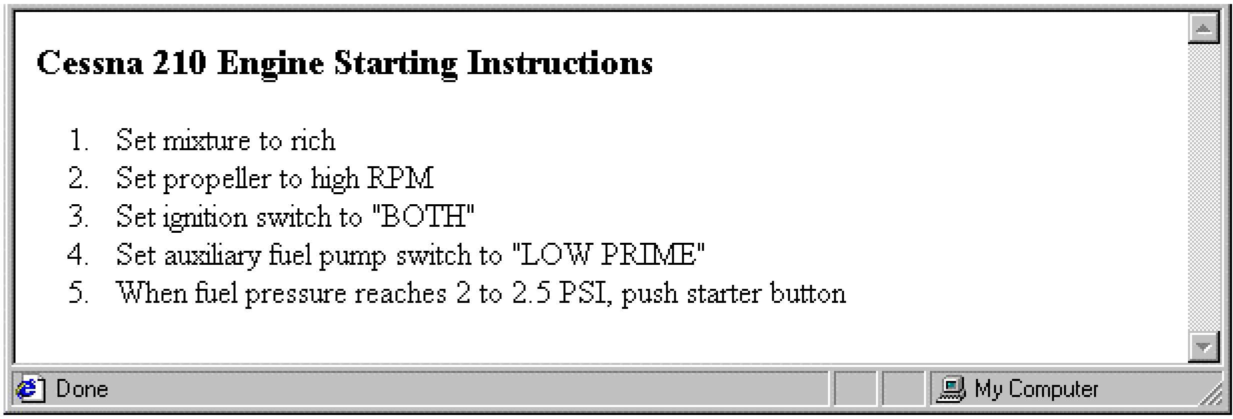
- *Ordered lists*

  - The list is the content of the `<ol>` tag

  - Each item in the display is preceded by a sequence value

## 2.7 Lists (continued)

```
<h3> Cessna 210 Engine Starting Instructions
 </h3>
<ol>
    <li> Set mixture to rich </li>
    <li> Set propeller to high RPM </li>
    <li> Set ignition switch to "BOTH" </li>
    <li> Set auxiliary fuel pump switch to
         "LOW PRIME" </li>
    <li> When fuel pressure reaches 2 to 2.5
         PSI, push starter button </li>
</ol>
```



### Cessna 210 Engine Starting Instructions

1. Set mixture to rich
2. Set propeller to high RPM
3. Set ignition switch to "BOTH"
4. Set auxiliary fuel pump switch to "LOW PRIME"
5. When fuel pressure reaches 2 to 2.5 PSI, push starter button

- *Nested lists*

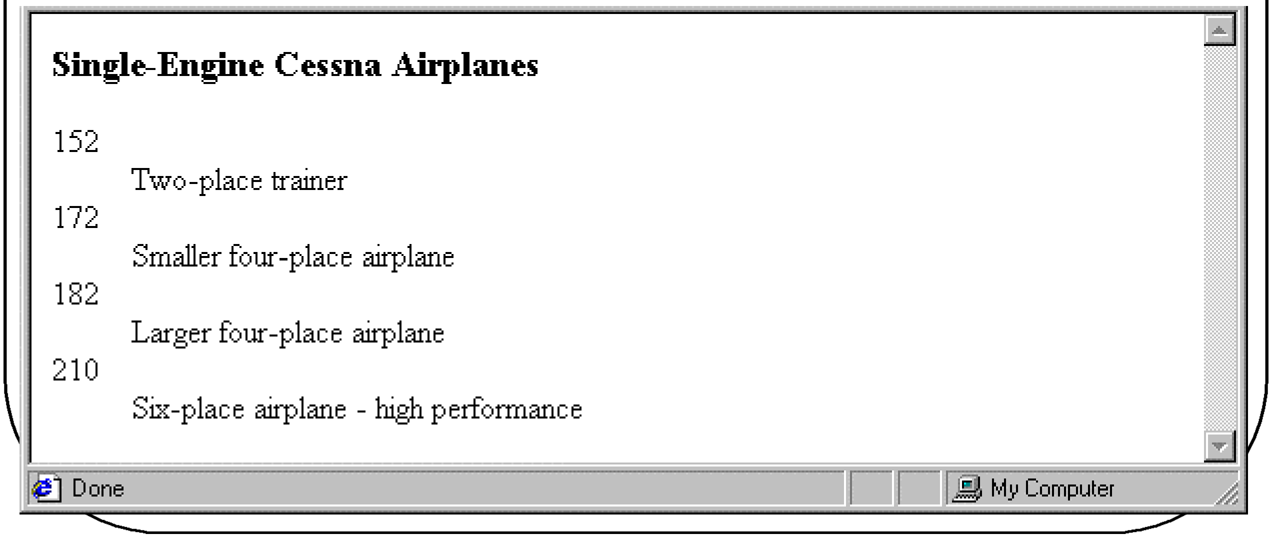
  - Any type list can be nested inside any type list

  - The nested list must be in a list item

# 2.7 Lists (continued)

- *Definition lists (for glossaries, etc.)*
    - List is the content of the `<dl>` tag
    - Terms being defined are the content of the `<dt>` tag
    - The definitions themselves are the content of the `<dd>` tag

```
<h3> Single-Engine Cessna Airplanes </h3>
<dl >
    <dt> 152 </dt>
    <dd> Two-place trainer </dd>
    <dt> 172 </dt>
    <dd> Smaller four-place airplane </dd>
    <dt> 182 </dt>
    <dd> Larger four-place airplane </dd>
    <dt> 210 </dt>
    <dd> Six-place airplane - high performance
    </dd>
</dl>
```



