

INFO 1300: Introductory Design and Programming for the Web

Welcome!

Instructor:

Dr. Kyle Harms (he/him)

Course Website:

<https://infosci.cornell.edu/courses/info1300/2026fa/>

Copyright 2026, Kyle J. Harms

Agenda

1. About this course.
2. Course reflection + motivations discussion.
3. Course structure and policies.
4. Static websites

Homework: class preparation exercise due **before** *next* class.

Welcome!

Diversity, inclusion, and belonging are all core values of this course.

Because a sense of belonging can also affect students' mental health and wellness, I want you to know that **all** participants in this course *belong here and deserve to be treated with respect.*

I am here to help you learn.

Don't be embarrassed to ask for help – it's my job.

Advice: Handwritten Notes

- Learning and memory retention are enhanced by handwritten notes compared to typing notes (Ihara et al., 2021).
- Distraction caused by off-task laptop use reduces learning for both the user and those sitting near them (Hall et al., 2020; Sana et al., 2013).
- Lower comprehension and poorer academic performance among students who multitask (or switch-task) with digital technologies (Glass & Kang, 2019; Kraushaar & Novak, 2010; Walker, 2017, 2017; Wood et al., 2012).

Activity: Meet Your Peers!

Turn to your left and right and **introduce yourself** to your neighbors.

Share **one** thing that you enjoy about the Finger Lakes region of Upstate New York **or** one thing that you want to do in the Finger Lakes before your graduate.

Share your contact information (for when you miss class and need to get caught up on what you missed, or want to get together for a study session while working on your homework, etc.)

Why? Students that have a supportive peer network are more likely to succeed in a course. Create your network today!

INFO 1300:

Introductory Design *and*
Programming for the Web

Activity: Prior Experience Reflection

Take a moment on reflect on the following questions:

- What do you know about static front-end web **design**?
- What do you know about static front-end web **development**?

Complete **Part I** on the handout.

Primary Course Objective

Understand the foundational principles of designing and implementing usable static, accessible front-end websites.

Web Design

- **User-centered design**
 - Information organization
 - Visual design principles
 - Interaction design
 - Usability & user testing
- **Accessibility**

Web Programming

- **Static *front-end* development**
 - Structuring content (HTML)
 - Styling content (CSS)
 - Client-side interactivity (JS)
 - Using reference docs + generative AI
- **Accessibility**

Prerequisites: *None*

This is an **introductory** level course. I assume the following...

- You have no prior **design** experience.
- You have no prior **programming** experience.

Prior programming experience?

Warning! This class is probably not a good fit for you.

Course Objectives

- Develop accessible client-side/front-end static websites using HTML, CSS, and JavaScript.
- Structure content with HTML, style it with CSS, and add client-side interactivity with JavaScript.
- Plan and implement complex layouts using CSS Flexbox and CSS Grid.
- Apply user-centered design methods to create websites for specific audiences.
- Design and implement usable, responsive websites for narrow and wide screen devices.

Course Objectives

- Organize content and navigation using information architecture principles.
- Design usable websites leveraging the visual design principles of color, contrast, typography, proximity, alignment, hierarchy, repetition and consistency.
- Orally communicate and defend design and coding decisions to project stakeholders.
- Gain experience with developer best practices: utilizing reference documentation, version control using Git, and authoring documentation using Markdown.

Course Objectives

- Troubleshoot programming problems independently.
- Utilize generative AI tools effectively as a coding partner through prompt engineering and critical evaluation to assist in development.
- Demonstrate a high standard of professionalism.

Activity: Motivation Reflection

Take a moment on reflect on the following questions:

- What do you want to get out of this course?
- What are your expectations for this course?

Complete **Part II** on the handout.

Activity: Discussion

- What do you hope to get out of this course?
- What are your expectations for this course?
- Why learn any of this stuff when AI can do it for you?
- What skills/experience do employers expect from you?

When I call on you, please **say your name** first, then answer.

