WEB TECHNOLOGY

For Class- B.Pharmacy 2nd Semester

Subject- COMPUTER APPLICATIONS IN PHARMACY (BP205T)

RAMAKANT JOSHI

School of Studies in Pharmaceutical Sciences, Jiwaji University, Gwalior

Internet and WWW

- Inter-network and World Wide Web
- Interlinked hypertext documents accessed using HTTP Protocol
- Client - Server architecture

Why Internet?
Use of internet

- Email
- Social Networking, Chat
- Information sharing
- Getting updates – News around the world
- Entertainment – Games, Videos and Music
- Virtual classrooms
- Remote Access
- Online Jobs

Why Websites?
Offline Apps vs. Online Apps

ONLINE APPS

- No need to install
- Just login and use
- Available from anywhere where Internet connection is available
- Operating system independent
- No piracy issues

Why Websites?
Offline Apps vs. Online Apps

OFFLINE APPS

- Ease of use
- Generally have more features
- Easier to develop but difficult to update

Technologies Overview
List of Technologies

Client Side Technologies

- HTML, CSS, JavaScript, VBScript
- XHTML, DHTML, WML, AJAX
- FLASH

Server Side Technologies

- ASP, PHP, Perl, JSP
- ASP.NET, Java
- MySQL, SQL Server, Access

Technologies Overview
List of Technologies

Some More Advanced Technologies

- XML, XSLT, RSS, Atom
- X-Path, XQuery, WSDL
- XML-DOM, RDF
- Ruby on Rails, GRAIL Framework
- REST, SOAP

How to choose a Technology?

Depends on:

- What is the type of content?
- Who is your audience?
- Who will modify your content?
- What are your Future Plans?
- Availability of technology?
- Your previous experience?
- Portability and Data sharing

HTML
Hyper Text Markup Language

- Documents
  - Document = page = HTM file = topic
  - Content (text, images)
  - Tags (display commands)
- Other terms
  - Window: browser display window
  - URL: Uniform Resource Locator
  - Hyperlink: hypertext jump to a resource
  - Resource: URL, image, mailto, external file

