CS120

Introduction to Web Site Development

Lecture 1 - Introduction to HTML

Gabor Kondas
California State University, Los Angeles

(Based on slides by John Hurley)

Introduction

Gabor Kondas

Email: (Start subject with **CS120**)
g_kondas@hotmail.com

Office Hours (ET A225):

Monday:	11:30am-1:30pm
Wednesday:	11:30am-1:30pm

Course Page:

http://csns.calstatela.edu/wiki/content/gkondas/courses/cs120/fall2014

Grading

Undergraduate

A, B, C, NC

Graduate

A, B, NC

A	94 - 100
A-	90 - 93
B+	85 - 89
B	80 - 84
B-	77 - 79
C+	74 - 76
C	70 - 73

Grading

Labs: 60%

Final: 40%

Class Meetings

Saturday

ET B9

Lecture

9:10AM - 10:50AM

Lab

10:50AM - 1:20PM

Lab Assignments

- One assignment per week
  - Due before the next week's lecture. Start it in the lab and finish it at home.
  - Linked from course page, hand in via CSNS
  - Starting with lab 2, you will turn in an html file with a link to your work on our server, as well as a copy of your actual work.

- Labs
  - Even though I do not expect you to finish the assignment in the lab, lab attendance is mandatory. Let me know in advance if you won't be able to attend a lab for some good reason.
  - I may take attendance intermittently and penalize students who do not show up for labs.

Cheating

- You **may** discuss general techniques of web development with other students
- You **may** give or receive help understanding assignments and debugging work
- You **may** copy examples from the lecture notes and then change them to meet the assignment requirements
- You **may not** copy code or allow anyone to copy your code
- You **may not** copy text from any source for short answer questions on labs and exams

Cheating

OK for lab work:

- Copying examples from the lecture notes and modifying them to fulfil the assignment
- “The problem with your table is that you forgot to put in a close table tag”
- “That doesn’t work because you say classname where you need className”
- “That’s not right because the tags are in the wrong order”
- “I did that with an unordered list”
- “You create a multiplication table by using nested loops and multiplying the loop counters”

Cheating:

- Copying code from other students or internet sources
- Copying text from other students for short-answer or essay questions
- Copying text from the internet or a book for short-answer and essay questions

Course Topics

- Web Servers, HTTP
- HTML
- CSS
- JavaScript
- HTML DOM

Web Servers

A web server is a computer program that:

- Is bound to (at least) one network port
  - a port is used to uniquely identify the process or application to which the communication is directed (on a single physical network connection)
  - HTTP uses port 80
  - HTTPS uses port 443
- Listens to network port
- Accepts/Declines connection requests on that port
- Processes IP requests from clients
- Sends IP responses back to the clients

Web Servers

- There are a variety of web servers in common use
- Servers may run in any operating system.
  - The majority of servers run on LINUX and other UNIX-like OS
  - Servers running under Windows are also common

- The most popular is Apache HTTP Server
- MS IIS, mostly used for MS ASP.NET

- In CS 320 you will use Apache Tomcat, which is specialized for Java web programming

Web Servers

- We will provide you with an account on the cs1. calstatela.edu server for use in this class
- Future classes, like CS 320, will also provide you with server accounts
- For business or other extracurricular web sites, you will need to use a commercial web host.
  - These can be found easily with a WWW search
  - Expect to pay around $100 per year for basic services
    - there will be a traffic limit
    - extra services, like e-commerce, will cost more

Network Protocols

- A network protocol is a specification that defines what all of the valid requests and responses are for communication and data transfer between two computers

- Examples
  - (HTTP) GET, POST
  - FTP
  - SFTP

HTTP Request and Response

Request (from client to server)

```
GET /index.html HTTP/1.0
```

Response Info (from server to client)

```
HTTP/1.1 200 OK
Date: Mon, 21 Sep 2009 20:36:40 GMT
Content-Length: 9490
Content-Type: text/html; charset=utf-8

HTTP/1.1 404 Not Found
Date: Mon, 21 Sep 2009 20:44:59 GMT
Content-Type: text/html; charset=utf-8
```

HTTP Request and Response

The request and response take unpredictable routes across the internet.

traceroute (Unix-like OS) or **tracert** (Windows) can give you an idea of what the route could be, but you may send requests from the same source address to the same target and get different paths every time.

tracert www.yahoo.com

HTML

- Basic language of WWW documents
- HTML documents consist of text, including tags that describe document structure
- HTML is rendered to a graphic format by the browser
- HTML can be hand written, output from execution of programming language code, or generated by graphic editors or automatic programming tools
- Even if you would use tools to generate it, you have to understand it

Why “Hypertext”?

- The internet was originally developed in universities, the US military, and government research labs.
- Most pre-WWW uses of the internet involved text-based communication
    - email
    - remote logons to mainframe computers to run (command-line) apps
    - library catalogs and other databases
    - academic papers, reference sources
    - USENET
    - Robust (“not easily disrupted”) communication for the military

Why “Hypertext”?

