

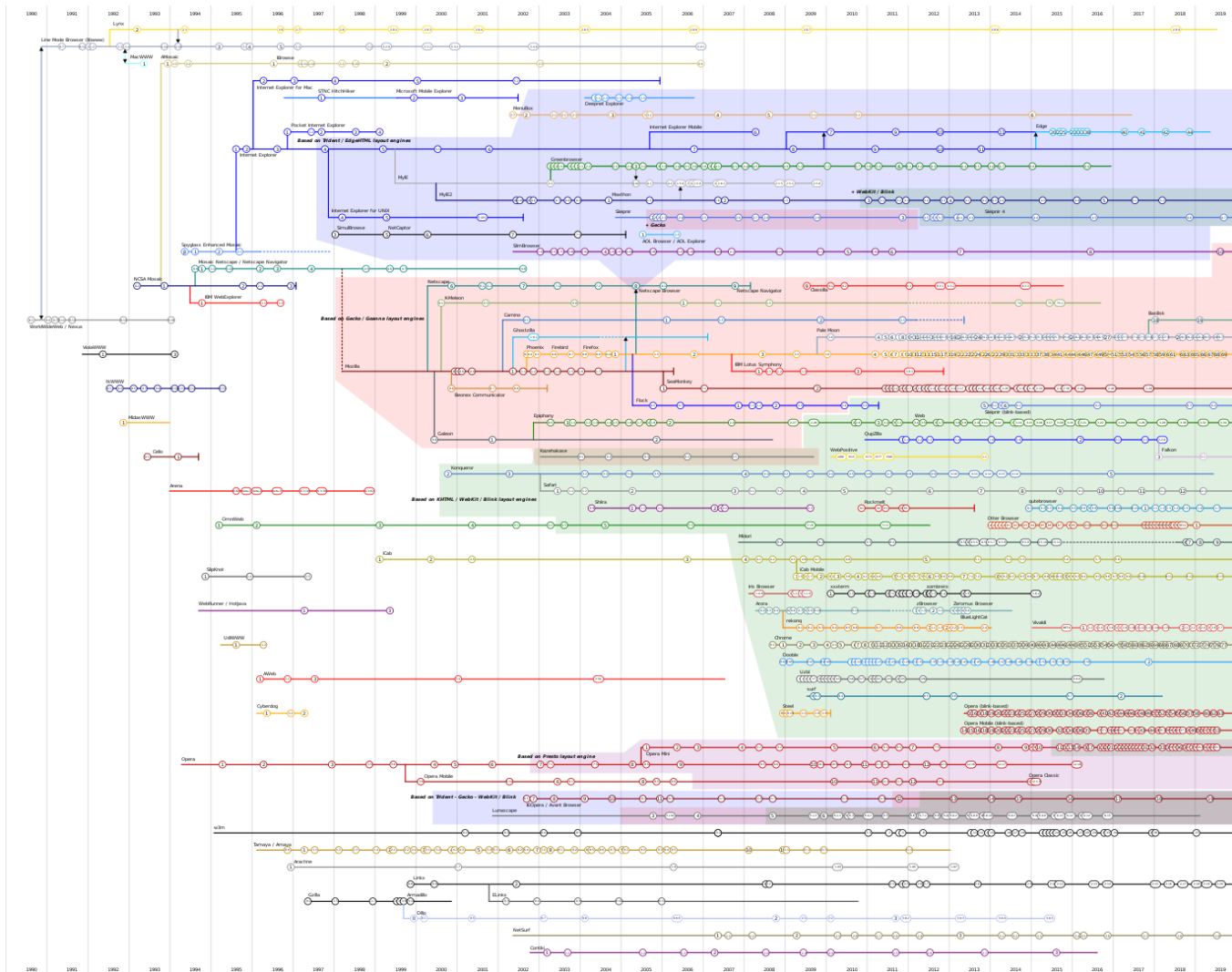
# HTML

# What is HTML

- Hypertext Markup Language (HTML) is a language in which one can describe:
  - The display and format of text
  - The display of graphics
  - Pointers to other html files
  - Pointers to files containing graphics, digitized video and sound
  - Forms that capture information from the viewer
- HTML was developed by Tim Berners-Lee of CERN around 1990
- HTML is understood by WWW browsers—e.g. Internet Explorer, Firefox, Chrome, Safari, many others — which interpret and display the output to the viewer

# List of Web Browsers

[http://en.wikipedia.org/wiki/List\\_of\\_web\\_browsers](http://en.wikipedia.org/wiki/List_of_web_browsers)



# Versions of HTML

- Version 0, 1990, was the original, minimum set of HTML
- Version 1 adds highlighting and images
- Version 2, November 1995, all V.0 and V.1, plus forms
- Version 3.2, January 1997, released by W3CW, tables
- HTML 4.01, December 1999 Recommendation:  
<http://www.w3.org/TR/html401/>
- HTML5, October 2014 Recommendation, vocabulary and APIs:  
<http://www.w3.org/TR/html5/>
- HTML5, December 2017 HTML 5.2 Recommendation:  
<https://www.w3.org/TR/html52/>
- **HTML Living Standard, December 2019:**  
<https://html.spec.whatwg.org>
- W3C and WHATWG Agreement  
<https://www.w3.org/blog/news/archives/7753>
- "W3C stops independent publishing of a designated list of specifications related to HTML and DOM and instead will work to take WHATWG Review Drafts to W3C Recommendations"

NOTE: some of the examples shown here are extracted from the HTML4.0 specification. This document is copyrighted according to: "Copyright © World Wide Web Consortium. All Rights Reserved. <http://www.w3.org/Consortium/Legal/>"

## An Example

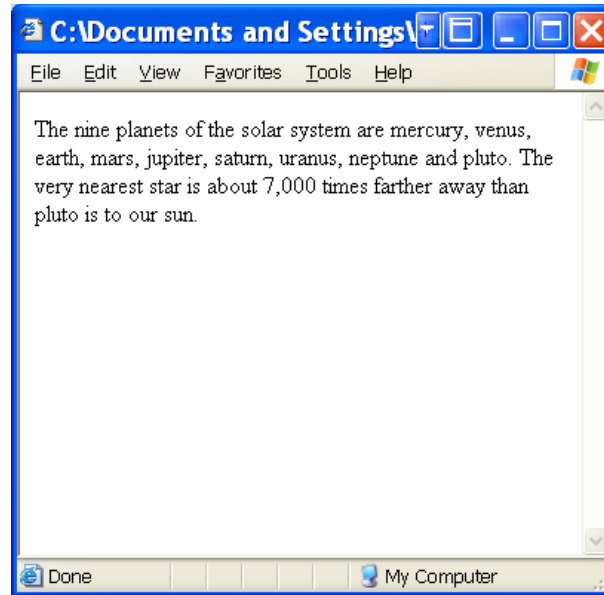
The nine planets of the solar system are  
mercury,  
venus,  
earth,  
mars,  
jupiter, saturn, uranus, neptune and pluto.

The very nearest star is about 7,000 times  
farther away than pluto is to our sun.

