



CS472 WAP

# Introduction to Web Programming

Layers of Abstraction

# Overview of Today

- Layers of the Web
- Structure of a Page
- Types of Text
- Lists and Tables

# Wholeness Statement

- The basic technologies that make up the Internet are the World Wide Web and the Hyper Text Markup Language (HTML). Many technologies are built on top of other technologies.
- *Life is found in layers and the TM Technique gives us access to the full range of our awareness and thoughts.*

Except where otherwise noted, the contents of this document are Copyright 2012 Marty Stepp, Jessica Miller, Victoria Kirst and Roy McElmurry IV. All rights reserved. Any redistribution, reproduction, transmission, or storage of part or all of the contents in any form is prohibited without the author's expressed written permission. Slides have been modified for Maharishi International University Computer Science course CS472 in accordance with instructors agreement with authors.

Maharishi International University  
Fairfield, Iowa © 2024



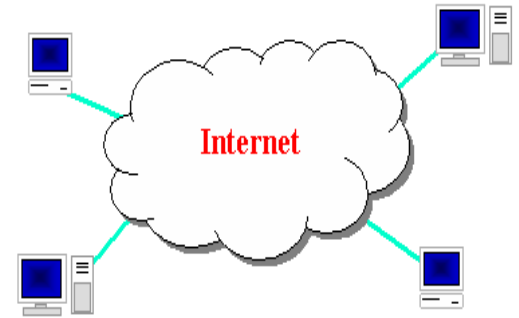
All rights reserved. No part of this slide presentation may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying or recording, or by any information storage and retrieval system, without permission in writing from Maharishi International University.

# Main Point Preview

- The internet is a global computer network that uses the IP protocol to uniquely identify computers on the network. Through the TCP protocol each IP address can work with multiple services at the same time. One of these services is the HTTP protocol which is used by the World Wide Web to transport HTML pages.
- There are many layers of the Internet. *Familiarity with the deep layers is necessary for optimal understanding and use of the surface layers. TM settles our mind so that it is more connected to the deep layers that are the basis of the more expressed layers.*

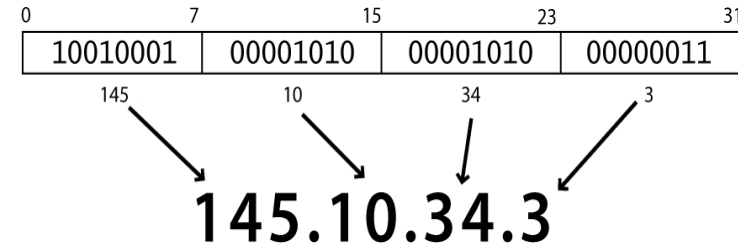
# The Internet

- A connection of computer networks using the Internet Protocol (IP)
- layers of communication protocols:  
IP → TCP/UDP → HTTP/FTP/...
- What's the difference between the Internet and the World Wide Web (WWW)?
  - The Web is the collection of web sites and pages around the world
  - The Internet is larger and also includes other services such as email, chat, online games, etc.



# Internet Protocol (IPv4)

- The underlying system of communication for all data sent across the Internet.
  - Each device has a 32-bit IP address written as four 8-bit numbers (0-255)
- There are two types of IP addresses
  - servers often have static IP address
  - Users usually get a dynamic IP address from their ISP
- Find out your local IP address:
  - In a terminal, type: ipconfig (Windows) or ifconfig (Mac/Linux)
- IPv6 addresses are 128-bit IP address written in hexadecimal and separated by colons.
  - An example IPv6 address could be written like this: 3ffe:1900:4545:3:200:f8ff:fe21:67cf





# Transmission Control Protocol (TCP)

- Adds multiplexing, guaranteed message delivery on top of IP
  - IP is stateless – no memory of sent messages or success of delivery
  - TCP is stateful – tracks status of messages until determine success of delivery
- Multiplexing: multiple programs using the same IP address
  - port: a number given to each program or service
  - port 80: web browser (port 443 for secure browsing)
  - port 25: sending email
  - port 22: ssh
  - port 21: File transfer (FTP)
  - port 5190: AOL Instant Messenger
  - [more common ports](#)
- Some programs (games, streaming media programs) use simpler UDP protocol instead of TCP

