Full Stack Web development

Lecture 1: Course Overview and Basics of Computer Usage

Today and Tomorrow:

- Introductions
- Course Overview
- Basics of Computer Usage
- Basics of the World Wide Web

Instructor – David Adkins

- **Engineering Manager – Facebook**
- Former VP, Technology at The Buffalo News
- More than 20 years experience developing websites in a variety of languages (PHP, .Net, GoLang, Python, JavaScript, React, etc)
- MS in Computer Science focused on Artificial Intelligence

Instructor – Eliot Budde

- **Software Engineer – Town News**
- Former Developer at The Buffalo News
- 5 years experience developing front and back-end websites in React, Golang, PHP and other technologies
- MA in English

Introductions

- Name
- In three sentences, tell us your story.
- Past web development, programming, or computer experience?
- What do you think make good qualities for a web developer?

OR

- What qualities do you possess that you think will make you a good developer?

Course Overview

- No letter grading, you get what you put into the class. The goal of this class is to build a portfolio that you can show potential employers and to give you the skills necessary for an entry level position.
  - All homework must be completed to receive your certificate of completion.

- Attendance will be taken during each class. If you cannot attend class, please let your instructors know.

- Expectation of a minimum of 10 hours of additional studying each week, outside of lecture/lab time.

- Very interactive, bring computers every class to write live code. Ask questions and clarifications. Be present, put phones away and give your instructors' your attention.

Lecture Format

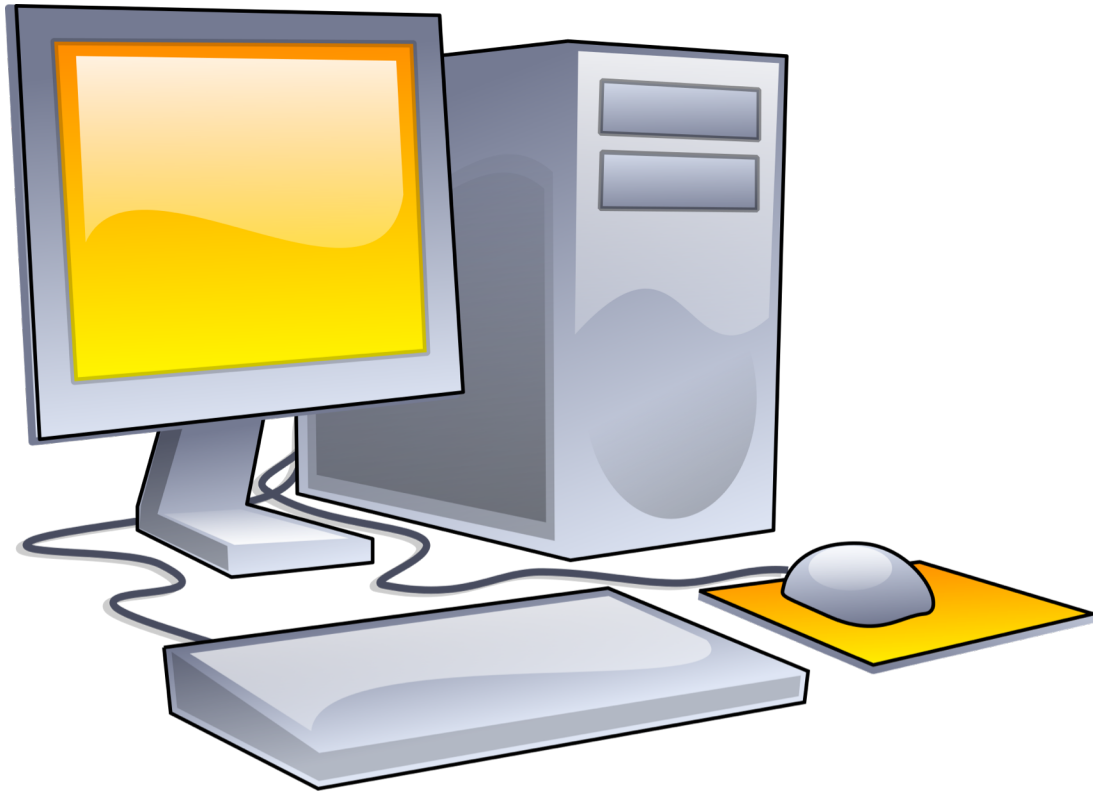
- Lectures are recorded, so if you miss one you can catch up.
  - Very Intensive, if you miss class, you will get left behind quickly.
  - Most lectures are a combination of slides and writing code, you can write the code at the same as the speaker, but its recommended to focus on the speaker and go back after class to follow along with the coding examples.
- Labs
  - Generally there will be labs in each module to help reiterate the lectures. Labs are required and we will explain how they work in the first lab class. You will write code while the instructor checks in on your.
- Homework due every Monday at the start of class, which is generally turning in a lab assignment.
- Each week, the lecture slides will be posted online ahead of class.
  - Before each class, you should review the lecture slides.

