Introduction to Web Engineering SENG2050/6050

Lecture 1b
An overview of WWW, HTML, etc.

Lecture 1b: An Overview

- How Internet works
- How the World Wide Web (WWW) works
- HTML

Review...

Web Engineering

- Tools and methodologies to design and implement complex web sites
- Technologies to support the on-line input and update of content and various categories of designers, developers and users
- Technologies and architectures to support access from various forms of devices
- Technologies to support context-aware access to information

The Internet

- A network of networks
- Adjacent networks are linked by gateways/routers
- The Internet "network" is a packet switching network
- As the network communicates, it needs rules for communicating: Protocols
  - ✓The Internet Protocol (IP) is a data-oriented protocol used for communicating data across. IP divides data into packets and controls their delivery.
  - ✓The Transfer Control Protocol (TCP) ensures that "lost" packets are resent.

The Internet

… Protocols

- ✓The Internet Control Message Protocol (ICMP) is used between routers to communicate information about network load, dead connections, etc. – this is used by IP to determine the best route for each packet.
- ✓User Datagram Protocol (UDP) is not as reliable as TCP but it is faster.

➢Messages can then jump from network to network until they reach their destinations

Transaction Control Protocol (TCP)

- Breaks each message up into manageable pieces before sending – called packets.
- Reassembles the packets into the original message at the receiving end.
- Ensures that all packets are delivered – if a packet goes missing, TCP makes sure it is resent.
- Gives the illusion that there is a continuous connection between the sending and receiving computers.

Internet Protocol (IP)

- IP is responsible for determining how each packet is delivered to the receiving computer – includes how the address of the receiving computer is given.
- Each packet can take a different route across the Internet.
- IP may split a packet into smaller packets to cross a particular network.
- Does not maintain connections nor guarantee delivery – that's why we need TCP.

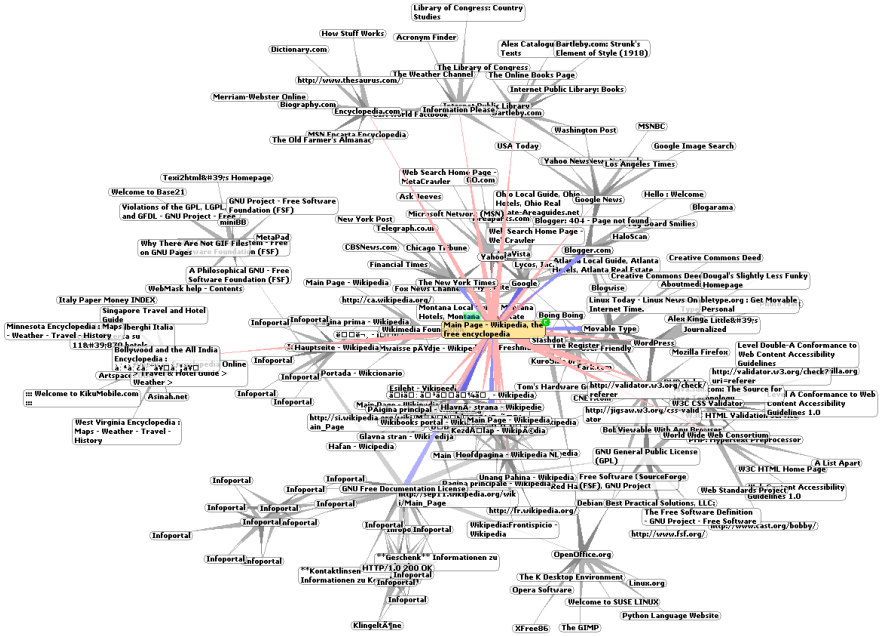
The IP Address

- Each device connected to the Internet has an IP number
  - ✓Part of the IP standard
  - ✓Usually written as four 8-bit decimals
  - ✓Addresses can be permanent (like a fixed Web server) or temporary (like the address given by an ISP)
- IP numbers ⇔ IP names
  - ✓17.112.152.32 ⇔ www.apple.com
  - ✓.com = Top-level domain (TLD)
  - ✓apple.com = Second-level domain (SLD)

The World Wide Web

- **WWW ≠ Internet**

  It is a system of interlinked, hypertext documents accessed via internet.