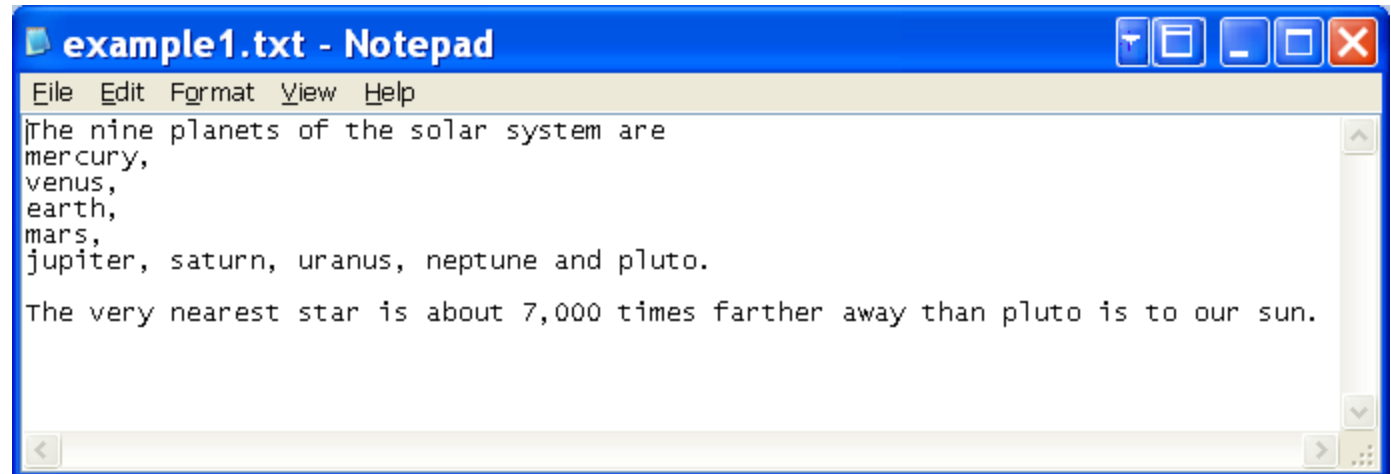
- Save this file as example1.txt
- Save this file as example1.html

# Browser Output - Example 1

example1.html



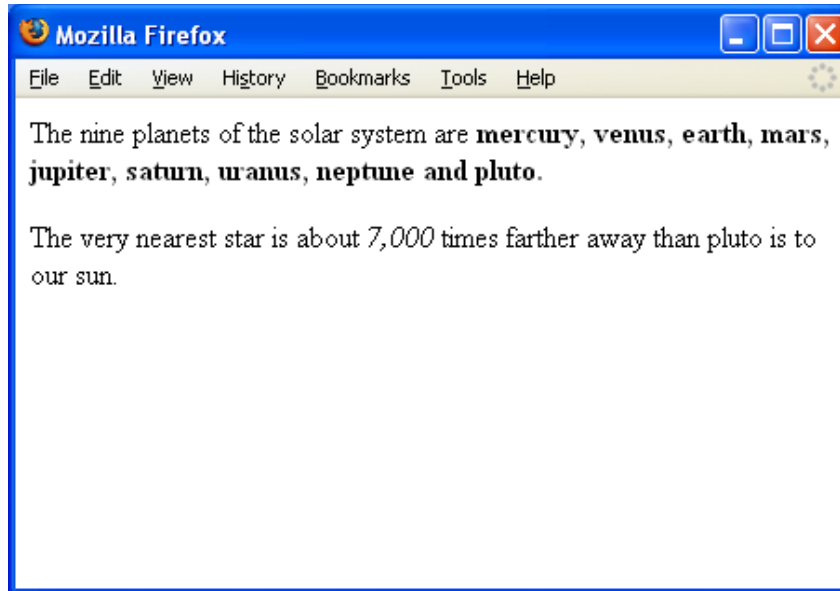
example1.txt



## Example 2: Adding Some Markup

```
<P>The nine planets of the solar system are  
  <B>mercury, venus, earth, mars, jupiter,  
  saturn, uranus, neptune and pluto.</B></P>
```

```
<P>The very nearest star is about  
  <I>7,000</I> times farther away than pluto  
  is to our sun. </P>
```



# General Structure

- HTML documents have a head and body
  - head should contain a title
  - body may have paragraphs
- A leading line indicates which version of HTML this document conforms to

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN"
```

```
"http://www.w3.org/TR/REC-html40/strict.dtd">
```

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>The Solar System</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<P>The nine planets of the solar system are...
```

```
</BODY>
```

```
</HTML>
```

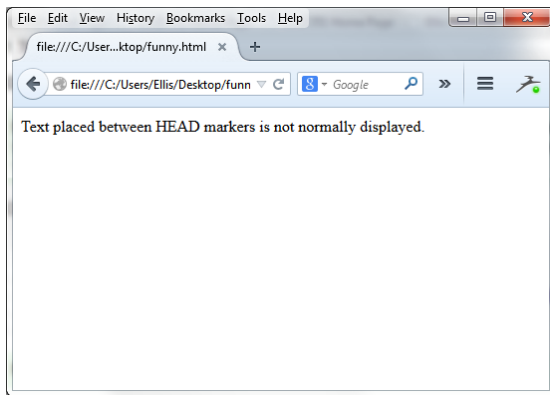


## Browsers Are Tolerant

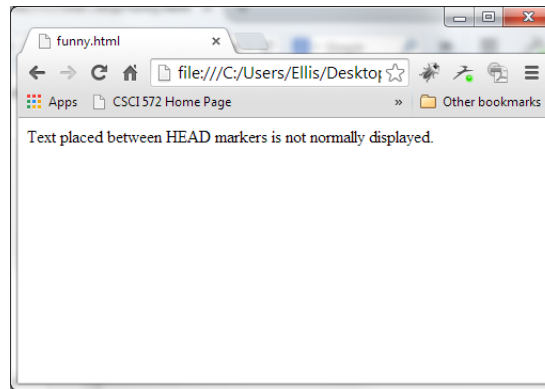
- Browsers follow the rule of being tolerant of mistakes in the input
  - They ignore markup they don't understand
- E.g. Internet Explorer/Firefox are tolerant browsers
  - They do not insist that the HTML document begin and end with `<HTML>`
  - `<HEAD>` and/or `<BODY>` tags are not required
  - But, there is no guarantee that this behavior will be the same for all browsers

## Example - Browsers are Tolerant

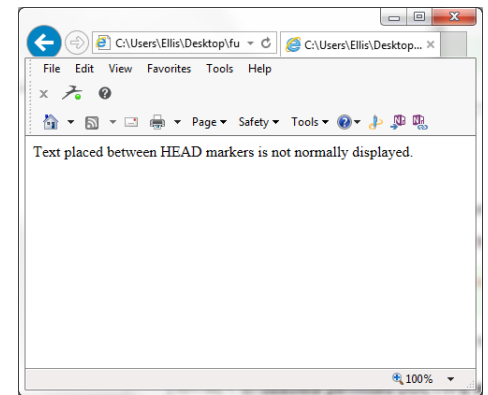
- Suppose the entire document is one line, such as:  
`<HEAD>Text placed between <ODDTAG> HEAD markers is not normally displayed.</HEAD>`



Firefox



Chrome



Internet Explorer

# HTML Elements

- Each element consists of a start tag, content, and an end tag
- E.g. `<BODY> some text </BODY>`
- A slash (/) after the “<” indicates an end tag
- Some elements do not require end tags,  
e.g. `<P>` paragraph tag
- Some elements do not require content,  
e.g. `<HR>`

horizontal rule tag places a straight line across the page

# Attributes

- Elements may have parameters, called attributes
- Attributes are placed in the start tag before the final ">"
- Attributes have the form name=value

E.g. <H1> is the first heading tag

```
<H1 id="Chapter1">
```

Start of Chapter 1

```
</H1>
```

- Attribute values are often enclosed in quotes, either double or single
- Quotes are not required when the value contains only letters, digits, hyphens, and periods.
- Attribute names are case insensitive, but not necessarily attribute values