**Single-Engine Cessna Airplanes**

152
    Two-place trainer
172
    Smaller four-place airplane
182
    Larger four-place airplane
210
    Six-place airplane - high performance

# 2.8 Tables

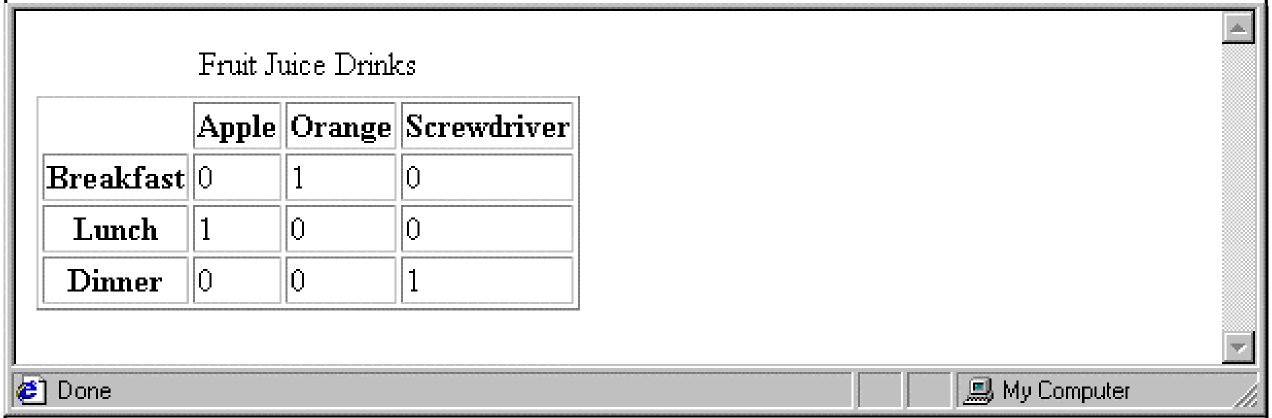
- A table is a matrix of cells, each possibly having content
  - The cells can include almost any element
  - Some cells have row or column lables and some have data
- A table is specified as the content of a `<table>` tag
- A `border` attribute in the `<table>` tag specifies a border between the cells
  - If `border` is set to `"border"`, the browser's default width border is used
  - The `border` attribute can be set to a number, which will be the border width
  - Without the `border` attribute, the table will have no lines!
- Tables are given titles with the `<caption>` tag, which can immediately follow `<table>`

## 2.8 Tables (continued)

- Each row of a table is specified as the content of a `<tr>` tag
- The row headings are specified as the content of a `<th>` tag
- The contents of a data cell is specified as the content of a `<td>` tag

```
<table border = "border">
<caption> Fruit Juice Drinks </caption>
    <tr>
        <th> </th>
        <th> Apple </th>
        <th> Orange </th>
        <th> Screwdriver </th>
    </tr>
    <tr>
        <th> Breakfast </th>
        <td> 0 </td>
        <td> 1 </td>
        <td> 0 </td>
    </tr>
    <tr>
        <th> Lunch </th>
        <td> 1 </td>
        <td> 0 </td>
        <td> 0 </td>
    </tr>
</table>
```

# 2.8 Tables (continued)



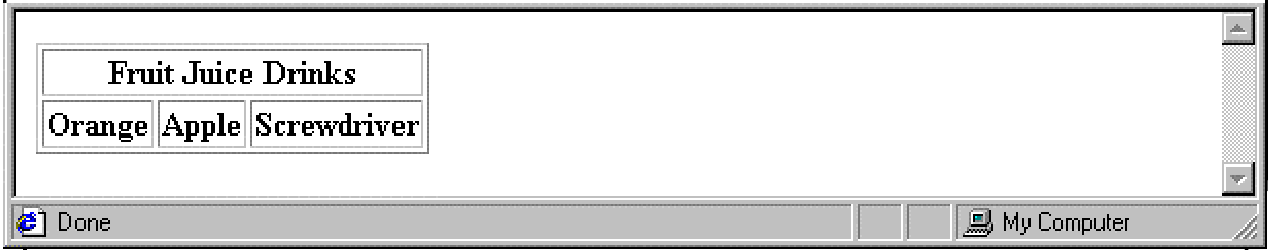
Fruit Juice Drinks

| | Apple | Orange | Screwdriver |
|---|---|---|---|
| **Breakfast** | 0 | 1 | 0 |
| **Lunch** | 1 | 0 | 0 |
| **Dinner** | 0 | 0 | 1 |

- **A table can have two levels of column labels**

  - **If so, the `colspan` attribute must be set in the `<th>` tag to specify that the label must span some number of columns**

```
<tr>
    <th colspan = "3"> Fruit Juice Drinks </th>
</tr>
<tr>
    <th> Orange </th>
    <th> Apple </th>
    <th> Screwdriver </th>
</tr>
```