HTML

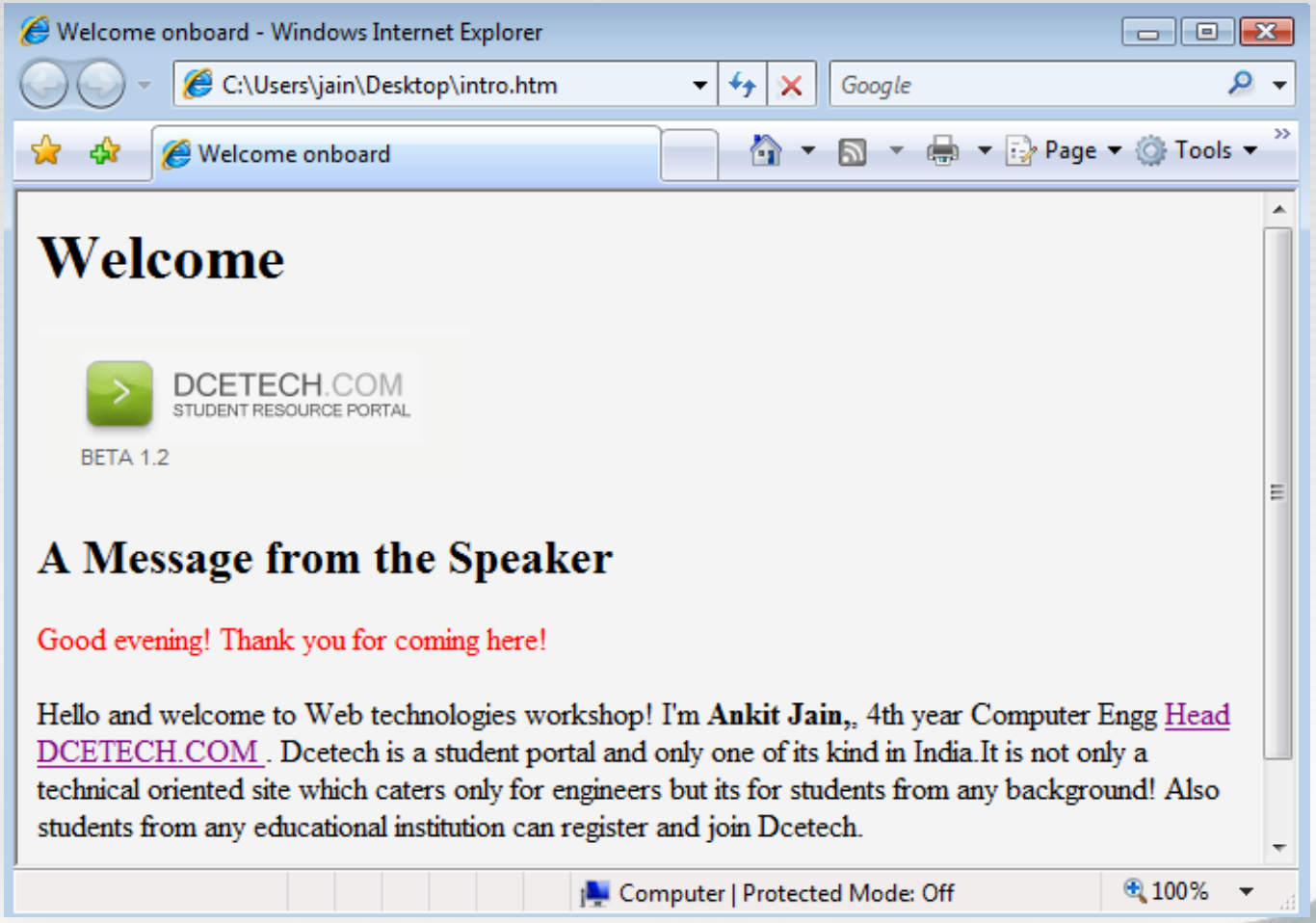
HTML pages are tag-based documents

- Really plain ASCII text files
- Don't look like documents they represent
- Tags indicate how processing program should display text and graphics
- Processed by browsers “on the fly”
- Tags usually appear in pairs
- Most have reasonable names or mnemonics
- Most can be modified by attributes/values

That's how this…

```
<html>
 <head><title>Welcome onboard</title></head>
  <body bgcolor="#f4f4f4">
  <h1>Welcome</h1>
  <img src="dcetech.gif" width="222" height="80" alt="DCETECH"
   BORDER="0" />
  <h2>A Message from the Speaker </h2>
  <p><font color=red>Good evening! Thank you for coming here!
   </font></p>
  <p>Hello and welcome to Web technologies workshop! I'm <b>Ankit
   Jain,</b>, 4th year Computer Engg   <a
   href="http://dcetech.com"> Head DCETECH.COM </a>. Dcetech is a
   student portal and only one of its kind in India.It is not
   only a technical oriented site which caters only for engineers
   but its for students from any background! Also students from
   any educational institution can register and join Dcetech.
   </p>
. . .
</body>
</html>
```

