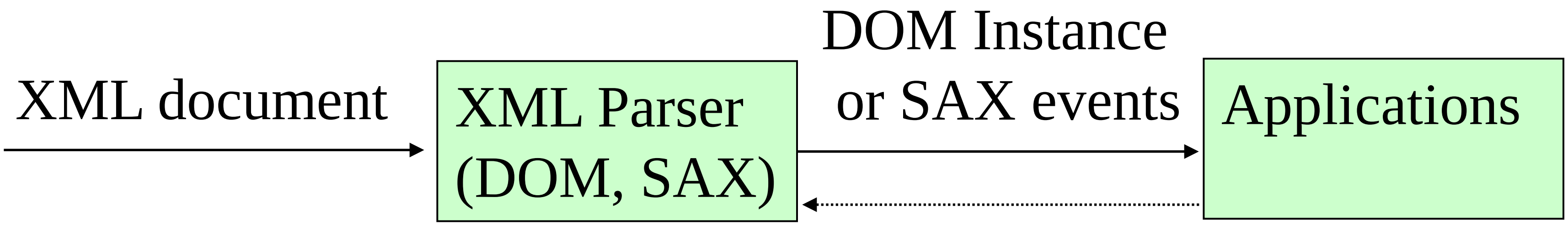
CSC309: Introduction to Web Programming

Lecture 3

Wael Aboulsaadat

Programming APIs



- DOM
  http://www.w3.org/DOM/
  http://www.w3schools.com/dom/

- SAX
  http://www.saxproject.org/

SAX: Simple API for XML

- http://www.saxproject.org/
- Event based
- Instead of reading the entire file in memory and building a tree, SAX reads a stream of tokens and triggers events, e.g.,
  - startDocument
  - startElement
  - endElement
  - endDocuments
  - Characters
- Applications write handlers for events.
- Supports “document order” access to data
- Read-only access, No update-in-place

SAX: an example

```
<imdb>
    <show year=“1993”>
        <title>Fugitive, The</title>
        <review>
            <suntimes>
                <reviewer>Roger
            Ebert</reviewer> gives
                    <rating>two thumbs
                        up</rating>
                    ! A fun action movie,
            Harrison Ford at his best.
            </suntimes>
         </review>
      …..
```

```
startElement(“imdb”, null)
  startElement(“show”, (“year”, “1993”))
      startElement(“title”,)
      characters(“Fugitive, The”)
      endElement(“title”)
      startElement(“review”, null)
          startElement(“suntimes”, null)
               startElement(“reviewer”)
               characters(“Roger Ebert”)
               endElement(“reviewer”)
               characters(“ gives ”) ...
               startElement(“rating”, null)
               characters(“two thumbs up”)
               endElement(“rating”)
                characters(“! A fun movie…”)
          endElement(“suntimes”)
      endElement(“review”)  ….
```

How can you find all show titles using a SAX parser?

DOM vs SAX

- DOM
  - □ XML represents a tree model and DOM is very natural to understand.
  - □ Supports navigation to document
  - □ Enable dynamic update, add, and delete to document content

- SAX
  - □ Lightweight
  - □ Good for applications that read large XML documents once.

    E.g. filter stock quotes, network alerts, load XML documents into storage systems

Link to XML parsers download site

- http://xml.apache.org/
- Xerces projects implement DOM and SAX parser in Java, C++, Perl.
  - http://xml.apache.org/xerces2-j/
  - http://xml.apache.org/xerces-c/
  - http://xml.apache.org/xerces-p/

XML Standards Landscape

- Schema languages
  - □ DTDs
  - □ XML Schemas
- Programming APIs
  - □ DOM
  - □ SAX
- Query languages
  - □ XPath
  - □ XQuery
  - □ XSLT

Common Querying Tasks

- Filter, select XML values
  - Navigation, selection, extraction
- Merge, integrate values from multiple XML sources
  - Joins, aggregation
- Transform XML values from one schema to another
  - XML construction

XML Query Languages

- XPath
  - □ http:www.w3.org/TR/xpath
  - □ Language for navigation, selection, extraction
  - □ Used in XSLT, XQuery, XPointer, Xlink, XML Schema, et al

- XQuery
  - □ http://www.w3.org/TR/xquery/
  - □ Strongly-typed query language
  - □ Additional join, transformation and construction ability

- XSLT
  - □ http://www.w3.org/TR/xslt
  - □ Transform XML to XML, HTML, Text

XHTML

What is HTML?

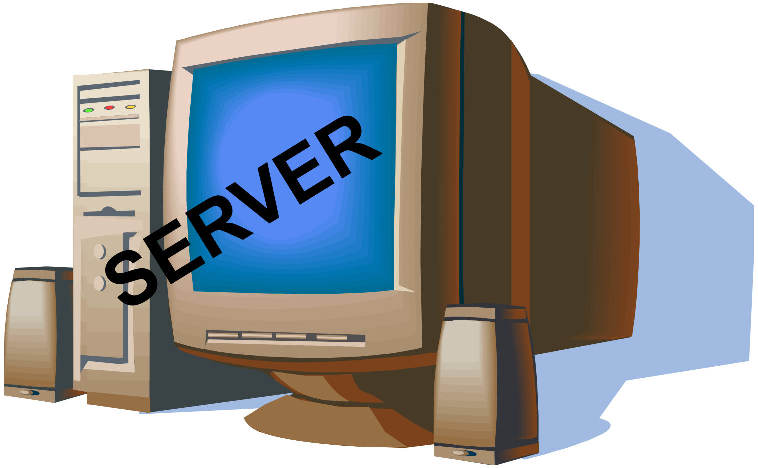
- Key part of the original proposal for the Web
- Subset of SGML
  - Standard Generalized Markup Language – used to prepare computer generated documents for publishing on a printing press
- HTML documents are straight ASCII text documents. Not a proprietary document format
  - Straight ASCII code to keep the size of the file down.

HTML, continued

- Can be read and created by simple text editors
  - Notepad on Windows platform
  - SimpleText on Macintosh platform
  - vi on UNIX (or Linux) platform
- Not a programming language
  - HTML is a markup language in the sense of printing and publishing
  - You markup your document and a Web browser interprets your markup code.
- HTML is platform independent

HTML is a Client-Side Application

- Specifies the structural aspects of a document
- Leaves the question of how to display the different elements in the document entirely up to the Web browser
- So document might look different in different browsers.
- This keeps the HTML files relatively small.



Server Computer is only responsible for conveying the structural components of the document.

The Client Computer determines the formatting characteristics.

Factors that Can Influence the Display of Your Web Page

The user’s;

- Browser
- Monitor Size
- Current Monitor Resolution
- Installed Fonts
- Browser Window Size
- Viewing Hardware; laptop, palm device, etc.
- Operating System (Mac and PCs interpret colors differently to the screen)

XHTML

- HTML is being replaced by a new language → XHTML
- Very similar to HTML
- STRICT coding standards, so that many different types of browsers can display the information.
- For this course we will follow those rules

XHTML Rules

- All tag and attribute names must be in lowercase
- Empty tags must be written with an extra slash at the end
- You can never omit an end tag
- Attributes must always have a value.
- Attribute values must always be quoted

HTML Document Structure

```
<html>
     <head>
          <title> Page Title Goes Here </title>
     </head>

     <body>
          content goes here
     </body>
</html>
```

Background attributes in <body>

Bgcolor

- Specifies a background-color for a HTML page.

  <body bgcolor="#000000">

  <body bgcolor="rgb(0,0,0)">

  <body bgcolor="black">

Background

- Specifies a background-image for a HTML page

  <body background="clouds.gif">

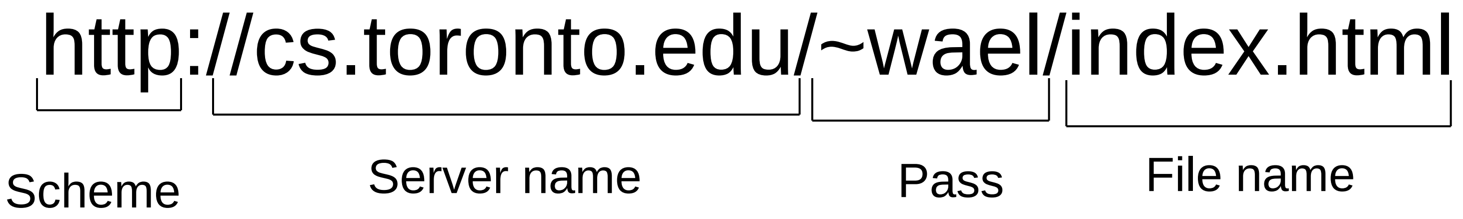
  <body background="http://www.w3schools.com/clouds.gif">

Linking

- Links inserted using the A (anchor) element
  - Requires HREF attribute which specifies the URL you would like to link to
    - <A HREF = “*address*”>…</A>
    - Can link to email addresses, using
    - <A HREF = “mailto: *emailaddress*”>…</A>
    - Note quotation mark placement
- Example:

<a href=“http://cs.toronto.edu/~wael”>Wael</a>

Uniform Resource Locator (URL)



http://cs.toronto.edu/~wael/index.html

Scheme

Server name

Pass

File name

URL can be

- Absolute – contain all parts of URL;
- Relative – present path and file name relatively current file.