- Text with internal/external links for cross referencing within and between text documents
- Sites like Wikipedia carry on in this vein.
- The first real hypertext apps originated in the 1970s
- Inventors of WWW (1991) expected it to be used mostly for text documents, and wanted to take advantage of the potential to make large amounts of print information available from diverse sources

HTML

- HTML
  - HyperText Markup Language
  - Markup: annotation. In this case, we mean annotation to describe document structure and formatting
  - Publishing language of the World Wide Web
- Standardized by the World Wide Web Consortium (W3C)
    - HTML 5 (2012)
    - Newest HTML specification, being released in sections

Writing HTML

- There are many ways to hand-write HTML.
- The easiest way is to use a **plain text** editor like **Notepad++** or **Sublime Text**, both of which are significantly better than Notepad

- **DO NOT** use a word processor like MS Word or Wordpad
- In this class, **DO NOT** use any tool that generates HTML for you.

Writing HTML

- Save your web pages as plain text with .htm or .html filename extensions. Be careful not to let the editor add .txt or other extension to the end of the file

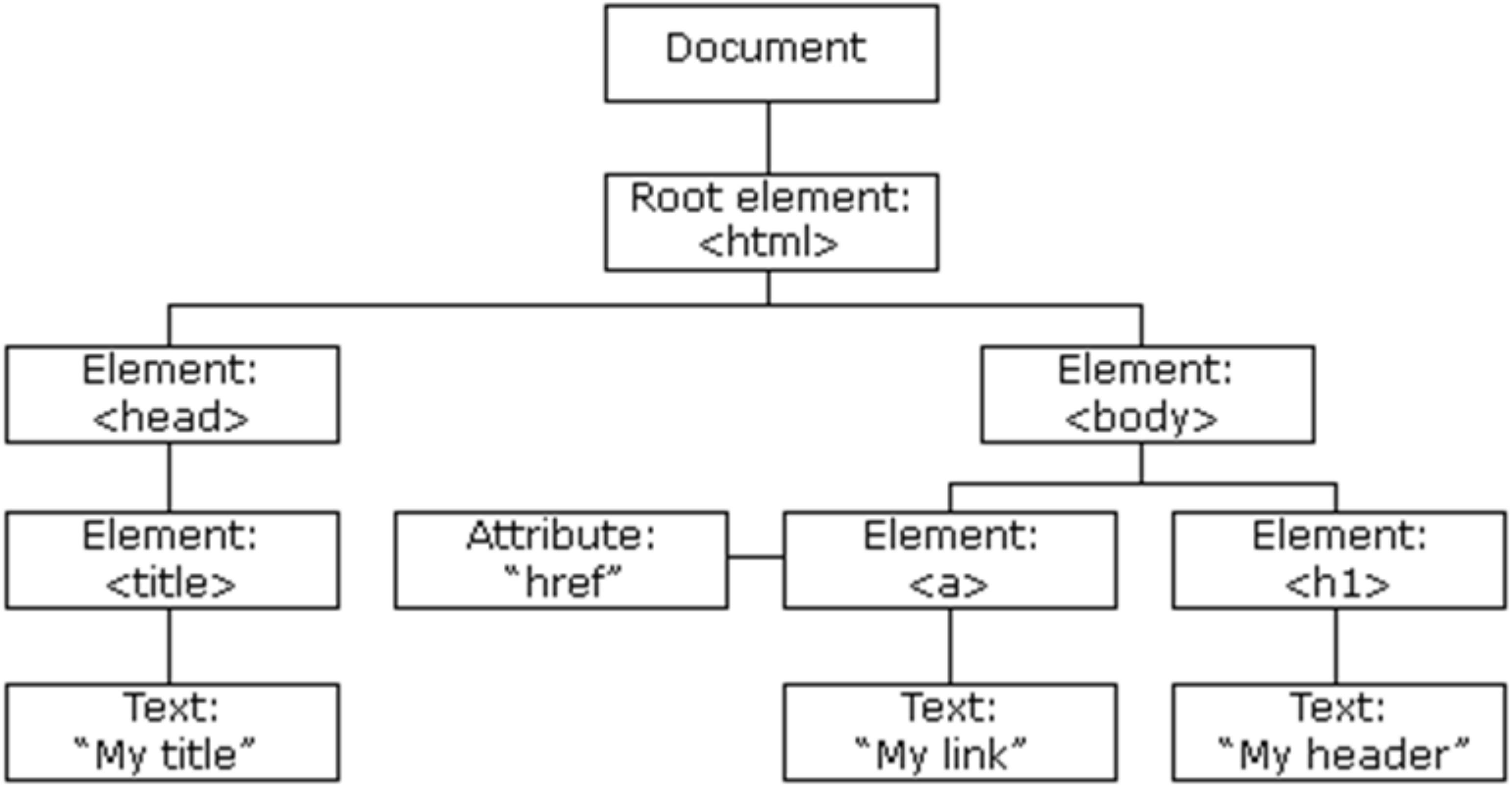
- If possible, save your files in a location that does not contain any spaces in the path. You may not be able to do this on the lab computers

- **DO NOT** put spaces in your filenames
- **DO NOT** capitalize any letter in an html file name