# Web Servers and Browsers

- **Web server:** software that listens for web page requests
  - [Apache](#)
  - Microsoft Internet Information Server (IIS)
- **Web browser:** fetches/displays documents from web servers
  - [Mozilla Firefox](#)
  - Microsoft [Internet Explorer](#) (IE)
  - Apple [Safari](#)
  - [Google Chrome](#)
  - [Opera](#)
  - [Browser usage statistics](#)

# Domain Name System (DNS)

- A set of servers that map written names to IP addresses
  - Example: [www.miu.edu](http://www.miu.edu) → 192.249.118.206

# Uniform Resource Locator (URL)

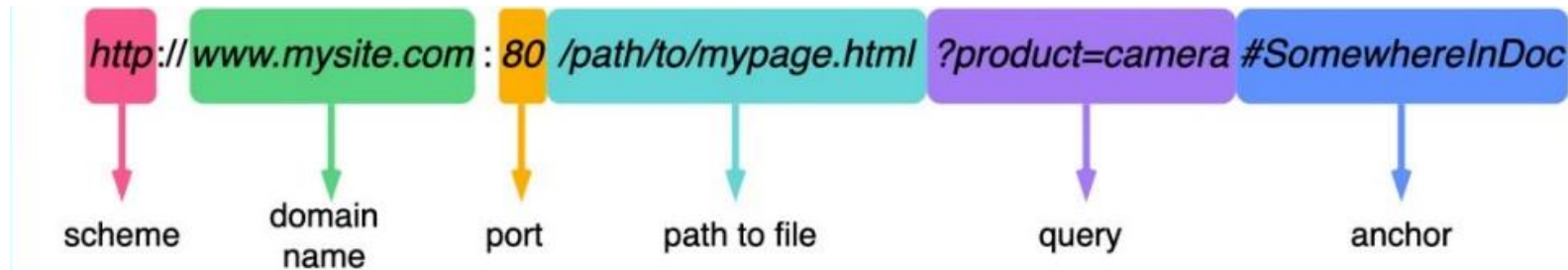
- An identifier for the location of a document on a web site
  - Example URL: <http://www.abc.com/info/index.html>

~~~~ ~~~~~~  
protocol    host            path

- Upon entering this URL into the browser, it will:
  - Ask the DNS server for the IP address of [www.abc.com](http://www.abc.com)
  - Connect to that IP address at port 80
  - Ask the server to GET /info/index.html
  - Display the result page on the screen

# More Advanced URLs

- **Anchor:** jumps to a given section of a web page
  - <http://www.textpad.com/download/index.html#downloads7>
  - fetches index.html then jumps down to <a name="downloads7"></a>
- **Port:** for web servers on ports other than the default 80
  - <http://www.cs.miu.edu:8080/secret/money.txt>
- **Query string:** a set of parameters passed to a web program
  - <http://www.google.com/search?q=happy+person&start=10>



# Hypertext Transport Protocol (HTTP)

- The protocol consists of a set of commands that a computer can send to a server to request files
- Example HTTP commands (your browser sends these internally):
  - **GET**: Requests a specific file or resource from the server
  - **POST**: Submits form information to the server (not idempotent)
  - **PUT**: Uploads a file to the server (idempotent)
  - **HEAD**: Requests information about a file from the server, but not the file's entire contents.
- ...

# HTTP Request / Response

- **Request:**

```
GET /index.html HTTP/1.1  
HOST: mumstudents.org
```

- **Response:**

```
HTTP/1.1 200 OK  
Date: Sun, 31 May 2020 12:10:06 GMT  
Server: Apache  
Last-Modified: Mon, 03 Oct 2016 15:51:02 GMT  
ETag: "3ae-53df7e933e013"  
Accept-Ranges: bytes  
Content-Length: 942  
Vary: Accept-Encoding  
Content-Type: text/html
```

