Announcements

- Class Web Site:
  - You can find this link at the end of the main passport site
- E-mail Account
  - Get your own e-mail account if you don’t have one
- Announcements’ section in web site
- Rules regarding Forum Use
- Academic Integrity
- Reminder to your parents
- Be on time
- Slides
Fundamentals: Client/Server

- Client and server are two terms frequently used
- Client/Server Model
- Client/Server model when talking about software
- Client/Server model when talking about hardware
Fundamentals: IP Addresses

- **IP Address** - Unique address for machine on internet
  - Get from ISP when connecting to internet
  - Allows network to find your machine

- **Format**
  - 32-bit unsigned integer ➔ 128.8.128.8

- **Domain Name**
  - Text name corresponding to the numeric IP address
  - Example: wikipedia.org

- **Name and address for local machine**
  - localhost
  - 127.0.0.1

- **Running out of 32-bit IP addresses**
Fundamentals: Web Server

- Web Server –
  - computer program that delivers (serves up) web pages.
  - It is like a person that is in charge of a warehouse

- Four popular Web Server Programs
  - IIS – Internet Information Services
  - Sun Java System Web Server

- Web server statistics

- Some web server statistics (by domain)
Fundamentals: DNS

- DNS – Domain Name Systems
- Protocol for translating domain names to IP addresses
  - Example: cs.umd.edu → 128.8.128.44
- Multiple DNS servers on internet
- DNS server may need to query other DNS servers
  - *edu* DNS server queries *umd.edu* server to find *cs.umd.edu*
Fundamentals: URLs

- URL – Uniform Resource Locators
- Represent web resources
  - Arbitrary files
  - Web pages
- Examples
  - https://login.yahoo.com/
  - file://dir/my.txt
Fundamentals: URL Structure

- URL consists of
  - Protocol
    - http
    - ftp
    - https (secure http)
    - file
    - ...
  - IP address (or domain name)
  - Port (optional most of the time)
  - path
Firefox

- Browser we will use

- Extensions we would like to have
  - Error Console
HTML

- Language used to define web pages
- What the server sends to the browser
- Browser reads HTML and renders the page
  - May require downloading data from server (e.g., images)
HTTP

- Hypertext Transfer Protocol (HTTP) – protocol that defines how user agents (e.g., browser) and web server can communicate
- HTTP is a request/response protocol between clients and servers
- Some methods (operations) defined as part of the protocol
  - GET – Use to download a resource (e.g., image, web page). Most common method used.
  - HEAD – Returns only the header
  - POST – Submits data (e.g., form data) to the server
- Do not confuse with HTML
- Demo
Creating Web Pages

- HTML - Hypertext Markup Language
- HTML Standard
  - Developed by the World Wide Web Consortium (W3C)
  - [http://www.w3.org](http://www.w3.org)
  - Latest version HTML 4.01
- Document is described through a series of commands and directives present in a text file.
- HTML goal is to describe structure only. Presentation should be left to cascading style sheets.
- When interpreted by an HTML **viewer**, those commands determine the appearance of the page
- HTML documents are entirely ASCII text
- Commands are explicitly inserted
- Great HTML/CSS tutorial site:
Three versions of HTML

- HTML 4.01 Strict (excludes deprecated tags and attributes)
- HTML 4.01 Transitional (less restrictive including appearance elements)
- HTML 4.01 Frameset (identical to transitional but allows `<body>` to be replaced with `<frameset>`)  

Web Standards Project ([www.webstandards.org](http://www.webstandards.org])

- Industry watchdog convincing web browsers developers to adhere to web standards.

HTML 4.01 is the last version for HTML. Next version is XHTML 1.0

XHTML

- Uses same tags as HTML 4.01
- Enforces rules like closing tags, tags in lowercase, and others.

We will use XHTML in this class
Validation

- You can use W3C Markup Validation Service (http://validator.w3.org/) to validate your html.
- Also through firefox you can use tidy for html validation. Tidy also provides suggestions for code that cannot be validated.
**HTML Tags**

- Tag – specifies a command or directive. It surrounds content and apply meaning to that content.