## Comments in HTML

- Comments start with  
    `<!--`  
    and end with  
    `-->`
- Comments cannot be nested
  - White space is permitted between the `--` and the closing angle bracket, `>`
  - It is not permitted between the opening angle bracket, exclamation point, and the `--`
- E.g.  
    `<BODY>`  
    `<!-- This is a comment`  
    `and hence not displayed -->`  
    But this will be displayed  
    `</BODY>`

# Complete Set of <BODY> tag attributes

- `id`, assigns a unique name to an element

e.g. `<P id=mystart>This is my starting paragraph...`

- `class`, assigns one or more names to an element
- `lang`, a language code that identifies a natural language spoken, written, or otherwise used
- `title`, a short description of the body
- `style`, inline display information
- `bgcolor`, background color
- Events include

`onload, onunload onclick, ondblclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout, onkeypress onkeydown, onkeyup`

- Deprecated elements in include:

`background, text, link, vlink, alink`

See <http://www.w3.org/TR/html4/struct/global.html#edef-BODY>

# <BODY> Using Style Sheets

- Deprecated example

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<HTML>
<HEAD>
  <TITLE>A study of population dynamics</TITLE>
</HEAD>
<BODY bgcolor="white" text="black" link="red" alink="fuchsia" vlink="maroon">
  ... document body...
</BODY>
</HTML>
```

- Using Style Sheets

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<HTML>
<HEAD>
  <TITLE>A study of population dynamics</TITLE>
  <STYLE type="text/css">
    BODY { background: white; color: black}
    A:link { color: red } A:visited { color: maroon } A:active { color: fuchsia }
  </STYLE>
</HEAD>
<BODY>
  ... document body...
</BODY>
</HTML>
```

# Composing HTML

- Conventional editors let you compose HTML directly
  - e.g. emacs, vi, NotePad, TextPad, etc.
  - use the tools when you are writing HTML directly
- Word Processors include a File SaveAs option which saves your document in HTML format
  - e.g. Microsoft Word
  - Do not use!
- There are several free HTML-specific editors, e.g.
  - Brackets, <http://brackets.io>
  - Microsoft Expression Web 4 (Free Version, Win7-8),
    - <http://www.microsoft.com/en-us/download/details.aspx?id=36179>
- There are several commercial HTML/CSS suites, e.g.
  - Microsoft Blend for Visual Studio
  - Adobe Dreamweaver CC (Creative Cloud)
- For a complete list see  
[http://en.wikipedia.org/wiki/List\\_of\\_HTML\\_editors](http://en.wikipedia.org/wiki/List_of_HTML_editors)



# Lists

- HTML lists are very common in everyday web development.
- Perhaps the most commonly used are
  - unordered lists (`<ul>`) and
  - ordered lists (`<ol>`)
- but there are a few other list options
  - definition list (`<dl>`), and
  - the menu (`<menu>`) element,
  - both were deprecated in HTML4, but reintroduced in HTML5
- All lists follow the same pattern:
  - *<start tag of list>*
  - *<list item tag>*
  - *<list item tag>*
  - *<list item tag>*
  - *</ end tag of list>*

# Definition Lists

<DL>

    <DT>light year<DD>the distance light travels  
in one year

    <DT>asteroids<DD>are small, irregular shaped  
objects, mostly occurring between Mars and  
Jupiter

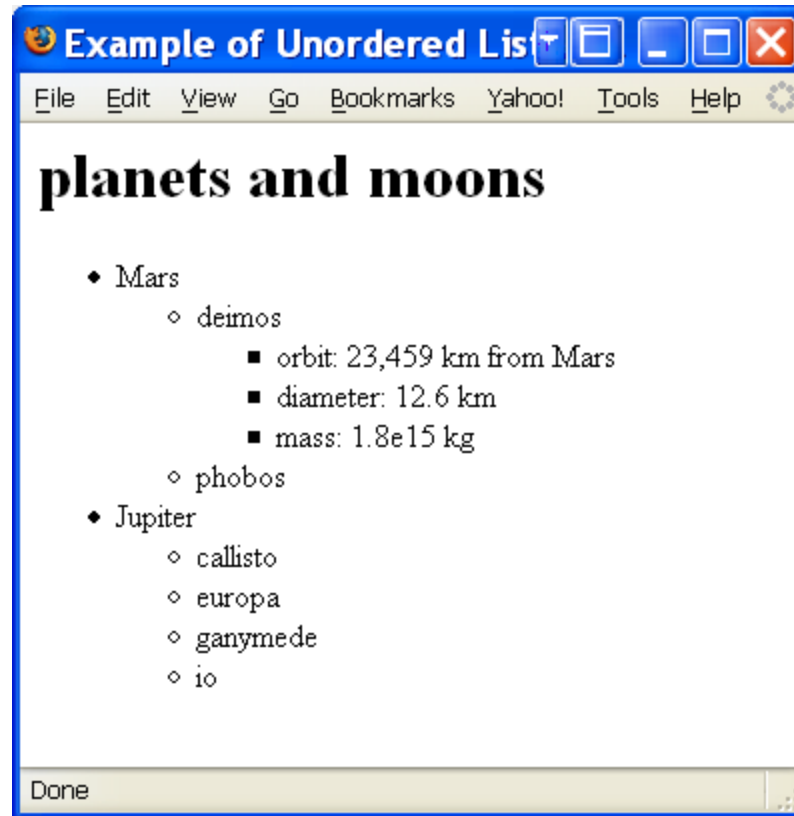
</DL>



## Example - Unordered Lists

```
<HTML><HEAD><TITLE>
Example of Unordered Lists</TITLE></HEAD>
<BODY><H1>planets and moons</H1>
<UL>
<LI>Mars
  <UL><LI> deimos
    <UL>    <LI>orbit: 23,459 km from Mars
    <LI>diameter: 12.6 km
    <LI>mass: 1.8e15 kg
    </UL>
  <LI>phobos
  </UL>
<LI>Jupiter
<UL><LI>callisto<LI>europa<LI>ganymede<LI>io</UL>
</UL></BODY></HTML>
```

# Browser Output of Unordered Lists



# Ordered Lists

- Has the general form

```
<OL><LI> first list item<LI> second list item</OL>
```

- START attribute can initialize the sequence to a number other than 1
- TYPE attribute can be used to select the numbering style

Type	Name	Style
1	arabic	1, 2, 3, ...
a	lower alpha	a, b, c, ...
A	upper alpha	A, B, C, ...
i	lower roman	i, ii, iii
I	upper roman	I, II, III,

- However, the type attribute is deprecated and list styles should be handled through style sheets, e.g.

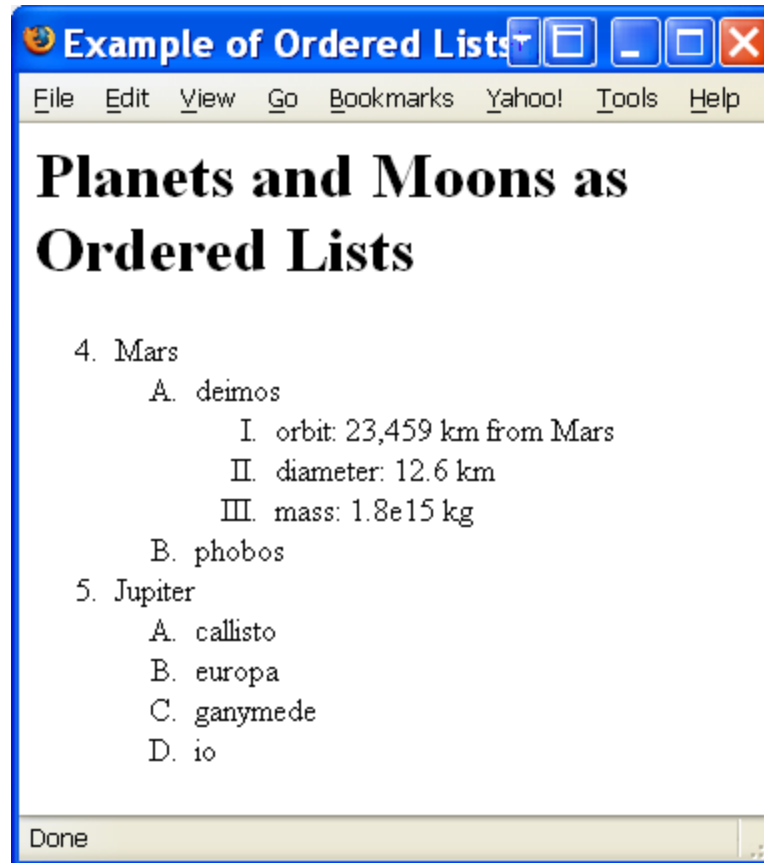
```
<STYLE type="text/css">
```

```
OL.withroman {list-style-type: lower-roman}</STYLE>
```