Scheme

- http – Hypertext Transfer Protocol to access Web-pages
- ftp – File Transfer Protocol to download the file from the net
- mailto – to send electronic mail
- File – to access file on a local hard disk (File scheme uses ///).
- and others…

Relative URL (examples)

- A file from the same folder as current file:
  - “file.html”
- A file from a subfolder of current folder:
  - “images/picture.gif”
- A file from another folder at the same hierarchical level:
  - “../info/data.html”
- same conventions as in UNIX!

Links inside document: anchors

- Place the cursor in the desirable part of a page, where the link should bring visitors

1) Create an anchor

<A NAME=“*anchor_name*”>*Label text*</A>

Label text is a text or image that should be referenced, i.e. where the link should bring the visitor to.

2) Link to the anchor, use

- □ <A HREF=“#*anchor_name*”>*Label text* </A> or
- □ <A HREF=“*URL*#anchor_name”>*Label text* </A>

Text

- Put text on a webpage
  - □ <p>Today is my first day at my new job, I’m so excited!</p>
  - □ Output: Today is my first day at my new job, I’m so excited!
- Put text in center of a page
  - □ <center>Hello</center>
  - □ Output: Hello
- Put text on the right of a page
  - □ <p align=“right”>Hello</p>
  - □ Output: Hello

Text Styling

- Underline style
  - <U>…</U>
- Emphasis (italics) style
  - <EM>…</EM>
- Strong (bold) style
  - <STRONG>…</STRONG>
- Superscript: <SUP> element
- Subscript: <SUB> element

Font

- To change text size
  - □ <font size=“+3”>Hello</font>
  - □ Output: Hello
- To change text color
  - □ <font color=“red”>Hello</font>
  - □ Output: Hello
- Using both
  - □ <font size=“+3” color=“red”>Hello</font>
  - □ Output: Hello

Headings

- There are 6 heading commands.

<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>

Character Entities

Result	Description	Entity Name
	Non-breaking space	&nbsp;
<	Less than	&lt;
>	Greater than	&gt;
&	Ampersand	&amp;
“	Quotation mark	&quot;
©	Copyright	&copy;

E.g. Ampersand in URL

- <!-- This is invalid! -->

  <a href=“http://www.wc.com/?chapter=1&section=2">...</a>

- This example generates an error for "unknown entity section" because the "&" is assumed to begin an entity.

- To avoid problems with both browsers, always use &amp; in place of &:

<a href=“http://www.wc.com/?chapter=1&amp;section=2">...</a>

List

- Unordered list
  - □ Code:

    ```
    <ul>
    <li>Coffee</li>
    <li>Milk</li>
    </ul>
    ```

  - □ Output:
    - Coffee
    - Milk

- Ordered list
  - □ Code:

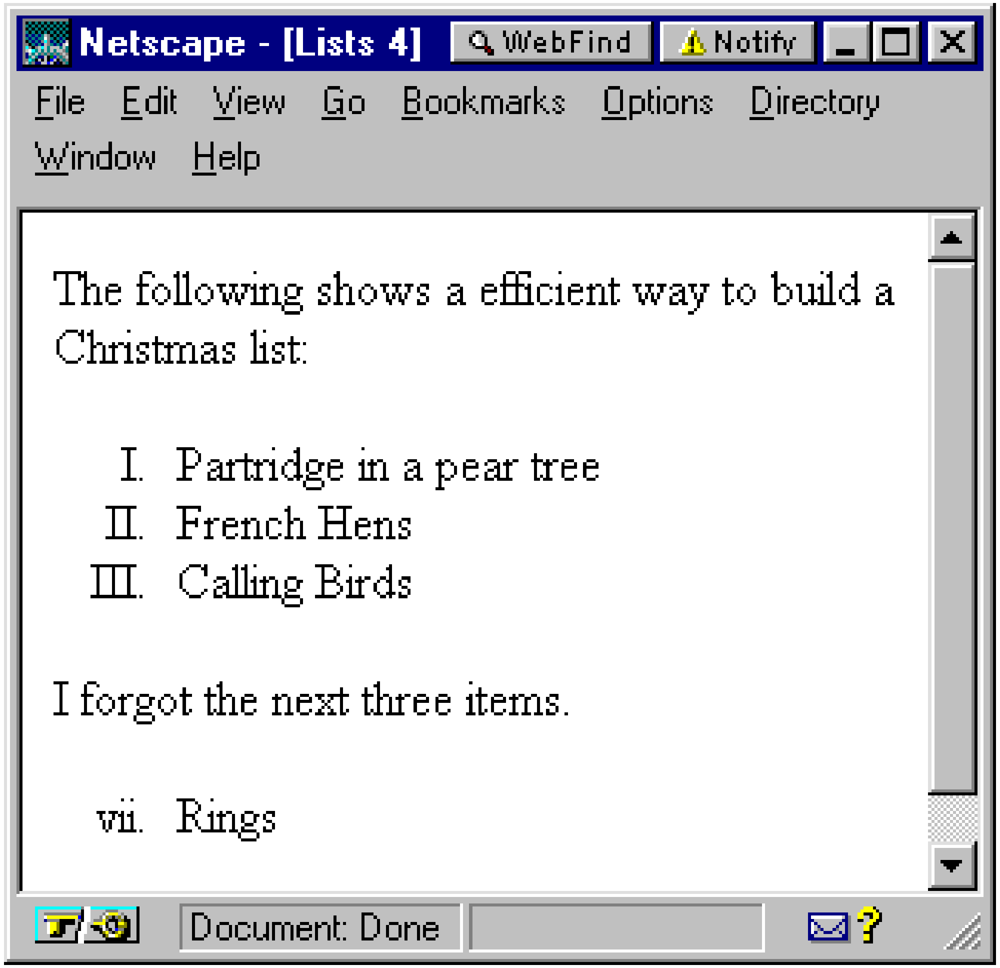
    ```
    <ol>
    <li>Coffee</li>
    <li>Milk</li>
    </ol>
    ```

  - □ Output:
    1. Coffee
    2. Milk

<OL> TYPE Attribute

Example

```
<HTML>
<HEAD>
<TITLE>Lists 4 </TITLE>
</HEAD>
<BODY>
The following shows a efficient way to build
a Christmas list:
<OL TYPE=I>
<LI>Partridge in a pear tree
<LI>French Hens
<LI>Calling Birds
</OL>
I forgot the next three items.
<OL TYPE=i START=7>
<LI>Rings
</OL>
</BODY>
</HTML>
```



The following shows a efficient way to build a Christmas list:

I. Partridge in a pear tree
II. French Hens
III. Calling Birds

I forgot the next three items.

vii. Rings

Tables

A table is a matrix formed by the intersection of a number of horizontal rows and vertical columns.

	Column 1	Column 2	Column 3
Row 1			
Row 2			
Row 3			

Tables (continue…)

The intersection of a column and row is called a cell. Cells in the same row or column are usually logically related in some way.

	Column 1	Column 2	Column 3
Row 1	Cell	Cell	Cell
Row 2	Cell	Cell	Cell
Row 3	Cell	Cell	Cell

Tables (continue…)

Container

<TABLE> … </TABLE>

Attributes:

BORDER= n – the border thickness in pixels

WIDTH=x – width of the table or a cell within the table in pixels or relative size to the screen display (0% to 100%)

Tables (continue…)

