

CSC309 Programming on the Web

Summer 2015

About Me

Mashiyat Ahmed Shah Mashiyat

□ 4th Year PhD student

- Worked in the industry for several years
 - 🗆 In Start Up,
 - \Box In mid-size company, and
 - □ In large consultancy firm

Teaching Assistants

Soheil Hassas Yeganeh Email: soheil@cs.toronto.edu

Mohammad Hossein Danesh Email: danesh@cs.toronto.edu

Sukwon Oh

Email: sukwon0709@gmail.com

Any One here Do not Like Web Programing?

What do you like about web programming?

Why I like Web

-- Massive reach! Facebook has 1.23B users (People have multiple accounts)

Facebook monthly users

1m 2004			SOU	RCE: FACEBOOK
6m 2005				
12m 2006				
58m 2007				
2008 145m				
2009	360m			
2010	608m			
2011		845m		
2012			1,056m	
2013				1,230m

Fast Scaling Power

1. AAPL, \$479 billion

2. GOOG, \$404 billion

- 3. MSFT, \$316 billion
- 4. IBM, \$195 billion
- 5. FB, \$184 billion
- 6. ORCL, \$175 billion

7. AMZN \$171 billion

- 8. QCOM, \$130 billion
- 9. INTC, \$123 billion

10. CSCO, \$116 billion <u>http://www.usatoday.com/</u> March 13, 2014



Fast Scaling Power

1. AAPL, \$479 billion

2. GOOG, \$404 billion

- 3. MSFT, \$316 billion
- 4. IBM, \$195 billion
- 5. FB, \$184 billion
- 6. ORCL, \$175 billion

7. AMZN \$171 billion

8. QCOM, \$130 billion

9. INTC, \$123 billion

10. CSCO, \$116 billion <u>http://www.usatoday.com/</u> March 13, 2014

→ Alibaba.com \$168 Billion

What CSC309 is about?

This course provides an introduction to **concepts and technologies of web development**

- Static/dynamic client-side programming
- Static/dynamic server-side programming
- Web architectures
- □ Web development processes
- Security, Scalability, Usability on the Web

CSC309 Objectives and Outcomes

Enabling Knowledge

Ability to define and explain protocols, concepts and technologies that enable web applications

Critical Analysis and Problem Solving

Ability to analyse, design and implement web applications that accommodate specific requirements and constraints with regard to issues of usability, performance and security

Communication

Ability to work as part of a team or individually, explore relevant alternatives and make decision recommendations How many of you have prior web programming experience?

Course Overview

- The Internet and the Web
- HTML Basics
- Cascading Style Sheets (CSS)
- Client-side Scripting (JavaScript)
- Server-Side Programming (NodeJS)
- E/R Model and Database Design
- Asynchronous Communication (Ajax) & JSON
- Web Architectures
- Web Security Issues
- Web Performance Issues
- Other Topics (Going live!, Monetization Strategies, ...)

Self Study Topics

- Almost Everything! The purpose of the class is not to introduce all topics in detail
- We introduce concepts and technologies in some detail in class but is left to you to excel by
 - Reading online material
 - Practice, Practice, Practice
 - Incorporating them to your projects

Lecture Materials

14

Lecture Notes and material are mostly from,

- Eyal de Lara, Professor, DCS
- Manos Papagelis, Instructor of this course in some previous terms.
- Juan Gonzalez, Founder/CEO at Videogami
- "Web Programming Step by Step" by Marty Stepp, Jessica Miller, Victoria Kirst.
- And of of curse from Web



Course Information

Instructors:

- Ahmed Mashiyat (mashiyatATcs.toronto.edu)
- Tutorial (BA1220): Tuesdays, 2pm--3pm
- Lectures (BA1220): Thursdays, 2pm--4pm

Website:

http://www.cs.toronto.edu/~mashiyat/csc309/index.htm

Discussion & Communication (register online):
 Piazza: https://piazza.com/class/i9e3qpq0p144m6

Communication

Office hours :

Mashiyat (BA3289): Thursdays, 12pm--2pm

Email (I prefer communication through piazza):

- Email must include your name
- Subject must include "309"
- Your Email should:
 - State your question clearly, with enough context
 - Have your sign (Name, Student number and CDF login are the most useful)

Course Textbooks

Course Textbooks:

 Robert W. Sebesta. Programming the World Wide Web, 7/E. Addison-Wesley, 2011. (Sebesta)

Marty Stepp, Jessica Miller, Victoria Kirst. Web Programming Step by Step, 2/E. 2012. (Webstepbook)

Online Resources!