## Example - Ordered Lists

```
<HTML><HEAD><TITLE>
Example of Ordered Lists</TITLE></HEAD>
<BODY><H1>Planets and Moons as Ordered Lists</H1>
<OL START=4>
<LI>Mars
  <OL type=A><LI>deimos
    <OL type=I><LI>orbit: 23,459 km from Mars
      <LI>diameter: 12.6 km
      <LI>mass: 1.8e15 kg
    </OL>
  <LI>phobos
  </OL>
<LI>Jupiter
<OL type=A><LI>callisto<LI>europa<LI>ganymede
<LI>io</OL></OL></BODY></HTML>
```

# Browser Output



## Table Elements

- `<TABLE>`, a tag used to define a table
- `<CAPTION>`, a tag to label a table
  - Its attributes are `ALIGN = top, bottom, left, right`
- `<TH></TH>` or `<TD></TD>`, tags that identify a table header cell and table data cell
  - Headers are the same as data except they use bold font and are centered
- `<TR>`, a tag that identifies a container for a row of table cells
  - Same attributes as TH and TD



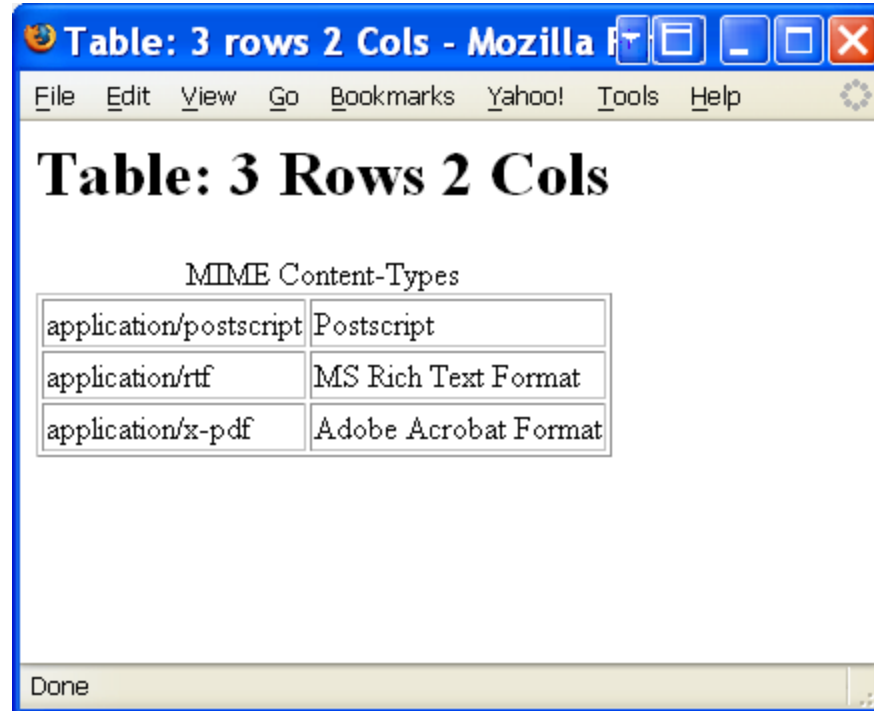
## Facts about Tables

- Table data can be text, lists, images, forms, figures, or even tables
- Table headers are typically rendered in boldface, and table data is typically rendered in the regular font and point size
- A table has an optional caption followed by rows
- Table rows are said to contain table headers and table data
- The browser will set the number of columns to be the greatest number of columns in all of the rows
- Blank cells are used to fill extra columns in the rows

## Example - 3 rows x 2 cols

```
<HTML>
<HEAD>
<TITLE>Table:  3 rows 2 Cols</TITLE>
</HEAD>
<BODY>
<H1>Table:  3 Rows 2 Cols</H1>
<TABLE BORDER="2">
<CAPTION>MIME Content-Types</CAPTION>
<TR><TD>application/postscript</TD><TD>Postscript</T
D>
<TR><TD>application/rtf</TD><TD>MS Rich Text
Format</TD>
<TR><TD>application/x-pdf</TD><TD>Adobe Acrobat
Format</TD>
</TABLE></BODY></HTML>
```