```
<!DOCTYPE html>  
<html>  
...
```

# Inside Developer Tools

The screenshot shows a web browser window displaying the Maharishi International University website. The browser's address bar shows the URL `https://www.cs.miu.edu/mum/`. The website header includes the university's logo and the text "MS in Computer Science COMPUTER PROFESSIONALS PROGRAM<sup>SM</sup> Formerly Maharishi University of Management". Below the header is a navigation menu with links: Home, People /Directory, Graduate Courses, Undergrad Courses, Quick Links, and News.

The Firefox Developer Tools are open, and the Network tab is selected. The Network tab displays a list of network requests. The table below shows the details of these requests:

| Status | Method | Domain         | File                                    | Cause      | Type | Transferred | Size      | 0 ms | 2.26 s |
|--------|--------|----------------|-----------------------------------------|------------|------|-------------|-----------|------|--------|
| 304    | GET    | www.cs.miu.edu | video-lightbox.js?ver=3.1.6             | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | nivo-slider.css?ver=3.15.3              | stylesheet | css  | cached      | 2.21 KB   | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | public.css?ver=3.15.3                   | stylesheet | css  | cached      | 6.56 KB   | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | default.css?ver=3.15.3                  | stylesheet | css  | cached      | 2.19 KB   | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | effect.min.js?ver=1.11.4                | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | effect-slide.min.js?ver=1.11.4          | script     | js   | cached      | 962 B     | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | effect-highlight.min.js?ver=1.11.4      | script     | js   | cached      | 789 B     | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | effect-fold.min.js?ver=1.11.4           | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | effect-blind.min.js?ver=1.11.4          | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | bg-show-hide.js?ver=4.9.7               | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | skip-link-focus-fix.min.js?ver=20130115 | script     | js   | cached      | 557 B     | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | jquery.cycle2.min.js?ver=2.1.6          | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | custom.min.js?ver=1.0                   | script     | js   | cached      | 294 B     | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | navigation.min.js?ver=20120206          | script     | js   | cached      | 919 B     | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | wp-embed.min.js?ver=4.9.7               | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | jquery.nivo.slider.pack.js?ver=3.15.3   | script     | js   | cached      | 0 B       | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | gp_photo-950x400.jpg                    | img        | jpeg | cached      | 80.15 KB  | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | eGraduation-2016-46-950x400.jpg         | img        | jpeg | cached      | 91.13 KB  | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | dsc09834-1-2-950x400.jpg                | img        | jpeg | cached      | 82.05 KB  | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | eIMG_0515-950x400.jpg                   | img        | jpeg | cached      | 67.95 KB  | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | eIMG_1235-1-950x400.jpg                 | img        | jpeg | cached      | 82.48 KB  | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | eIMG_1238-1-950x400.jpg                 | img        | jpeg | cached      | 106.31 KB | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | cropped-cs-miu-logo-v1-5.jpg            | img        | jpeg | cached      | 14.89 KB  | 0 ms | 0 ms   |
| 404    | GET    | www.cs.miu.edu | fontawesome-webfont.woff2?v=4.7.0       | font       | html | 7.13 KB     | 6.93 KB   | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | arrows.png                              | img        | png  | cached      | 824 B     | 0 ms | 0 ms   |
| 304    | GET    | www.cs.miu.edu | bullets.png                             | img        | png  | cached      | 1.25 KB   | 0 ms | 0 ms   |
| 404    | GET    | www.cs.miu.edu | fontawesome-webfont.woff?v=4.7.0        | font       | html | 7.12 KB     | 6.93 KB   | 0 ms | 0 ms   |
| 200    | GET    | www.cs.miu.edu | cropped-cropped-logo.png                | img        | png  | cached      | 24.07 KB  | 0 ms | 0 ms   |
| 200    | GET    | www.cs.miu.edu | cropped-cropped-logo.png                | img        | png  | cached      | 24.07 KB  | 0 ms | 0 ms   |

At the bottom of the Network tab, a summary bar shows: 42 requests, 817.65 KB / 23.76 KB transferred, Finish: 2.26 s, DOMContentLoaded: 1.75 s, load: 2.21 s.