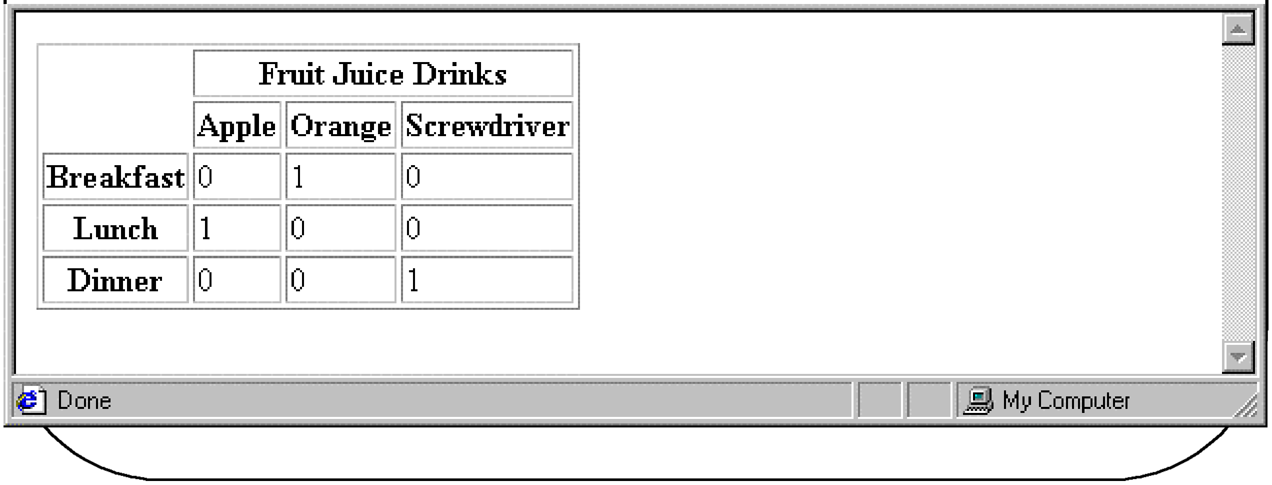


| Fruit Juice Drinks | | |
|---|---|---|
| **Orange** | **Apple** | **Screwdriver** |

# 2.8 Tables (continued)

- If the rows have labels and there is a spanning column label, the upper left corner must be made larger, using `rowspan`

```
<table border = "border">
<caption> Fruit Juice Drinks and Meals
</caption>
    <tr>
        <td rowspan = "2"> </td>
        <th colspan = "3"> Fruit Juice Drinks
         </th>
    </tr>
    <tr>
        <th> Apple </th>
        <th> Orange </th>
        <th> Screwdriver </th>
    </tr>
    …
</table>
```



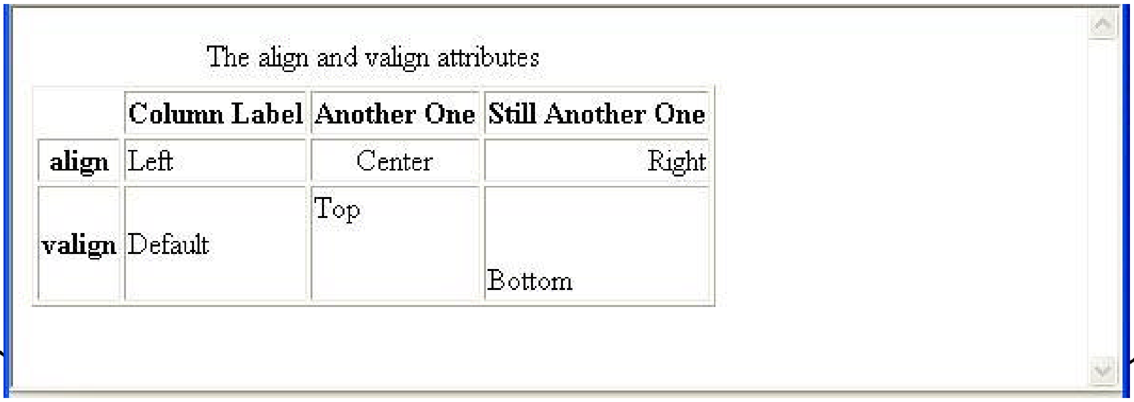
| | Fruit Juice Drinks | | |
|---|---|---|---|
| | Apple | Orange | Screwdriver |
| Breakfast | 0 | 1 | 0 |
| Lunch | 1 | 0 | 0 |
| Dinner | 0 | 0 | 1 |

## 2.8 Tables (continued)

- The `align` attribute controls the horizontal placement of the contents in a table cell
    - Values are `left`, `right`, and `center` (default)
    - `align` is an attribute of `<tr>`, `<th>`, and `<td>` elements
- The `valign` attribute controls the vertical placement of the contents of a table cell
    - Values are `top`, `bottom`, and `center` (default)
    - `valign` is an attribute of `<th>` and `<td>` elements

# 2.8 Tables (continued)

```
<table border = "border">
<caption> The align and valign attributes
</caption>
    <tr align = "center">
        <th> </th>
        <th> Column Label </th>
        <th> Another One </th>
        <th> Still Another One </th>
    </tr>
    <tr>
        <th> align </th>
        <td align = "left"> Left </td>
        <td align = "center"> Center </td>
        <td align = "right"> Right </td>
    </tr>
    <tr>
        <th> <br /> valign <br /> <br /> </th>
        <td> Default </td>
        <td valign = "top"> Top </td>
        <td valign = "bottom"> Bottom </td>
    </tr>
</table>
```