Turns into this…

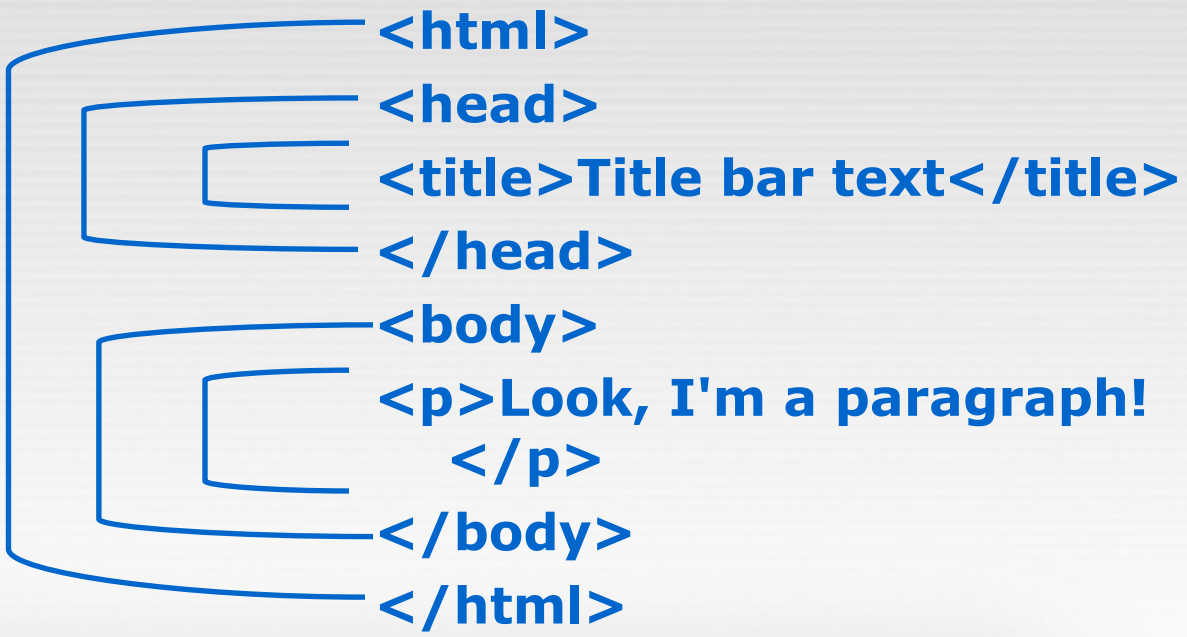


Some HTML Tags example

START TAG	*END TAG*
<HTML>	</HTML>
<HEAD>	</HEAD>
<TITLE>	</TITLE>
<BODY>	</BODY>
<H1>, <H2>, …	</H1>, </H2>, …
<IMG …>	</IMG> (optional)
<A …>	</A>
<P>	</P>
 	(none; "empty" tag)
<OL>	</OL>
<UL>	</UL>
<LI>	</LI>

Basic Structure of HTML document

Example of basic tag positioning



Attributes and Values

- Properties, traits, or characteristics that modify the way a tag looks or acts
  - Usually in pairs: `<body bgcolor="teal">`
  - Sometimes not: `<option selected>`
- Most HTML tags can take attributes
  - Format of value depends on attribute
  - `width="150" ... href="page3.htm"` *not* `width="page3.htm" ... href="150"`

Tables

```
<table border="1">
 <tr>
   <td>Row 1, Cell 1</td>
   <td>Row 1, Cell 2</td>
 </tr>
 <tr>
   <td>Row 2, Cell 1</td>
   <td>Row 2, Cell 2</td>
 </tr>
</table>
```

Row 1, Cell 1	Row 1, Cell 2
Row 2, Cell 1	Row 2, Cell 2

Some Common Text Tags

- Heading levels
  - h1 – h6, numbers inverse to text size

```
<h1>Heading One</h1>
<h2>Heading One</h2>
```

- Paragraph
  - Probably the most common tag

```
<p>Yada yada yada...</p>
```

- Line break (an empty tag)
  - Used when <p>'s white space not wanted

```
This line will break<br>right there
```

- Note white space or lack thereof in HTML source ***does not matter!***

Ordered & Unordered Lists

- Ordered (numbered)
  - Use `<ol>...</ol>` tags
- Unordered (bulleted)
  - Use `<ul>...</ul>` tags
- List Items make up both lists
  - Use same `<li>...</li>` tags
- Lists can contain almost anything
  - Text, images, paragraphs, links
  - Even other (nested) lists, same type or not