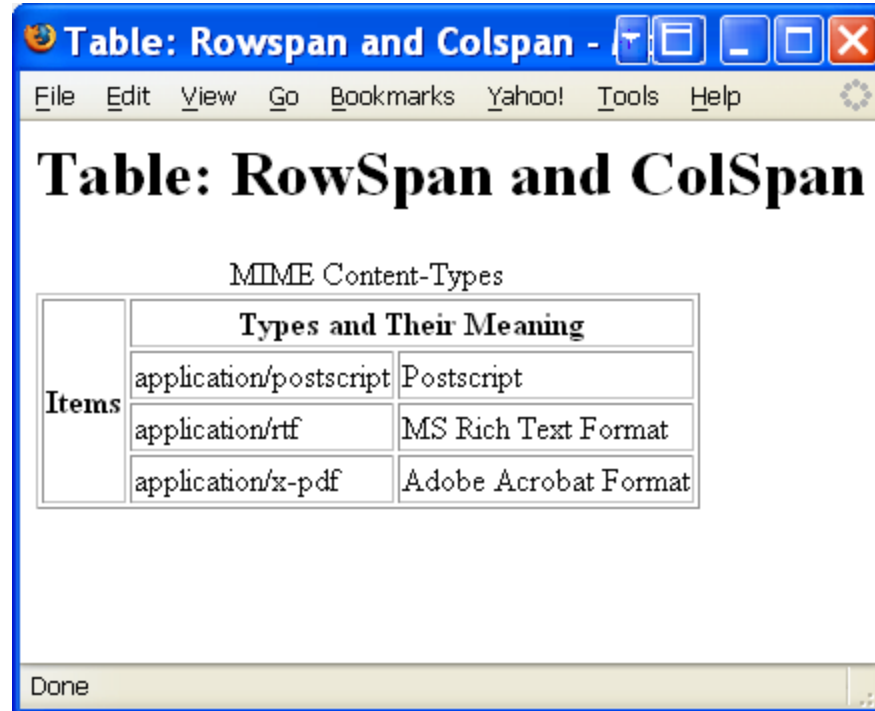
# Browser Output



## Table Example Rowspan colspan

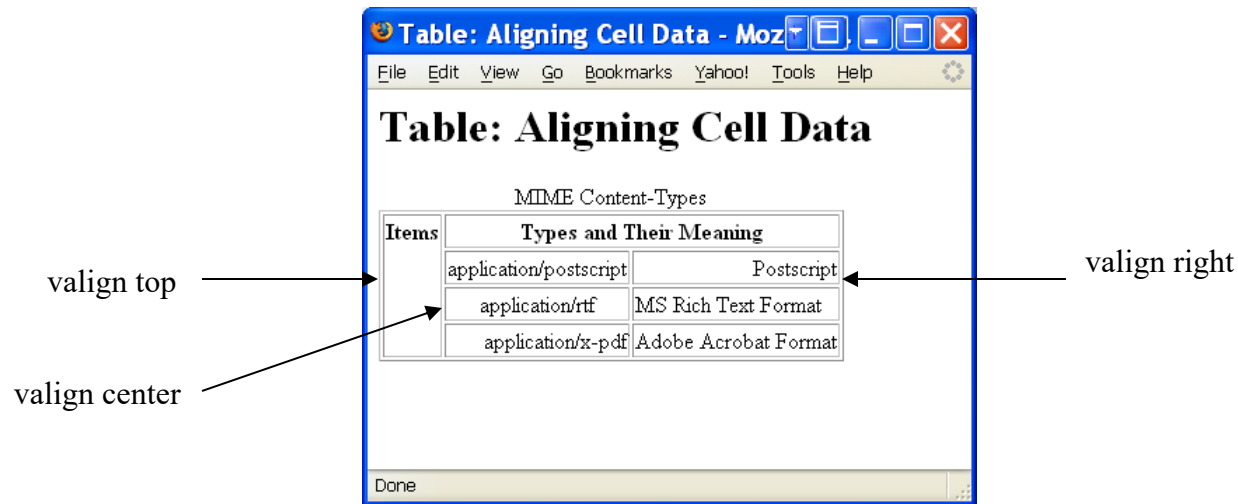
```
<HTML><HEAD>
<TITLE>Table: Rowspan and Colspan</TITLE>
</HEAD>
<BODY>
<H1>Table: RowSpan and ColSpan</H1>
<TABLE BORDER="2">
<CAPTION>MIME Content-Types</CAPTION>
<TR><TH ROWSPAN=4>Items</TH><TH colspan=2>Types and
Their Meaning</TH>
<TR><TD>application/postscript</TD><TD>Postscript</TD>
<TR><TD>application/rtf</TD><TD>MS Rich Text
Format</TD>
<TR><TD>application/x-pdf</TD><TD>Adobe Acrobat
Format</TD>
</TABLE></BODY></HTML>
```

# Browser Output



# Arranging Data in a Table

- Originally data in a table could be manipulated using attributes: align left, align right, align center, valign top, valign middle and valign bottom



- the above attributes are **deprecated** in **HTML5** in favor of Cascading Style Sheets (CSS) settings
- What does "deprecated" mean?
- see the next slide lecture on CSS

## Table Example: Surprise Quiz

```
<HTML><HEAD><TITLE>Table:  Pop Quiz</TITLE>
</HEAD><BODY><H1>Table:  Pop Quiz</H1>
<H2>Draw the table described here</H2>
<TABLE BORDER="2">
<TR><TD>Data1</TD><TD rowspan=2>Data2</TD>
<TD>Data3</TD>
<TR><TD>Data4</TD><TD>Data5</TD></TABLE>
<H2>Draw the table described here</H2>
<TABLE BORDER>
<TR><TH Rowspan=2  colspan=2></TH>
      <TH colspan=2>Average</TH>
<TR><TH>Cost</TH><TH>Time</TH>
<TR><TH Rowspan=2>Projects</TH>
      <TH>P1</TH><TD>100</TD><TD>7</TD>
<TR><TH>P2</TH><TD>250</TD><TD>15</TD>
</TABLE></BODY></HTML>
```

### Table: Pop Quiz

**Draw the table described here**

Data1	Data2	Data3
Data4		Data5

**Draw the table described here**

		Average	
		Cost	Time
Projects	P1	100	7
	P2	250	15

# HTML Character Set

- HTML uses the **Universal Character Set (UCS)**, defined in [ISO10646]. This standard defines a repertoire of thousands of characters used by communities all over the world.
  - Its latest specification (**Unicode 12.1.0**), dated May 7, 2019, can be found at <http://www.unicode.org/versions/Unicode12.1.0/>
  - Includes Unicode Emoji (with skin tone diversity)
  - <http://www.unicode.org/emoji/charts/emoji-released.html>

HTML must also specify how characters are encoded during transmission.

- Commonly used character encodings on the Web include
  - ISO-8859-1 (also referred to as "Latin-1",
  - ISO-8859-5 (which supports Cyrillic),
- A browser is informed of the encoding by a line  
Content-Type: text/html; charset=EUC-JP



# Character references

- Character references in HTML may appear in two forms:
  - **Numeric** character references (either decimal or hexadecimal)
    - `&#229;` (in decimal) represents the letter "a" with a small circle above it (used, for example, in Norwegian).
    - `&#60;` represents left angle bracket
    - `&#62;` represents right angle bracket
    - `&#38;` represents ampersand sign
    - `&#34;` represents double quote
  - Character **entity** references.
    - `"&lt;"` represents the `<` sign.
    - `"&gt;"` represents the `>` sign.
    - `"&amp;"` represents the `&` sign.
    - `"&quot;"` represents the `"` mark.

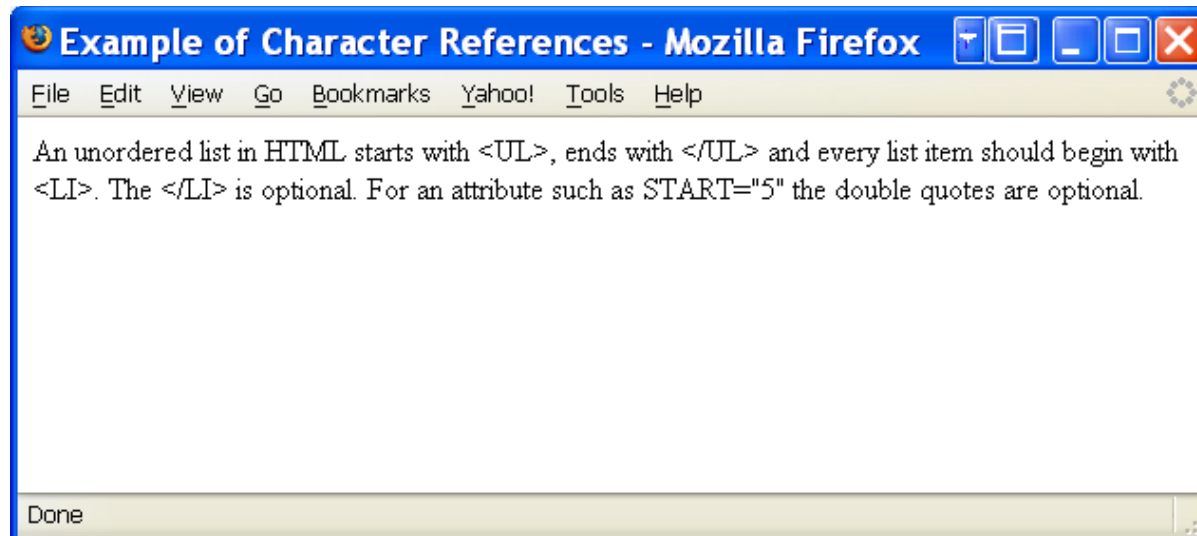
## Example - Character References

```
<HTML><HEAD><TITLE>Example of Character  
References</TITLE></HEAD>
```

```
<BODY>
```

An unordered list in HTML starts with `<ul>`, ends with `</ul>` and every list item should begin with `<li>`. The `</li>` is optional. For an attribute such as `START="5"` the double quotes are optional.

```
</BODY></HTML>
```



# Anchors Away

- An anchor is a way to designate a **link to another document** or to a specific place in the **same document**
- Begins with `<A>` and ends with `</A>`
- The link location is given by the **HREF attribute** (Hypertext Reference); e.g.,

See `<A HREF="http://cs-server.usc.edu:45678/index.html">Class Home Page</A>` for a description of the class project

- Hypertext links are displayed using underlining, color, and/or highlighting
  - Depends on the browser defaults or the use of style settings
  - Once a link is taken, it should change color
  - HREF, stands for **H**ypertext **REF**erence

# Syntax of Anchor Names

- An anchor name is the value of either the name or id attribute when used in the context of anchors.
- Anchor names must observe the following rules:
  - **Uniqueness:** Anchor names must be unique within a document. Anchor names that differ only in case may not appear in the same document.
  - **String matching:** Comparisons between fragment identifiers and anchor names must be done by exact (case-sensitive) match.