The World Wide Web

1. A scheme for locating the documents

   URI: Universal/Uniform Resource Identifier

   URL: Universal/Uniform Resource Locator

   - http://www.globis.ethz.ch
   - ftp://ftp.inf.ethz.ch

   URN: Universal/Uniform Resource Name

2. A protocol for accessing documents (HTTP)

   HTTP: Hypertext Transfer Protocol

3. A “hypertext” language linking together documents (HTML)

   HTML: Hypertext Mark-Up Language

The World Wide Web

Getting documents from the remote computer to your computer requires specialized software:

- **WWW Server**
  - ✓Runs on the remote computer
  - ✓Listens for requests for documents
  - ✓Responds with requested document (if valid request)
- **WWW Browser**
  - ✓Displays documents sent by Server
  - ✓Lets user request new documents, either directly as a URI, or by selecting a link in the current document

Web Terms I

- **User**

  Human user of the World Wide Web
- **Client**

  Software sends HTTP-requests to a Web server
- **Browser**

  Client that can visualize HTML + ...
- **Server**

  Software that waits for HTTP-requests and answers with HTTP-replies
- **Site**

  A collection of web pages (usually belonging to one organization)

Web Terms II

➢Page

A single HTML page.

- It may contain other document types (e.g. images, sounds, ...)

➢Homepage

Entry page to interconnected pages that have a common content.

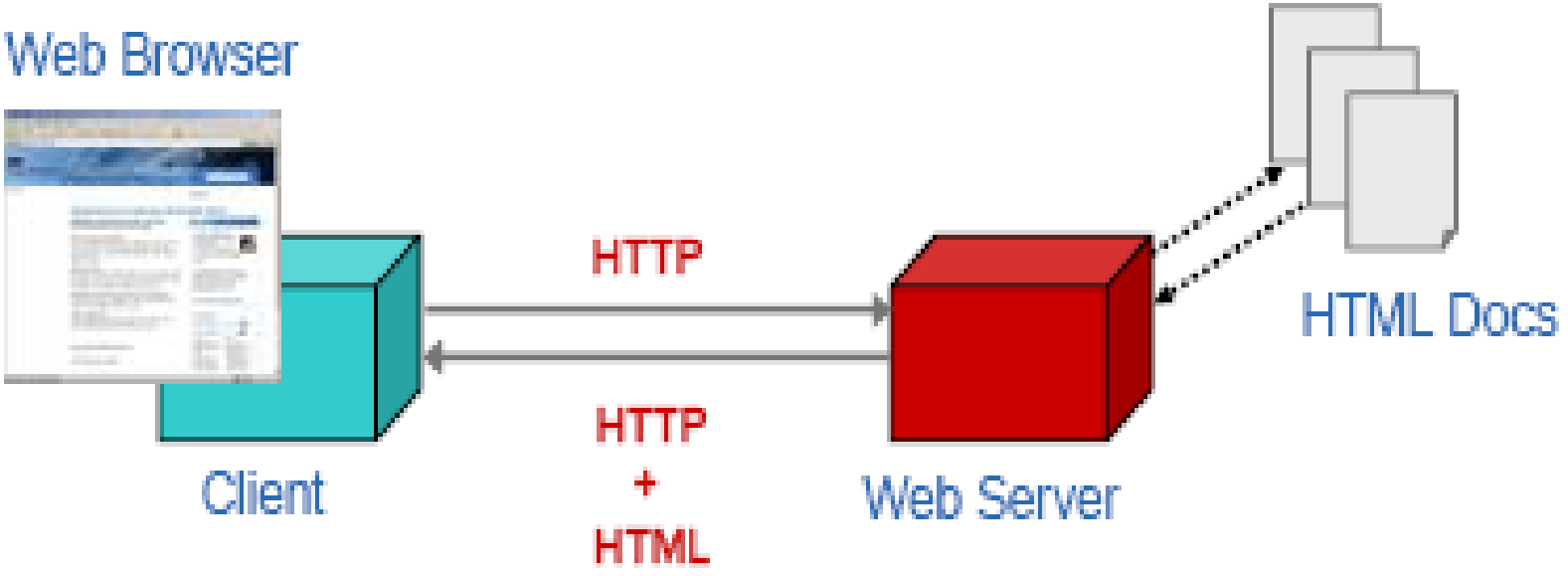
- Homepage of a company or organization
- Homepage on a specific topic
- Personal homepage

➢Portal

Specific entry page to the web.

