CISM 190: Intro to Tech

April 2017

Week 4

Agenda for today

- VR discussion
- Quiz time
- Lecture: file systems, HTML, JavaScript
- Exercise: build a file system
- Exercise: build your first webpage!
- Lab time

Lecture Rules

- There are no stupid questions
- Learning is the objective
- Everyone learns together
- Help each other

Discussion: Virtual Reality

Virtual/Augmented/Mixed Reality (VR/AR/MR)

What is virtual, augmented, mixed reality? What are the differences between the three?

What are current consumer applications of VR/AR/MR?

What commercial applications of VR/AR/MR might we see in the next 3 years?

What are the benefits and risks of using VR/AR/MR in prison?

How could those other benefits and risks be applied to other groups? (e.g. teachers, police, social workers)

Moodle log in

To log into Moodle:

1. Open Chrome. It should automatically redirect you to moodle.un-loop.org (then turn into an IP address)

2. Your username is your last name and first initial
    1. Example: burgundyr

3. Your password is a tilde (~), the first 4 letters of your last name with the first letter in CAPS, and the last 3 numbers of your DOC number
    1. Example: ~Bur1234

4. You should see “Unloop Intro to Tech” on your dashboard

Quiz time!

Today's Class

- File Systems & files
- Browsers
- HTML
- JavaScript

File Systems

- I know, *very* exciting
- For the purpose of understanding:
  - File System = Computerized Filing Cabinet
- What is it *really:*
  - A file system is a collection of files, folders, directories and associated pointer data, security data and other information necessary to manage the files.

File Systems, continued!

- What are common file systems?
  - NTFS (NEW TECHNOLOGY FILE SYSTEM) WINDOWS NT, XP, VISTA, WINDOWS 7, WINDOWS 10)
  - FAT16 (FILE ALLOCATION TABLE 16 BIT)
  - FAT32 (FILE ALLOCATION TABLE 32 BIT)
  - ISO 9660 (INTERNATIONAL STANDARDS ORGANIZATION) AKA CDFS
- File systems reside on:
  - Disk Drives
  - Flash Drives
  - CD ROMS/DVDs
  - Network Drives

WHAT IS A FOLDER?

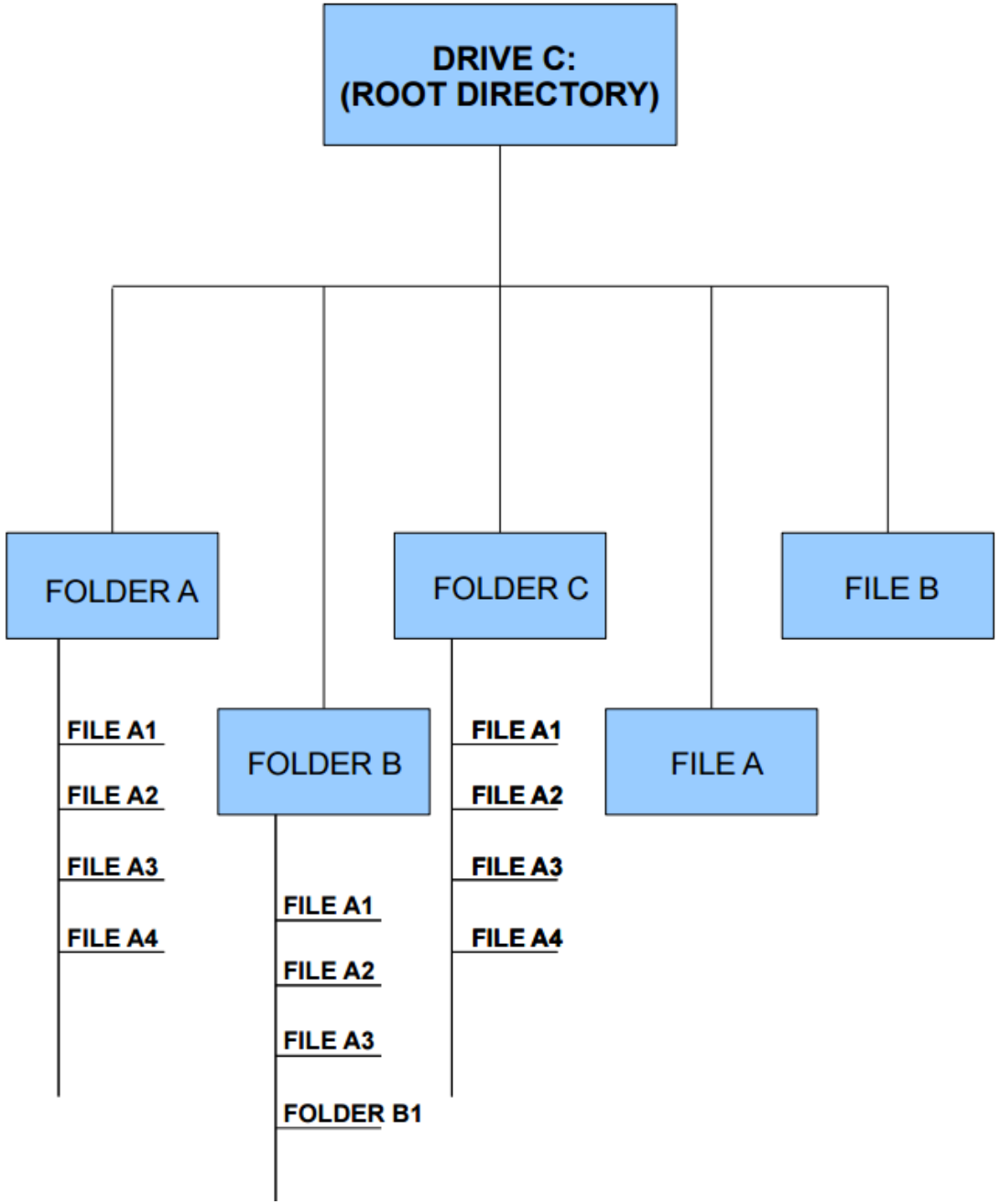
A folder is a container in a file system that contains files and folders. Folders are used to organize the information that is stored on the storage device. Folders are also called **directories**.