# Defining Anchors Using the id Attribute

- The id attribute may be used to create an anchor at the start tag of any element (including the A element).
- Example: the **id attribute** places an anchor in an H2 element.

You may read more about this in

```
<A href="#section2">Section Two</A>.
```

```
. . . more text . . .
```

```
<H2 id="section2">Section Two</H2>
```

```
. . . more text
```

```
<P>Please refer to <A href="#section2">Section Two</A> above for  
more details.
```

- The id and name attributes share the same name space. They cannot both define an anchor with the same name in the same document. Try this example in IE and Firefox:

```
<P><A href="#label1">...</A>
```

```
...more document...
```

```
<P><A name="LABEL1">...</A>
```

← **NOTE: browsers work differently**

# Examples of Anchors

- `<A href=myfile.html>` a file in the same directory and same domain as the current page
- `<A href=http://nunki.usc.edu/mydocs/book.doc>` a file in directory mydocs on machine nunki.usc.edu, a WWW site
- `<A href=news:comp.compilers>` the newsgroup computers.compilers
- `<A href=mailto:sales@xyz.com>` opens an e-mail window for sending a message
- `<A HREF="ftp://ds.internic.net/rfc/rfc1866.txt">`  
Download RFC1866 `</A>` executes the ftp program to fetch a file
- Given the current position, this `<HREF>` moves up one directory, connects to Docs/Style/ and displays the document Overview.html  
`<A HREF="../Docs/Style/Overview.html"> ... </A>`
- Connects to lycos and runs pursuit with three arguments  
`<A HREF="http://www.lycos.com/cgi-bin/pursuit?Hypertext+Markup+Language"> ... </A>`

## Anchor Titles

- The title attribute may be set to add information about the nature of a link.
- This information may be spoken by a user agent, rendered as a tool tip, cause a change in cursor image, etc.

```
<BODY>
```

```
...some text...
```

```
<P>You'll find a lot more in <A href="chapter2.html"
    title="Go to chapter two">chapter two</A>.
```

```
See also this <A href="../images/solarsystem.gif"
    title="GIF image of solar system">view of the
solar system.</A>
```

```
</BODY>
```

# Universal Resource Identifier

- URIs typically consist of three pieces:
  - The **scheme** of the mechanism used to access the resource.
  - The name of the machine **hosting** the resource.
  - The name of the **resource** itself, given as a **path**E.g. `http://www.usc.edu/dept/cs/index.html`
- Fragment identifiers are URIs that refer to a location within a resource  
e.g. `http://www.usc.edu/dept/cs/index.html#section2`
- Relative URIs have a path that refers to a resource on the same machine as the current document, e.g. `".."` means one level up
- See p. 24 in:  
`http://www.ietf.org/rfc/rfc3986.txt`
- See also:  
`https://en.wikipedia.org/wiki/Fragment_identifier`



## The <LINK> Element

- Should only appear in the HEAD
- It may appear any number of times
- It conveys relationship information that may be rendered in a variety of ways (e.g., a tool-bar with a drop-down menu of links, external files)
- Example - The current document is "Chapter2.html". The rel attribute specifies the relationship of the linked document with the current document.

```
<HTML> <HEAD>
```

```
  <TITLE>Chapter 2</TITLE>
```

```
  <LINK rel="Index" href="../index.html">
```

```
  <LINK rel="Next" href="Chapter3.html">
```

```
  <LINK rel="Prev" href="Chapter1.html">
```

```
</HEAD>
```

```
...the rest of the document...
```

## How is <LINK> Used

- To provide a variety of information to search engines:
  - Links to **alternate versions** of a document, written in another human language, e.g.

```
<LINK lang="fr" title="La documentation en Fran&ccedil;ais"
      type="text/html" rel="alternate" hreflang="fr"
      href="http://domain/manual/french.html">
```

- Links to alternate versions of a document, designed for **different media**

```
<LINK media="print" title="The manual in postscript"
      type="application/postscript"
      rel="alternate"
      href="http://domain/manual/usermanual.ps">
```

- Links to the starting page of a collection of documents.
- Links to style sheets and "**media queries**" used in Responsive Web Design

# Creating Graphics

- Digital cameras & Smartphones
  - Snap and the image is digitized and can be transferred to a computer
  - Typical resolutions are 1280x720, 1920x1080
- Graphic editors
  - Permit the combination of text, drawing, and color
  - For example, Corel PhotoPaint, Adobe Photoshop
- Scanners
  - Convert text and graphics into machine readable form

# Image Formats

- Four image formats are always supported by Web browsers
  - **x-pixelmaps** (obsolete)
    - Similar to x-bitmaps, but 8 bits are given to each pixel, permitting 256 colors in the image
  - Graphic Interchange Format (**GIF**)
    - Support black and white, grayscale, and color
    - Patented by Unisys (expired, abandoned)
  - Joint Photographic Experts Group (**JPEG**)
    - Designed for photographic images
    - Includes image compression
  - Portable Network Graphics (**PNG**)
    - An open, extensible image format with Lossless Compression
    - Patent-free replacement for GIF and TIFF
    - W3C Recommendation: <http://www.w3.org/TR/PNG/>