- A table is formed row by row
- To define a row

  <TR>…</TR>

  is used
- Within a row table cells with data is determined by

  <TD>…</TD>

  or with headers by

  <TH>…</TH>

Table

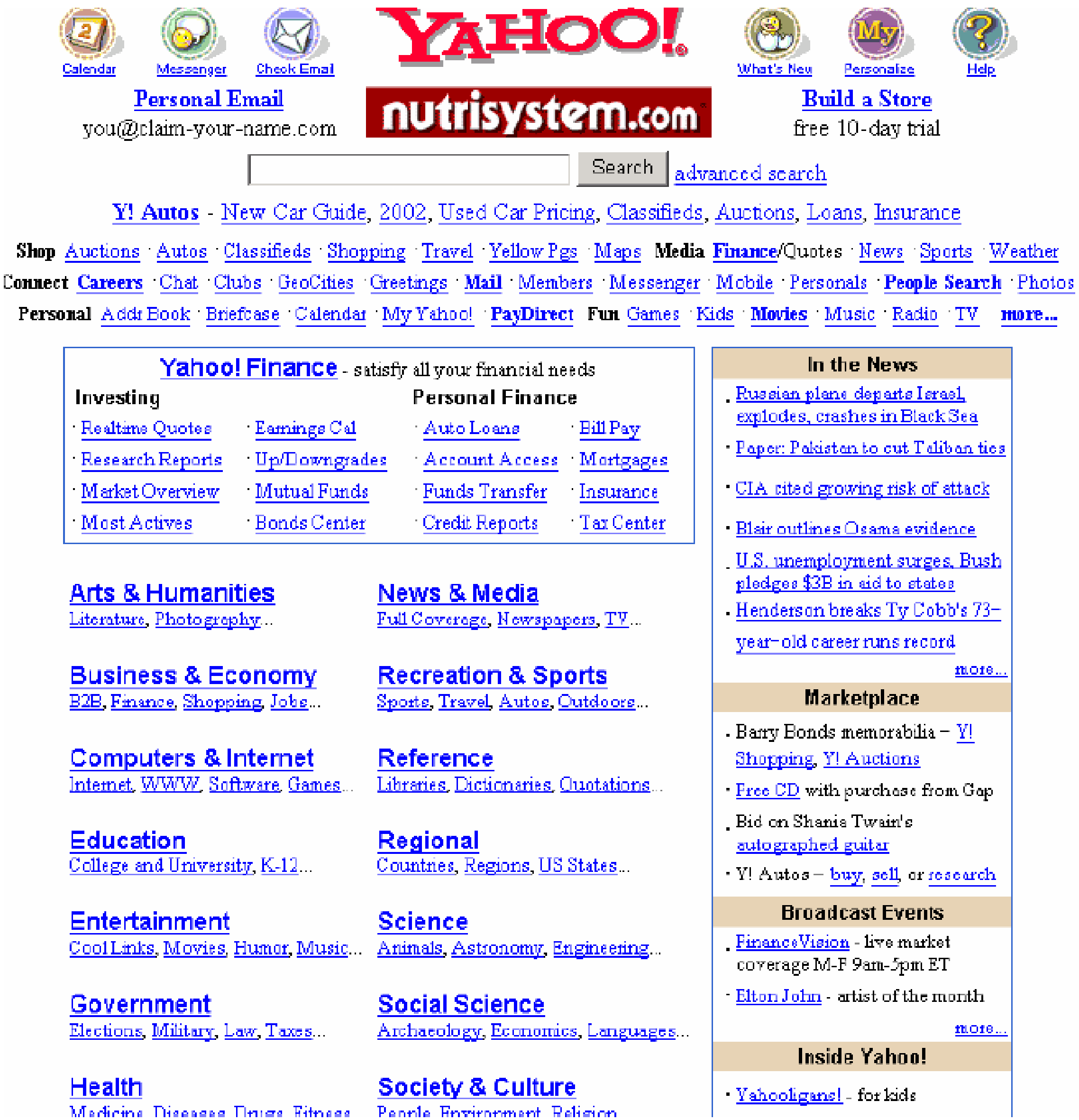
```
<table border=“1">
<tr>
      <th>Heading</th>
      <th>Another Heading</th>
</tr>
<tr>
      <td>row 1, cell 1</td>
      <td>row 1, cell 2</td>
</tr>
<tr>
      <td>row 2, cell 1</td>
      <td></td>
</tr>
</table>
```

Heading	Another Heading
Row 1, cell 1	Row 1, cell 2
Row 2, cell 1	

Tables (more complicated)

- To span a cell across a few columns, use the attribute COLSPAN=*n*, where *n* is number of columns is used

- To span a cell across a few rows use the attribute ROWSPAN=*n*, where *n* number of rows is used



Personal Email
you@claim-your-name.com

Build a Store
free 10-day trial

advanced search

Y! Autos - New Car Guide, 2002, Used Car Pricing, Classifieds, Auctions, Loans, Insurance

Shop Auctions · Autos · Classifieds · Shopping · Travel · Yellow Pgs · Maps **Media** Finance/Quotes · News · Sports · Weather

Connect Careers · Chat · Clubs · GeoCities · Greetings · Mail · Members · Messenger · Mobile · Personals · People Search · Photos

Personal Addr Book · Briefcase · Calendar · My Yahoo! · PayDirect **Fun** Games · Kids · Movies · Music · Radio · TV · more...

Yahoo! Finance - satisfy all your financial needs

Investing		Personal Finance	
· Realtime Quotes	· Earnings Cal	· Auto Loans	· Bill Pay
· Research Reports	· Up/Downgrades	· Account Access	· Mortgages
· Market Overview	· Mutual Funds	· Funds Transfer	· Insurance
· Most Actives	· Bonds Center	· Credit Reports	· Tax Center

Arts & Humanities
Literature, Photography...

Business & Economy
B2B, Finance, Shopping, Jobs...

Computers & Internet
Internet, WWW, Software, Games...

Education
College and University, K-12...

Entertainment
Cool Links, Movies, Humor, Music...

Government
Elections, Military, Law, Taxes...

Health
Medicine, Diseases, Drugs, Fitness...

News & Media
Full Coverage, Newspapers, TV...

Recreation & Sports
Sports, Travel, Autos, Outdoors...

Reference
Libraries, Dictionaries, Quotations...

Regional
Countries, Regions, US States...

Science
Animals, Astronomy, Engineering...

Social Science
Archaeology, Economics, Languages...

Society & Culture
People, Environment, Religion...

In the News

- Russian plane departs Israel, explodes, crashes in Black Sea
- Paper: Pakistan to cut Taliban ties
- CIA cited growing risk of attack
- Blair outlines Osama evidence
- U.S. unemployment surges, Bush pledges $3B in aid to states
- Henderson breaks Ty Cobb's 73-year-old career runs record

more...

Marketplace

- Barry Bonds memorabilia – Y! Shopping, Y! Auctions
- Free CD with purchase from Gap
- Bid on Shania Twain's autographed guitar
- Y! Autos – buy, sell, or research

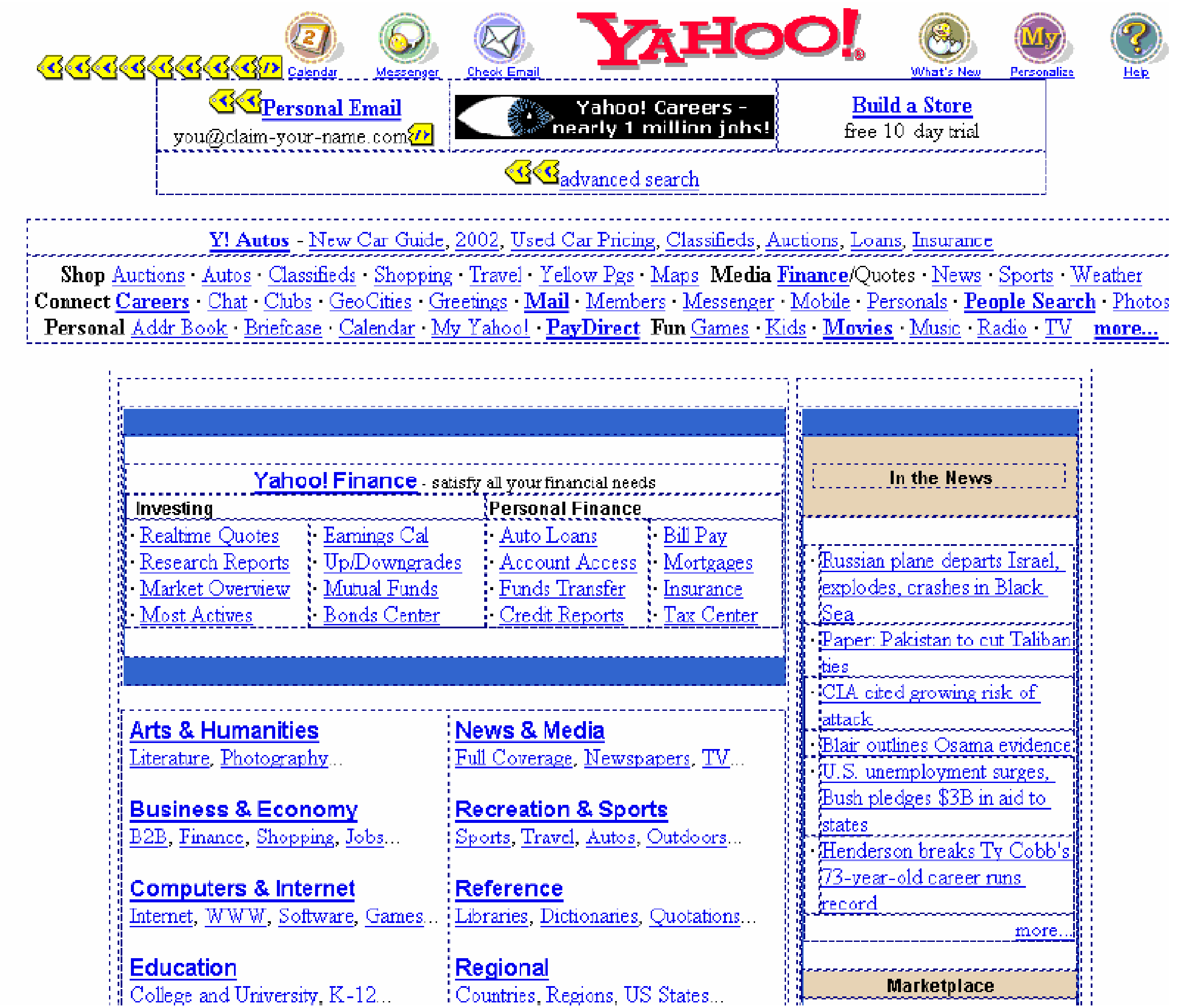
Broadcast Events

- FinanceVision - live market coverage M-F 9am-5pm ET
- Elton John - artist of the month

more...