Attributes and Values

- Properties, traits, or characteristics that modify the way a tag looks or acts
  - Usually in pairs: `<body bgcolor="teal">`
  - Sometimes not: `<dl compact>`
- Most HTML tags can take attributes
  - Format of value depends on attribute
  - `width="150" ... href="page3.htm"` *not* `width="page3.htm" ... href="150"`

The Anchor Tag (1)

- The tag that puts the HT in HTML
  - <a> ... (useless by itself)
  - Must have attributes to be useful
- HREF (Hypertext REFerence) attribute
  - Makes a jump to someplace (URL)

```
<a href="mypage.htm">My Page</a>
<a href="www.google.com">Google</a>
```

  - Link text is underscored by default
- ***Whatever*** is between <a> and is hot (clickable)
  - Clicking makes the link go somewhere or do something

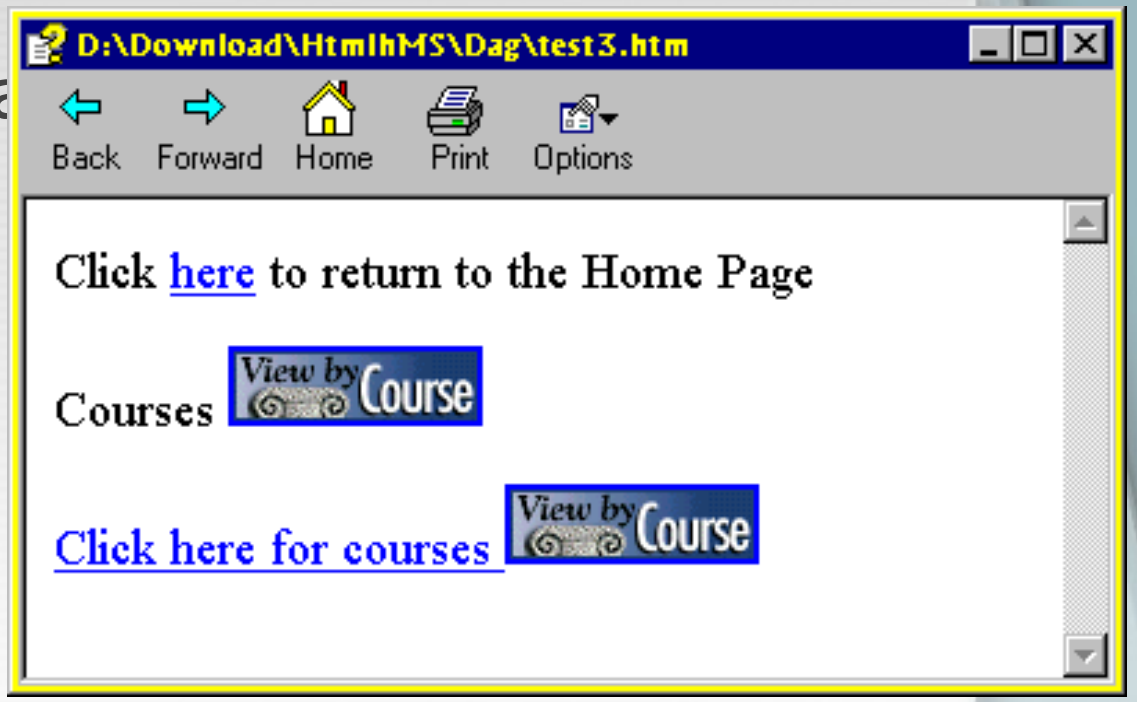
The Anchor Tag (2)

- Some link exa

text only

image only

both



Images (1)

- Used in pages for various reasons
  - Clarification, navigation, peripheral training
- Images not *in* page; *referenced* by page
  - Graphics are separate, required files
  - Usually GIFs or JPGs, sometimes others
  - Can be anywhere in document body: in paragraphs, headings, lists, anchors, etc.
- Place image with <img> tag
  - Like <a>, <img> is useless by itself
  - All work is done with attributes/values