# HTTP Status Codes

- When you request a document from a web server, it sends this document back to you, along with a number called HTTP status code.

| Number                        | Meaning                                     |
|-------------------------------|---------------------------------------------|
| 200                           | OK                                          |
| 301-303                       | page has moved (permanently or temporarily) |
| 403                           | you are forbidden to access this page       |
| 404                           | page not found                              |
| 500                           | internal server error                       |
| <a href="#">complete list</a> |                                             |

# MIME Types

- A two-part identifier which many web protocols use to categorize each type of data.

| MIME Type                        | File extension(s) | Description           |
|----------------------------------|-------------------|-----------------------|
| application/octet-stream         | .exe              | Executable programs   |
| audio/mpeg                       | .mp3, .mpg        | MPEG or MP3 music     |
| image/gif, image/jpeg, image/png | .gif, .jpg, .png  | GIF, JPEG, PNG images |
| text/css                         | .css              | Style sheets          |
| text/html                        | .html, .htm, .php | Web pages             |
| <a href="#">MIME Type list</a>   |                   |                       |

# Web Languages / Technologies

- Hypertext Markup Language (HTML): used for writing web pages
- Cascading Style Sheets (CSS): stylistic info for web pages
- JSP/EJB/Spring, Node/Express, Ruby, .Net, PHP Hypertext Processor (PHP) ...: dynamically create pages on a web server
- JavaScript, jQuery, React, Angular: interactive and programmable web pages
- Asynchronous JavaScript and XML (Ajax): accessing data for web applications
- Extensible Markup Language (XML): metalanguage for organizing data
- JavaScript Object Notation (JSON): lightweight data-interchange format that is largely replacing XML in modern web apps

# How Browsers Display a Page

- . User machines have IP address on the Internet
- . Server machines have IP address and Domain Name
- . Domain names and IP addresses are registered at global DNS Server
- . When the user opens a browser window and asks for [www.test.com](http://www.test.com)
- . First, the browser will check the local DNS (host file) for the IP address of that domain
- . If not found, it will connect to ISP and ask it for the DNS
- . Once retrieved, the browser will send another request to that server
- . Requests are delivered by the IP protocol, collected by the TCP protocol, and processed by HTTP or HTTPS protocol
- . The server will send the browser a response with HTML code.
- . The browser will interpret the HTML code line by line and start building the web page.
- . For every resource not found in the browser cache, the browser will send a new request to the server again asking for that resource and so on.

# Main Point

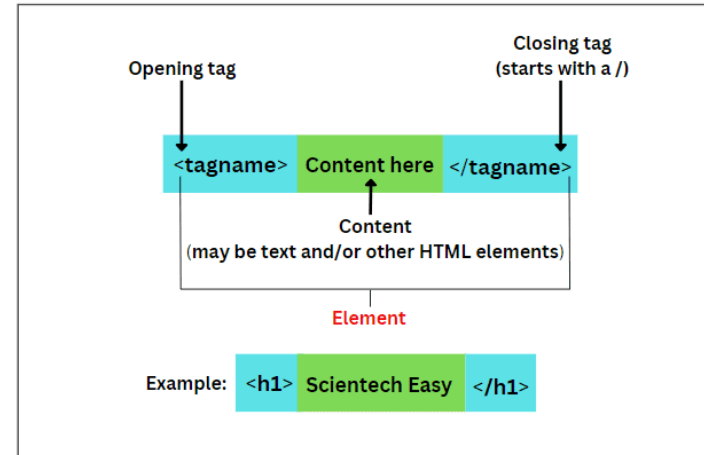
- The internet is a global computer network that uses the IP protocol to uniquely identify computers on the network. Through the TCP protocol each IP address can work with multiple services at the same time. One of these services is the HTTP protocol which is used by the World Wide Web to transport HTML pages.
- There are many layers of the Internet. *Familiarity with the deep layers is necessary for optimal understanding and use of the surface layers. TM settles our mind so that it is more connected to the deep layers that are the basis of the more expressed layers.*

# Main Point Preview

- The Hyper Text Markup Language uses tags to demarcate different sections of a text. An HTML page always starts with a `<html>` tag, inside of which it has a `<head>` tag to describe the page, and a `<body>` tag of the contents that will actually be displayed. These are the tags you will use for every HTML page. This is a foundational concept
- *Well begun is half done. Start with a good foundation and build upon that.*

# Hypertext Markup Language (HTML)

- Describes the content and structure of information on a web page
- Surrounds text content with opening and closing tags
- Each tag and its content is called an element
  - Syntax: **<tag>**content **</tag>**
  - Example: **<p>**This is a paragraph**</p>**
- Most whitespace is insignificant in HTML
  - (ignored or collapsed to a single space)
- The newest version is HTML5



# Structure of an HTML5 page

- An HTML page is saved into a file ending with extension **.html**
- **DOCTYPE** tag tells browser the HTML version.
  - In this case, HTML5.
- The **header** describes the page and the **body** contains the page's contents

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8"/>
    <title></title>
  </head>
  <body>
    Page contents
  </body>
</html>
```