Inside Yahoo!

- Yahooligans! - for kids



Calendar · Messenger · Check Email · YAHOO! · What's New · Personalize · Help

Personal Email
you@claim-your-name.com

Yahoo! Careers - nearly 1 million jobs!

Build a Store
free 10 day trial

advanced search

Y! Autos - New Car Guide, 2002, Used Car Pricing, Classifieds, Auctions, Loans, Insurance

Shop Auctions · Autos · Classifieds · Shopping · Travel · Yellow Pgs · Maps **Media** Finance/Quotes · News · Sports · Weather
Connect Careers · Chat · Clubs · GeoCities · Greetings · Mail · Members · Messenger · Mobile · Personals · People Search · Photos
Personal Addr Book · Briefcase · Calendar · My Yahoo! · PayDirect **Fun** Games · Kids · Movies · Music · Radio · TV more...

Yahoo! Finance - satisfy all your financial needs

Investing		Personal Finance	
· Realtime Quotes	· Earnings Cal	· Auto Loans	· Bill Pay
· Research Reports	· Up/Downgrades	· Account Access	· Mortgages
· Market Overview	· Mutual Funds	· Funds Transfer	· Insurance
· Most Actives	· Bonds Center	· Credit Reports	· Tax Center

Arts & Humanities
Literature, Photography...

Business & Economy
B2B, Finance, Shopping, Jobs...

Computers & Internet
Internet, WWW, Software, Games...

Education
College and University, K-12...

News & Media
Full Coverage, Newspapers, TV...

Recreation & Sports
Sports, Travel, Autos, Outdoors...

Reference
Libraries, Dictionaries, Quotations...

Regional
Countries, Regions, US States...

In the News

- Russian plane departs Israel, explodes, crashes in Black Sea
- Paper: Pakistan to cut Taliban ties
- CIA cited growing risk of attack
- Blair outlines Osama evidence
- U.S. unemployment surges, Bush pledges $3B in aid to states
- Henderson breaks Ty Cobb's 73-year-old career runs record

more...

Marketplace

Cell Attributes

- FONT – establishes the font of a cell
- ALIGN – determines horizontal alignment of cell content, accept values:

  “left”, “center”, or “right”
- VALIGN - determines vertical alignment of cell content, accept values:

  “top”, “middle”, “bottom”, or “base line”

Purposes to use tables

- To present tabular data
- To create multicolumn text
- To create captions for images
- To create side bars

Cells may contain various HTML containers:

Images, Hyperlinks, Text, Objects, even Tables

Image Formats

- .gif
  - Graphics Interchange Format
- .jpeg or .jpg
  - Joint Photographic Experts Group
- .png
  - Portable Network Graphics

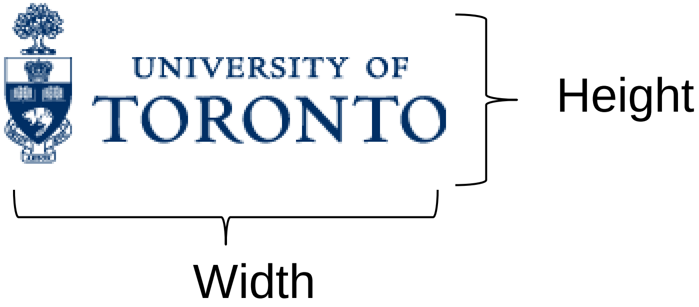
Inserting Image

- Place all images in the same directory/folder where you web pages are
- <img src> is a single tag
- <img src=“image.gif”>
  - □ Output:

- Turn an image into a hyerlink
  - □ <a href=“http://www.utoronto.ca”><img src=“image.gif”></a>
  - □ Output:

Image Size

- Computer images are made up of “pixels”
- <IMG HEIGHT=“100" WIDTH=“150" SRC="image.gif">



Forms

- <form> is just another kind of HTML tag

- HTML forms are used to create (rather primitive) GUIs on Web pages
  - □ Usually the purpose is to ask the user for information
  - □ The information is then sent back to the server

- A form is an area that can contain form elements
  - □ The syntax is: <form ***parameters***> ***...form elements...*** </form>
  - □ Form elements include: buttons, checkboxes, text fields, radio buttons, drop-down menus, etc
    - Other kinds of HTML tags can be mixed in with the form elements
  - □ A form usually contains a Submit button to send the information in he form elements to the server
  - □ The form's ***parameters*** tell JavaScript how to send the information to the server (there are two different ways it could be sent)
  - □ Forms can be used for other things, such as a GUI for simple programs

Forms and JavaScript

- The JavaScript language can be used to make pages that “do something”
  - You *can* use JavaScript to write complete programs, but...
  - Usually you just use snippets of JavaScript here and there throughout your Web page
  - JavaScript code snippets can be attached to various form elements
    - For example, you might want to check that a zipcode field contains a 5-digit integer before you send that information to the server
- Microsoft calls its version of JavaScript “active scripting”
- HTML forms can be used without JavaScript, and JavaScript can be used without HTML forms, but they work well together
- JavaScript for HTML is covered in a separate lecture

The <form> tag

- The <form ***arguments***> ... </form> tag encloses form elements (and probably other HTML as well)
- The arguments to form tell what to do with the user input
  - □ action="***url***" (required)
    - Specifies where to send the data when the Submit button is clicked
  - □ method="get" (default)
    - Form data is sent as a URL with ?form_data info appended to the end
    - Can be used *only* if data is all ASCII and not more than 100 characters
  - □ method="post"
    - Form data is sent in the body of the URL request
    - Cannot be bookmarked by most browsers
  - □ target="***target***"
    - Tells where to open the page sent as a result of the request
    - ***target=*** _blank means open in a new window
    - ***target=*** _top means use the same window

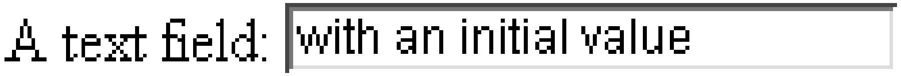
The <input> tag

- Most, but not all, form elements use the input tag, with a type="…" argument to tell which kind of element it is
  - type can be text, checkbox, radio, password, hidden, submit, reset, button, file, or image
- Other common input tag arguments include:
  - name: the name of the element
  - value: the "value" of the element; used in different ways for different values of type
  - readonly: the value cannot be changed
  - disabled: the user can't do anything with this element
  - Other arguments are defined for the input tag but have meaning only for certain values of type

Text input

A text field:

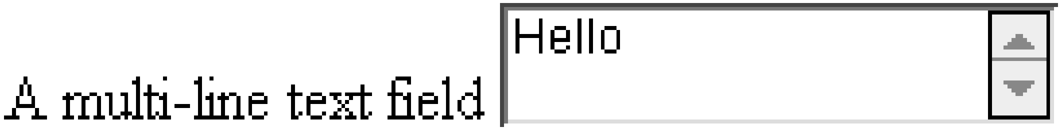
```
<input type="text" name="textfield" value="with an initial value">
```



A text field: with an initial value

A multi-line text field

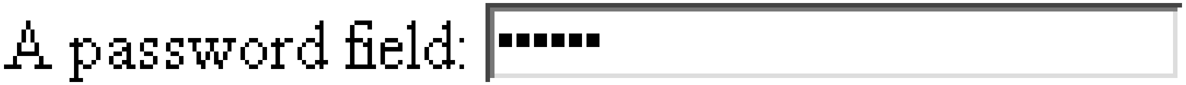
```
<textarea name="textarea" cols="24" rows="2">Hello</textarea>
```



A multi-line text field Hello

A password field:

```
<input type="password" name="textfield3" value="secret">
```