Course Structure

This is a project-oriented class with exams.

- 2 exams during class + 1 cumulative final exam

Before every class:

- class preparation exercise

An assignment due every Monday:

- 8 homework assignments
- a project with 4 milestones

Every Friday during sections:

- 6 practice quizzes

Expected Workload: ~8 hours outside class a week for a satisfactory grade (C).

Resources

Class – (Primary Resource) In-class activities and lectures.

Your Notes – (Primary Resource) Your notes are your study guide.

Readings – Online resources and documentation.

Office Hours – The instructor and teaching/course assistants are available to help you. (Limited office hours available. More available upon request.)

GitHub Copilot – *Reference AI tool* for coding activities/assignments.
(Not to be used a substitute for the above resources.)

(Timely) Attendance Required

- **As per university policy, lecture attendance is required.**
- Complete the in-class activity handout and physically submit it **before** leaving the classroom to be counted as **present** for that class.
- If you arrive after the start of lecture (class starts exactly at 1:25:00pm), but within 10 minutes of the start time (1:35:00pm), you will be counted as **half** present for that class.
- If you arrive more than 10 minutes after the start of lecture (after 1:35:00pm), you will be counted as **absent** for that class.

Required Materials

Required Materials: Pen or pencil, paper, a functioning laptop, an active github.com account, and a [GitHub Copilot Pro](#) subscription (\$10/month + additional credits).

Textbooks: *None*

This course requires GitHub Copilot Pro, with the costs paid directly by the student, estimated at ~\$40/semester. (Free GitHub Copilot Pro accounts are available through [GitHub Education](#) for verified students.) Note that AI pricing is difficult to accurately estimate and can change with little notice. Contact info1300@cornell.edu if this cost presents a hardship for you.

Syllabus

The syllabus documents everything you need to know about this course. **Refer to it first, if you're not sure what to do.**

By being here, you have earned the right to be held to a high standard of professionalism.

Grades

Your grades are posted to Canvas. The course grade in Canvas is your *current* grade.

Component	Weight
<i>Attendance</i>	0% [†]
<i>Class Prep + Homeworks</i>	10%
<i>Practice Quizzes</i>	10% [‡]
<i>Project + Oral Defense</i>	20%
<i>Exam 1</i>	20%
<i>Exam 2</i>	20%
<i>Final Exam</i>	20%

† If you arrive to class before it starts and attend at least 86% of all lectures, your final course percentage will be rounded up to the next whole number; you may miss 4 lectures without penalty.

‡ Lowest practice quiz score is dropped when calculating your final grade.

This course does not offer *extra credit*.

Grading

- Grades are **earned**, not given.
- Your grade is a reflection of your mastery of the course content and your ability to apply that knowledge to the assignments.
- Your grade is not a reflection of your effort, your potential, or your worth as a person.

Course Policies

- Email info1300@cornell.edu to privately contact the instructor. (No undergrad CAs access.)
- I strive to respond to each email within **2 business days** (Monday-Friday, excluding holidays and breaks), during *business hours* (Monday-Friday, 9am-4pm).
- You are responsible for all information presented in lecture, including information not available elsewhere, such as in the slides, course website, Canvas, etc.
- Late work receives 0 credit without an accommodation.
- Accommodations must be arranged in advance.

Course Policies

- Email the course for accommodations. The sooner, the better (per university policy, accommodations are not applied retroactively.)
- *Only* the instructor may clarify assignment instructions/requirements or course content.
- All assignments are graded once.
- TA/CAs cannot “pre-grade” your work or tell you if your work is correct; TA/CAs do not have the authority to endorse solutions, provide clarifications, or regrade your assignment.
- You are encouraged to collaborate/work with your peers in this class so long as you do your own work.

Flexibility

This course is structured to support the diverse needs of all students.