The align and valign attributes

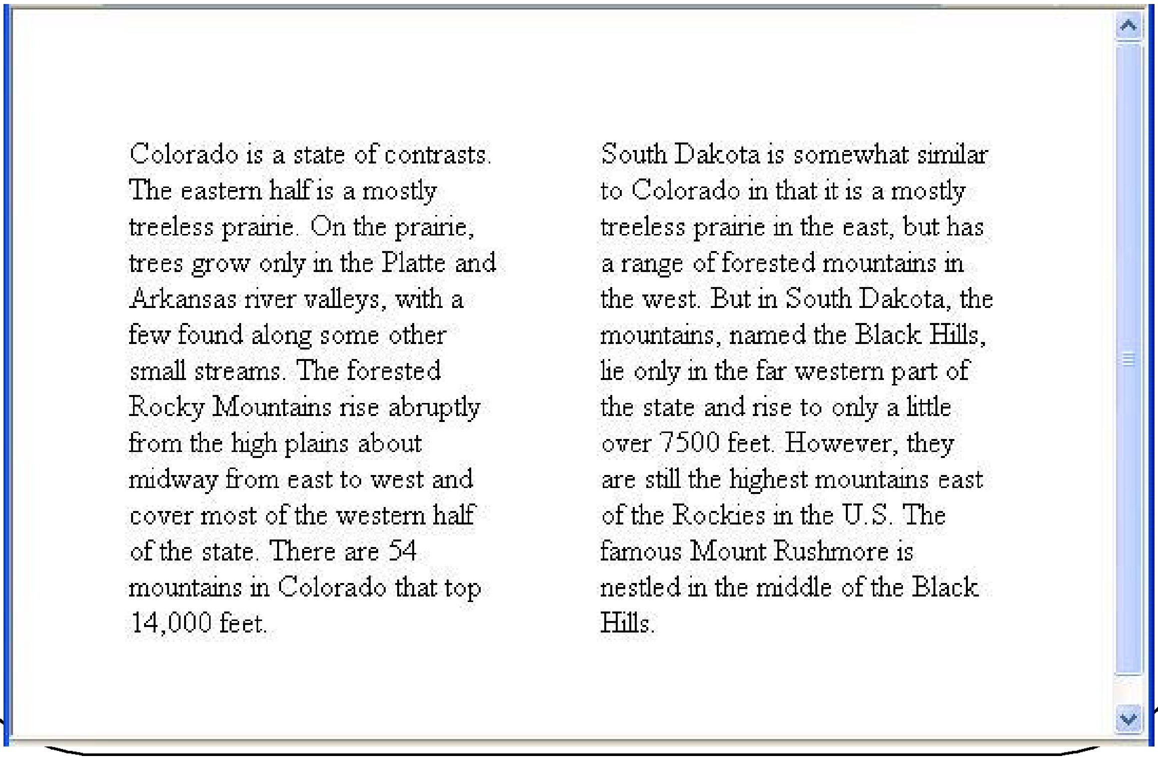
| | Column Label | Another One | Still Another One |
|---|---|---|---|
| **align** | Left | Center | Right |
| **valign** | Default | Top | Bottom |

## 2.8 Tables (continued)

- The `cellspacing` attribute of `<table>` is used to specify the distance between cells in a table

- The `cellpadding` attribute of `<table>` is used to specify the spacing between the content of a cell and the inner walls of the cell

```
<table cellspacing = "50">
   <tr>
      <td> Colorado is a state of …
      </td>
      <td> South Dakota is somewhat…
      </td>
   </tr>
</table>
```



| | |
|---|---|
| Colorado is a state of contrasts. The eastern half is a mostly treeless prairie. On the prairie, trees grow only in the Platte and Arkansas river valleys, with a few found along some other small streams. The forested Rocky Mountains rise abruptly from the high plains about midway from east to west and cover most of the western half of the state. There are 54 mountains in Colorado that top 14,000 feet. | South Dakota is somewhat similar to Colorado in that it is a mostly treeless prairie in the east, but has a range of forested mountains in the west. But in South Dakota, the mountains, named the Black Hills, lie only in the far western part of the state and rise to only a little over 7500 feet. However, they are still the highest mountains east of the Rockies in the U.S. The famous Mount Rushmore is nestled in the middle of the Black Hills. |

# 2.9 Frames

- Frames are rectangular sections of the display window, each of which can display a different document

- One common use of frames is to have a list of links in a left frame and use the right frame to display the destination document of the chosen link

- The `<frameset>` tag specifies the number of frames and their layout in the window

  - `<frameset>` takes the place of `<body>`
    - Cannot have both!

  - `<frameset>` must have either a `rows` attribute or a `cols` attribute, or both (usually the case)
    - Default is 1

  - The possible values for `rows` and `cols` are numbers, percentages, and asterisks

    - A number value specifies the row height in pixels - Not terribly useful!

    - A percentage specifies the percentage of total window height for the row - Very useful!

## 2.9 Frames (continued)

- An asterisk after some other specification gives the remainder of the height of the window

- Examples:

```
 <frameset rows = "150, 200, 300">

 <frameset rows = "25%, 50%, 25%">

 <frameset rows = "50%, 20%, *" >

 <frameset rows = "50%, 25%, 25%"
           cols = "40%, *">
```

- The `<frame>` tag specifies the content of a frame

- The first `<frame>` tag in a `<frameset>` specifies the content of the first frame, etc.
    - Row-major order is used

    - Frame content is specified with the `src` attribute

    - Without a `src` attribute, the frame will be empty (such a frame CANNOT be filled later)

- If `<frameset>` has fewer `<frame>` tags than frames, the extra frames are empty

## 2.9 Frames (continued)

- Scrollbars are implicitly included if needed (they are needed if the specified document will not fit)