Course Prerequisites

19

□ Make sure you have the prerequisites!

- CSC209H1 Software Tools and Systems Programming
- CSC343H1 Introduction to Databases (Automatically Waived)

Note: Students that would like to attend CSC309 but miss one of the prerequisites would need to email the instructor asking for a course prerequisite waiver. (There is a CGPA prerequisite as well, which can only be waived by the Undergraduate Office.)



- Assignment 1: Individual Project
 - Personal Website
 - More details to follow
- Assignment 2: Team Project (2 students)
 Online Game development

 - Start looking for partners
- Assignment 3: Team Project (4 students)
 Three Phases

Class Participation

You will marked on:

- Class participation
 - Its not attendance, however highly encouraged
 - Ask interesting questions (in class or in Piazza)
 - Answer quiz that I will post in piazza
 - Most importantly help each other in finding answers to the questions posted.
 - Be a good team mate (Also counts towards group assignments)



- All deliverables will be submitted electronically using MarkUs
- Deliverables are due at 11:59 p.m. on the due date
 check website for final due dates
- Late Work Policy: Accepted up to 2 days after the due date with a 10% penalty per day
- Demo day: to showcase assignment 3

Plagiarism

- 23
- "The work you submit must be your own, done without participation by others. It is an academic offense to hand in anything written by someone else without acknowledgement."
- You are not helping your friend when you give him or her a copy of your work
- You are hurting your friend when you ask him or her to give you a copy of their work

What is Cheating?

Cheating is

- copying parts or all of another student's assignment
- including code from books, web sites, other courses without attribution
- getting someone else to do substantial parts of your assignment
- giving someone else your solution
- Cheating is not
 - helping to find a bug in a friend's code (be careful)
 - helping each other understand example code

Course Marking Scheme

Work	Weight	Comment	
Assignment 1	10%	Personal Website	
Assignment 2	15%	Online Game	
Assignment 3	30%	1 st Phase – 5% 2 nd Phase – 5% 3 rd Phase -20%	
Final	35%	You must get >=40% to pass the course	

A Few Do's and Don'ts

🗆 Do

- ask questions if you don't understand something
- work together to understand concepts/projects
- use tutorials and office hours
- read material online & practice
- Don't
 - hand in other peoples' work (it's cheating)
 - harass others (see the University's policies)
 - distract or disrupt the class (it's immature)

Today's Overview

- Course Administrivia
- Introduction
 - History of the Internet Video (~8min)
 - http://www.youtube.com/watch?v=9hlQjrMHTv4
 - The Internet and the World Wide Web
- Simple Web Request

THE INTERNET AND THE WORLD WIDE WEB

What is the internet?

- A collection of computer networks that use a protocol to exchange data
- Is the World Wide Web (WWW) and the internet the same?



Brief history

- Began as a US Department of Defense network called ARPANET (1960s-70s)
- Packet switching (in the 60s)
- E-mail is born on 1971
- □ TCP/IP beginning on 1974 (Vinton Cerf)
- □ USENET (1979)
- By 1987: Internet includes

nearly 30,000 hosts







Brief history (cont.)

- WWW created in 1989-91 by Tim Berners-Lee
- Popular web browsers released:
 - Netscape 1994
 - 🗖 IE 1995
- □ Amazon.com opens in 1995
- Google January 1996
- Wikipedia launched in 2001
- MySpace opens in 2003
- Facebook February 2004



The future of the internet?



Key aspects of the internet

- Sub-networks are independent
- Computers can dynamically join and leave the network
- Built on open standards
- Lack of centralized control (mostly)
- Everyone can use it with simple, commonly available software

People and organizations

- Internet Engineering Task Force (IETF): internet protocol standards
- Internet Corporation for Assigned Names and Numbers (ICANN): decides top-level domain names
- World Wide Web Consortium (W3C): web standards



Internet Protocol (IP)

- Simple protocol for data exchange between computers
- IP Addresses:
 - □ 32-bit for IPv5
 - 128-bit for IPv6



Transmission Control Protocol (TCP)

- 36
- Adds multiplexing and reliable delivery on top of IP
 - Multiplexing: multiple programs using the same IP address
 - Reliability: guaranteed, ordered and error-checked delivery
- Port: a number given to each program or service
 port 80: web browser (port 443 for secure browsing)
 - 🗖 port 25: email
 - port 22: ssh
- Some programs (games, streaming media programs) use simpler UDP protocol instead of TCP
Web Browser vs Web Server



Web Browser requests and parses

documents from web servers

□Mozilla Firefox

□Microsoft Internet Explorer (IE)

□Apple Safari

□Google Chrome

Web Server listens for web page requests

□Apache

Microsoft Internet Information Server (IIS)

□...