- Portal of a provider (e.g. Dodo, AOL, ...)
- Portal of content/new provider (e.g. ninenews.com, cnn.com, ...)
- Portal of search engines/catalogs (e.g. yahoo, lycos, ...)

Graphic Structure



HTTP

- Works on top of TCP/IP
- HTTP operates by the exchange of lines of printable ASCII text
- Requesting a document
  - ✓Type of request: GET, POST, HEAD, PUT, DELETE, TRACE, OPTIONS
  - ✓Headers – look like MIME (email) headers
  - ✓Data
- **Responding to a request**
  - ✓Status: 200 = OK, 403 = forbidden, 404 = not found
  - ✓Headers
  - ✓Data

HTTP

- HTTP/1.0
  - ✓One object transferred by connection
  - ✓Even if several documents are to be downloaded from the same server
  - ✓Even if the same document was recently downloaded

Inefficient

- HTTP/1.1
  - ✓Persistent connections
  - ✓Proxies
  - ✓Caching

Much better!!!

HTTP/1.1

Persistent connections

- ✓After a document is downloaded from a Web server, the TCP connection remains open
- ✓Documents from the same server (e.g., inline images for a Web page) can be downloaded using the same connection
- ✓Much more efficient, as breaking and then re-establishing a TCP connection requires a significant amount of work
- ✓The connection is broken only when the client (explicit disconnect) or server (timeout) decides so
- ✓Can leave the server vulnerable to denial-of-service attack if the timeout is too long ☹

HTTP/1.1

Proxies

- ✓A proxy is a computer that handles Internet requests on behalf of computers within an intranet
- ✓Proxies can restrict access to the Internet, blocking spam, pornography, file sharing protocols, …
- ✓Proxies can be used with HTTP/1.0, but it requires special configuration of the intranet
- ✓HTTP/1.1 has built-in support for proxies

HTTP/1.1

Caching

- ✓A cache is a place to store things “closer” to where they will be needed next
- ✓In computers, caching trades faster access against extra cost for storage
- ✓Usually stores the most-recently used
- ✓A CPU cache stores words from main memory in special fast memory, close to the CPU
- ✓A disk cache stores pages from disk in main memory
- ✓A network cache stores documents on local disk

HTTP/1.1

Caching…

- ✓User Cache stores recently accessed documents on the user's computer
  - Probably you have a University website stored in your cache
- ✓Proxy Cache stores documents for an entire intranet
- ✓Server Cache stores recently used documents within a distributed service
  - The Google search engine is actually thousands of computers that share the task of answer queries – each of these computers probably has a locally cached copy of the Google Home page and other common pages

Hypertext Markup Language

- HTML is the originally proposed language for creating Web pages
- A HTML file is (in general) plain ASCII text, containing some information and some markup tags. A markup tag is enclosed between the < and > "special" characters.
- There have been several versions of HTML
  - ✓HTML 2.0/ 3.2/ 4.01
  - ✓HTML 1.0/ 1.1
    - Most of the features of HTML 4.01, but a much stricter (cleaner) syntax
    - Compatible with XML

HTML

HTML defines rules for placing “tags” within the text to describe the structure of a hypertext document

- ✓These tags are not displayed by the browser
- ✓Some tags describe the semantics of the document
- ✓Some tags advise a browser on visual presentation
- ✓Some tags tell the browser to include (inline) other documents – image/video/sound
- ✓Some tags establish a link (externally) to other documents

HTML

- Simple and Easy to Publish on Web
- Structure, Content or Presentation
  - Wide use of table elements for formatting layout
  - Address elements
- Problems of
  - link maintenance
  - document interpretation
  - Does not contain meaning
- Flexible
  - unknown tags ignored by browsers… easy to extend with customized tags