Tools

We will be using a variety of tools in this class and you will be working in groups often. Homework for next class should be to become familiar with these tools.

- Google Jamboard – Shared whiteboard https://jamboard.google.com
- Codeshare.io – Shared coding environment https://codeshare.io
- Google Docs – Shared text documents https://docs.google.com

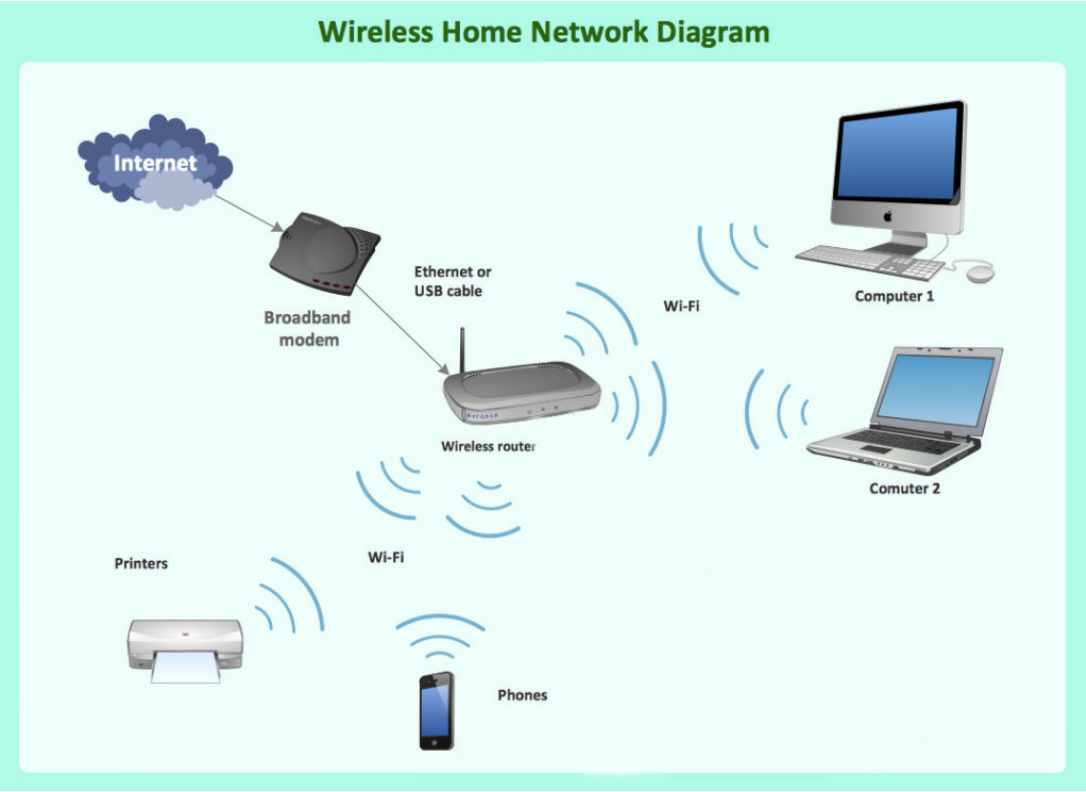
Computers & Networking



Source: Wikipedia.com

- Monitor
- Computer
  - CPU
  - Memory (RAM)
  - Graphics
  - Storage (Hard Drive)
- Mouse
- Keyboard

How does a computer connect to the Internet?



