WELCOME TO INFO6250

Web Development Tools and Methods

- The Hows and Whys of the Web
- Building a Multiple Page Web Application
  - NodeJS backend
  - HTML
  - CSS
  - JS frontend
- Building REST-based services (NodeJS backend)
- Building a Single Page Application (using React)
- Basic Best Practices for the Web

WHY THE INSTRUCTOR IS CREDIBLE

Brett Ritter `<b.ritter@neu.edu>`

- first job as WebDev in 1995
- webdev ever since
- multiple languages, frameworks, platforms
- both frontend and backend
- full-time job is in webdev, teaching is a side-gig
  - tell me where to improve my teaching

NOT PERFECT

I have a truly terrible memory.

Terrible

You have my permission to remind me, and keep reminding me, until something is done or I explicitly say "stop".

I am partially tone-deaf.

If I ask you to repeat yourself, my poor hearing is why.

FUNNY

I tell jokes.

Fortunately, they are all hilarious and you will laugh.

Out loud.

Try it now.

GET BETTER

We will keep practicing on that

EXPECTATIONS

Your Job is to Learn

My Job is to Teach

EXPECTATIONS - REQUIREMENTS

Your Job is to Learn

- Read materials
- Follow Instructions

My Job is to Teach

- Make material clear

EXPECTATIONS - REAL REQUIREMENTS

Your Job is to Learn

- Read materials
    - For realsies, then ask questions
- Follow Instructions
    - For realsies, then ask questions

My Job is to Teach

- Make material clear
    - I have to know when it is not clear
        - You need to ask questions

SLACK

We communicate via Slack

- **http://rebrand.ly/seainfo6250-slack**
- You can email me
    - but it will be slower
    - and it is terrible to talk about code
- Use it web, desktop, or phone
    - desktop/phone recommended for notifications
- This is a useful job skill

USING SLACK

Respond to the question

CONFIGURING SLACK FOR CODE

Update your Slack Preferences:

MENTIONING CODE IN SLACK
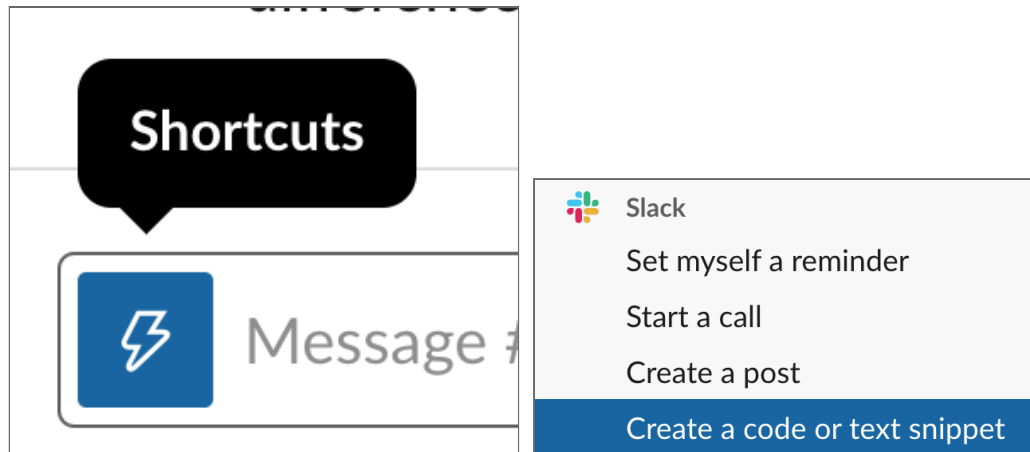
Send a message to **#questions** that says:

(notice the *backtick* characters)

`this has *bold*` but `` `this does *not*` ``

CODE BLOCKS IN SLACK

triple backticks ``` before and after your message

Or use "Create a code or text snippet"

CLASS GITHUB

We use git and github.com

Each of you get a personal repository

- **https://rebrand.ly/seainfo6250-github**

You must have/get a github.com account

GIT VS GITHUB

`git` is the source control/version control system.