- If a `name` attribute is included, the content of the frame can be changed later (by selection of a link in some other frame)

```
<!-- frames.html
     An example to illustrate frames
     -->
<html>
<head>
<title> Frames </title>
</head>
<frameset cols = "20%, *">
   <frame src = "contents.html” />
   <frame src = "fruits.html"
          name = "descriptions” />
</frameset>
</html>
```

- Note: the `Frameset` standard must be specified in the `DOCTYPE` declaration (rather than `Strict`)

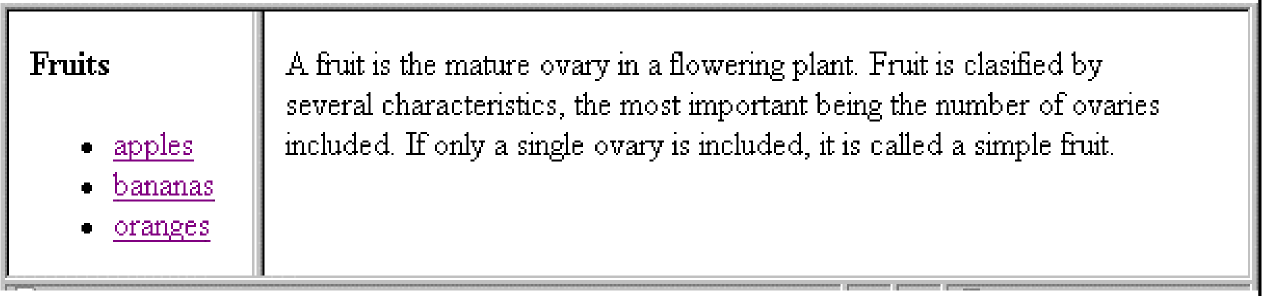
# 2.9 Frames (continued)

```
<!-- contents.html
     The contents of the first frame of
     frames.html, which is the table of
     contents for the second frame
     -->
<html>
<head> <title> Table of Contents Frame </title>
</head>
<body>
<h4> Fruits </h4>
<ul>
   <li> <a href = "apples.html"
           target = "descriptions">
           apples </a>
   <li> <a href = "bananas.html"
           target = "descriptions">
           bananas </a>
   <li> <a href = "oranges.html"
           target = "descriptions">
           oranges </a>
</ul>
</body>
</html>
```

**Note: This use of frames is deprecated because of some potential problems with it**

- **This document must specify `Transitional`, rather than `Strict`**

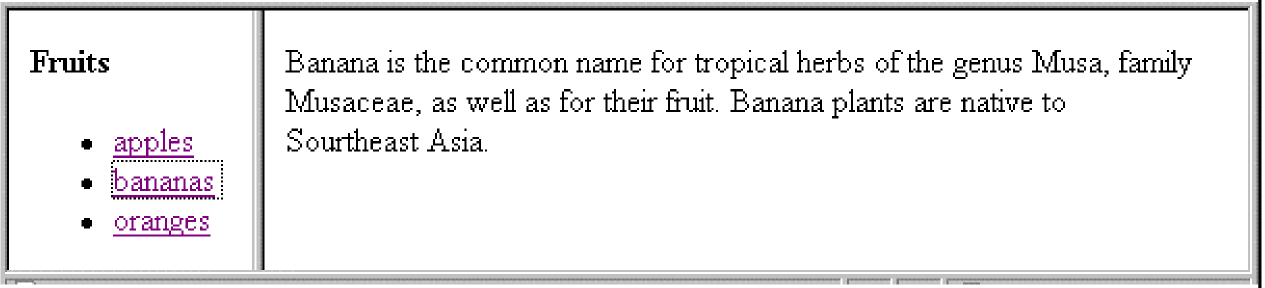
# 2.9 Frames (continued)



**Fruits**

- apples
- bananas
- oranges

A fruit is the mature ovary in a flowering plant. Fruit is clasified by several characteristics, the most important being the number of ovaries included. If only a single ovary is included, it is called a simple fruit.



