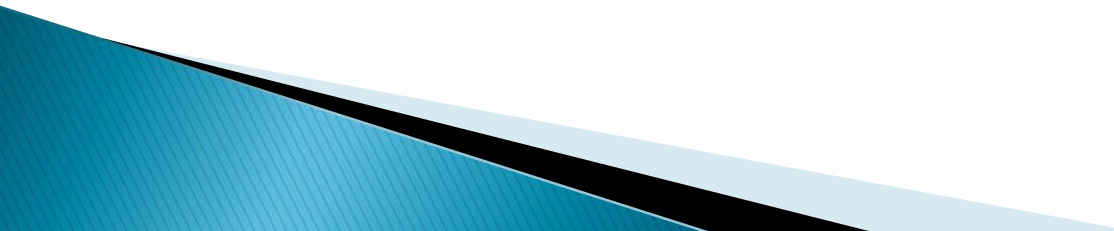


HTTP

- ▶ Hypertext Transport Protocol
 - ▶ Language of the Web
 - protocol used for communication between web browsers and web servers
 - ▶ TCP port 80 (443 secure)
 - ▶ RFC 2616 (ver 1.1)
- 

URI,URN,URL

- ▶ Uniform Resource Identifier
 - Information about a resource
- ▶ Uniform Resource Name
 - The name of the resource with in a namespace
- ▶ Uniform Resource Locator
 - How to find the resource, a URI that says how to find the resource

HTTP – URLs

▶ URL

- Uniform Resource Locator

- protocol (http, ftp, news)
- host name (name.domain name)
- port (usually 80 but many on 8080)
- directory path to the resource
- resource name
- `http://xxx.myplace.com/www/index.html`
- `http://xxx.myplace.com:80/cgi-bin/t.exe`

HTTP – methods

▶ Methods

◦ GET

- retrieve a URL from the server
 - simple page request
 - run a CGI program
 - run a CGI with arguments attached to the URL

◦ POST

- preferred method for forms processing
- run a CGI program
- parameterized data in sysin
- more secure and private

HTTP – methods

▶ Methods (cont.)

◦ PUT

- Used to transfer a file from the client to the server

◦ HEAD

- requests URLs status header only
- used for conditional URL handling for performance enhancement schemes
 - retrieve URL only if not in local cache or date is more recent than cached copy

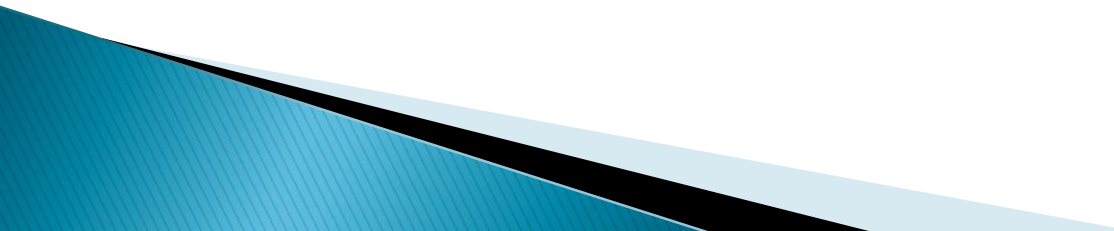
HTTP Request Packets

- ▶ Sent from client to server
- ▶ Consists of HTTP header
 - header is hidden in browser environment
 - contains:
 - content type / mime type
 - content length
 - user agent – browser issuing request
 - content types user agent can handle
- ▶ and a URL

HTTP Request Headers

- ▶ Precede HTTP Method requests
- ▶ headers are terminated by a blank line
- ▶ Header Fields:
 - From
 - Accept
 - Accept-Encoding
 - Accept Language

HTTP Request Headers (cont.)

- Referer
 - Authorization
 - Charge-To
 - If-Modified-Since
 - Pragma
- 

From:

- ▶ In internet mail format, the requesting user
- ▶ Does not have to correspond to requesting host name (might be a proxy)
- ▶ should be a valid e-mail address

Accept:

- List of schemes which will be accepted by client
- `<field> = Accept: <entry> * [, <entry>]`
- `<entry> = <content type> * [; <param>]`
- `<param> = <attr> = <float>`
- `<attr> = q / mxs / mxb`
- `<float> = <ANSI-C floating point >`
- `Accept: text/html`
- `Accept: audio/basic q=1`
- if no Accept is found; plain/text is assumed
- may contain wildcards (*)