Source: http://beonservices.com

- Your computer connects to a modem, either through a wired connection or a wireless connection.
- That modem connects to an Internet Service Provider (ISP), like Spectrum or Verizon.
- The ISP connects your computer to the Internet.

But How Does that work?

- IP Addressing
  - IPv4 (255.255.255.255)
    - Decimal (0-9)
  - IPv6 (ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff)
    - Hexidecimal (0-f)
- Localhost (127.0.0.1)
- Domain Names (DNS)
  - TLD (.com, .net, .org, etc)
  - Subdomains (dev.davidadkins.com)

Protocols

- So, we have domains and IP addresses, but what are protocols?

- HTTP(S) – secure and non-secure web traffic
- FTP(S) – secure and non-secure file transfer
- SSH – secure shell access to servers (more on this later)

Web Browsers

Usage share of all browsers

Browser	StatCounter[16] April 2020	NetMarketShare[17] April 2020	Wikimedia[18] November 2019
Chrome	62.48%	65.96%	48.7%
Safari	19.94%	17.26%	22.0%
Firefox	4.21%	3.19%	4.9%
Samsung Internet	3.40%	2.68%	2.7%
UC	1.97%	0.82%	0.3%
Opera	1.73%	0.89%	1.1%
Edge	2.67%	3.03%	1.9%
IE	1.41%	2.13%	3.9%
AOSP	0.57%	0.47%	0.2%
Others	1.62%	3.57 %	14.3%

Source: Wikipedia.com

- Which web browser do you use most often?

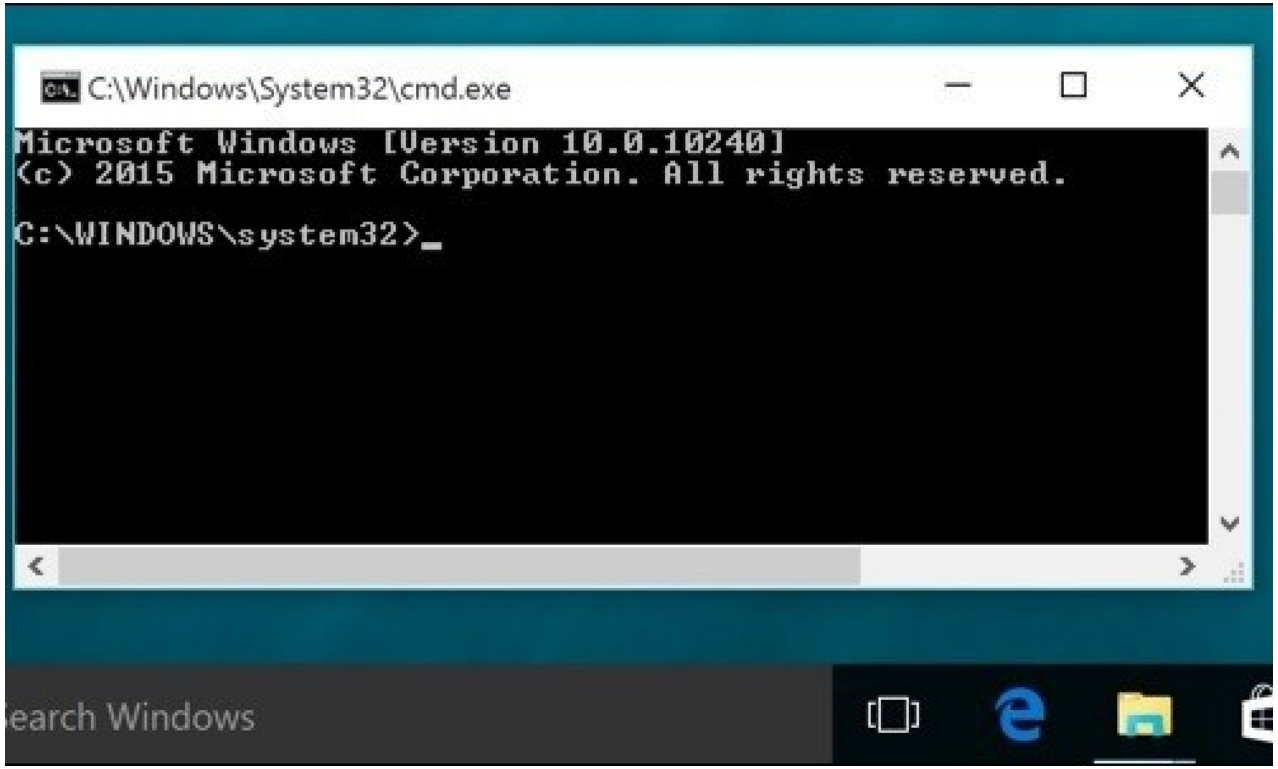
Web Browser Demo (Chrome / Firefox)