**Fruits**

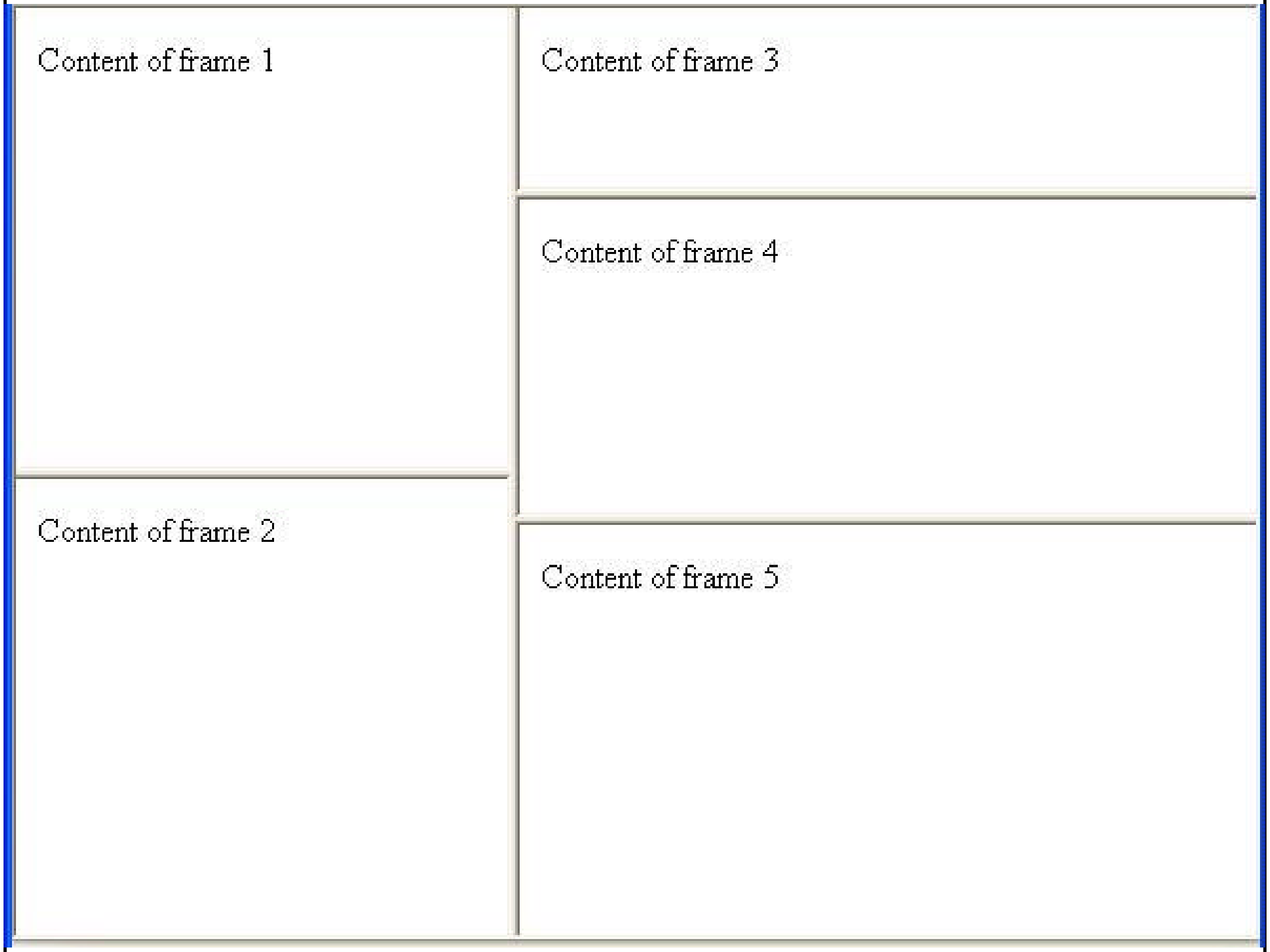
- apples
- bananas
- oranges

Banana is the common name for tropical herbs of the genus Musa, family Musaceae, as well as for their fruit. Banana plants are native to Sourtheast Asia.

- Nested frames - to divide the screen in more interesting ways

```
<frameset cols = "40%, *">
    <frameset rows = "50%, *">
        <frame src = "frame1.html" />
        <frame src = "frame2.html" />
    </frameset>
    <frameset rows = "20%, 35%, *">
        <frame src = "frame3.html" />
        <frame src = "frame4.html" />
        <frame src = "frame5.html" />
    </frameset>
</frameset>
```

# 2.9 Frames (continued)



Content of frame 1

Content of frame 2

Content of frame 3

Content of frame 4

Content of frame 5

# 2.10 Forms

- A form is the usual way information is gotten from a browser to a server

- HTML has tags to create a collection of objects that implement this information gathering

  - The objects are called *widgets* (e.g., radio buttons and checkboxes)

- When the Submit button of a form is clicked, the form's values are sent to the server

- All of the widgets, or components of a form are defined in the content of a `<form>` tag

  - The only required attribute of `<form>` is `action`, which specifies the URL of the application that is to be called when the Submit button is clicked

```
action =
  "http://www.cs.ucp.edu/cgi-bin/survey.pl"
```

  - If the form has no action, the value of `action` is the empty string

# 2.10 Forms (continued)

- The `method` attribute of `<form>` specifies one of the two possible techniques of transferring the form data to the server, `get` and `post`

  - `get` and `post` are discussed in Chapter 10

- *Widgets*

  - Many are created with the `<input>` tag

    - The `type` attribute of `<input>` specifies the kind of widget being created

    1. Text

       - Creates a horizontal box for text input

       - Default size is 20; it can be changed with the `size` attribute

       - If more characters are entered than will fit, the box is scrolled (shifted) left

## 2.10 Forms (continued)

- If you don’t want to allow the user to type more characters than will fit, set `maxlength`, which causes excess input to be ignored

```
<input type = "text" name = "Phone"
       size = "12" >
```

2. *Checkboxes* - to collect multiple choice input

- Every checkbox requires a `value` attribute, which is the widget’s value in the form data when the checkbox is ‘checked’

  - A checkbox that is not ‘checked’ contributes no value to the query string

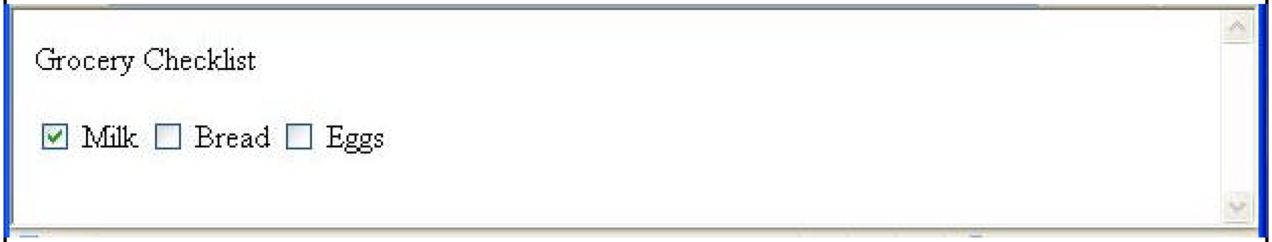
- By default, no checkbox is initially ‘checked’