Images (2)

- Main attribute: SRC
  - Tells page where to find the image
  - File name can be local, relative, or full
  - Case sensitivity depends on server
  - Animated GIFs added same as static

```
<img src="smiley.gif">
<img src="./pix/mypic.jpg">
<img src="http://www.myweb.com/mail.gif">
```

Tables (1)

- Powerful, flexible information delivery
  - Used to reflect or impart structure
- A table is a container
  `<table> ... </table>`
- That contains other containers (rows)
  `<tr> ... </tr>`
- That contain other containers (cells)
  `<td> ... </td>` **(data cells)**
  `<th> ... </th>` **(heading cells)**
- That contain data – or other containers
  - Text, graphics, lists, even other tables!

Tables (2)

- Basic table markup

```
<table border="1">
   <tr>
      <td>Row 1, Cell 1</td>
      <td>Row 1, Cell 2</td>
   </tr>
   <tr>
      <td>Row 2, Cell 1</td>
      <td>Row 2, Cell 2</td>
   </tr>
</table>
```

Row 1, Cell 1	Row 1, Cell 2
Row 2, Cell 1	Row 2, Cell 2

CSS Concepts

- Styles are named sets of formatting commands
  - [18pt, Arial, left aligned] "Section head"
  - [Bold, italic, blue] "Glossary term"
  - [Normal, 10pt, Verdana] "Body text"
  - [Italic, orange, small caps] "Bob"
- Style sheets are control documents that are referenced by content documents
  - MS Word, other editors & desktop publishing programs have done it for years
  - DOT : DOC :: CSS : HTM

Why Use CSS?

- HTML formatting is awkward and imprecise
  - Originally intended to deliver well organized text (aimed at structure, not formatting)
  - Over time, formatting elements were added that solved some problems, but created many more
- W3C proposed Cascading Style Sheets
  - Separate format from content
  - Enforce consistency
  - Greatly simplify control & maintenance

What's "Cascading" All About?

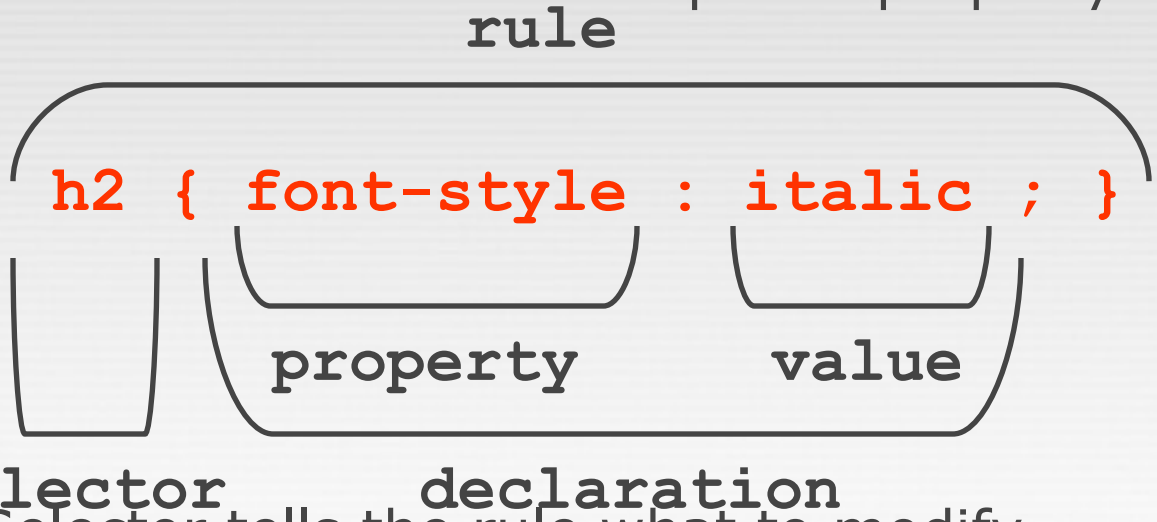
- Three places to put style commands
  - External: Affects all documents it's attached to
  - Internal: Affects only document it appears in
  - Inline: Affects only text it's applied to
- Cascading means styles' "pecking order"
  - Precedence is: Inline > Internal > External
  - Seems backward, but it couldn't work any other way; for example…
  - Picture a document whose style sheet specifies Verdana as the font, with one paragraph style in Courier New, with one bold word or phrase

What Can CSS Control?