# Metadata

- Data about data - Information describes the page itself
  - Placed in the head section of your HTML page
- meta tags often have both the name and content attributes
- HTML5 introduced a way for designers to control the viewport
  - (the user's visible area of a web page), through the <meta> tag

```
<meta charset="utf-8"/>
```

```
<meta name="description" content="Learn HTML"/>
```

```
<meta name="keywords" content="HTML,CSS"/>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
```

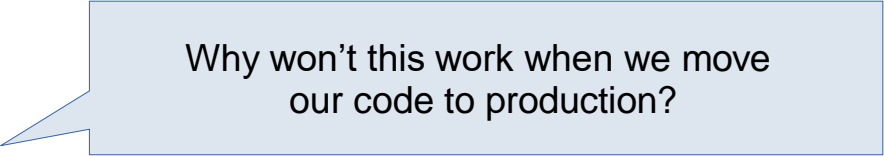
# Relative vs Absolute URL

- **Relative URL**

- index.html (path relative)
- graphics/image.png
- ../about.html (directory relative)
- ../../stories/stories.html
- /image.png (root relative)

- **Absolute URL**

- <http://www.mysite.com>
- C:\website\images\image.png



Why won't this work when we move our code to production?

# Favorites Icon (favicon)

- The link tag, in the head section, attaches another file to the page
- In this case, an icon for the browser title bar and bookmarks
  - E.g. [https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5\\_link\\_sizes](https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_link_sizes)

```
<link href="filename" type="MIME type" rel="relationship"/>  
<link href="yahoo.gif" type="image/gif" rel="shortcut icon"/>
```



# Common Mistake

- Placing a leading “/” on a relative URL
- Say you are at index.html and want to link to smiley which is at:
  - <http://www.myexample.com/mysite/files/index.html>
  - <http://www.myexample.com/mysite/files/images/smiley.gif>

- The following fails:

``

Tries to GET:

<http://www.myexample.com/images/smiley.gif>

- The following works:

`` (correct for ‘mysite’)

# Block Level Elements



- A block-level element always starts on a new line and takes up the full width available (stretches out to the left and right as far as it can).
- Examples of block-level elements:

`<div>`

`<h1> - <h6>`

`<p>`

`<form>`

# Paragraph <p>



- Creates a paragraph in a block of text
  - Placed in the body of the page
  - See examples at [w3schools](https://www.w3schools.com/html/html_tags.asp)

<p>

Lorem quis lorem. Pellentesque ultrices nunc id mauris  
posuere pulvinar.

</p>

# Headings <h1>,<h2>, ..., <h6>

- Headings are used to give a title (heading) to major areas of the page (block)

Note: only use one <h1> tag per page, as it describes “the page”

```
<h1>Maharishi University</h1>
```

```
<h2>Department of Computer Science</h2>
```

```
<h3>WAP Course</h3>
```

Maharishi University

Department of Computer Science

WAP Course

# Horizontal Rule <hr />



- A line to separate sections
  - It represents a paragraph-level thematic break, e.g. a scene change in a story, or a transition to another topic within a section of a reference book

```
<h1>CS472 Web Programming</h1>
```

```
<p>Very good, fun course - HTML, CSS, JavaScript, JSP, Servlets</p>
```

```
<hr/>
```

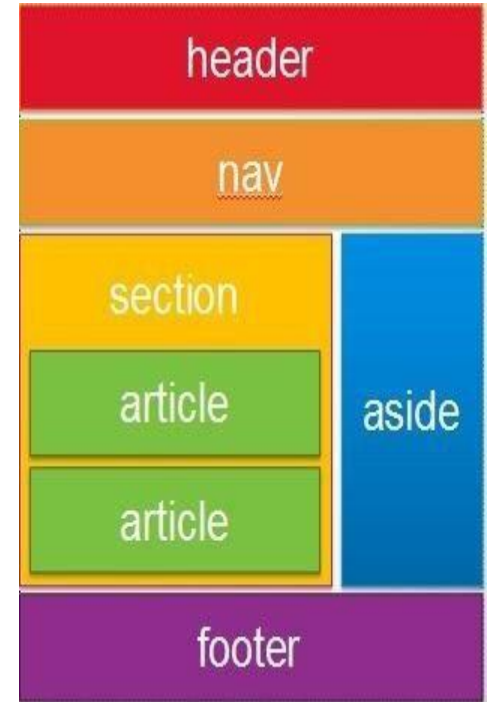
```
<h1>CS421 Modern Programming Practices</h1>
```

```
<p>Profound course - OOAD, Java8 features</p>
```