Domain Name Server (DNS)

- Set of servers that map (translate) written names to IP addresses
 - **Example:** www.cs.toronto.edu \rightarrow **128.100.3.40**
- Many systems maintain a local cache called a hosts file
 - Windows: C:\Windows\system32\drivers\etc\hosts
 - Mac: /private/etc/hosts
 - Linux: /etc/hosts

Uniform Resource Locator (URL)

- Identifies the path to a document on the web server
 Example URL: http://www.cs.toronto.edu/~mashiyat/csc309/index.htm
 - Upon entering this URL into the browser, it would:
 ask the DNS server for the IP address of the URL
 connect to that IP address at port 80
 request the document from the server by sending GET mashiyat/csc309/index.htm
 parse and display the resulting page on the screen

Hypertext Transport Protocol (HTTP)

- Set of commands understood by a web server and sent from a browser
- Some HTTP commands (your browser sends these internally):
 - GET filename: download
 - POST filename: send a web form response
 - PUT filename: upload

HTTP Error Codes

41

When something goes wrong, the web server returns a special "error code" number

Common error codes:

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

Internet Media ("MIME") types

MIME type	file extension
text/html	.html
text/plain	.txt
image/gif	.gif
image/jpeg	.jpg
video/quicktime	.mov
application/octet-stream	.exe

Enabling Web Technologies

- Hypertext Markup Language (HTML) Used for writing web pages
- Cascading Style Sheets (CSS)
 - Stylistic info for web pages
- JavaScript

Interactive web pages (on the client)

Server side Programming language (Python, Nodejs, etc.)

Dynamic and data-driven pages (on the web server)

Enabling Web Technologies (cont.)

- eXtensible Markup Language (XML): Markup language for describing and exchanging data
 Asynchronous JavaScript and XML (Ajax)
 - Creating asynchronous Web applications

•••



Simple Web Request



Thanks to Karen Reid for material in these slides

The request

- How do we tell the server what we want?
- □ How do we even find the server?
- How do the server and browser talk to each other?

HTTP Request



How do we find the server?

- Every computer on the Internet has an Internet address.
- Called an IP address (Internet Protocol)
- □ An IP address is 4 numbers separated by dots.

www.tkf.toronto.on.ca = 207.245.2.3

Domain Name Servers



This is getting complicated!



Actually, it's worse than that

The web page for www.tkf.toronto.on.ca doesn't really live at 207.245.2.3

request

GET / HTTP/1.1 Connection: Keep-Alive Host:www.tkf.toronto.on.ca HTTP/1.1 301 Moved Permanently Date: Tues, 23 July 2002 Server: Apache/1.3.22(unix) Location: http://www.kites.org Content-Type: text/html

reply

So we need to go look up www.kites.org

Now what?

- □ Okay, we have the address.
- What do we do with it?
- Let's look at how two computers communicate.
- HTTP is a high-level protocol
- □ HTTP is specific to the web.
- Computers communicate for other reasons.

Protocols

- Computers use several layers of general protocols to communicate.
- To understand why these layers are important, think about how a company sends you an invoice for a purchase.

Protocols

Invoice: Customer: Karen Reid Order No: 5379				
Qty: 1 Ath 2 128				
		TOTAL	596.74	





Karen Reid	Feb 18, 2001			





TCP/IP: Protocol Framework

TCP/IP Layers

TCP/IP Prototocols

Application Layer	HTTP	FTP		Telnet		SMTP		DNS
Transport Layer	тср			UDP				
Network Layer	IP		ARP		ICMP		IGMP	
Network Interface Layer	Ether	Ethernet		Token Ring			Other Link-Layer Protocols	

TCP/IP: Sending/Receiving Data



TCP/IP

□ Transmission Control Protocol.

Tells us how to package up the data.



Example analogy:

You can think that HTTP is what goes in the envelop, TCP is the envelop, and IP is the truck.

TCP Connection



TCP: Three-way handshake



Details



When something goes wrong

- □ A packet might not arrive
 - traffic overload
 - bit corruption
- Receiver asks for missing packets to be resent. Want to send data as fast as possible.
- Strategies:
 - Send packets as fast as possible (too many lost)
 - Send packets at a certain rate (can go faster)
 - Wait for the ack (too slow)
 - Window-based (adaptive)

TCP Congestion Control

- □ Window-based:
 - some number of packets allowed to be sent and not ack'd
 - as successful ack's arrive, grow window
 - □ if packet loss is detected, cut window size



TCP Congestion Control



Time

The Big Picture



- Client-Server model: a client process wants to talk to a server process
- Client must find server DNS lookup
- Client must find service on server ports
- □ Finally **establish a connection** so they can talk

Routing



Plus the redirect (20 hops)



At least 5 different cities



68

Putting it together



How many messages?