HTML

A simple example

```
<html>
  <head>
    <title>A simple HTML file</title>
  </head>
  <body>
    <h1>An Announcement</h1>
    <p>Listen up, I have something to say!</p>
  </body>
</html>
```

HTML Syntax

- Some tags have content
  - ✓E.g. `<h1>Learning HTML</h1>`
  - ✓The tag `<h1>` means “the most important heading in the document”
  - ✓So, the content `Learning HTML` is the most important heading in the document
- Some tags do not
  - ✓E.g. <hr>
  - ✓The tag <hr> means “insert a horizontal rule into the document at this point”

HTML Syntax

Some tags have required attributes

- E.g. `<img src="smiley.gif" alt=":-)"/>`
- The tag `<img/>` means “insert an image into the document at this point”
- The image URI is specified by the `src` attribute
- The `alt` attribute contains alternative text to be displayed by browsers that cannot display the image

HTML Syntax

Some tags have optional attributes

- E.g. `<img src="diagram.jpg" alt="Figure 1" id="fig1" />`
- The `id` attribute allows a browser to jump directly to a tag within a document by giving its id
  - ✓Among other things

HTML Syntax

In a well-formed HTML document…

- Tag and attribute names are in all lowercase letters
- Every opening <*tag*> must have a matching closing </*tag*>, or be in the <*tag* /> form
- Leave a space before the / in <*tag* /> – it provides compatibility with older browsers
- Attribute values must be enclosed in quotes "…"

HTML Syntax

It is a good idea to include a DOCTYPE at the start of any HTML document

```
<!DOCTYPE html>
```

This tells the browsers what type of HTML document it is dealing with.

In this case HTML 5

HTML Syntax

You should comment HTML like any program

```
<!-- comment text, possibly spanning many lines -->
```

Good commenting helps understanding, maintenance and improvement of the system

HTML Structure

- The root tag is the <html> tag
- Inside the <html> tag there is a <head> and a <body> section.
- <head> contains information about the document – this is not usually displayed by the browser
- <body> contains the displayed content of the document

HTML Head

The <head> tag contains information about the document

- <title> – the title of the document
  - ✓Sometimes displays at the top of the browser (not in the content area), or sometimes not at all
  - ✓Displayed by search engines when linking to the document
- There are many other tags that can go in the `<head>` section – they will be explained as they are needed

HTML Body

HTML body tags can be loosely divided into two groups

1. Tags that start and end a new “block” of text
   - ✓Includes headings, paragraphs, lists, tables, …
2. Tags that do not
   - ✓Includes definitions, abbreviations, quotes, emphases, hyperlinks, inline images, …

- HTML has rules on which tags can be nested inside which other tags
  - ✓An HTML document which is well-formed and obeys these rules is said to be **valid**

HTML Block Tags

- `<h1>…</h1>` – the most important heading in the document (there will usually be only one of these; not necessarily the same as `<title>`)
- `<h2>` – headings slightly less important than `<h1>`
- `<h3>`, `<h4>`, `<h5>`, `<h6>`
  - ✓Think of a Web page as a chapter in a book
  - ✓The chapter heading is the most important, `<h1>`
  - ✓Sections are the next most important, `<h2>`
  - ✓Then subsections, `<h3>`, sub-subsections, `<h4>`, and so on

HTML Block Tags

- `<p>…</p>` – a paragraph of text
- `<blockquote>…</blockquote>` – a block of text quoted from someone/somewhere else
- `<address>…</address>` – a paragraph of text that is an (postal) address
- `<pre>…</pre>` – a block of preformatted text – whitespace is passed through unchanged!
- `<hr />` – draw a horizontal rule between blocks

HTML Block Tags

- `<div>…</div>` – a generic division of the document – used to group and apply formatting
- Nesting
  - ✓`<blockquote>` must contain other block tags
  - ✓`<div>` can contain other block tags
  - ✓the others cannot contain other block tags

HTML List Tags

- `<ol>…</ol>` – an ordered list of items
- `<ul>…</ul>` – an unordered list of items
  - ✓ `<li>…</li>` – an item within a list
- `<dl>…</dl>` – a definition list
  - ✓ `<dt>…</dt>` – a term to be defined
  - ✓ `<dd>…</dd>` – the definition of a term
- List tags should only contain the appropriate item tags as immediate children
  - ✓ `<dt>` can only contain inline tags
  - ✓ `<li>` and `<dd>` are not so limited – nested lists!