This course provides built-in accommodations:

- 4 automatically excused absence accommodations
- 3, one-time use slip day accommodations
- 1, one-time resubmission accommodation

These accommodations are specifically intended to support *most* student accommodation needs (e.g. disability, religious observance, Title IX, athletic participation, military service, medical and mental health concerns, family emergencies, late enrollment, etc.)

Accommodations

Use the existing course structure and built-in accommodations.

If you believe that your individual needs and circumstances require additional consideration, email info1300@cornell.edu and let us know how the built-in structure and accommodations are not providing the **access** you require.

Note that all accommodations must be arranged well in advance – as per university policy, we are unable to retroactively apply accommodations to late assignments, etc.

Privacy

We respect your privacy.

Please keep your personal information private. You need not provide any personal information when requesting accommodations, etc.

In the interest of respecting the privacy of all students, **you may not record lecture or take photographs in the classroom** without the *explicit* permission of the instructor.

Academic Integrity

- Unless otherwise stated by the assignment, all work should be your own.
- If found guilty of violating the [Code of Academic Integrity](#), the minimum penalty is a 0 for the offending assignment. The maximum penalty is failing the course.

Generative AI

GenAI use is permitted, but you may not upload copyrighted or proprietary class materials (slides, readings, recordings, lecture notes, etc.) unless otherwise specified.



PP

ChatGPT vs. GitHub Copilot

ChatGPT is a general-purpose conversational AI.

GitHub Copilot is a coding-specific AI.

Discussion:

- Do you need to know industry/domain-specific AI tools?
- Can you get by with copy-pasting from ChatGPT or Claude?

Advise: Recent Graduate

“ You’re getting hired because of the value you are adding. ”

– Celeste Naughton (Information Science '25) on what students should know about genAI and industry.

Wait-List

If you are on the wait-list or trying to enroll in this course, you should:

- act like you are enrolled
- attend class
- submit the homeworks (on time)

You have access to the *entire* course infrastructure now, including the course website, Canvas, and GitHub repository.

Accommodations are not provided for *not being enrolled in the course*.

How the Web Works:

What is **HTML**?

“Content is King”

- Bill Gates, 1996

Web Design **and** Web Programming

Where do you start?

The Content!

How to Create a Website

1. **Design** the website.
2. Code the website.

1. Designing a Website

1. Identify the website's audience by their goals. (Not your goals.)
2. Assemble/create **content** to address the audience's goals.
3. Organize **content** into web pages. (Websites have *multiple* pages.)
4. Plan the website's design.
5. Evaluate the site's design. Does the website address the audience's goals?

2. Coding a Website

For *each* webpage on the website:

1. Structure the **content** in HTML.
2. Style the **content** with CSS using your design plan.
3. Add interactivity with JavaScript to enhance the user's experience with the **content**.

How do websites work?

HTML Structures Content

HTML: Hyper**T**ext **M**arkup **L**anguage

Structure **content** with `<tags>`:

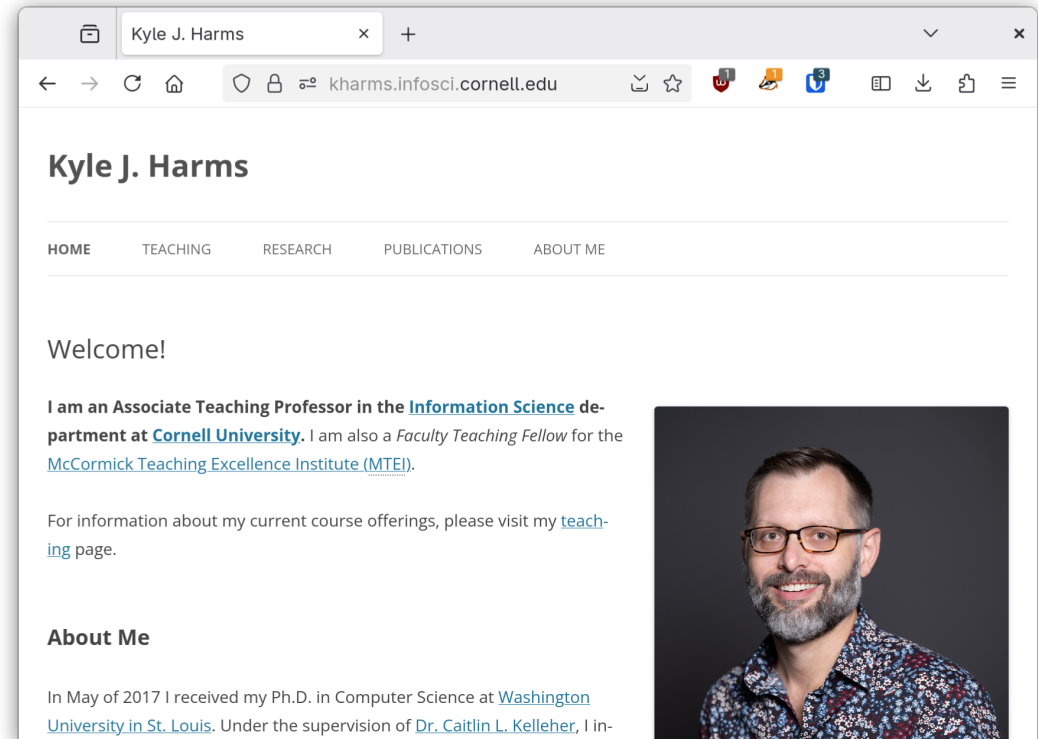
```
<tag>  
  content  
</tag>
```

Example: Content Structured in HTML

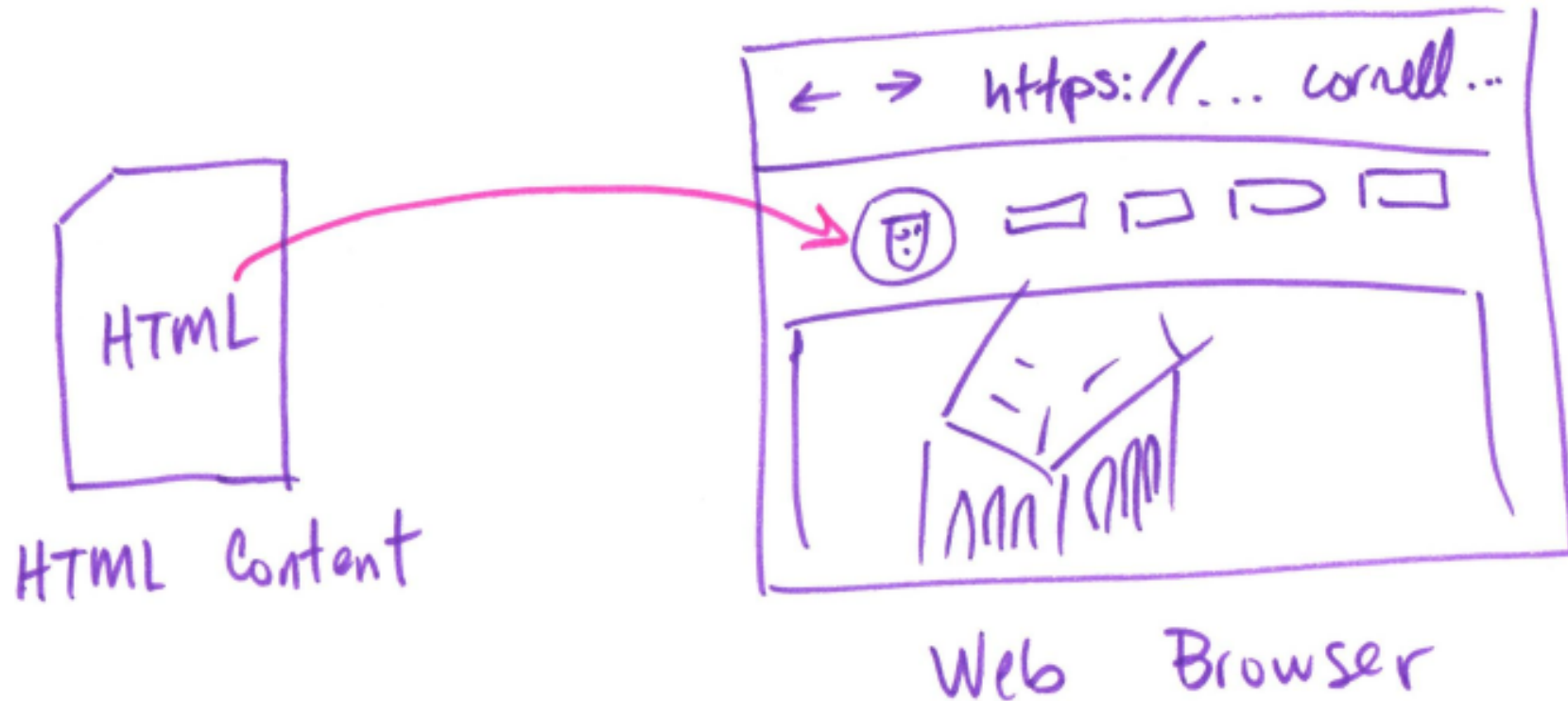
```
<h2>
  Welcome!
</h2>

<p>
  I am an Associate Teaching Professor...
</p>

<p>
  For information about...
</p>
```