- Let’s look at Chrome and Firefox and some of the tools available to you as a developer.

Command Line Tools

- In Windows, click the Start button, then in the search box type **cmd** and hit enter.

- You should see the window at the right.



Command Line Tools

Common Commands:

(If there's a '/' on the left is windows, right is equivalent on Mac)

cd – change directory

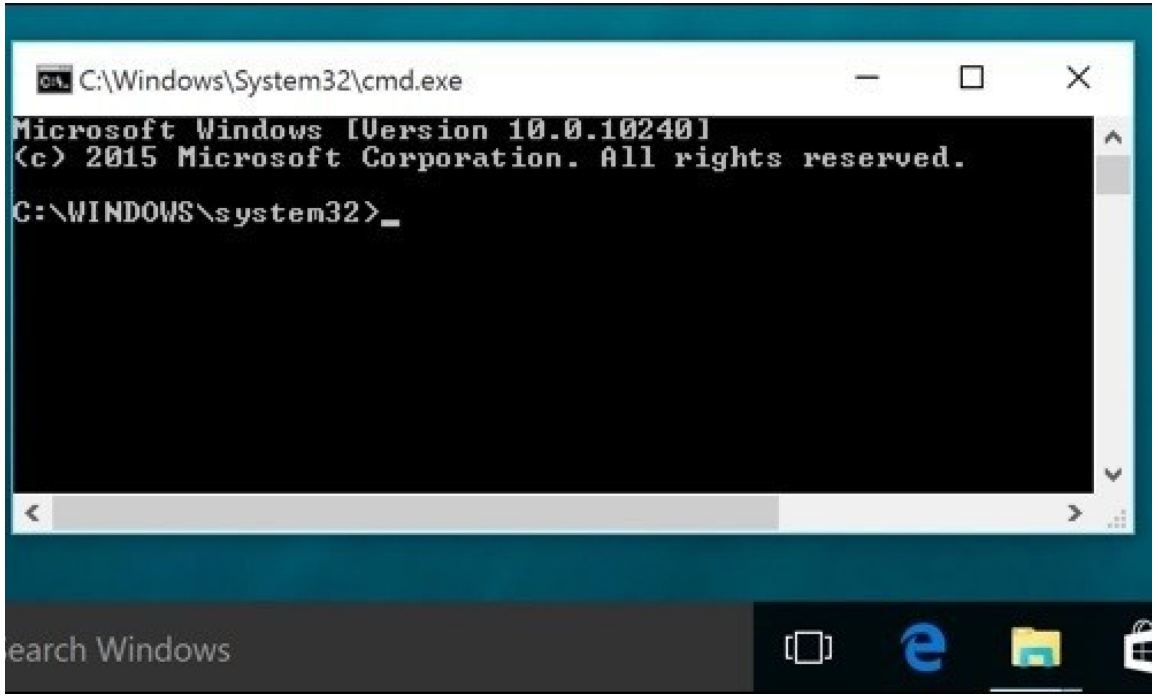
dir / ls - list

copy / cp – copy

move / mv – move

del / rm – delete

More Information: https://www.thomas-krenn.com/en/wiki/Cmd_commands_under_Windows