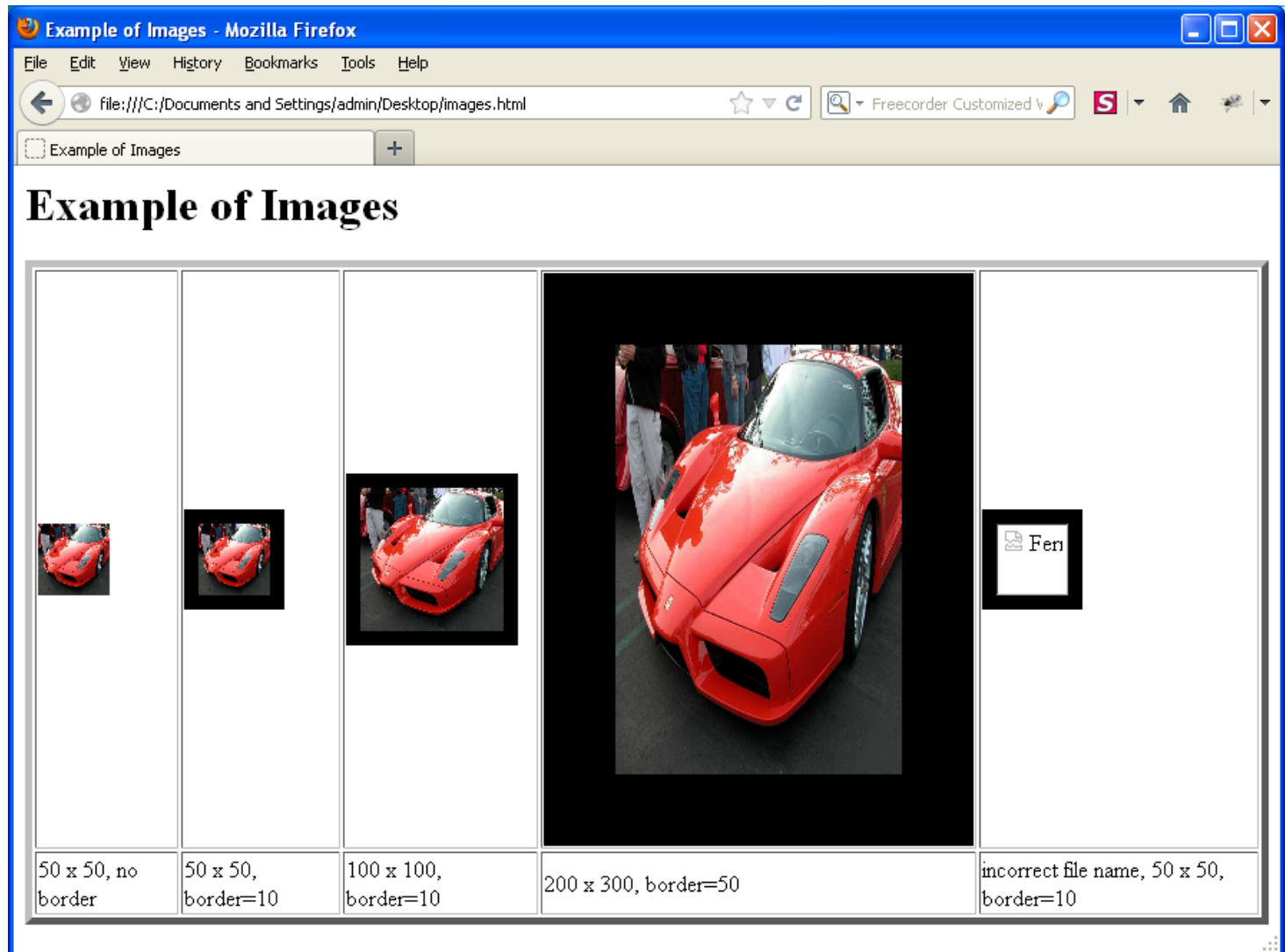
## <IMG> Element

- The IMG element embeds an image in the current document, e.g. `<IMG SRC="file.gif">`
- Some attributes of `<IMG>` include
  - **SRC**, the HREF or name of the image
  - `Align=top`, `middle`, or `bottom` to align text around an image
  - `border` to place a border around an image
    - If set to 1 or higher, places a black box around the image
    - If set to 0, leaves no surrounding box
  - **height** and **width** to control the dimensions of the image
  - `Alt` to replace an image with text, if the image is unavailable or a text browser is used; e.g.,  
`<IMG SRC="eiffel.gif" ALT="picture of the Eiffel Tower">`

## Example - Images

```
<HTML><HEAD><TITLE>Example of Images</TITLE></HEAD>
<BODY>
<H1>Example of Images</H1>
<table border=5><tr>
<td><IMG SRC="ferrari.jpg" ALT="Ferrari" BORDER=0 WIDTH=50 HEIGHT=50>
<td><IMG SRC="ferrari.jpg" ALT="Ferrari" BORDER=10 WIDTH=50 HEIGHT=50>
<td><IMG SRC="ferrari.jpg" ALT="Ferrari" BORDER=10 WIDTH=100 HEIGHT=100>
<td><IMG SRC="ferrari.jpg" ALT="Ferrari" BORDER=50 WIDTH=200 HEIGHT=300>
<td><IMG SRC="xferrari.jpg" ALT="Ferrari" BORDER=10 WIDTH=50
HEIGHT=50></tr>
<tr>
<td>50 x 50, no border</td>
<td>50 x 50, border=10</td>
<td>100 x 100, border=10</td>
<td>200 x 300, border=50</td>
<td>incorrect file name, 50 x 50, border=10</td>
</tr>
</table>
</BODY></HTML>
```

# Browser Output



## Active Images

- Active images are images that can be clicked and, just like an anchor, they act as a hypertext link

```
<A HREF="http://sunset.usc.edu:8080/index.html">
```

```
<IMG SRC="USCimage.gif"> </A>
```

- Active images have a border around them and the cursor changes shape when passed over



# Image Maps

- Image maps are active images with multiple clickable regions
- each region can be associated with a specific action (e.g., retrieve a document, run a program, etc.)
- When the region is activated by the user, e.g. by a mouse click, the action is taken
  - the pixel coordinates are interpreted by the browser (**USEMAP** attribute).

# An ImageMap Example from Wikipedia



To see the ImageMap work  
go to

[http://en.wikipedia.org/wiki/Image\\_map](http://en.wikipedia.org/wiki/Image_map)  
and click on each of the individuals  
at the table

```

```

```
<map id="ImageMap_1_2013620197"  
name="ImageMap_1_2013620197">
```

```
<area title="Dr Johnson – Dictionary writer"  
alt="Dr Johnson – Dictionary writer"  
coords="133,343,124,287,159,224,189,228,195,291,222,  
311,209,343,209,354,243,362,292,466,250,463"  
shape="poly" href="/wiki/Samuel_Johnson">
```

```
<area title="Boswell – Biographer" alt="Boswell"  
coords="76,224,84,255,43,302,62,400,123,423,121,361,  
137,344,122,290,111,234,96,225"  
shape="poly" href="/wiki/James_Boswell">
```

```
<area title="Sir Joshua Reynolds – Host"  
alt="Sir Joshua Reynolds – Host"  
coords="190,276,208,240,229,228,247,238,250,258,286,  
319,282,323,223,323,220,301,200,295"  
shape="poly" href="/wiki/Joshua_Reynolds">
```

# Imagemaps

- Add USEMAP attribute to <IMG> tag to indicate a client-side imagemap, e.g.

```
<IMG SRC="images/banner.gif" USEMAP="#bannerbar">
```

- Different regions of the image are described using <MAP> tag, e.g.

```
<MAP NAME="bannerbar">
```

```
<AREA SHAPE="RECT" COORDS="10,10,50,50" HREF="p1.html">
```

```
<AREA SHAPE="RECT" COORDS="50,10,90,50" HREF="p2.html">
```

```
<AREA SHAPE="RECT" COORDS="90,10,130,50" HREF="p3.html">
```

```
<AREA NOHREF SHAPE=default>
```

```
</MAP>
```

- Possible values for SHAPE are:
  - default**: Specifies the entire region.
  - rect**: Defines a rectangular region.
  - circle**: Defines a circular region.
  - poly**: Defines a polygonal region.