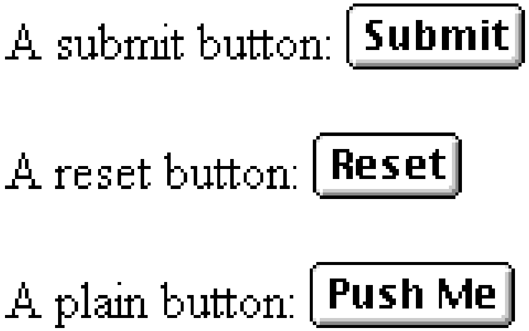


A password field: ■■■■■■

- Note that two of these use the input tag, but one uses textarea

Buttons

- A submit button:
  ```
  <input type="submit" name="Submit" value="Submit">
  ```
- A reset button:
  ```
  <input type="reset" name="Submit2" value="Reset">
  ```
- A plain button:
  ```
  <input type="button" name="Submit3" value="Push Me">
  ```



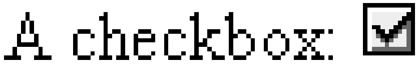
- submit: send data
- reset: restore all form elements to their initial state
- button: take some action as specified by JavaScript

- Note that the type is input, not “button”

Checkboxes

- A checkbox:

```
<input type="checkbox" name="checkbox”
        value="checkbox" checked>
```

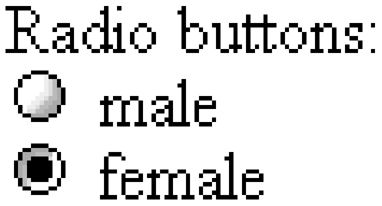


A checkbox: ☑

  - type: "checkbox"
  - name: used to reference this form element from JavaScript
  - value: value to be returned when element is checked
  - Note that there is *no text* associated with the checkbox—you have to supply text in the surrounding HTML

Radio buttons

```
Radio buttons:<br>
    <input type="radio" name="radiobutton" value="myValue1">
    male<br>
    <input type="radio" name="radiobutton" value="myValue2" checked>
    female
```



Radio buttons:

○ male

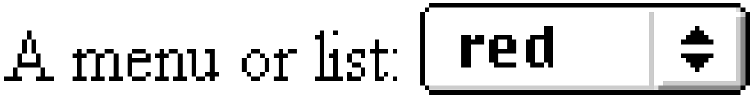
● female

- If two or more radio buttons have the same name, the user can only select one of them at a time
    - This is how you make a radio button “group”
- If you ask for the value of that name, you will get the value specified for the selected radio button
- As with checkboxes, radio buttons do not contain any text

Drop-down menu or list

- A menu or list:

```
<select name="select">
    <option value="red">red</option>
    <option value="green">green</option>
    <option value="BLUE">blue</option>
</select>
```

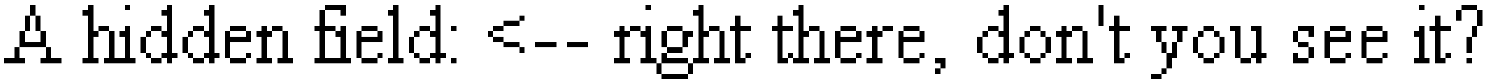


A menu or list: red

- Additional arguments:
  - size: the number of items visible in the list (default is "1")
  - multiple: if set to "true", any number of items may be selected (default is "false")

Hidden fields

- <input type="hidden" name="hiddenField" value="nyah"> &lt;-- right there, don't you see it?

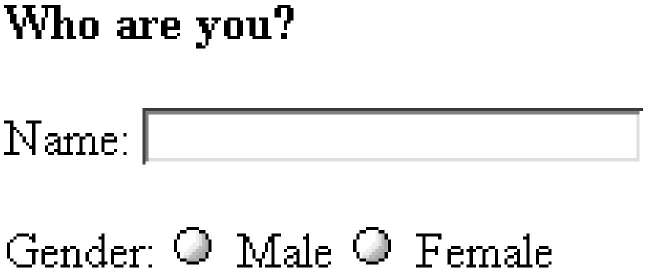


A hidden field: <-- right there, don't you see it?

- What good is this?
  - All input fields are sent back to the server, including hidden fields
  - This is a way to include information that the user doesn’t need to see (or that you don’t want her to see)
  - The value of a hidden field can be set programmatically (by JavaScript) before the form is submitted

A complete example

```
<html>
<head>
<title>Get Identity</title>
<meta http-equiv="Content-Type" content="text/html;
         charset=iso-8859-1">
</head>
<body>
<p><b>Who are you?</b></p>
<form method="post" action="">
  <p>Name:
    <input type="text" name="textfield">
  </p>
  <p>Gender:
    <input type="radio" name="gender" value="m">Male
    <input type="radio" name="gender" value="f">Female</p>
  </form>
</body>
</html>
```



Who are you?

Name:

Gender: ○ Male ○ Female

Hexadecimal Color Codes

- color=“#000000”

#000000–	black
#FFFFFF –	white
#0000FF–	blue (default link color)
#FF0000–	red (default active link color)
#80080 –	purple (default visited link color)

How to publish a web page?

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title></title>
<meta http-equiv="Content-Type" content="text/html;charset=utf-8"/>
</head>
<body>
</body>
</html>
```

How to publish a web page?

- http://validator.w3.org/

How to publish a web page?

- After you assemble a web page and have validated it, it must be transferred to a “server” or a “host”

- Use ftp to put the files on the server
  - in the right directory…

CSS

Cascading style sheets

- Make it easier to have a consistent style on all of your web pages.
- Create a file like "myformat.css" containing various information about font types, sizes, and colors and such.
- Include a reference to it in each html file.

A Basic Example

- Suppose you want to bind a style rule to the <h1> element so that it appears as 28-point text. The following rule would result in the desired display:

```
h1 {font-size: 28pt;}
```

A Basic Example

- More rules such as setting the color to red or the font face to Impact also could be added simply by separating each style property with a semi-colon.

```
h1 {font-size: 28 pt; color: red;
     font-family: Impact;}
```

What does a style sheet look like?

- A style sheet is made up of **rules** that look something like this:

```
h3 {font-family: Arial; font-style: italic;
    color: green;}
```

What does a style sheet look like?

- Each rule begins with a **selector**, which is h3 in this example. A selector is normally one or more HTML elements (tags), and the **declarations** inside the rule describe how those elements should appear. A declaration is simply a CSS **property** followed by a **value**. For example, the declaration "font-style: italic;" is composed of the property "font-style" followed by the value "italic". So, this example states that every <h3> HTML tag should use an Arial italic font and be colored green.

Style Sheet Basics

- You can also use **classes** as selectors, which aren't tied to specific HTML elements. For example, consider this rule:

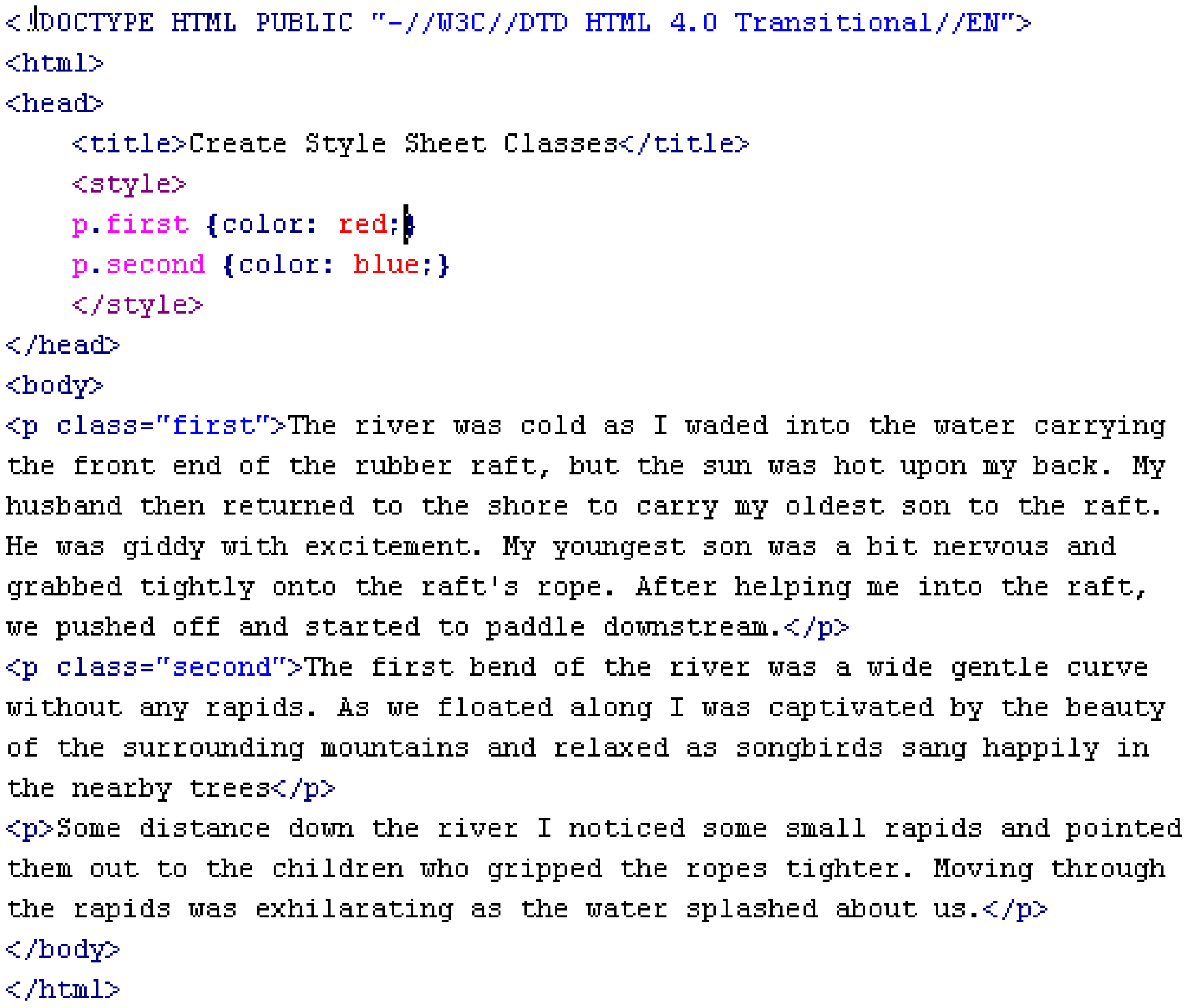
```
.greenitalic {font-family: Arial;
    font-style: italic; color: green;}
```

Style Sheet Basics

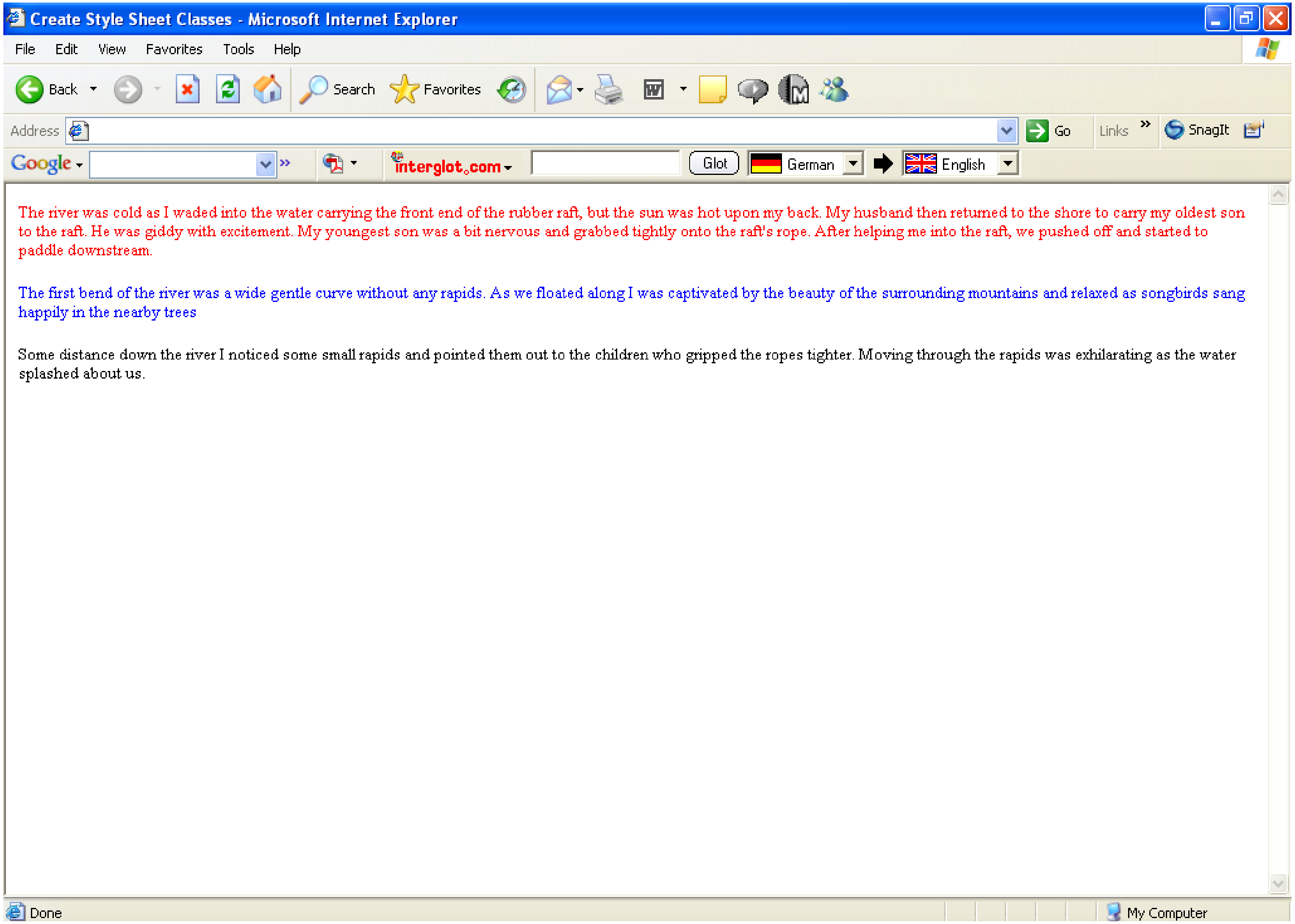
- To apply a class to an HTML tag, you use the the **class** attribute (which was introduced in HTML 4.0). For example, to apply the above style to an <h3> tag, you'd use:

  <h3 class="greenitalic">this is greenitalic<h3>

  (note that the period before the class name is *not* included).



```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
    <title>Create Style Sheet Classes</title>
    <style>
    p.first {color: red;}
    p.second {color: blue;}
    </style>
</head>
<body>
<p class="first">The river was cold as I waded into the water carrying
the front end of the rubber raft, but the sun was hot upon my back. My
husband then returned to the shore to carry my oldest son to the raft.
He was giddy with excitement. My youngest son was a bit nervous and
grabbed tightly onto the raft's rope. After helping me into the raft,
we pushed off and started to paddle downstream.</p>
<p class="second">The first bend of the river was a wide gentle curve
without any rapids. As we floated along I was captivated by the beauty
of the surrounding mountains and relaxed as songbirds sang happily in
the nearby trees</p>
<p>Some distance down the river I noticed some small rapids and pointed
them out to the children who gripped the ropes tighter. Moving through
the rapids was exhilarating as the water splashed about us.</p>
</body>
</html>
```



The river was cold as I waded into the water carrying the front end of the rubber raft, but the sun was hot upon my back. My husband then returned to the shore to carry my oldest son to the raft. He was giddy with excitement. My youngest son was a bit nervous and grabbed tightly onto the raft's rope. After helping me into the raft, we pushed off and started to paddle downstream.

The first bend of the river was a wide gentle curve without any rapids. As we floated along I was captivated by the beauty of the surrounding mountains and relaxed as songbirds sang happily in the nearby trees

Some distance down the river I noticed some small rapids and pointed them out to the children who gripped the ropes tighter. Moving through the rapids was exhilarating as the water splashed about us.

webformat.css

```
BODY {color: black; background-color: #ffe; font-family: arial;
   font-size: 14px}

H1 {color: #039; font-family: arial; font-size: 25px}
H2 {color: #039; font-family: arial; font-size: 22px}
H3 {color: #039; font-family: arial; font-size: 17px}
H4 {color: #039; font-family: arial; font-style: italic;
   font-size: 14px; font-weight: normal}
H4.indent {color: #039; font-family: arial; font-style: italic;
   font-size: 14px; font-weight: normal; margin-left: 14px}
H5 {font-family: arial; font-size: 10px}
A {color: #90C; text-decoration: none}
A:link  {color: #90C; text-decoration: none}
A:visited  {color: #90C; text-decoration: none}
```

And more

```
P.vlarge {font-family: arial; font-size: 20px}
P.large {font-family: arial; font-size: 17px; margin-left: 14px}
P.lgnoindent {font-family: arial; font-size: 17px; margin-left: 0}
P.noindent {font-family: arial; font-size: 14px; margin-left: 0}
P.small {font-family: arial; font-size: 10px; margin-left: 0}
P {color: black; font-family: arial; font-size: 14px; margin-left: 14px}
CODE {color: blue; font-family: courier; font-size: 14px}
CODE.large {color: blue; font-family: courier; font-size: 17px} margin-left: 14px}
```

How to link to CSS file?