WHAT IS A FILE?

A file is a collection of sequential bytes of data. A file may contain: • PROGRAM(S) • DATA • MUSIC • PICTURES • MOVIES, etc

Application programs are executable code that perform useful work for the user. Examples are – Windows Explorer, Internet Explorer, Wordpad, Turbo Tax, etc.

Data files contain information that is useful and of value to the user of the personal computer e.g. text documents, pictures, financial data, (stock, income tax data, bank account, etc.)



File systems are like trees!

- Root = drive/directory
- Branches = folders
- Files = leaves

Using File Explorer

Panes

- Navigation Pane
- Content Pane
- Preview Pane
- Ribbon

PLUS:

- Search

With File Explorer we can copy, delete and move files. Simply right click on the file name and select the Cut, Copy, Delete, or Rename options.

Naming Files and Folders

- **Consistent, Descriptive**
- Options:
  - Project Name
  - Location
  - User name
  - Date/time (YYYYMMDD, DDMMYY)
  - Version Numbers (v1, v2)
- Try not to make file names too long, since long file names do not work well with all types of software.
- Special characters such as ~ ! @ # $ % ^ & * ( ) ` ; < > ? , [ ] { } ' " and | **should be avoided.**
- You *can* use:
  - Underscore: david_Almeida
  - Dashes:
  - CapitalizationLikeThis (called camel case)

File Types

- You can’t remember them all…
- Here’s a sample list
  - .doc
  - .csv
  - .xls
  - .html
  - .js
  - .pdf
  - .jpeg
  - +10000000 more

File Paths

- For understanding:
  - File Path is the directions to your file
- Example:
  - C:\Users\david\Downloads\Unloop

What would happen if we tried to name a file: david/almeida.html?

Exercise: Build a file system

- In your student drive, set-up a series of folders to organize your work in Web Development (**you will need to create folders!)**
  - Example:
    - In Class Assignments
    - Homework
    - Materials
    - Archive
- Rename files using naming best practices
- Shortcut keys:
  - Ctrl +x = cut
  - Ctrl +c = copy
  - Ctrl + p = paste
- Grading - you will be graded on:
  1. Are the files in your folder logically organized?
  2. Are files and folders named using best practices?