HTML Inline Tags

- `<em>…</em>` – the enclosed text should be emphasized
- `<strong>…</strong>` – the enclosed text should be strongly emphasized
  - ✓Using multiples of `<em>` and `<strong>` does not make it very strongly emphasized!
- `<dfn>…</dfn>` – the enclosed text is a definition
- `<code>…</code>` – the enclosed text is a piece of program code
- `<samp>…</samp>` – the enclosed text is a sample of (program) output
- `<kbd>…</kbd>` – the enclosed text represents input that would be typed on a keyboard

HTML Inline Tags

- `<cite>…</cite>` – the enclosed text is a citation
- `<q>…</q>` – the enclosed text is a short quotation from another source
- `<var>…</var>` – the enclosed text is a variable
- `<sup>…</sup>` – the enclosed text is a mathematical superscript
- `<sub>…</sub>` – the enclosed text is a mathematical subscript

HTML Inline Tags

- `<abbr>…</abbr>` – the enclosed text as an abbreviation
  - ✓`<abbr title="Limited">Ltd.</abbr>`
- `<acronym>…</acronym>` – the enclosed text as an acronym
  - ✓`<acronym title="World Wide Web">WWW</acronym>`
- `<br />` – the end of a “logical” line of text (such as in a postal address)
- You should not use inline tags as immediate children of `<body>`
- Inline tags can only contain other inline tags

HTML & Visual Formatting

All tags except `<div>`, `<abbr>` and `<acronym>` imply suggestions to the browser about how to draw their content

- Headings change the font size and weight
- `<blockquote>` indents the margins

THESE ARE ONLY SUGGESTIONS!

- A browser can ignore them
- It may not even support them – e.g., a mobile phone

HTML & Visual Formatting

- Before using a tag, think about whether it is really what you mean in the given context
- **Never use a tag just because it gives you the desired visual formatting!!**
- It won't always work
- There are much better ways to do it

HTML & Visual Formatting

There are old tags which are only for adding visual formatting to documents (without adding implicit semantic meaning)

- `<i>…</i>` – draw in an italic font-style, if possible
- `<b>…</b>` – … bold font-weight
- `<tt>…</tt>` – … typewriter-text font
- `<big>…</big>` – … one font size bigger
- `<small>…</small>` – … one font size smaller

HTML & Visual Formatting

- There is a single tag that gives full control over the (inline) visual formatting of text in HTML
  - ✓It is called `<font>`
  - ✓**DO NOT USE THIS**
  - ✓There is also `<blink>`, which was so annoying that some browsers had an option to disable it

If you use these in your assignments, then you will lose marks

HTML Hyperlinks

```
<a href="uri">content</a>
```

- A hypertext “anchor”
- An inline tag
- Creates a hyperlink from the *content* to the document identified by *uri*
- When the link is activated in the browser, the linked document is requested and (if all going well) displayed in the browser

HTML Hyperlinks

- URI = Uniform Resource Identifier
- URL = Uniform Resource Location, a type of URI
  - ✓A “physical” location
  - ✓*protocol*`:` `/` `/`*host*`:`*port*`/`*path*`/`*name*`?`*params*`#`*target*
- Protocol = `http`, `ftp`, `file`, `mailto`, …
- Host = name or IP address of the server
- Port = number identifying where the Web server is listening for requests

HTML Hyperlinks

- Path = where the document is on the server
- Name = the name of the document
- Params = parameters to be passed to the server
  - ✓Used to dynamically generate documents
- Target = location within the document
  - ✓The browser should move to display the tag with the given `id` attribute

HTML Hyperlinks

You can add an identity to any tag within the body of a HTML document, so that a hyperlink can jump directly to it

```
<h2 id="company">About the Company</h2>
<p>
  By hyperlinking to about.html#company
  you can jump directly to the above
 heading
</p>
```