```
<html>
<head>
<title>309 CSS Example</title>
<link rel=“stylesheet” type="text/css" href=“webformat.css">
</head>

<body>

<p class=“noindent”>Text not indented </p>

</body>
</html>
```

JavaScript

background

- JavaScript was originally called LiveScript (pre-’95) and was developed by Netscape. It changed names to JavaScript and became a joint venture with Sun. ECMA developed a standard for JavaScript in the late 90s which is also an ISO standard and the official name of the language is ECMAScript. ECMA is at version 3 which corresponds to Netscape’s 1.5.

Parts of JavaScript

- **Core:** operators, expressions, statements and subprograms
- **Client-side:** browser control and user interaction (ie., mouse-clicks, input)
- **Server-side:** language features for use on a server – not covered in this text.

Client-side JavaScript

- XHTML-embedded scripting language.
- Client-side JavaScript still substitutes for some server-side functionality.
- The browser interprets the JavaScript.
- Client-side JavaScript cannot substitute for server-side file networking and database operations.
- Applets – which are also client-side code (albeit delivered as byte-code to a client by a server) can often be replaced by JavaScript.
- Since Applets can access nearly the full java API(s) they are much more powerful than JavaScript, but also harder to learn.

interaction

- Mouseclicks and form data can trigger java script functions.
- DOM allows JavaScript to access and modify the CSS properties and content of any element of a displayed XHTML – enabling dynamic presentation. (text chapter 6)

interpreter

- Browsers use their JavaScript interpreters to process scripts embedded in html.
- Client-side input data checking is an important role for JavaScript. Without this, form information goes to the server, which must locate errors and relay them back to the client.

Placement in head or body of html

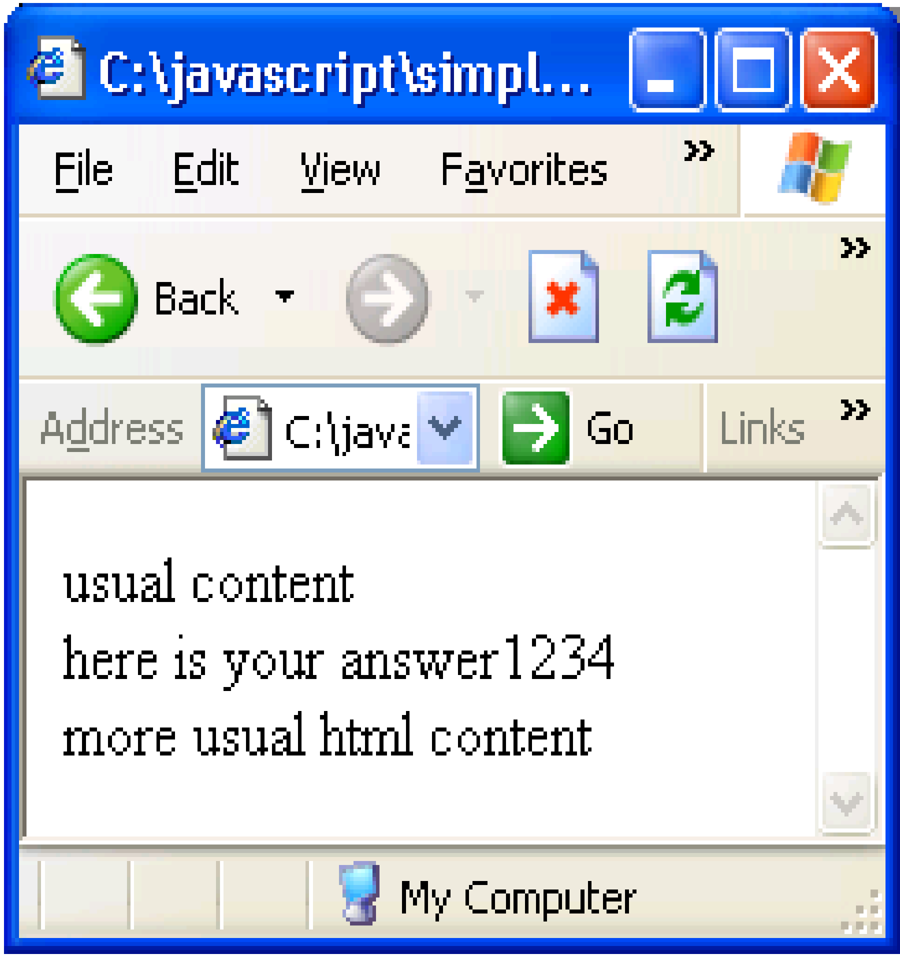
- Typically function definitions or code for handling events is placed in the head (and accessed if/as needed).
- Script meant to be processed just once as part of document content goes in the body. Scripts in the head are cataloged but not interpreted at that time. Scripts in the body are processed as encountered.

output to the html document

- Document.write(“here is your answer”,answer,”<br/>”);
- Basically writes (dynamic) html content

A really simple script

```
<html>
<body>
usual content
<script type="text/javascript">
<!--
var answer=1234;
document.write("<br/>here is
    your answer",answer,"<br/>");
//now end script
// -->
</script>
more usual html content
</body>
</html>
```



usual content

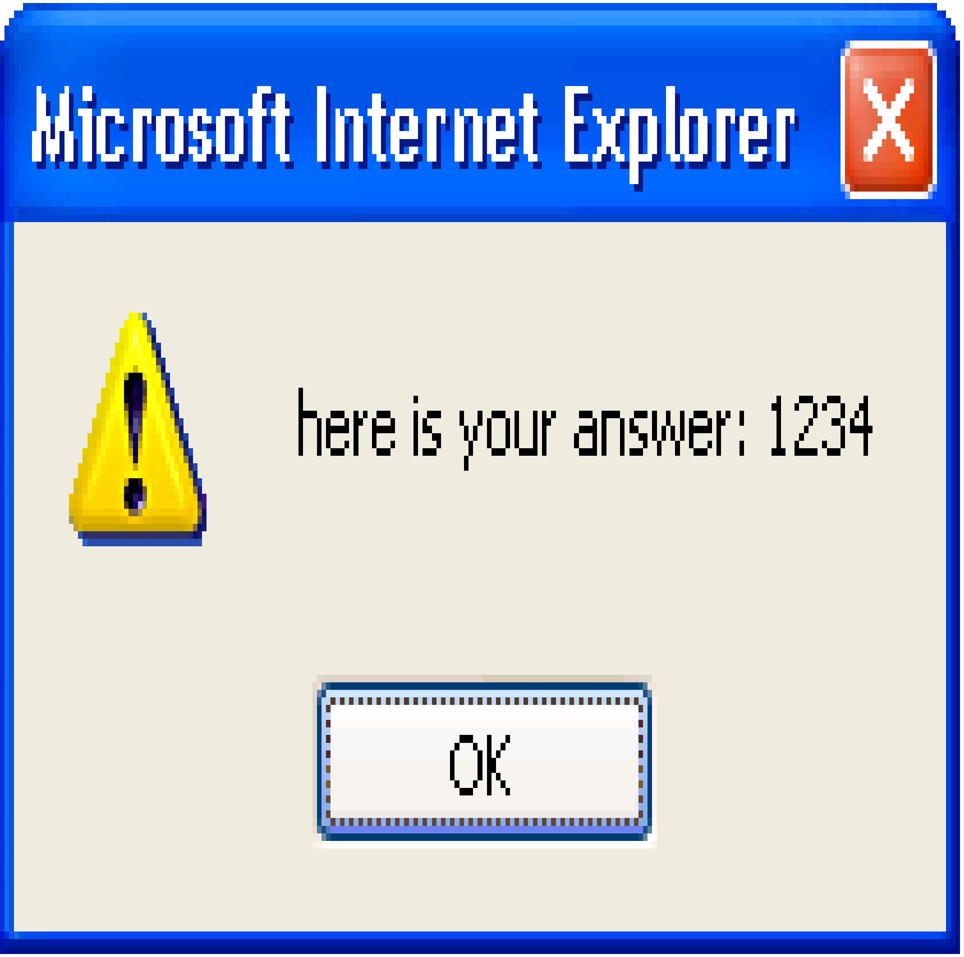
here is your answer1234

more usual html content

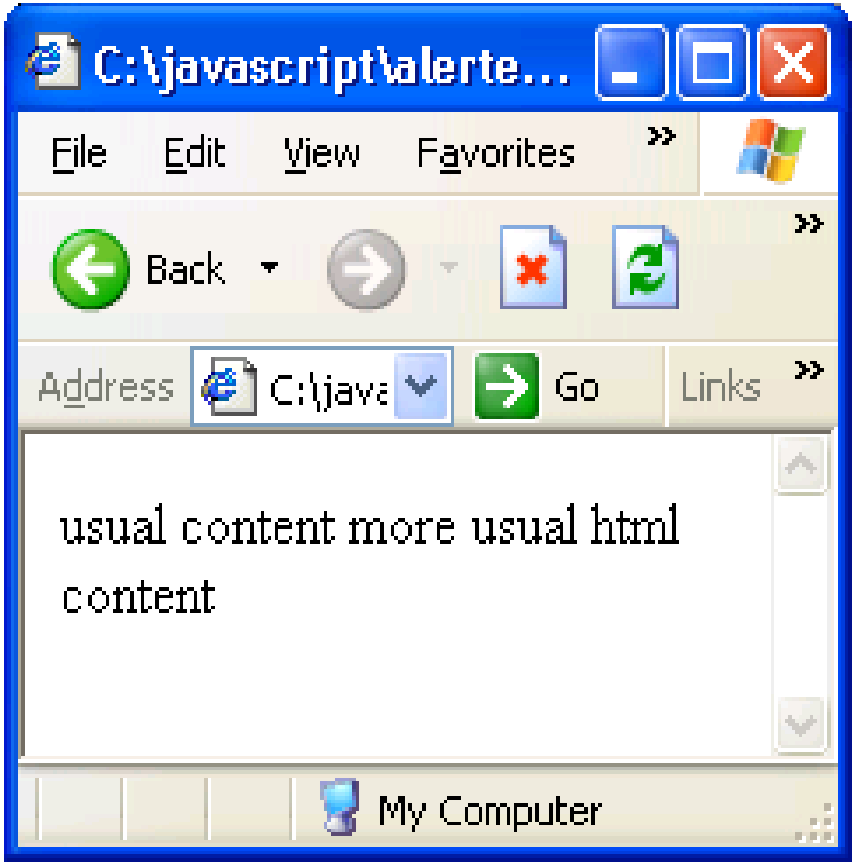
the earlier example using an alert instead of document.write(..)

