

## Announcements

- Due dates extended:
  - Project 1B—Wednesday by 10pm
    - 1-1-1 rule Thursday by 10pm
  - Lab 5—Friday by 10pm

11/2/2009 D.A. Clements, MLIS, UW iSchool 1

## Announcements

- Vocabulary for the week has been posted in GoPost
- Reading
  - Ch 18 for today and Wednesday
  - Ch 21 for Friday

11/2/2009 D.A. Clements, MLIS, UW iSchool 2

## Programs Defined

- A *program* is an algorithm written for a specific programming language and specific circumstances

11/2/2009 D.A. Clements, MLIS, UW iSchool 3

## TYPES OF PROGRAMMING LANGUAGES

11/2/2009 D.A. Clements, MLIS, UW iSchool 4

## High- vs. Low-Level Languages

Translated ↓	HIGH-LEVEL	Humans Understand		
		Fortran	C, C++, C#	JavaScript
		High-Level Language		
		Assembly Language		
		Machine Language		
	LOW-LEVEL	Computers Understand		

11/2/2009 D.A. Clements, MLIS, UW iSchool 5

## Videos

- High- and low-level programming languages

11/2/2009 D.A. Clements, MLIS, UW iSchool 6

## Human-Understandable Code

- Today's programs are written in "high-level" language (HLL) that we can understand (and debug)
- HLL use "real" words—if, while, when, until, push, pop, print, set, etc.
  - Words look like English
  - Have a precisely defined meaning for the computer
  - Make it easier for us to understand (and troubleshoot)
- For example:
 

```
if (today=='Wednesday')
    print "I have lecture today!"
else
    print "Tonight is time to study!"
```

11/2/2009 D.A. Clements, MLIS, UW iSchool 7

## High-Level Languages (HLL)

- **Video: High-Level Programming Languages**

11/2/2009 D.A. Clements, MLIS, UW iSchool 8

## Assembly language

- The lowest level language humans can understand
- Example
  - LOOP: MOV.B r0, #80 ;initialize counter

11/2/2009 D.A. Clements, MLIS, UW iSchool 9

## What computers understand

- Machine code
  - Assembly code is translated to binary:
    - 0011 0000 1000 0000
  - Binary is how computer stores information
    - all zeroes and ones
      - Magnetized or not
      - Off or on
      - Bumpy surface on the CD or not

11/2/2009 D.A. Clements, MLIS, UW iSchool 10

## Machine code

```
01101110001011100111011001111011000110011
00111101101010111001110110011001000101110
01110110011110110001100110011110110101011
10000101110011101100111101100011001100111
10110101011100001011100111011001111011000
11001100111101101010111000010111001110110
01111011000110011001111011010101110000101
11001110110011110110001100110011110110101
01110000101110011101100111101100011001100
11110110101011100001011100111011001111011
00011001100111101101010111000110011001111
```

11/2/2009 D.A. Clements, MLIS, UW iSchool 11

## Translating human to machine

Translated ↓	HIGH-LEVEL	Humans Understand		
		Fortran	C, C++, C#	JavaScript
		High-Level Language		
		Assembly Language		
		Machine Language		
	LOW-LEVEL	Computers Understand		

11/2/2009 D.A. Clements, MLIS, UW iSchool 12

## Compiled Languages

- Compiled languages are translated to machine code (assembly language) before they are run.
- Whenever you make changes to your program, you have to recompile the program *again*.
- Because they already speak the computer's language, they run faster.
- Sometimes, they run by themselves—.exe files (NotePad2.exe)—or run with an engine (the Java virtual engine).

11/2/2009 D.A. Clements, MLIS, UW iSchool 13

## Compiled Languages

- Examples:
  - Java
  - C family
  - Visual Basic
  - COBOL
  - ForTRAN
  - many others

11/2/2009 D.A. Clements, MLIS, UW iSchool 14

Also called *scripting* languages

## INTERPRETED LANGUAGES

11/2/2009 D.A. Clements, MLIS, UW iSchool 15

## Interpreted Languages

- An interpreter translates from JavaScript to machine language while the Web browser renders the page
- The interpreter is part of the Web browser.
  - The JavaScript interpreter is available in all major Web browsers

11/2/2009 D.A. Clements, MLIS, UW iSchool 16

## How the Interpreter Works

- The interpreter translates the script to machine language while the program runs!
  - Two tasks at once—translating and running the program!
  - Scripts run slower than compiled programs

11/2/2009 D.A. Clements, MLIS, UW iSchool 17

## The Advantages...


- Scripted languages are interpreted *on the fly*—while they are running!
  - Make changes while the program is running!
  - Provides a dynamic, responsive, and interactive experience for the user

Scripts respond to user input

Font Size


normal | large | largest

11/2/2009 D.A. Clements, MLIS, UW iSchool 18



All about  
**JAVASCRIPT**

11/2/2009 D.A. Clements, MLIS, UW iSchool 19




## JavaScript

- Java was developed by Sun Microsystems and is seen on the Web mostly in Java Applets.

JavaScript is **not** Java!