- □ It depends on the size of the web page we retrieve.
- If the web page is 100 Kbytes (small!) it will be broken up into ~80 IP packets.

10 (DNS) + 22 (Connect with toronto.on.ca) + 6 (DNS) + + 80 * 20 hops = 1638 messages!

Types of Connection (TCP/UDP)

Connection oriented model

- Like phone calls
- Uses Transmission Control Protocol (TCP)
- Defined ordering of messages and acks
- Connectionless model
 - Like sending letters via postal service
 - Uses User Datagram Protocol (UDP)
 - More efficient and good for sending broadcasts to many machines

CASCADING STYLE SHEETS (CSS)
Overview

- Basic CSS
- CSS Properties
- More CSS Syntax



The good, the bad and the... ugly!

<pre>{font face="Arial">Shashdot.</pre>
News for nerds!! You will <i></i> never , <u></u> EVER
be
<pre>\$\$ font size="+4" color="red">BORED here!</pre>

BORED

- Tags such as b, i, u, and font are discouraged in strict XHTML
- Why is this bad?

Cascading Style Sheets (CSS)

- Describes the appearance, layout, and presentation of information on a web page
 HTML describes the content of the page
 - Describes how information is to be displayed, not what is being displayed
 - Can be embedded in HTML document or placed into separate .css file

Basic CSS rule syntax

<pre>selector { property: value; property: value;</pre>	
<pre>property: value; }</pre>	CSS
<pre>p { font-family: sans-serif; color: red; }</pre>	CSS

- A CSS file consists of one or more rules
- Each rule starts with a selector
- A selector specifies an HTML element(s) and then applies style properties to them
 - a selector of * selects all elements

Aside: Favorites icon ("favicon")

<link icon"</link 	href="filename" />	type=" <i>MIME type</i> " .	rel="shortcut <i>HTML</i>
<link icon" <i>HTML</i></link 	href="yahoo.gif" />	type="image/gif"	rel="shortcut
🥭 Yaho	oo! - Microsoft Internet	Explorer	



- The link tag, placed in the HTML page's head section, can specify an icon
 - this icon will be placed in the browser title bar and bookmark/favorite

Attaching a CSS file <link>

HTML	

□ A page can link to multiple style sheet files

In case of a conflict (two sheets define a style for the same HTML element), the latter sheet's properties will be used

Embedding style sheets: <style>

<style type="text/css"></td></tr><tr><td></td></tr><tr><td></td></tr><tr><td></style>
HTML

- CSS code can be embedded within the head of an HTML page
- □ Bad style and should be avoided when possible (why?)

Inline styles: the style attribute

This is a paragraph
HTML
output

- Higher precedence than embedded or linked styles
- Used for one-time overrides and styling a particular element
- □ Bad style and should be avoided when possible (why?)

Cascading Style Sheets

- Properties of an element cascade together in this order (precedence):
 - browser's default styles
 - external style sheet files (in a <link> tag)
 - embedded/internal style sheets (inside a <style> tag in the page's header)
 - inline style (the style attribute of the HTML element)

W3C CSS Validator

CSS	
outpu	ut

- jigsaw.w3.org/css-validator/
- checks your CSS to make sure it meets the official CSS specifications



CSS properties for colors

color: red; background-color: yellow

CSS

This paragraph uses the style above

property	description
color	color of the element's text
background-color	color that will appear behind the element

Specifying colors

p { color: red; }
h2 { color: rgb(128, 0, 196); }
h4 { color: #FF8800; }
CSS

This paragraph uses the first style above

This h2 uses the second style above.

This h4 uses the third style above.

- color names: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white (white), yellow
- **RGB codes:** red, green, and blue values from 0 (none) to 255 (full)
- **hex codes:** RGB values in base-16 from 00 (0, none) to FF (255, full)

Grouping styles

p, h1, h2 { color: green;	
} h2 {	
<pre>background-color: yellow;</pre>	
}	CSS

This paragraph uses the above style.

This h2 uses the above styles.