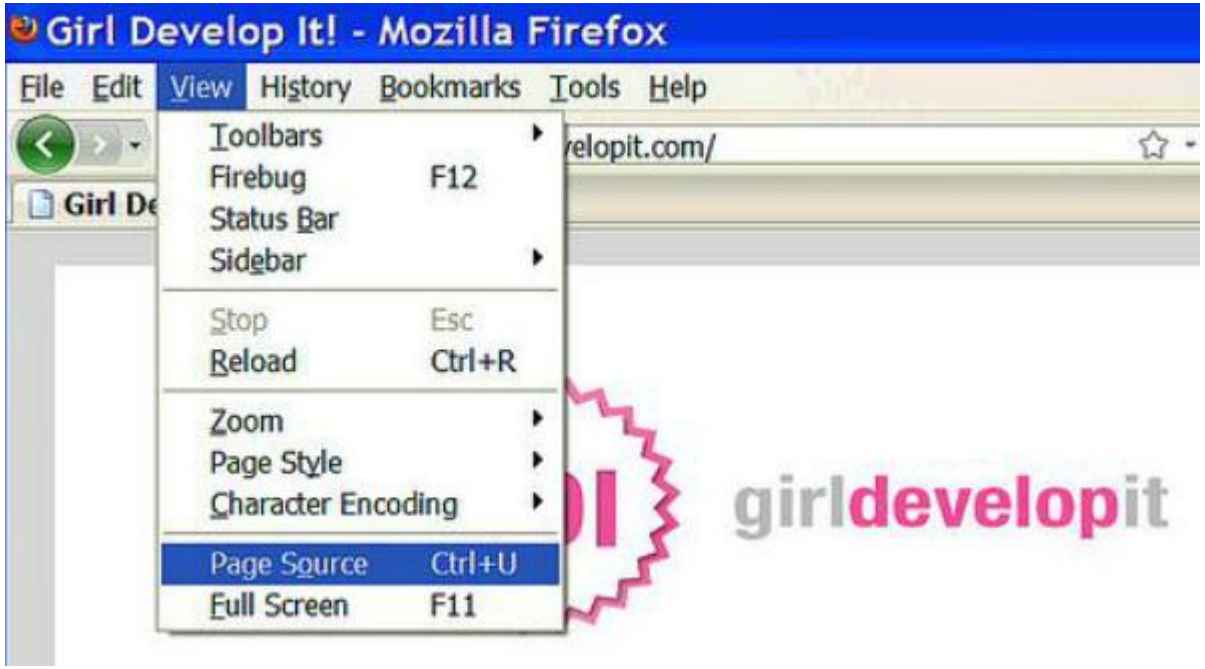
Web Development Files

- HTML (Hypertext markup language)
- CSS (Cascading Style Sheets)
- JavaScript

How do you make an HTML file?
What about a CSS file?
What about a JavaScript file?

Browsers

- Browsers are programs that render content on the web, most often HTML pages
- When you are looking at a browser rendering content (like a webpage), you can click “view source” to look at the HTML file



HTML

- Hypertext Markup Language
- The content of your page!
- It’s a text file, containing markup tags
- Those tags tell the browser what to display

HTML

```
1  <!DOCTYPE HTML>
2  <html>
3  <head>
4  <title>JavaScript Hello World Example</title>
5  </head>
6  <body>
7  <h1>Hello World</h1>
8  </body>
9  </html>
```

- Each tag has a
  - Start tag <body>
  - End tag </body>
  - Some content in between
  - Optional attributes
- Think of a tag as a “command” to the browser
- Head and Body
  - <head> is a description of the page purpose/content. Nothing is rendered
  - <body> tag contains the actual content of a page

```
7  <h1>Header 1</h1>
8  <h2>Header 2 </h2>
9  <h3>Header 3</h3>
```

What goes in the “body”

- Headers: <h1> </h1>, <h2> </h2>, etc.

```
7  <h1>Header 1</h1>
8  <h2>Header 2 </h2>
9  <h3>Header 3</h3>
```

Header 1

Header 2

Header 3

- Paragraphs:
- Line Breaks:
 </br>

```
7  <p>This is my very first paragraph. Obviously, I'm thrilled</p>
8  <p>This is my second paragraph. <br>
9      I'm a little less thrilled. <br> But at least it has line breaks</p>
```

This is my very first paragraph. Obviously, I'm thrilled

This is my second paragraph.
I'm a little less thrilled.
But at least it has line breaks

What goes in the “body”, lists!

- Lists: <li>
- Unordered Lists <ul>
- Ordered Lists: <ol>

```
 7  <ol>
 8  <li>Ordered</li>
 9  <li>List</li>
10  </ol>
11
12  <ul>
13  <li>Unordered</li>
14  <li>List</li>
15  </ul>
```

1. Ordered
2. List

- Unordered
- List

Attributes

- A great way to embed links
  - <a>

```
1  <!DOCTYPE HTML>
2  <html>
3  <head>
4  <title>Attribute Example</title>
5  </head>
6  <body>
7  <a href="http://twitter.com/girldevelopit">Twitter</a>
8  </body>
9  </html>
```

Exercise 1: Create an HTML File

- Create & save a HTML document as “index.html” in your “websiteLab” folder
- Add your metadata using the <html></html> tag
- Add a title that says “Your name’s website” (in the <title></title> tags)
- Add a header that says “Hello World” (use <h1></h1> tags)
- Add a paragraph detailing why you love breakfast as a meal (use tags)
- **Bonus**
  - Add a list of your favorite breakfast foods (mess with ordered/unordered)
  - Try to minimize the number of times you use the mouse! Can you write a tag without it?