- To initialize a checkbox to ‘checked’, the `checked` attribute must be set to `"checked"`

# 2.10 Forms (continued)

- *Widgets* (continued)

```
Grocery Checklist
<form action = "">
  <p>
  <input type = "checkbox"  name = "groceries"
         value = "milk"  checked = "checked">
   Milk
  <br/>
  <input type = "checkbox"  name = "groceries"
         value = "bread"> Bread
  <br/>
  <input type = "checkbox"  name = "groceries"
         value= "eggs"> Eggs
  </p>
</form>
```



Grocery Checklist

☑ Milk □ Bread □ Eggs

3. *Radio Buttons* - collections of checkboxes in which only one button can be ‘checked’ at a time

- Every button in a radio button group MUST have the same name

## 2.10 Forms (continued)

- *Widgets* (continued)

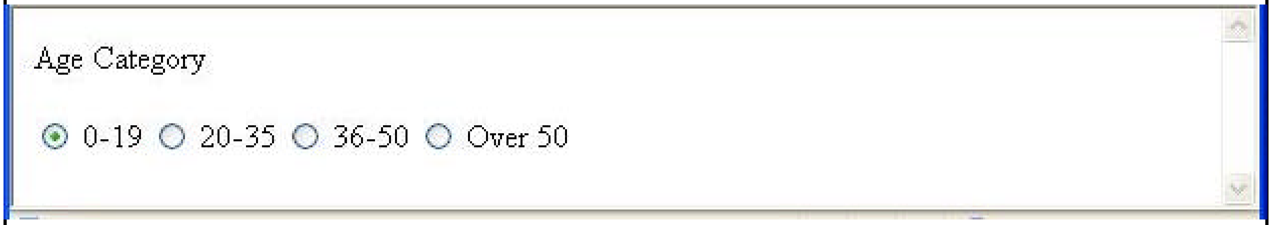
3. *Radio Buttons* (continued)

  - If no button in a radio button group is ‘pressed’, the browser often ‘presses’ the first one

```
Age Category
<form action = "">
  <p>
  <input type = "radio"  name = "age"
   value = "under20" checked = "checked"> 0-19
  <br/>
  <input type = "radio"  name = "age"
         value = "20-35"> 20-35
  <br/>
  <input type = "radio"  name = "age"
         value = "36-50"> 36-50
  <br/>
  <input type = "radio"  name = "age"
         value = "over50"> Over 50
  </p>
</form>
```

# 2.10 Forms (continued)

- *Widgets* (continued)



Age Category

● 0-19 ○ 20-35 ○ 36-50 ○ Over 50

4. Menus - created with `<select>` tags

   - There are two kinds of menus, those that behave like checkboxes and those that behave like radio buttons (the default)

     - Menus that behave like checkboxes are specified by including the `multiple` attribute, which must be set to `"multiple"`

   - The `name` attribute of `<select>` is required

   - The `size` attribute of `<select>` can be included to specify the number of menu items to be displayed (the default is 1)

     - If `size` is set to > 1 or if `multiple` is specified, the menu is displayed as a pop-up menu

# 2.10 Forms (continued)

- *Widgets* (continued)

4. Menus (continued)

- Each item of a menu is specified with an `<option>` tag, whose pure text content (no tags) is the value of the item

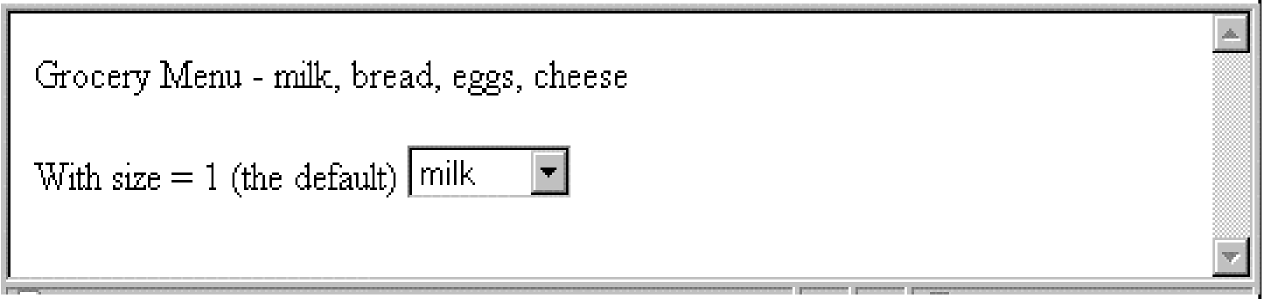
- An `<option>` tag can include the `selected` attribute, which when assigned `"selected"` specifies that the item is preselected

```
Grocery Menu - milk, bread, eggs, cheese

<form action = "">
   <p>
   With size = 1 (the default)
   <select name = "groceries">
         <option> milk </option>
         <option> bread </option>
         <option> eggs </option>
         <option> cheese </option>
   </select>
   </p>
</form>
```