- A style can select multiple elements separated by commas
- The individual elements can also have their own styles

CSS comments /*...*/



- CSS (like HTML) is usually not commented as rigorously as programming languages such as Java
- □ The // single-line comment style is NOT supported in CSS
- The <!-- ... --> HTML comment style is also NOT supported in CSS

CSS properties for fonts

property	description
font-family	which font will be used
font-size	how large the letters will be drawn
font-style	used to enable/disable italic style
font-weight	used to enable/disable bold style

Complete list of font properties

font-family

p { font-family: Georgia; }		
h2 { font-family: "Courier New"; }	CSS	

output

□ Enclose multi-word font names in quotes

More about font-family

	CSS	

- We can specify multiple fonts from highest to lowest priority
- Generic font names:
 - serif, sans-serif, cursive, fantasy, monospace
- If the first font is not found on the user's computer, the next is tried
- Placing a generic font name at the end of your font-family value, ensures that every computer will use a valid font

font-size

font-size: 24pt;		
	CSS	

- units: pixels (px) vs. point (pt) vs. m-size (em)
 16px, 16pt, 1.16em
- vague font sizes: xx-small, x-small, small, medium, large, x-large, XX-large, smaller, larger
- percentage font sizes, e.g.: 90%, 120%

font-size

font-size: 2	4pt;	
	CSS	

- pt specifies a number of points, where a point is 1/72 of an inch onscreen
- **px** specifies a number of pixels on the screen
- em specifies number of m-widths, where 1 em is equal to the font's current size

font-weight, font-style

р {	
<pre>font-weight: bold; font-style: italic;</pre>	
}	CSS
output	-

Either of the above can be set to normal to turn them off

CSS properties for text

property	description
text-align	alignment of text within its element
text-decoration	decorations such as underlining
line-height, word-spacing, letter-spacing	gaps between the various portions of the text
text-indent	indents the first letter of each paragraph

Complete list of text properties

text-align

blockquote { text-align: h2 { text-align: center; <i>CSS</i>	<pre>justify; } }</pre>
output	

text-align can be left, right, center, or justify



text-decoration

<pre>p { text-decoration: underline; </pre>	
}	CSS

output

- can also be overline, line-through, blink, or none
- effects can be combined:

text-decoration: overline underline;

The list-style-type property

{ list-style-type: lower-roman;

CSS

Possible values:

- i. none : No marker
- ii. disc (default), circle, square
- iii. Decimal: 1, 2, 3, etc.
- iv. decimal-leading-zero: 01, 02, 03, etc.
- v. lower-roman: i, ii, iii, iv, v, etc.
- vi. upper-roman: I, II, III, IV, V, etc.
- vii. lower-alpha: a, b, c, d, e, etc.
- viii. upper-alpha: A, B, C, D, E, etc.
- x. lower-greek: alpha, beta, gamma, etc.
- others: hebrew, armenian, georgian, cjk-ideographic, hiragana...

Body styles

<pre>body { font-size: 16px; }</pre>		
	CSS	

- □ Applies a style to the entire body of your page
- □ Saves you from manually applying a style to each element

Inheriting styles

100

body { font-family: sans-serif; background-color: yellow; } p { color: red; background-color: aqua; } a { text-decoration: underline; } h2 { font-weight: bold; text-align: center; } CSS

This is a heading

A styled paragraph. <u>Previous slides</u> are available on the website.

• A bulleted list

- when multiple styles apply to an element, they are inherited
- a more tightly matching rule can override a more general inherited rule

Styles that conflict

p, h1, h2 { color: blue; font-style: italic; h2 { color: red; background-color: yellow; } CSS

This paragraph uses the first style above. *This heading uses both styles above.*

when two styles set conflicting values for the same property, the latter style takes precedence

OL

CSS properties for backgrounds

property	description
background-color	color to fill background
background-image	image to place in background
background-position	placement of bg image within element
background-repeat	whether/how bg image should be repeated
background-attachment	whether bg image scrolls with page
background	shorthand to set all background properties

background-image

body {
background-image: url("images/draft.jpg");
}
CSS
This is the first paragraph
This is the second paragraph...
It occupies 2 lines

background image/color fills the element's content area

background-repeat

body

background-image: url("images/draft.jpg");
background-repeat: repeat-x;

CSS



can be repeat (default), repeat-x, repeat-y, or no-repeat

background-position

body

background-image: url("images/draft.jpg"); background-repeat: no-repeat; background-position: 370px 20px;

This is the first paragraph

This is the second paragraph... It occupies 2 lines



- value consists of two tokens, each of which can be top, left, right, bottom, center, a percentage, or a length value in px, pt, etc.
- value can be negative to shift left/up by a given amount