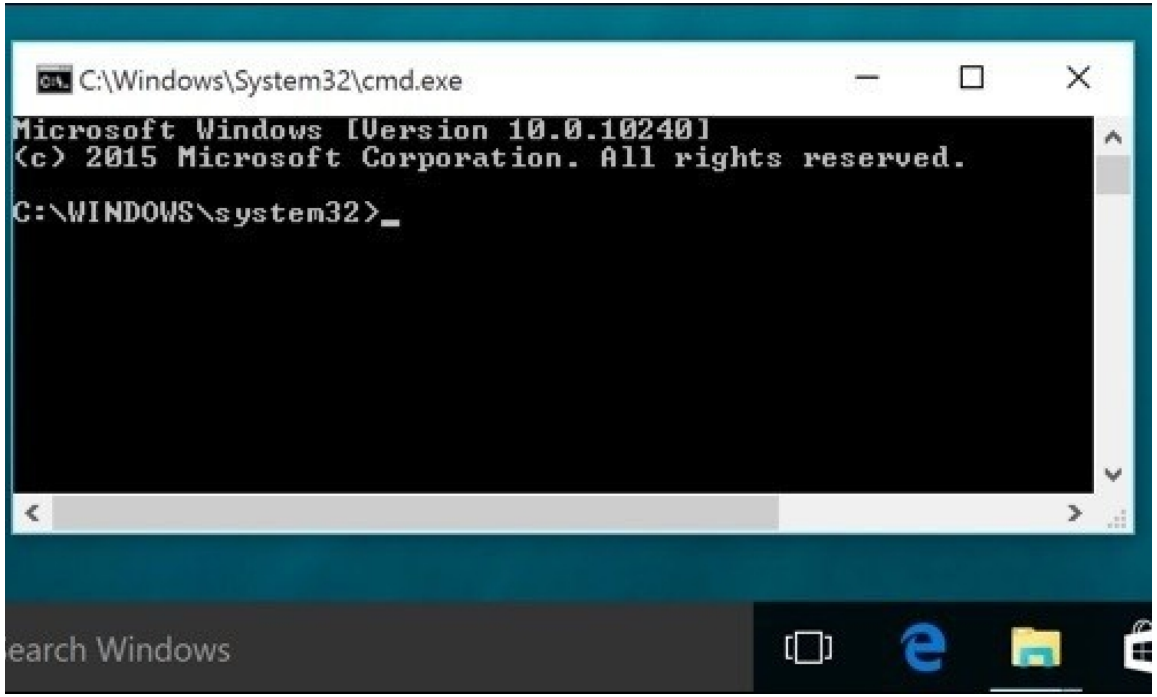


Command Line Tools

Tutorial:

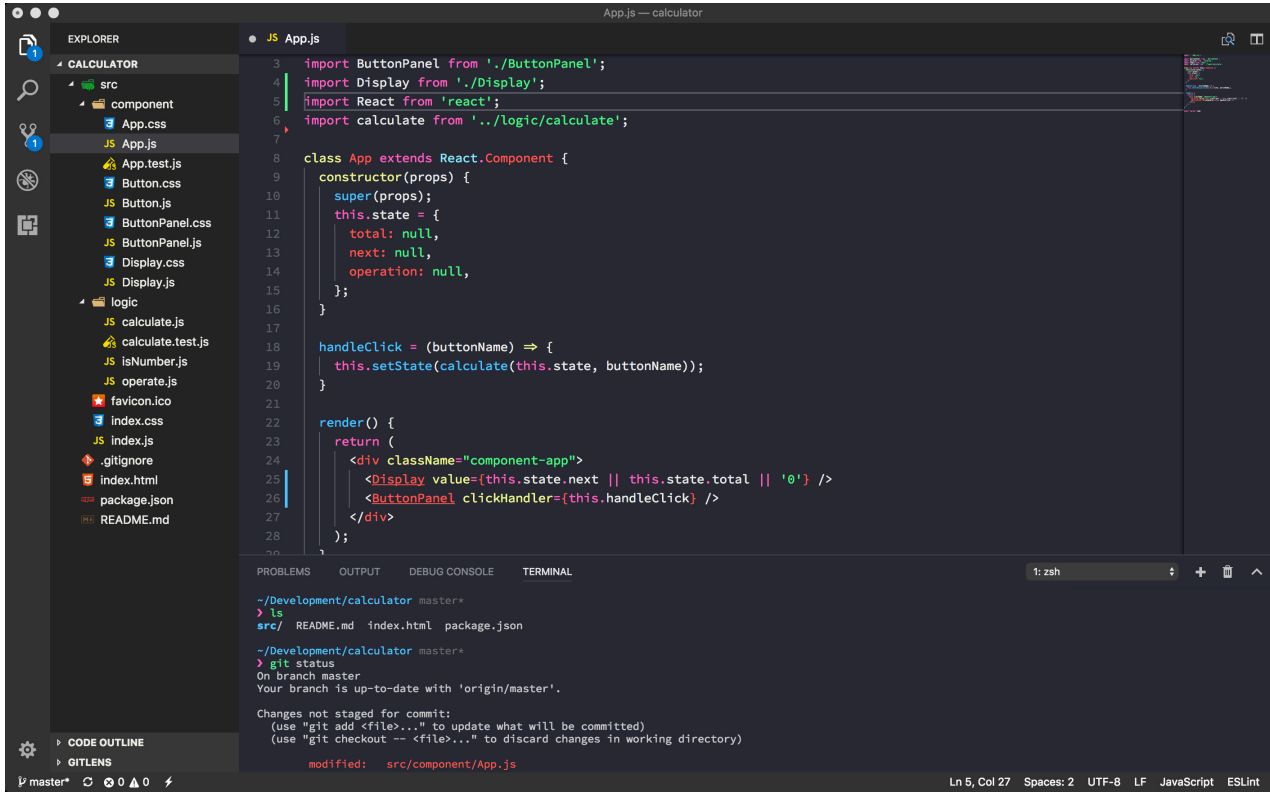
https://www.youtube.com/playlist?list=PL6gx4Cwl9DGDV6SnbINlVUd0o2xT4JbMu

9 Videos that show the basics of how the command line in Windows works. It's about an hour and worth it!



Integrated Development Environments

- An IDE is a tool that is used for writing code. It offers lots of extensions that we will show throughout this class to make it easier to develop webpages.
- In this class we will use a variety of online tools and most in-class demos will be done with VSCode (https://code.visualstudio.com)
- Quick Demo



What is Front-End development?

- Building web applications that perform logic on a user's device directly (computer, phone, etc.)
- Communicate with back-end systems that handle secure data and do complex analysis
- Focuses on design and user-interaction, not on algorithmic logic and databases.
- Works heavily with APIs and third-party services
- Requires hyper-focus on user experience and less on behind-the-scenes databases and structure.
- The "heroes" that make back-end developers success show.

What is Back-End development?

- Working with databases and storing data
- Big focus on algorithms, how to make code fast and efficient
- Must understand how to abstract problems and build for use cases that may not be clearly seen at the beginning of a project
- Builds APIs for front-end developers to interact with
- Requires hyper-focus on security, structure and solving problems for current and future use-cases
- The “heroes” that make front-end developers jobs easier.

So Full Stack Is?

- The ability to do switch between front and back-end development easily.
- The ability to “walk a mile in their shoes” and understand what the other developer needs to be successful.
- A ”jack-of-all-trades” that can easily be team lead for both types of developers or can be a one person army for a small company.
- **So why are we specializing in the second course?**

Web Development History

- https://www.webdesignmuseum.org/web-design-history

What to Expect From This Class

- What we're going to build?
  - Article page
  - Calculator
  - Small 5-page website
  - Basic database driven website
  - A small API to handle requests from a front-end system
  - Interactive web application
- What we're going to learn?
  - HTML
  - CSS
  - JavaScript
  - PHP
  - SEO / SMO
  - Working with and Building APIs
  - Web Hosting
  - Security
  - MySQL Databases
  - Git / Debugging Tools

Our Expectations

- Be present
- Stay up to date, if you miss a class, watch it before the next class
- Participate and complete all assignments and all tasks outside of class.
- Ask Questions! There are no bad questions only bad answers.
- Use lab time wisely and if you finish a task early use that time to expand on a project or try to figure out something new.
- Instructors have full-time jobs so don’t wait until the last minute to ask questions, do it early enough to give us time to respond.