- tracks files
- changes to files
- many devs can have repos
- can pass files/changes between many repos

`github.com` provides a central place for repos. It has competitors (example: gitlab.com)

github uses git, git does not require github or a github competitor, though we will use github.

GITHUB FLOW

git is *decentralized.* github provides centralization(ish)

See `readings/basic-git.md` in your repo

- You make changes in a "feature branch" locally.
- You send that branch to github
- You create a Pull Request (PR) to merge your branch into main **on github**
- I/TA review and approve your request
    - We might request changes first
- I/TA merge your branch into main
    - On the job you will probably do this
- You update your local main

OTHER GIT-BASED FLOWS

Other flows of changes and branches exist

- this one is the one we will use

Not all version control systems (VCS) are decentralized the way git is.

Github acts as a central point for communications

LOCAL AND REMOTES

"Local" means your computer (or anyone's)

When I give notes or assignments:

- I pull latest from github to my local copy
- I update my local copy
- I push the changes the github

You submit the same way:

- You pull changes from github to your local copy
- You make changes to your local copy
- You push the changes to github and create PR

KEY GIT NOTES

- Always do work in the correct branch
- Always check `git status` before `git commit`
- Always check `git status` before `git push`
- When creating a PR, always check the file list
- Before creating a new feature branch
  - Always switch to `main` and pull latest

If you follow these instructions

- each assignment is distinct and will not conflict.

DO AND DO NOT

- DO ask questions
- DO not worry about bothering me
  - I will tell you BEFORE that happens
- DO NOT worry about looking uninformed
  - You are literally students
- DO NOT expect to catch up by working harder
  - Good attitude, but...
  - Time, not you, is the problem
  - Easier to fix earlier rather than later
  - Your employer will follow the same rules

A UNIQUE WARNING ABOUT THE WEB

You WILL be expected to learn a lot of detail online

BUT a **lot** of info about web tech is outdated

- Don't use any sources older than 3 years ago

Really. **3 years max**. Or extra work and wrong work.

COMMON QUESTIONS

- Do I need to know HTML/CSS?
- Do I need to know programming?
- Do I need to know JS?
- Can I use another language?
- Can I use this outside library?
- Can I use this other framework?
- Can I use this other IDE?
- Can the instructor send class materials out?

CLASS PREREQUISITES

- Do I need to know HTML/CSS?
  - You are expected to know the basics
  - The 6150 class is great and recommended, but not required
  - You CAN learn alongside class (but harder)
  - **https://developer.mozilla.org/en-US/docs/Learn**

CLASS PREREQUISITES (CONT)

- Do I need to know programming?

    - Basics (variables, conditionals, looping, functions) are expected
    - Be ready to learn how things are different

- Do I need to know JS (Javascript)?

    - No expectation

USING OTHER TOOLS

- Can I use another language?

    - No, we use only JS for my sanity

- Can I use Typescript?

    - No, we are rely on JS fundamentals

USING OTHER TOOLS (CONT)

- Can I use this outside library?
    - (example: Bootstrap)
    - No, we need to exercise the fundamentals
    - Exceptions will be explicitly given per assignment
- Can I use this other framework?
    - (example: Angular/Vue)
    - No, we use React for core principles
        - and my sanity
- Can I use this other IDE?
    - Yes, but I don't know that IDE

CLASS MATERIALS

Can the instructor send class materials out?

Slides and code samples will be added to your repositories after class.

- `git checkout main`
- `git pull origin main`

If you feel something is missing, ask for it via Slack.

BUT - anything that comes in the moment may not be saved to send out. Missing class is risky.

WHAT TO EXPECT

Section 1:

- Lots of fundamental concepts
- Rushing HTML/CSS/JS

Section 2:

- Network and services
- Asynchronous code

Section 3:

- Modern React
- Architectural concepts
- Job skills

SUMMARY - YOU

- will **remind me** if you need something
- can and will **use and check** our Slack
- have your own **github repo**
- have a **local copy** of the repo
- have installed and done **config** changes
- know how to **get** and **submit** work
- have **submitted a PR** that provides your info
- will **not use old** online information
- will **read** material
- will **ask questions**