The default “target” is the top of the page

HTML Images

`<img src="uri" alt="text" />`

- Loads the image file from *uri* and insert it into the document at this point
- If the image cannot be loaded or displayed, then display the alternative *text* instead
- `<img />` is an inline tag with no content
- Most modern browsers support GIF, PNG and JPEG images – do not use platform-specific image formats, like BMP, PCX and XBM

HTML Tables

```
<table>
  <tr>
    <td>Top-left</td>
    <td>Top-right</td>
  </tr>
  <tr>
    <td>Bottom-left</td>
    <td>Bottom-right</td>
  </tr>
</table>
```

Example

HTML Tables

- `<table>` – a block-level tag for arranging content into a table (rows and columns)
- A table can contain an optional `<caption>`
- Each row of a table is grouped inside `<tr>`
- Each "data cell" of a row is contained inside `<td>`
  - ✓Unless it is a "heading" for a row or column, which is inside `<th>`

HTML Tables

The rows of a table can be grouped

- ✓`<thead>` – header rows; copied at the top of a table if it spans multiple pages
- ✓`<tfoot>` – footer rows; copied at the bottom of a table if it spans multiple pages
- ✓`<tbody>` – body rows; the default

HTML Tables

- Given cells can "span" multiple rows or columns
  - ✓ `colspan` = number of columns spanned
  - ✓ `rowspan` = number of rows spanned
- Cells can contain any body tags
  - ✓ Even nested tables

HTML Tables

- Individual cells can be made to span several columns
  - ✓`<td colspan="3">a cell across 3 columns</td>`
- … or several rows
  - ✓`<td rowspan="2">a cell 2 rows tall</td>`
  - ✓The corresponding column in the next row should be omitted
- … or both
  - ✓`<td colspan="5" rowspan="3">a 5x3 cell</td>`

HTML Tables

- `<colgroup>` is placed at the start of `<table>` (after `<caption>`) to indicate that a number of columns belong to a "semantic group"

```
<table>
  <colgroup span="1" />
  <colgroup span="3" />
  <tr>
     <th>Dow</th>
     <td>2,824.15</td>
      <td>2,691.35</td>
     <td>2,695.10</td>
  </tr>
  <tr>
     <th>NAQDAQ</th>
      <td>1,925.14</td>
       <td>1,786.94</td>
       <td>1,810.45</td>
  </tr>
  …
```

The table contains 4 columns, grouped as 1 then 3

HTML Tables

- Tables are often used to control the layout of HTML document
  - ✓Dividing the page up into a grid of cells
  - ✓Each cell (`<td>`) can contain arbitrarily complex tags (even nested tables)
- This is not a good idea
  - ✓Can fail if the browser window is resized
  - ✓Difficult to allow for multiple screen sizes
  - ✓Destroys the semantic meaning of the “table”

HTML Forms

- Allow web pages with simple interaction:
  1. User enters data into form.
  2. Browser “submits” form to server.
  3. Server processes data and responds.
     - ✓Respond = update internal records, dynamically generate web page, …

HTML Forms

`<form action="`*uri*`">` – A way of adding *simple* interaction to Web pages

- ✓With <form>, special tags create “inputs” where the user can enter information
- ✓When the form is “submitted”, this information is sent to the Web server
- ✓The server responds with the new document, this could just be another static document, but most likely it is dynamically generated using the information entered by the user

A lot of this course is about how to implement this dynamic behavior on the server

HTML Forms

```
<input type="text"
       name="name"
       value="text"
       size="num"
       maxlength="num"
      readonly="readonly" />
```

- ✓Allows the user to enter a single line of text
- ✓`value` = the default input value
- ✓`size` = the visible length of the input (in characters)
- ✓`maxlength` = the maximum number of characters that can be entered
- ✓`readonly` = the user cannot change the input value

HTML Forms

<input type="*text*" placeholder="*text*"/>

- Placeholder gives the textbox default text that disappears on focus.
- Useful to guide the user on the expected input
- i.e.
  - placeholder="John Doe…"
  - placeholder="123 SomeStreet st"
  - etc.