Accept-Encoding

- ▶ Like Accept but list is a list of acceptable encoding schemes
- ▶ Ex
 - Accept-Encoding: x-compress;x-zip

User-Agent

- ▶ Software product used by original client
- ▶ `<field> = User-Agent: <product>`
- ▶ `<product> = <word> [/<version>]`
- ▶ `<version> = <word>`
- ▶ Ex.
 - User-Agent: IBM WebExplorer DLL /v960311

Referer

- ▶ For Server's benefit, client lists URL of document (or document type) from which the URL in request was obtained.
- ▶ Allows server to generate back-links, logging, tracing of bad links...
- ▶ Ex.
 - Referer: `http://www.w3.com/xxx.html`

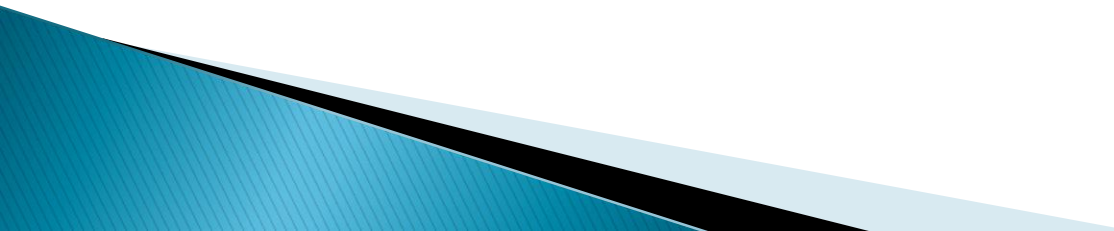
Authorization:

- ▶ For Password and authentication schemes
- ▶ Ex.
 - Authorization: user fred:mypassword
 - Authorization: kerberos kerberosparameters

ChargeTo:

- ▶ Accounting information
- ▶ Accounting system dependent

Pragma:

- ▶ Same format as accept
 - ▶ for servers
 - ▶ should be passed through proxies, but used by proxy
 - ▶ only pragma currently defined is no-cache; proxy should get document from owning server rather than cache
- 

Modified-Since:

- ▶ Used with GET to make a conditional GET
- ▶ if requested document has not been modified since specified date a Modified 304 header is sent back to client instead of document
 - client can then display cached version

Response Packets

- ▶ Sent by server to client browser in response to a Request Packet

Status Header

- ▶ “HTTP/1.0 sp code”
- ▶ Codes:
 - 1xx – reserved for future use
 - 2xx – successful, understood and accepted
 - 3xx – further action needed to complete
 - 4xx – bad syntax in client request
 - 5xx – server can’t fulfill good request

HTTP Response Headers

- ▶ Sent by server to client browser
- ▶ Status Header
 - Entities
 - Content-Encoding:
 - Content-Length:
 - Content-Type:
 - Expires:
 - Last-Modified:
 - extension-header
- ▶ Body – content (usually html)

Status Codes

- ▶ 200 OK
- ▶ 201 created
- ▶ 202 accepted
- ▶ 204 no content
- ▶ 301 moved perm.
- ▶ 302 moved temp
- ▶ 304 not modified
- ▶ 400 bad request
- ▶ 401 unauthorized
- ▶ 403 forbidden
- ▶ 404 not found
- ▶ 500 int. server error
- ▶ 501 not impl.
- ▶ 502 bad gateway
- ▶ 503 svc not avail

Statelessness

- ▶ Because of the Connect, Request, Response, Disconnect nature of HTTP it is said to be a stateless protocol
 - i.e. from one web page to the next there is nothing in the protocol that allows a web program to maintain program “state” (like a desktop program).
 - “state” can be maintained by “witchery” or “trickery” if it is needed

Maintaining program “state”

- ▶ Hidden variables (`<input type=hidden>`)
- ▶ Sessions
 - Special header tags interpreted by the server
 - Used by ASP, PHP, JSP
 - Implemented at the language api level

Choosing Text Editor

- HTML Editors are excellent tools for experienced web developers; however; it is important that you learn and understand the HTML language so that you can edit code and fix “bugs” in your pages.
- For this Course, we will focus on using the standard Microsoft Windows text editors, NotePad. We may use also textpad.

Creating a Basic Starting Document

- The HEAD of your document point to above window part. The TITLE of your document appears in the very top line of the user's browser. If the user chooses to "Bookmark" your page or save as a "Favorite"; it is the TITLE that is added to the list.
- The text in your TITLE should be as descriptive as possible because this is what many search engines, on the internet, use for indexing your site.