```
<html>
<body>
usual content
<script type="text/javascript">
<!--
var answer=1234;
alert("here is your answer: " +answer);
// -->
</script>
more usual html content
</body>
</html>
```

First the alert, then the page

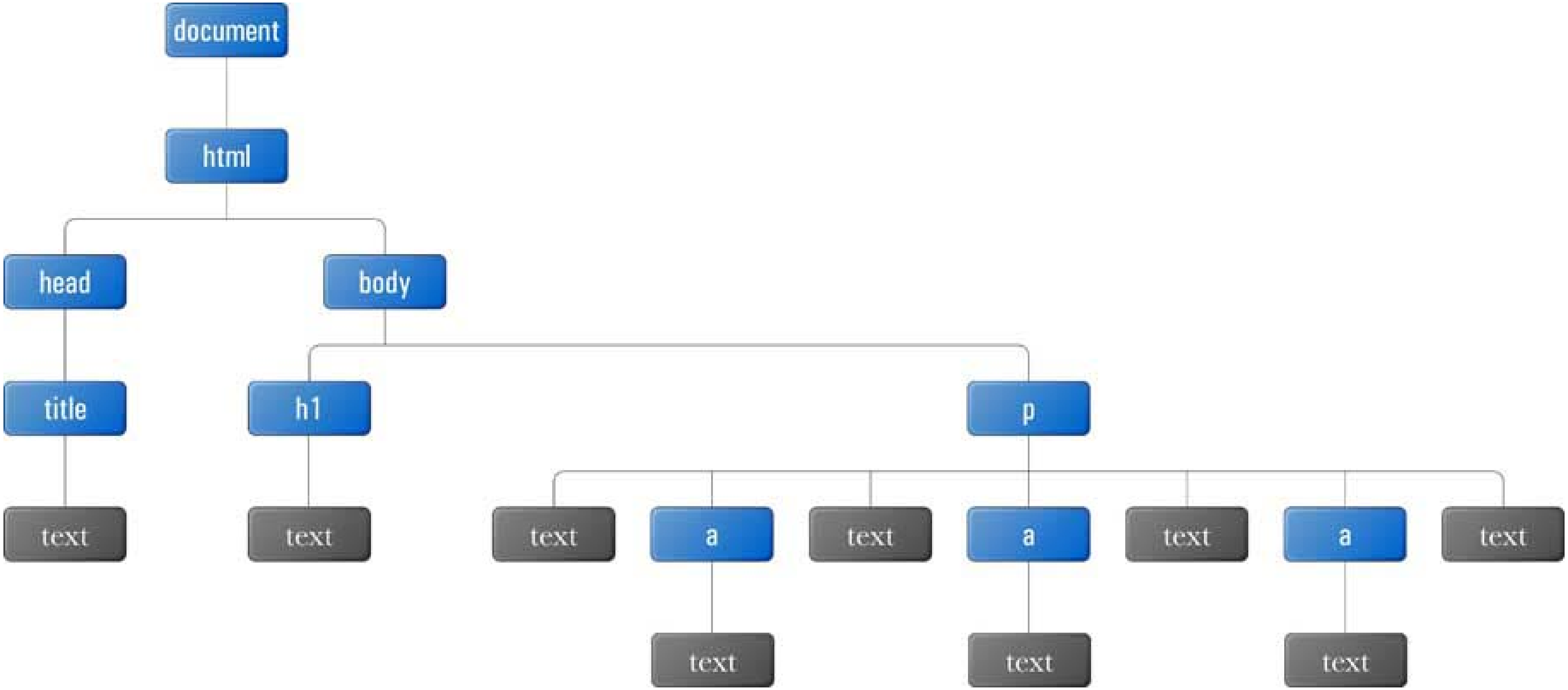


here is your answer: 1234

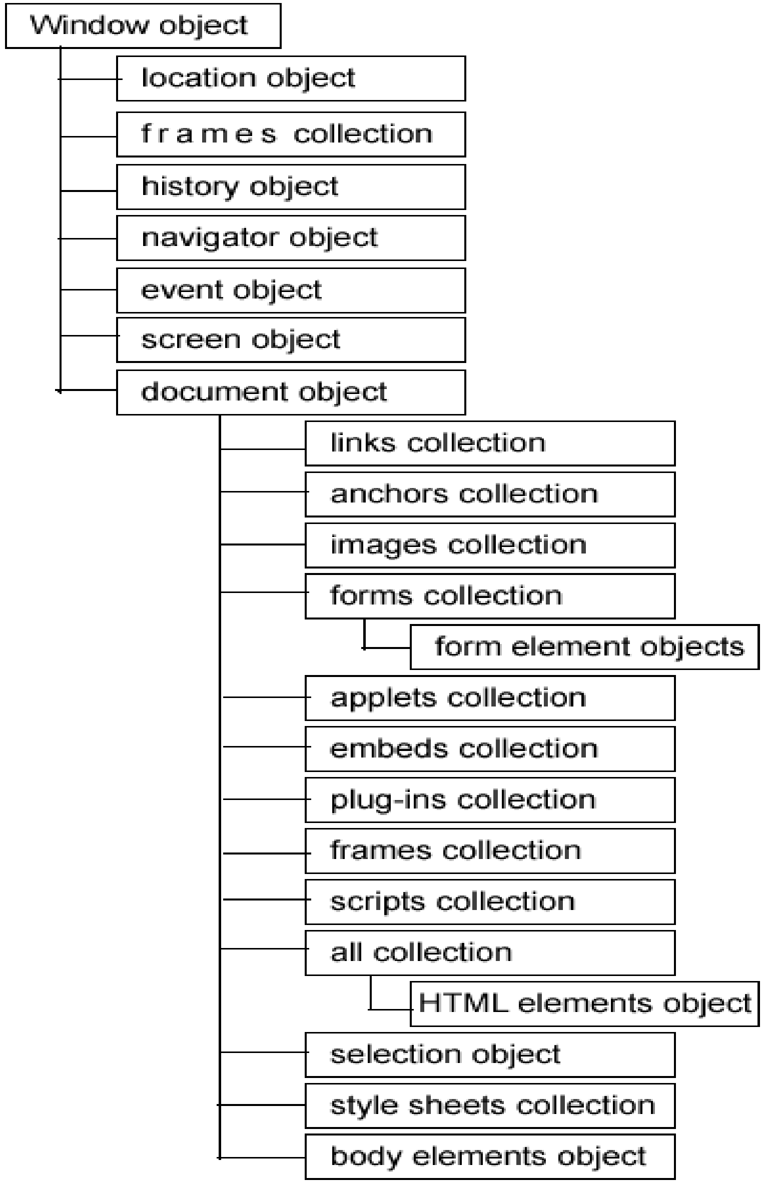


usual content more usual html content

JavaScript Objects



JavaScript Objects



- Window object
  - location object
  - frames collection
  - history object
  - navigator object
  - event object
  - screen object
  - document object
    - links collection
    - anchors collection
    - images collection
    - forms collection
      - form element objects
    - applets collection
    - embeds collection
    - plug-ins collection
    - frames collection
    - scripts collection
    - all collection
      - HTML elements object
    - selection object
    - style sheets collection
    - body elements object