Note: When you are writing code, it can be time consuming to go back and forth
Use the printed out “Hotkeys” as much as possible.

Comments

- Comments are lines that will be ignored by your code.
- Options
  - /* Comment here */
- Useful to document how code works or why it is there – makes your code “readable”
- COMMENT ALL THE TIME!

- **Exercise: Add a comment to your HTML file**

At this stage you should have…

An HTML file named “index.html” in your “websiteLab” folder in your share

The title of the page should be “yourname’s website”

There should be a header that says “Hello world”

There should be a paragraph that says a little something about breakfast

There should be a comment in your file

And maybe some lists and stuff

JavaScript, what does it do?

- Makes your code interactive
- Drawing, animation
- Keeps tracks of users with cookies!
  - JavaScript can change HTML content
  - JavaScript can change HTML attributes
  - JavaScript can change CSS style
  - JavaScript can hide HTML elements
  - JavaScript can show hidden HTML elements

JavaScript, Output

- Writing into an window alert box

```
<script>
window.alert(5 + 6);
</script>
```

- Writing into the HTML output

```
<script>
document.write(5 + 6);
</script>
```

- Writing into an HTML element

```
<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = 5 + 6;
</script>
```

- Writing into the browser console

```
Console.log(5+6)
```

JavaScript, Statements

- Each instruction in JS is a “statement”, like:
  - Console.log(‘Hello World!’);
- Variable
  - Used to store values

```
Var x = 2;
Console.log(x);
```

  - Values can be updated

```
Var x = 2;
x = x + 1;
```

JavaScript, Write your first program!

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Web Page</h1>

<script>
window.alert(5 + 6);
</script>

</body>
</html>
```

Exercise 3: Write a JavaScript program

- Write a program that has an output of your own name using JavaScript
- Bonus:
  - Use a different output method than your first try
  - Use a variable
  - Use a function

At this stage you should have…

Required: (3 points each)

1. An HTML file named “index.html” in your “websiteLab” folder in your share
2. The title of the page should be “yourname’s website”
3. There should be a header that says “Hello world”
4. There should be a paragraph that says a little something about breakfast
5. There should be a comment in your file
6. An output of your own name on the page (window alert or html output)

Optional: (.5 point each)

1. An ordered list
2. An unordered list
3. An output using a different method than you used before that says “Robots”

JavaScript Data Types

- **String**: an immutable string of characters
  - Var greeting = “hello there”;
- **Number**: whole or decimal number
  - Var pi = 3.14;
- **Boolean**: Represents a logical value of true or falce
  - Var dogsrule = true
- **Undefined**: represents a value that hasn’t been defined yet
  - Var notDefinedYet;
- **Null**: Represents an explicitly empty value
  - Var goodPickupLines = null;

JavaScript, Expressions

- Variables can also store the result of any expression
  - Var x = 2+2
  - Var y = x * 3

  - Var name = ‘Lindsey’;
  - Var greeting = ‘Hello’ + name;

JavaScript, Comments

- Again, comments are human-readable text ignored by computer
- Used for documentation
- Ways to comment
  - // comment here
  - **var x= 4** // comment after a line
  - /* or like you did in css */
- COMMENT ALL THE TIME!

JavaScript, Functions

- Functions are re-usable collections of statements

First, declare the function

```
function sayMyName() {
    console.log('Hi Lindsey!')
}
```

Then call on it

```
sayMyName();
```

JavaScript, Arguments/Parameters

- Functions can accepts any number of named parameters arguments
  - Parameters are the names listed in the function definition, EXAMPLE: num1
  - Arguments are the real values passed to and received by the function, EXAMPLE: 3

```
function addNumbers (num1,num2){
    var result = num1 + num2;
    return results;
}
addNumbers (3, 2);
```

For the rest of class today:

- Finish up the Blockly lab if you haven't already
- Finish today's exercise: first website
- Spend some time on your Unloop application
- If you finish all this stuff and want to push forward
  - Experiment with what we learned about HTML & JavaScript
  - Come see me for the Week 5 lab