Setting Document Properties

- Document properties are controlled by attributes of the **BODY** element. For example, there are color settings for the background color of the page, the document's text and different states of links.

Color Codes

Colors are set using “**RGB**” color codes, which are, represented as hexadecimal values. Each 2–digit section of the code represents the amount, in sequence, of **red**, **green** or **blue** that forms the color. For example, a **RGB** value with 00 as the first two digits has no red in the color.

The Body Element

The **BODY** element of a web page is an important element in regards to the **page's appearance**. Here are the attributes of the **BODY** tag to control all the levels:

TEXT="#RRGGBB" to change the color of all the text on the page (full page text color.)

- This element contains information about the page's background color, the background image, as well as the text and link colors.

Background Color

- It is very common to see web pages with their background color set to white or some other colors.
- To set your document's background color, you need to edit the `<BODY>` element by adding the `BGCOLOR` attribute. The following example will display a document with a **white** background color:

```
<BODY BGCOLOR="#FFFFFF"></BODY>
```

TEXT Color

- The TEXT attribute is used to control the color of all the normal text in the document. The default color for text is black. The TEXT attribute would be added as follows:

```
<BODY BGCOLOR="#FFFFFF"  
TEXT="#FF0000"></BODY>
```

In this example the document's page color is white and the text would be red.

LINK, VLINK, and ALINK

These attributes control the colors of the different link states:

1. LINK - initial appearance - default = Blue.
2. VLINK - visited link - default = Purple.
3. ALINK - active link being clicked - default = Yellow.

The Format for setting these attributes is:

```
<BODY BGCOLOR="#FFFFFF" TEXT="#FF0000"  
  LINK="#0000FF"  
  VLINK="#FF00FF"  
  ALINK="FFFF00"> </BODY>
```

Using Image Background

- The BODY element also gives you ability of setting an image as the document's background.
- An example of a background image's HTML code is as follows:

```
<BODY BACKGROUND="hi.gif"  
BGCOLOR="#FFFFFF"></BODY>
```


Previewing Your Work

- Once you have created your basic starting document and set your document properties it is a good idea to save your file.
- To save a file, in NotePad, follow these steps:
 1. Locate and click on the menu called “File”.
 2. Select the option under File Menu labeled “Save As”.
 3. In the “File Name” text box, type in the entire name of your file (including the extension name .html).

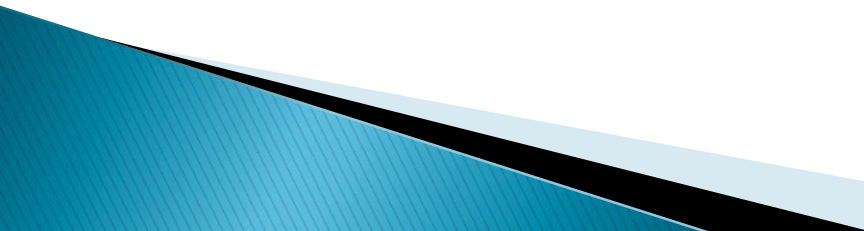
DHTML

- ▶ **DHTML is a combination of technologies used to create dynamic and interactive Web sites.**
 - **HTML** – For creating text and image links and other page elements.
 - **CSS** – Style Sheets for further formatting of text and html plus other added features such as positioning and layering content.
 - **JavaScript** – The programming language that allows you to access and dynamically control the individual properties of both HTML and Style Sheets

Why DHTML

- ▶ With DHTML you can create:
 - Animation
 - Pop-up menus
 - Inclusion of Web page content from external data sources
 - Elements that can be dragged and dropped within the Web page

HTML (Hyper Text Markup Language)

- ▶ An HTML file is a text file containing small **markup tags**
 - ▶ The markup tags tell the Web browser **how to display** the page
 - ▶ An HTML file must have an **htm** or **html** file extension
 - ▶ An HTML file can be created using a **simple text editor**
- 

SEG3210

DHTML Tutorial

DHTML

- ▶ **DHTML is a combination of technologies used to create dynamic and interactive Web sites.**
 - **HTML** – For creating text and image links and other page elements.
 - **CSS** – Style Sheets for further formatting of text and html plus other added features such as positioning and layering content.
 - **JavaScript** – The programming language that allows you to access and dynamically control the individual properties of both HTML and Style Sheets

Why DHTML

- ▶ With DHTML you can create:
 - Animation
 - Pop-up menus
 - Inclusion of Web page content from external data sources
 - Elements that can be dragged and dropped within the Web page