# HTML5 New Semantic Elements

- They generally have no default outward appearance on the page, instead they give insight into the structure of the page.
  - section: defines a section in a document.
  - header: specifies a header for a document or section.
  - footer: specifies a footer for a document or section.
  - nav: defines a set of navigation links.
  - aside: defines some content aside from the content it is placed in (like a sidebar).
  - article: specifies independent, self-contained content.
  - [More Semantic Tags](#)



# Main Point

- The Hyper Text Markup Language uses tags to demarcate different sections of a text. An HTML page always starts with a `<html>` tag, inside of which it has a `<head>` tag to describe the page, and a `<body>` tag of the contents that will actually be displayed. These are the tags you will use for every HTML page. This is a foundational concept
- *Well begun is half done. Start with a good foundation and build upon that.*

# Main Point Preview

- More common tags include inline elements, lists, tables. The most important concept is to use tags based on their semantics (meaning), not based on their visual effect (which can easily be changed). By using tags for their meaning, clients (including non visual) will better understand the meaning of your page and use it more effectively.
- *Clear and settled awareness allows us to better grasp the meaning of information and take the right actions for success.*

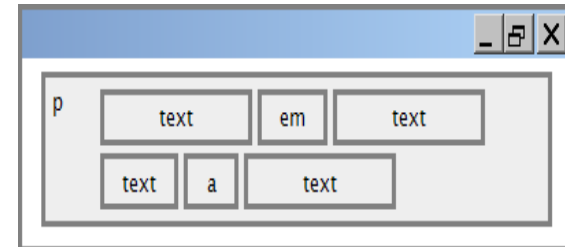
# Inline Elements



- An inline element does not start on a new line and only takes up as much width as necessary.
- Examples of inline elements:
  - `<span>`
  - `<a>`
  - `<img>`

# Block vs Inline Elements

- Block elements contain an entire large region of content
  - Examples: paragraphs, lists, table cells
  - The browser places a margin of whitespace between block elements for separation, normally a newline
- Inline elements affect a small amount of content
  - Examples: bold text, code fragments, images
  - The browser allows many inline elements to appear on the same line
  - Must be nested inside a block element



# Images `<img />`

- Inserts a graphical image into the page (inline)
  - Another get request
  - The **src** attribute specifies the image URL
  - HTML5 also requires an **alt** attribute
    - describing the image
  - The **title** attribute is an optional tooltip
    - (on ANY element)



```

```

# Links <a>

- Links, or "anchors", to other pages (inline)
  - href can be absolute or relative URL
  - Anchors are inline elements

In HTML5, you can wrap links around "block-level" elements

<p>

```
<a href="story1.html">Bruce Wayne, the
richest man in Gotham City, is the alter
ego of Batman.  Bam! </a>
```

</p>

# Line Break `<br />`

- Forces a line break in the middle of a block element (inline)

`<p>`Teddy said it was a hat, `<br />` So I put it on. `</p>`

`<p>`Now Daddy's saying, `<br />` Where the heck's the toilet plunger gone? `</p>`

`<br />` should not be used to separate paragraphs or used multiple times in a row to create spacing

- Warning: Don't over-use br
  - (guideline:  $\geq 2$  in a row is bad)



# Phrase elements : `<em>`, `<strong>`

- `em`: emphasized text
  - usually rendered in italic
- `strong`: strongly emphasized text
  - usually rendered in bold



```
<p>HTML is  
    <em>really</em>, <strong>REALLY</strong>  
    fun!  
</p>
```

As usual, the tags must be properly nested for a valid page.

HTML is *really*, **REALLY** fun!