HTML Forms

```
<input type ="password"
       name="name"
       value="text"
       size="num"
       maxlength="num"
      readonly="readonly" />
```

- Behaves the same a `type="text"`, except that the value is not displayed by the browser (usually hidden under asterisks)

```
<input type="hidden" name="name"
            value="text" />
```

- This input is not displayed by the browser

HTML Forms

```
<input type="checkbox" name="name"
  value="text" checked="checked" />
```

- A checkbox; is either “on” or “off”
- `checked="checked"` if it is initially “on”

```
<input type="radio" name="name"
value="text" checked="checked" />
```

- Like a checkbox, but if multiple radios have the same *name*, then only one can be “on” at any time

HTML Forms

```
<input type="file" name="name" accept="type-list"
/>
```

- The user should enter (or browse for) a local file
- When the form is submitted, this file is uploaded to the server
- *type-list* is a comma-separate list of MIME types that the server will accept for upload – e.g., `text/plain, text/html, application/msword`

HTML Forms

`<input type="submit" value="`***label***`" />`

- A button which, when clicked, submits the form

`<input type="reset" value="`***label***`" />`

- A button which, when clicked, resets the form's inputs to their default values

`<input type="button" value="`***label***`" />`

- A button which does nothing! (you'll find out why we need this later)

HTML Forms

```
<button type="type">content</button>
```

- Creates a button from almost any HTML markup
- *type* = `submit|reset|button`

```
<textarea name="name" cols="num"
 rows="num" readonly="readonly">
value</textarea>
```

- A large (`cols`×`rows`) text input
- The initial *value* is the content of the tag

HTML Forms

```
<select name="name" size="num"
    multiple="multiple">
```

- Contains a selection list
- `size` = the number of items shown in the menu (size = 1 creates a drop-down list, while size > 1 creates a scroll list)
- `multiple="multiple"` lets the user select multiple items – otherwise, only one item can be selected

```
<option value="value">label</option>
```

- One item within a `<select>`
- *label* is displayed in the list

```
<optgroup label="label">…</optgroup>
```

- Groups `<option>` tags into a submenu

HTML Forms

- When a form is “submitted”, the user’s inputs are sent to the server
- How this happens depends on the attributes of the `<form>` tag
  - ✓`action="`*uri*`"` – the server “document” that will process the form data
  - ✓`method="get|post"` – how the information will be sent (a HTTP `GET` or `POST` request)
  - ✓`enctype="`*type*`"` – how is the information encoded

HTML Forms

- `method="get"`
  - ✓The form data is appended to the HTTP request's URI – `http://`*server*`/`*page*`?`*name*`=`*value*`&`*name*`=`*value*`&...`
  - ✓Do not use this when submitting a password, as it will appear in **plain-text** in the location bar of your browser!
- `method="post"`
  - ✓*name*`=`*value*`&`*name*`=`*value*`&...` sent as the body of a HTTP POST request

HTML Forms

- `enctype="application/x-www-form-urlencoded"`
  - ✓Uses the *name*=*value*`&`*name*=*value*`&`... encoding
- `enctype="multipart/form-data"`
  - ✓Places each *name-value* pair in its own section of the HTTP request body
  - ✓Only works with `method="post"`
  - ✓You **must** use this encoding if uploading a file

HTML Editors & Validators

- A word on using WYSIWYG HTML editors (such as Frontpage, Pagemill, "SaveAs HTML" from Word, etc.)

DO NOT

- ✓They rarely produce well-formed HTML
- ✓They often include non-standard tags
- ✓You will not learn how HTML works, which will hurt you when you try to add JSP code to them

HTML Editors & Validators

- The bad news:
  - ✓**If you use a WYSIWYG HTML editor for your assignments, then you will get zero marks for the HTML!**
- The good news:
  - ✓You can and **SHOULD** use HTML validation services, such as…
  - ✓http://validator.w3.org/
  - ✓http://www.htmlhelp.com/tools/validator/

HTML Resources

- The Web is full of HTML tutorials, “how-to”s and FAQs
  - ✓A lot of these are based around old HTML standards
  - ✓Many will show you a quick way to achieve a particular end … but use techniques that will not adapt to different situations
- You can find some good HTML tutorials at:
  - ✓http://www.w3schools.com/HTML/default.asp
  - ✓http://www.html-5-tutorial.com/
  - ✓http://www.tutorialspoint.com/html5/

THE END

QUESTIONS??

THANKS!!