- General format:
  ```html
  <elementName attributes>
  ```

- Most HTML elements have two tags:
  - start tag and end tag
  - Example: `<h1> text </h1>`

- Tags and attributes will be in lowercase (XHTML requirement)

- Some tags are self-closed (ending them in `/`):
  ```html
  <hr />
  <br />
  <meta />
  <img />
  ```
Attributes

- An attribute extends or modify a tag
- Attributes
  - Only appear in the start tag
  - You can have several attributes in one tag each separated by spaces
  - Order is immaterial
  - Some take values which are specified after an =
- General format
  - `<ELEM ATTR=“attrValue”>Displayed Text</ELEM>`
- Example
  - `<img src="bear.gif" width="100" height="75" alt="bear image" />`
- All attribute values will be enclosed in “ “ for XHTML compliance.
NestedTags/Spaces/Comments

- Nested tags are possible but don’t overlap sets of them. Avoid the following:
  \[<i><b>Message</b></i></b>

- Browser Processing
  - Multiple spaces are converted to one space
    John Mary Peter
  - Line returns are ignored
  - Unrecognized tags are ignored
  - Comments
    - Represented by \(<!-- -->\) Note: (two sets of double -)
  - Examples
    - \(<!--The html code example starts at this point-->\)
  - Comments can not be nested
HTML Basic Skeleton

- An html document has two main parts.
  - **Header** – provides information about the document
  - **Body** – contents of the page

**Example 1 (htmlDoc1.html)**

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <title>Template</title>
  </head>

  <body>
    <!-- HTML CODE HERE -->
  </body>
</html>
```

- Let’s validate the above document
How to Develop HTML Documents

- **Text Editor**
  - Any text editor (e.g., wordpad, notepad, pico, etc.)

- **HTML Editors**
  - Utilities designed to write HTML
  - Examples: CoffeeCup HTML Editor, HTMLjive

- **Authoring tools**
  - Frontpage – Good for beginners
  - Dreamweaver – Fairly complex but powerful
  - NVU – Free and available for (Windows, Linux, Mac)

- List of editors can be found at:

- **Recommended:**
Frequently Used Tags

- `<head> </head>`
  - It does not generate displayed contents
  - Contains other tags (e.g., `<title> </title>`)
- `<title> </title>`
  - Part of the header
  - It is required
  - Search engines depend on it, so use meaningful titles
Frequently Used Tags

- Heading tags
  - `<h1> text </h1>`
  - `<h2> text </h2>` … and so on until
  - `<h6> text </h6>`
  - Higher numbers imply smaller headers
- Paragraph tag
  - `<p> paragraph </p>`
- Code – Use to define computer code
  - `<code> </code>`
- Horizontal Line - `<hr />`
Frequently Used Tags

- Emphasis
  - `<em> text here </em>` Text usually rendered in italics
  - `<strong> text here </strong>` Text usually rendered in bold

- Super/Sub script
  - `<sub> text here </sub>`
  - `<sup> text here </sup>`

- Quotations
  - `<q> quote here </q>`

- Line Breaks
  - `<br />`

- Verbatim (text displayed exactly as it appears)
  - `<pre> text here </pre>`

- Example: HtmlDoc.html
Lists

- **Unordered lists**
  - `<ul> </ul>` tags to represent beginning and end
  - `<li> </li>` to represent elements in the list
  - **Example: Lists.html**

- **Ordered lists**
  - `<ol> </ol>` tags to mark beginning and end
  - `<li> </li>` to represent elements in the list

- **Definition lists**
  - Consist of terms and definitions like in a glossary
  - Tags - `<dl> </dl>`
  - Terms specified using `<dt> </dt>` and definitions with `<dd> </dd>`
  - **Example: DefNestedLists.html**

- **Nested lists** (See previous example)
Image Inclusion

- We can include an image using the `img` tag
  `<img src="testudo.jpg" alt="Testudos' image" />

- **Example:** `Image.html`

- Although the width and height attributes are not required they are highly recommended. (They can also be set through CSS).
Links

- Link – connection between web resources
- Hypertext links are created using the `<a>` (anchor) tag
- The link can be text:
  - Notice that you need to specify the protocol (http://)
  - **Example: Link.html**
  - The URL can be absolute or relative
- The link can be an image:
  - `<a href="http://www.umd.edu"><img src="testudo.jpg" alt="Testudos' image" /></a>`