# Nesting Tags

- Tags must be correctly nested
  - a closing tag must match the most recently opened tag

The browser may render it correctly anyway, but it is invalid HTML

- Bad nesting:
  - `<p> What a <em> lovely </p> day </em>`
- Good nesting:
  - `<p> What a <em> lovely </em> day </p>`

# Comments <!-- -->

- Comments to document your HTML file
  - or "comment out" text
- Useful at the top of page and for disabling code

```
<!-- This is a comment -->
```

```
<p>
```

```
WAP courses are <!-- NOT --> a lot of fun!
```

```
</p>
```

# HTML Character Entities



- A way of representing any Unicode character within a web page
  - A complete list of HTML entities

| character(s) | entity                     |
|--------------|----------------------------|
| < >          | &lt; &gt;                  |
| é è ñ        | &eacute; &egrave; &ntilde; |
| ™ ©          | &trade; &copy;             |
| π δ Δ        | &pi; &delta; &Delta;       |
| ℥            | &#1048;                    |
| " &          | &quot; &amp;               |

How would &amp; display on a web page?

# HTML-encoding Text



- To display the following html code on a web page, its special characters must be encoded as shown below

```
<p> <a href="http://google.com/"> Google</a> </p>
```

&lt;p&gt;

&lt;a href=&quot;<http://google.com/>&quot;&gt; Google&lt;/a&gt;

&lt;/p&gt;

# Inline Quotations <q>



- A short quotation (inline)

<p>

Quoth the Raven, <q>Nevermore.</q>

</p>

Quoth the Raven, "Nevermore."

- Why not just write the following?

<p>Quoth the Raven, "Nevermore."</p>

Using <q> allows us to apply CSS styles to quotations

# Quotations <bblockquote>



- A lengthy quotation (block)

<p>As Lincoln said in his famous Gettysburg Address:</p>

<blockquote>

Fourscore and seven years ago, our fathers brought forth on this continent a new nation, conceived in liberty, and dedicated to the proposition that all men are created equal.

</blockquote>

As Lincoln said in his famous Gettysburg Address:

Fourscore and seven years ago, our fathers brought forth on this continent a new nation, conceived in liberty, and dedicated to the proposition that all men are created equal.

# Computer Code `<code>`



- A short section of computer code (usually shown in a fixed-width font) (inline)

wi

Wi

The `ul` and `ol` tags make lists.

`<p>`

The `<code>ul</code>` and `<code>ol</code>` tags make lists.

`</p>`



# Preformatted Text <pre>



- A large section of pre-formatted text (block)
  - a fixed-width font (usually Courier), and it preserves both spaces and line breaks.

<pre>

**Steve Jobs speaks loudly reality**  
**distortion Apple fans bow down**

</pre>

```
Steve Jobs speaks loudly reality
distortion Apple fans bow down
```

# Unordered Lists `<ul>` `<li>`



- `ul` represents a bulleted list of items (block)
- `li` represents a single item within the list (block)

```
<ul>  
  <li>No shoes</li>  
  <li>No shirt</li>  
  <li>No problem!</li>  
</ul>
```

- No shoes
- No shirt
- No problem!

# Ordered List <ol>



- ol represents a numbered list of items (block)
- We can make lists with letters or Roman numerals using CSS (later)

```
<p>Concise business model:</p>
<ol>
  <li>Find customers</li>
  <li>Make profits</li>
</ol>
```

Concise business model:

1. Find customers
2. Make profit!

# Nested Lists



```
<ul>
  <li>Simpsons:
    <ul>
      <li>Homer</li>
      <li>Marge</li>
    </ul>
  </li>
  <li>Family Guy:
    <ul>
      <li>Peter</li>
      <li>Lois</li>
    </ul>
  </li>
</ul>
```

Important: nested lists are inside list items.  
Not directly inside the other list!

- Simpsons:
  - Homer
  - Marge
- Family Guy:
  - Peter
  - Lois

# Definition Lists <dl> <dt> <dd>

- dl represents a list of definitions of terms (block)
- dt represents each term, and dd its definition



```
<dl>  
  <dt>newbie</dt>  
  <dd>one who does not have mad skills</dd>  
  <dt>own</dt>  
  <dd>to soundly defeat</dd>  
  <dt>frag</dt>  
  <dd>a kill in a shooting game</dd>  
</dl>
```

```
newbie  
    one who does not have mad skills  
own  
    to soundly defeat  
frag  
    a kill in a shooting game
```

# Abbreviation <abbr>



- An abbreviation, acronym, or slang term (inline)

```
<p> Safe divers always remember to check their  
    <abbr title="Self-Contained Underwater Breathing  
Apparatus">SCUBA</abbr> gear.  
</p>
```

Safe divers always remember to check their SCUBA gear.