# 2.10 Forms (continued)

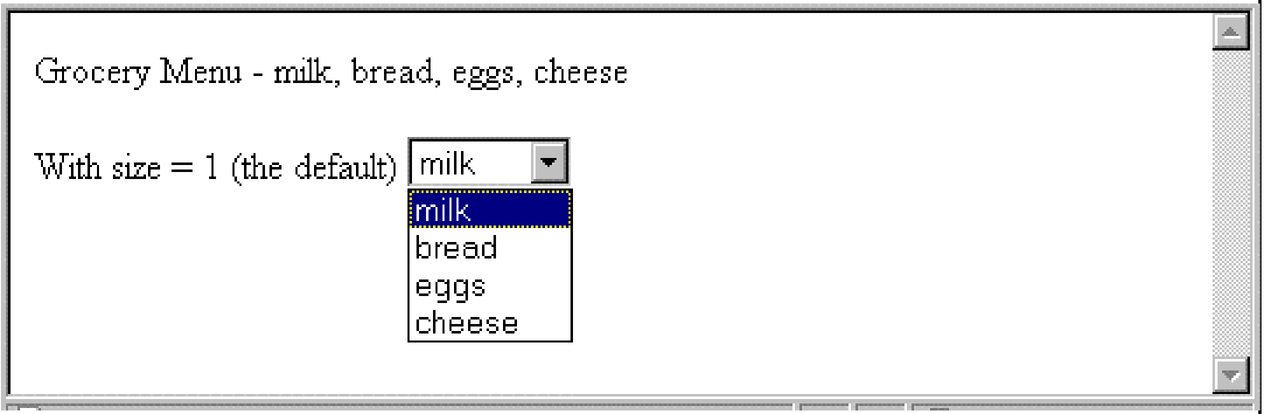
## - *Widgets* (continued)



Grocery Menu - milk, bread, eggs, cheese

With size = 1 (the default) milk

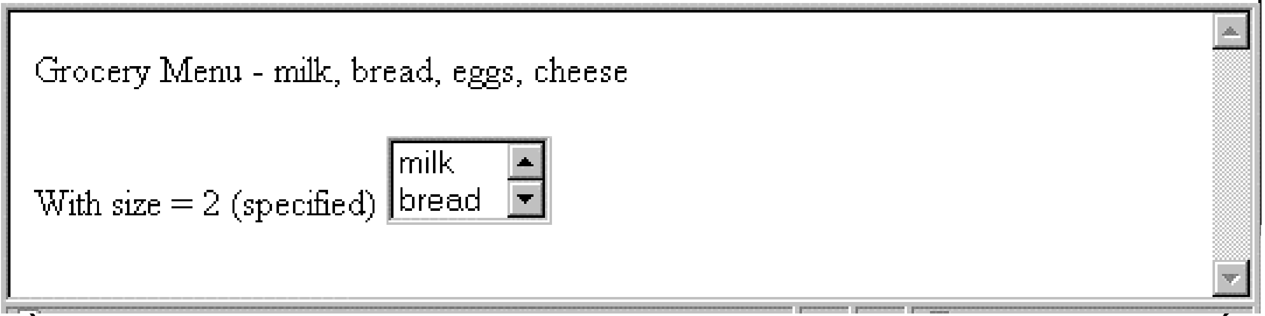
## - After clicking the menu:



Grocery Menu - milk, bread, eggs, cheese

With size = 1 (the default) milk

## - After changing `size` to 2:



Grocery Menu - milk, bread, eggs, cheese

With size = 2 (specified)

## 2.10 Forms (continued)

**- *Widgets* (continued)**

**5. Text areas - created with `<textarea>`**

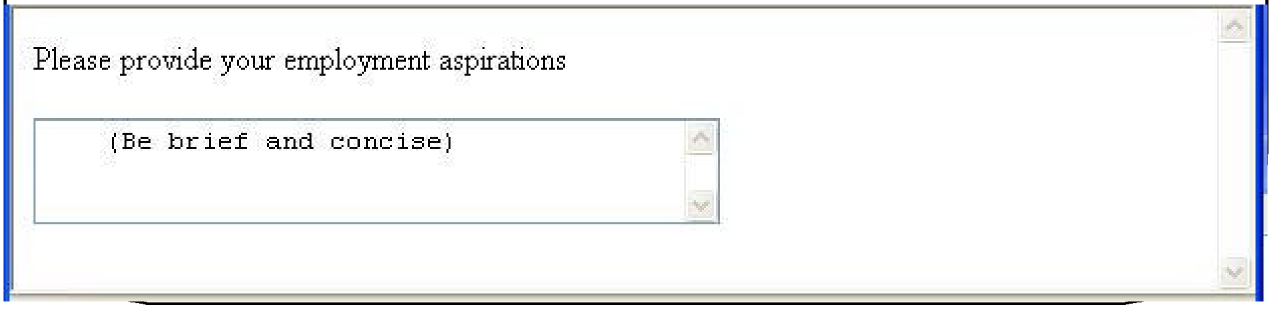
- Usually include the `rows` and `cols` attributes to specify the size of the text area

- Default text can be included as the content of `<textarea>`

- Scrolling is implicit if the area is overfilled

```
Please provide your employment aspirations

<form action = "">
   <p>
   <textarea name = "aspirations"  rows = "3”
             cols = "40">
   (Be brief and concise)
   </textarea>
   </p>
</form>
```



Please provide your employment aspirations

(Be brief and concise)

## 2.10 Forms (continued)

- *Widgets* (continued)

6. Reset and Submit buttons

    - Both are created with `<input>`

```
<input type = "reset"  value = "Reset Form">
<input type = "submit″  value = "Submit Form">
```

    - Submit has two actions:

        1. Encode the data of the form

        2. Request that the server execute the server-resident program specified as the value of the `action` attribute of `<form>`

    - A Submit button is required in every form

--> SHOW `popcorn.html`

--> SHOW Figure 2.29 (a browser display of the form)