- Almost everything
  - Page background, colors, images, fonts and text, margins and spacing, headings, positioning, links, lists, tables, cursors, etc.
- W3C intends CSS to "…relieve HTML of the responsibility of presentation."
  - Translation: "Don't bug us for new tags; change existing tags & make your own using CSS."
- Idea is to put all formatting in CSS
  - To that end, many tags are "deprecated" by CSS: <font>, <basefont>, , <strike>…

Coding CSS Rules

- Rules have very specific parts and syntax
  - Rules have two basic parts: selector and declaration
  - Declaration also has two parts: property and value



  - Selector tells the rule what to modify
  - Declaration tells the rule how to modify it

CSS Rule Placement

- In a separate .CSS file
  - Affects all pages to which it is linked
  - .CSS referenced by pages with <link> tag
- In the <head> of an .HTM page
  - Affects only page in which it appears
  - Rules are coded in <style></style> container
- In an HTML tag in page <body>
  - Affects only text to which it is attached
  - Declarations are coded as *attribute= "value"* pairs, e.g., style="color: blue;"

Linking To An External CSS

- Do not put <style></style> tags in the .CSS file; this will prevent it from working
- Add CSS rules as needed; break lines where necessary; format as desired
- Save as *filename*.css
- Reference .CSS in <head> of .HTM(s)

```
<head>
<link rel="stylesheet" type="text/css"
      href="mystyles.css">
</head>
```

Adding Styles To A Single Page

- Within document's <head>, insert a <style></style> container
- Code rules exactly as you would in an external .CSS

```
<head>
<style>
 h2 { font-style: italic; color: red; }
 p { font-family: "Verdana, Arial, sans-
      serif"; font-size: 12pt;
      color: blue; }
</style>
</head>
```

Adding Styles To An HTML Tag

- Within a tag's < >, code an *attribute* = *"value"* pair defining style characteristics
  - Tip: Watch out for nested quotes

```
<h1 style = "font: small-caps bold
italic; font-family: 'Verdana, Arial,
sans-serif'; color: #008080; text-
align: center;">Gettysburg Address
(First Draft)</h1>
<p style = "font-family: Times;
color: #800000; font-weight: bold;">
Four score and seven beers ago…</p>
```

JavaScript

- What JavaScript isn’t
  - Java (object-oriented programming language)
  - A "programmers-only" language
- What JavaScript is
  - Extension to HTML (support depends on browser)
  - An accessible, object-based scripting language
- What JavaScript is for
  - Interactivity with the user:
    * input (user provides data to application)
    * processing (application manipulates data)
    * output (application provides results to user)

Usage of JS

- Direct insertion into page (immediate)

```
<body><p>Today is
<script>document.write( Date() );
</script></p>
```

- Direct insertion into page (deferred)

```
<head>
<script>
   function dwd()
     {
     document.write( Date() );
     }
   </script>
</head>
   . . .
   <body>
<p>Today is <script>dwd(); </script></p>
```

Conclusion & Future Work

- Most Web pages – remote or local – are a combination of those technologies
  - Raw content, placed inside…
  - HTML tags, formatted with…
  - CSS rules, interactivity produced by…
  - JavaScript scripts on Clients sides and…
  - PHP scripts on server sides
- Newer technologies like DHTML, XHTML, and XML are based on these
  - A little knowledge now can prepare you for new technologies!