Web Browsers Render HTML



How to Build Websites

A Website

“ A website is a collection of web pages and related content. ”

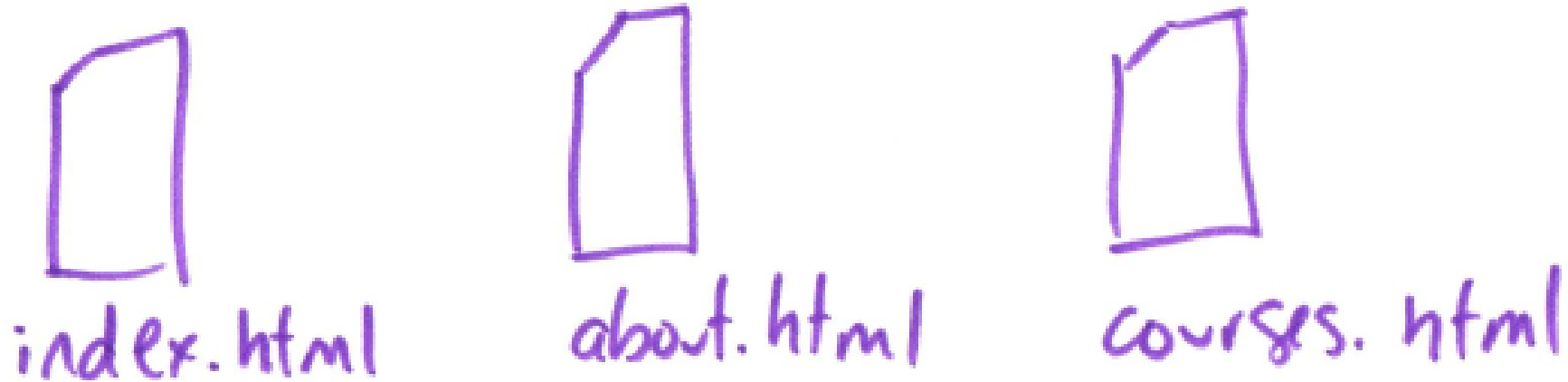
Static Websites (INFO 1300)

“ A website with fixed content; the website is the same for all users. ”

Dynamic Websites (INFO 2310)

“ A website with content that changes depending on several factors (user, time of day, etc.) ”

Static Website = Collection of Web Pages



Web page = HTML File

What's Next

- Due before next class:
 - Class 2 Preparation

See the course website for details:

<https://infosci.cornell.edu/courses/info1300/2026fa/>