11/2/2009 D.A. Clements, MLIS, UW iSchool 20



## Brief History of JavaScript

- Released with Netscape Navigator in 1995.
- Ratified by the European Equipment Manufacturer's Association (ECMA)
- Result:
  - ECMAScript is the core spec for the JavaScript language
  - Netscape, MS, and the others try to conform to the spec

11/2/2009 D.A. Clements, MLIS, UW iSchool 21




## Divergence from standards....

Developer	Name
Netscape (now Mozilla)	JavaScript
Microsoft	JScript

- Programmers call both **JavaScript**.
- Both comply differently with the standards

11/2/2009 D.A. Clements, MLIS, UW iSchool 22




## Javascript & the Web

*Adding interaction to a static HTML page*

D.A. Clements

11/2/2009 D.A. Clements, MLIS, UW iSchool 23



## Objectives

- Understand how JavaScript and HTML interact
- Understand where to place JavaScripts on the HTML page

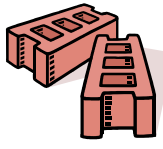
11/2/2009 D.A. Clements, MLIS, UW iSchool 24

## Programming Concepts

- Programming: Act of formulating an algorithm or program
- Basic concepts have been developed over last 50 years to simplify common programming tasks
- Concepts will be expressed in JavaScript

## The Web page

- Without JavaScript the Web page is like a brick; it just sits there!



11/2/2009 D.A. Clements, MLIS, UW iSchool 26

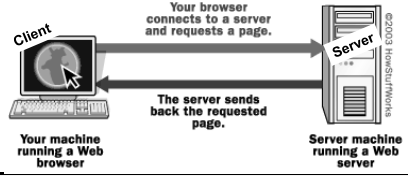
## Scripts

### Client-side scripts

- Your Web browser on your computer is the client

### Server-side scripts

- Web server
- Database server
- File server



11/2/2009 D.A. Clements, MLIS, UW iSchool 27

## Web browser and JavaScript

- The major Web browsers have a built-in interpreter that parses JavaScript
  - Parses: breaks into smaller pieces that can be translated into machine code

11/2/2009 D.A. Clements, MLIS, UW iSchool 28

## Placing JavaScripts on a Web page

Types of scripts:

- Body scripts
- Header scripts
- External scripts

11/2/2009 D.A. Clements, MLIS, UW iSchool 29

## Body Script

```

<html>
  <head>
    <title>Name of Page</title>
  </head>
  <body>
    <script type="text/javascript">
      //JavaScript goes here
    </script>
  </body>
</html>

```

11/2/2009 D.A. Clements, MLIS, UW iSchool 30

## Header Script

```

<html>
  <head>
    <title>Name of Page</title>
    <script type="text/javascript">
      //JavaScript goes here
    </script>
  </head>
  <body>
    Body content goes here
  </body>
</html>

```

11/2/2009 D.A. Clements, MLIS, UW iSchool 31

## Linking to External JavaScripts

- Linked in the <head>
- src gives path to file and its name

```

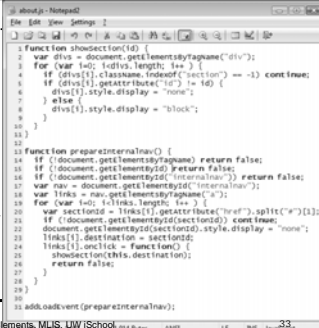
<html>
  <head>
    <title>Name of Page</title>
    <script type="text/javascript"
      src="basic.js"></script>
  </head>
  <body>
    Body content goes here
  </body>
</html>

```

11/2/2009 D.A. Clements, MLIS, UW iSchool 32

## External JavaScripts

- Make changes scripts in one place
- Reusable
  - Link to
    - any page,
    - every page, or
    - many sites



```

Function showSection(id) {
  var divs = document.getElementsByTagName("div");
  for (var i=0; i<divs.length; i++) {
    if (divs[i].className.indexOf("section") == -1) continue;
    if (divs[i].getAttribute("id") != id) {
      divs[i].style.display = "none";
    } else {
      divs[i].style.display = "block";
    }
  }
}

Function prepareInternalNav() {
  if (!document.getElementById("internalNav")) return false;
  if (!document.getElementById("internalNav")) return false;
  var nav = document.getElementById("internalNav");
  var links = nav.getElementsByTagName("a");
  for (var i=0; i<links.length; i++) {
    var sectionId = links[i].getAttribute("href").split("#")[1];
    if (!document.getElementById(sectionId).style.display = "none");
    links[i].onclick = function() {
      showSection(this.destination);
      return false;
    }
  }
}

addLoadEvent(prepareInternalNav);

```

11/2/2009 D.A. Clements, MLIS, UW iSchool 33

## Best Practice

- Best practice to separate Content from Action from Appearance
  - Put styles in external CSS
  - Put scripts in external JavaScript files
  - Leave only the HTML markup and content on the page
- Accomplishing that goal takes more experience....

11/2/2009 D.A. Clements, MLIS, UW iSchool 34

## Summary

- Understand how JavaScript and HTML interact
- Understand where to place JavaScripts on the HTML page

11/2/2009 D.A. Clements, MLIS, UW iSchool 35