HTML Syntax

- An HTML document consists of a tree of HTML elements
  - Each element begins with a start tag
  - Each element ends with an end tag
  - An element can define attributes within its start tag
- The html element is the “root” of the tree

HTML Syntax


Document
Root element: <html>
Element: <head>
Element: <body>
Element: <title>
Attribute: "href"
Element: <a>
Element: <h1>
Text: "My title"
Text: "My link"
Text: "My header"

HTML Example

Example

```
<!DOCTYPE html>
<html lang = "en">
   <head>
      <title>HTML TEST!</title>
   </head>
<body>
   <p>Hello HTML!</p>
   <p>This is HTML Syntax!</p>
</body>
</html>
```

XML

- eXtensible Markup Language
- XML defines a generic syntax used to mark up data with simple, human-readable tags
- XML is designed to be customized
- Like HTML, XML was developed from the SGML markup language

XHTML

- eXtensible HyperText Markup Language
- Version of HTML that is designed to adhere to XML rules
- Stricter syntax than traditional HTML
- Adherence to XML rules allows greater extensibility
- Parsable as XML, so it can be manipulated in sophisticated ways by clients; you'll learn about this in more advanced CS classes
- XHTML is superseded by HTML5

XHTML Example

```
<<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN" "http://www.
w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
    <title>Hello HTML!</title>
</head>
<body>
<p >Hello HTML!</p>
<p>This is XML Syntax!</p>
</body>
</html>
```

HTML 5

The latest specification of HTML, and is the standard form for this class.

Simple HTML Elements

- Root Element
    - The HTML Element
        - <html>head + body elements</html>
- Metadata Elements
    - The head Element
        - <head>One or more metadata elements</head>
    - The title Element
        - <title>Document title</title>

Simple HTML Elements

The Body Element

```
<body>Flow content (stuff to be rendered)</body>
```

The p Element

Description

- Used to mark paragraphs

Example

- <p>The is paragraph text. Type more here…</p>

The br Element

Description

- Marks a line break within phrasing content

Comments

- Use a self-closing tag: <br />

Example

- <p>John Doe<br />1 Main St.<br />
  Anytown, CA 90000</p>

The hr Element

Description

- Horizontal Rule, ie a line across the parent element

Comments

- Use a self-closing tag: May have a width attribute, eg <hr width = "75%" />

Example

```
<h1>Members</h1>

<hr  width = "80%" />

<p>John Doe<br />1 Main St.<br />
Anytown, CA 90000</p>
```

The h1 - h6 Elements

Description

Marks a new heading (h1 largest, h6 smallest).

Example

```
<h1>Chapter 1</h1>
<h2>Section 1.1</h2>
<h2>Section 1.2</h2>
<h3>Section 1.2.1</h3>
```

Logical vs. Physical Markup

- In the context of HTML, some elements show the structure of the content, while others just give directions for formatting it.
- This is a distinction between semantics (“logical”) and typography (“physical”)
- Physical formatting may be irrelevant if your page is used for some purpose you didn’t anticipate.
    - Translation into a language that uses a different alphabet.
    - Screen readers

b and i

Examples of “physical” formatting

Description

- Marks bold, italic, and centered phrasing content.

Comments

- Start Tag: Required
- End Tag: Required
- Self-Closing Tag: Forbidden

Example

```
<p>A <b>long</b> time ago, in a galaxy
<i>far, far</i> away…</p>
```

Strong and Em

“Logical” formatting; refers to meaning, not typography

Description

- Marks strong and emphatic content.

Example

- <p>A <strong>long</strong> time ago,
  in a galaxy <em>far, far</em> away…</p>

Strong and Em

The strong and em elements by default behave similarly to b and i respectively

- Strong and em describe meaning, not just typography
    - Later we will learn how to use CSS to make them render differently

More relevant for screen readers, machine translators, and applications that might parse your content and use it various ways

Parse: break down into components. A browser, translator, or search engine will do this with your documents as part of the process of rendering, translating, or indexing.

Center

- Center displays its content with center justification

- We will learn a better way to do this in a few weeks

Block vs Inline

- Block elements, like h1-h6, are displayed on their own lines

- Inline elements, like strong and em, do not require separate lines

Nested Elements

- An element may be inside another element; this is called nesting
- Elements can’t overlap in any other way

Nested Elements

This works:

```
<p>
    <strong>
      Really important text
    </strong>
</p>
```

This may work in some browsers, but it is BAD SYNTAX:

```
<p>
    <strong>
      Really important text
    </p>
</strong>
```

Nested Elements and Indentation

The browser doesn’t care about the indentation in your html file

This is hard for humans to read:

```
 <!DOCTYPE html><html><head> <title>Hello HTML!
</title></head><body><p>Hello HTML!</p><p>This is HTML5!
</p></body></html>
```

Nested Elements and Indentation

This is more readable:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Hello HTML!</title>
    </head>
    <body>
        <p>
            Hello HTML!
        </p>
        <p>
            This is HTML5!
        </p>
    </body>
</html>
```

Viewing Your HTML

- To check your work periodically, just point your browser to the .htm or .html file on disk
- If you use Notepad++, use Run/Launch in Firefox/IE/Chrome

- Next week, we will put pages on an actual web server