Self-Contained Underwater Breathing Apparatus

# HTML Tables <table> <tr> <td>



- A 2D table of rows and columns of data (block element)
- table defines the overall table, tr each row, and td each cell's data
- tables are intended for displaying large row/column data sets

```
<table>
  <tr>
    <td>1,1</td>
    <td>1,2 okay</td>
  </tr>
  <tr>
    <td>2,1 real wide</td>
    <td>2,2</td>
  </tr>
</table>
```

|               |          |
|---------------|----------|
| 1,1           | 1,2 okay |
| 2,1 real wide | 2,2      |

# Table Headers <th> Captions <caption>



- th cells in a row are headers
  - Appear bold by default
- A caption labels the meaning of the table

| Column 1      | Column 2 |
|---------------|----------|
| 1,1           | 1,2 okay |
| 2,1 real wide | 2,2      |

```
<table>
  <caption>My important data</caption>
  <tr>
    <th>Column 1</th>
    <th>Column 2</th>
  </tr>
  <tr>
    <td>1,1</td>
    <td>1,2 okay</td>
  </tr>
  <tr>
    <td>2,1 real wide</td>
    <td>2,2</td>
  </tr>
</table>
```



# The rowspan and colspan attributes



**colspan** makes a cell occupy multiple columns  
**rowspan** makes a cell occupy multiple rows

```
<table>
  <tr>
    <th>Column 1</th>
    <th>Column 2</th>
    <th>Column 3</th>
  </tr>
  <tr>
    <td colspan="2">1,1-1,2</td>
    <td rowspan="3">1,3-3,3</td>
  </tr>
  <tr>
    <td>2,1</td>
    <td>2,2</td>
  </tr>
  <tr>
    <td>3,1</td>
    <td>3,2</td>
  </tr>
</table>
```

Column 1 Column 2 Column 3

1,1-1,2

2,1

2,2

1,3-3,3

3,1

3,2

Column 1

Column 2

Column 3

1,1-1,2

2,1

2,2

1,3-3,3

3,1

3,2

# Don't use Tables for Layout

- Tables appear to be an easy way to achieve grid-like page layouts (used to be only way)
  - Some "newbie" web pages may still do this
  - But a table has semantics; it should be used only to represent an actual table of data
  - Instead of tables, use divs, widths/margins, flexbox, grid, floats, etc to perform layout

# W3C HTML Validator



- Checks your HTML code to see if it follows official HTML syntax
- More picky than the browser, which may render bad HTML correctly

<p>

[<a href="https://validator.w3.org/check?url=yourWebsiteURL">](https://validator.w3.org/check?url=yourWebsiteURL)

 [Validate" />](http://mumstudents.org/cs472/2013-09/images/w3c-html.png)

</a>

</p>

- Note that the validation website expects requests to come from a page that was served by a web server. Will not work with an html page opened from a file.

# Main Point

- More common tags include inline elements, lists, tables. The most important concept is to use tags based on their semantics (meaning), not based on their visual effect (which can easily be changed). By using tags for their meaning, clients (including nonvisual) will better understand the meaning of your page and use it more effectively.
- *Clear and settled awareness allows us to better grasp the meaning of information and take the right actions for success.*

# CONNECTING THE PARTS OF KNOWLEDGE WITH THE WHOLENESS OF KNOWLEDGE

## *Layers of Abstraction*

1. HTML is the basis of web programming; every web page is composed of HTML.
  2. To be an effective web programmer you also must understand the deeper underlying realities of HTTP, TCP, and DNS.
- 
3. Transcendental consciousness is when our mind is in contact with the deepest underlying reality, the unified field.
  4. Impulses within the Transcendental field: the infinite dynamism of the unified field constantly expresses itself as all the layers of the universe
  5. Wholeness moving within itself: In Unity Consciousness, one experiences that all these layers are ultimately composed of pure consciousness, our own Pure Awareness.
- 