## <META> Element

- Allows you to insert Name/Value pairs describing document properties, e.g.

```
<META NAME="Author" CONTENT="Ellis Horowitz">
```

- USC CS dept home page header

```
<META name = "description" content = "The Computer  
Science Department at the University of Southern  
California, Los Angeles (USC) provides education leading  
to the Bachelors, Masters and Ph.D. degrees in Computer  
Science.">
```

```
<META name = "keywords" content = "USC, computer science,  
computer science research, computer science teaching">
```

```
<META name = "author" content = "Ellis Horowitz">
```

## <META> Element

- Moving a Web page to a new site

```
<HTML><HEAD>
```

```
<META HTTP-EQUIV="REFRESH" CONTENT="5;  
    URL=http://www.usc.edu/dept/cs/">
```

```
<META NAME="GENERATOR" CONTENT="Mozilla/4.04 [en] (Win95; I)  
    [Netscape]">
```

```
<TITLE>This site has moved</TITLE></HEAD>
```

```
<BODY><CENTER>This site has moved to a new location which is:  
<A
```

```
    HREF="http://www.usc.edu/dept/cs/">http://www.usc.edu/dept  
    /cs/</A> <BR>
```

Your browser should automatically move to the correct URL in five seconds.

```
</CENTER></BODY></HTML>
```

# Meta Tag and Robot Exclusion

```
<meta name="robots" content="noindex,nofollow">
```

```
<title>...</title> </head> <body> ...
```

- The content of the Robots META tag contains directives separated by commas.
- The currently defined directives are
  - [NO]INDEX. The INDEX directive specifies if an indexing robot should index the page.
  - [NO]FOLLOW The FOLLOW directive specifies if a robot is to follow links on the page.
  - The defaults are INDEX and FOLLOW. The values ALL and NONE set all directives on or off: ALL=INDEX,FOLLOW and NONE=NOINDEX,NOFOLLOW.
- Some examples:

```
<meta name="robots" content="index, follow">
```

```
<meta name="robots" content="noindex, follow">
```

```
<meta name="robots" content="index, nofollow">
```

```
<meta name="robots" content="noindex, nofollow"> Note the  
"robots" name of the tag and the content are case  
insensitive.
```

# Validating Your HTML

- The reasons for validation
  - Browsers display HTML differently
  - Browsers treat HTML errors differently
- What validators do
  - Flag syntax errors with respect to HTML DTD
  - Compare your pages to HTML 4.x, XHTML, and even HTML 5 (experimental)
- Some tools are downloaded to your site; others read your Web page from a URL
- HTML validation tools can be found at:  
<http://search.yahoo.com/bin/search?p=html+validation>
- W3C Markup Validation Service:  
<http://validator.w3.org/>

# W3C Markup Validation Service

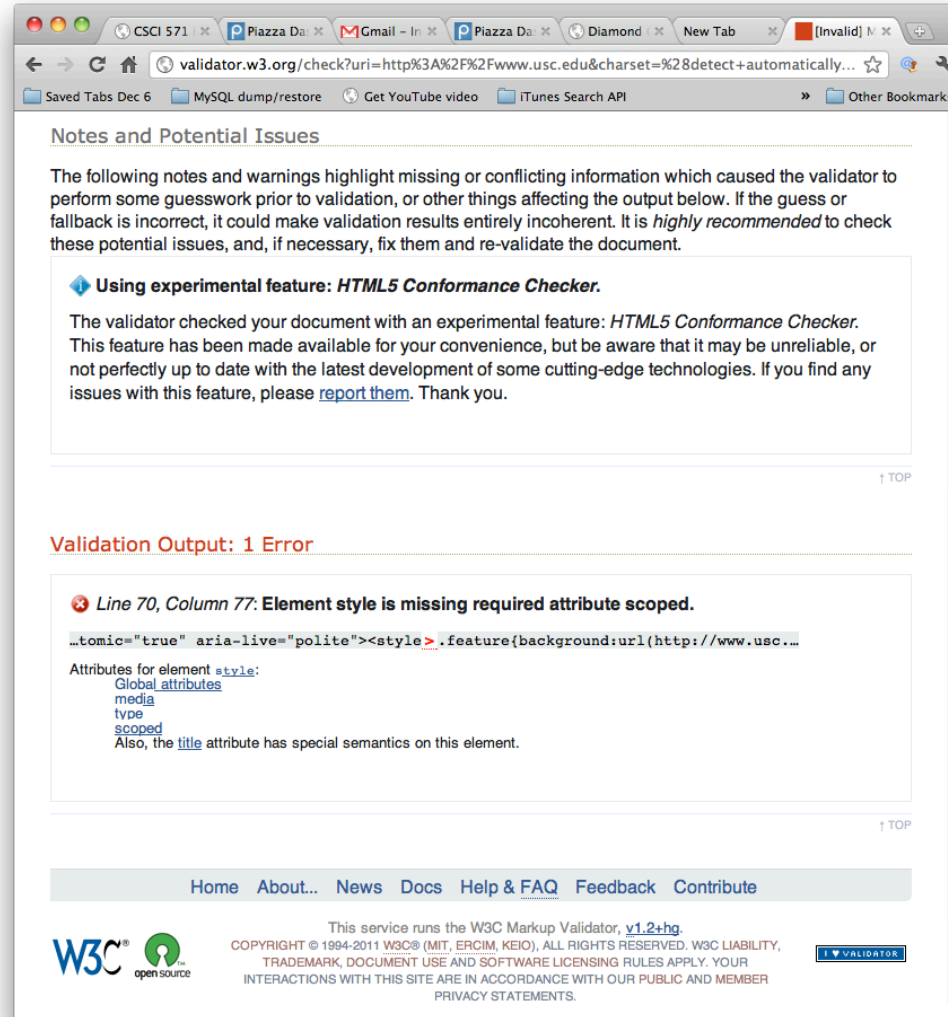
- Options: character encoding & Document type

The screenshot shows the W3C Markup Validation Service interface in a web browser. The browser's address bar shows the URL `validator.w3.org/#validate_by_uri+with_options`. The page has a blue header with the W3C logo and the text "Markup Validation Service" and "Check the markup (HTML, XHTML, ...) of Web documents". Below the header, there are three tabs: "Validate by URI" (selected), "Validate by File Upload", and "Validate by Direct Input". Under the "Validate by URI" tab, the text "Validate a document online:" is followed by an "Address:" label and a text input field containing `http://www.usc.edu`. Below this, there is a "More Options" section with several settings: "Character Encoding" set to "(detect automatically)" with a dropdown arrow and a checkbox "Only if missing"; "Document Type" set to "(detect automatically)" with a dropdown arrow and a checkbox "Only if missing"; radio buttons for "List Messages Sequentially" (selected) and "Group Error Messages by Type"; checkboxes for "Show Source", "Clean up Markup with HTML-Tidy", "Show Outline", "Validate error pages", and "Verbose Output". A "Check" button is at the bottom of this section. Below the form, a paragraph explains that the validator checks the markup validity of Web documents in HTML, XHTML, SMIL, MathML, etc., and provides links for specific content validation and broken links. At the bottom, there is a Mozilla logo and text stating that the W3C validators are developed with assistance from the Mozilla Foundation, supported by community donations, and a "Donate" link. A "Flattr" button with the number "2276" is also present. The footer includes navigation links (Home, About..., News, Docs, Help & FAQ, Feedback, Contribute), the W3C open source logo, and copyright information: "This service runs the W3C Markup Validator, v1.2. COPYRIGHT © 1994-2010 W3C® (MIT, ERCIM, KEIO). ALL RIGHTS RESERVED. W3C LIABILITY, TRADEMARK, DOCUMENT USE AND SOFTWARE LICENSING RULES APPLY. YOUR INTERACTIONS WITH THIS SITE ARE IN ACCORDANCE WITH OUR PUBLIC AND MEMBER PRIVACY STATEMENTS."



# Validating Your HTML

- Sample output



# HTML Tidy

- a W3C open source program for fixing HTML
- converts presentation markup to CSS
- cleans up HTML produced by Microsoft Word
- converts HTML to XHTML standards, e.g.

`<h1>heading`

`<h2>subheading</h3>`

is converted to

`<h1>heading</h1>`

`<h2>subheading</h2>`

- Supported by W3C Markup Validator Service  
located at:

<http://www.w3.org/People/Raggett/tidy/>