

SWE 4743: Object Oriented Design

Jeff Adkisson

Monday, Wednesday at 6:30 PM
Architecture, 175

Course Materials

D2L

- Presentations
- Grades
- Calendar
- Announcements
- Syllabus

GitHub

- Demos
- Semester Project Details
<https://github.com/jeff-adkisson/swe-4743-spring-2026-oo-design.git>

Contacting Me

- Use my KSU email address.
- Do not use D2L email. It is awful.
- Write from your KSU email address. I am not allowed to respond to your personal email address.

Write me at:

JAdkiss1@Kennesaw.edu

I usually respond within 24-48 hours. If I do not respond, it is generally because you wrote to tell me you missed lecture and asked “what did we do?”

Do not use D2L email.

Office Hours

- I do not have an office.
- I stay late after almost every lecture.
- Occasionally I cannot stay late.

If you need to talk to me, stay after lecture. I almost always stay late to help students.

If you cannot stay after, contact me by email in advance and we will work something out.

Grading

- Academic Work – 50%
- Practical Work – 50%

Category	Points/Weight
	<i>400 total points</i>
Academic Work – 50%	50% total
• Exam 1 / Written, In Class	25%
• Exam 2 / Written, In Class	25%
Assignments – 0%	
- Assignments will be given every week or <u>so</u> , but will not be graded. I will do a review of your assignment work on request.	0% <i>The work is its own reward.</i>
Practical Work (Team) – 50%	50% total
• Design	12.5% / 25% of practical work
• Implementation	37.5% / 75% of practical work

Rounding Policy

- I follow this policy to ensure rounding is fairly and evenly applied to all students without exception.

I calculate grades to 2 decimal places. Rounding will occur at the 3rd decimal position.

For example:

- 89.984 becomes 89.98
- 89.985 becomes 89.99
- 89.994 becomes 89.99
- 89.995 becomes 90.0

*TLDR: You need a **#9.995** or higher to receive the next highest grade*

Attendance

- Attendance is highly recommended.
- There are two in-person written exams. You must attend those.
- **Aside from the two exam dates, I will not take attendance.**

You are expected to attend each class session. This is an in-person course. The material will not be delivered online, and classes will not be recorded or broadcast. I will only post video lectures in the event of inclement weather, *or* I must cancel a lecture due to travel for my primary occupation.

You are responsible for knowing about announcements/assignments made in lecture, whether you are present.

There is no extra credit for attendance though you are far more likely to be successful if you regularly attend.

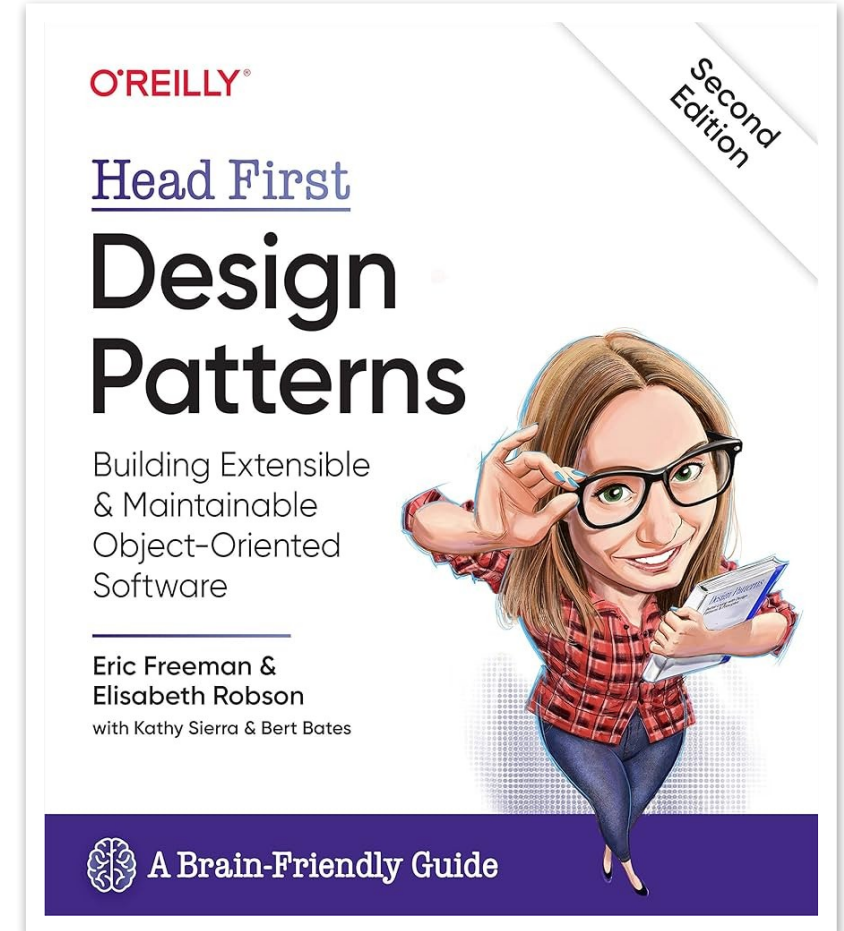
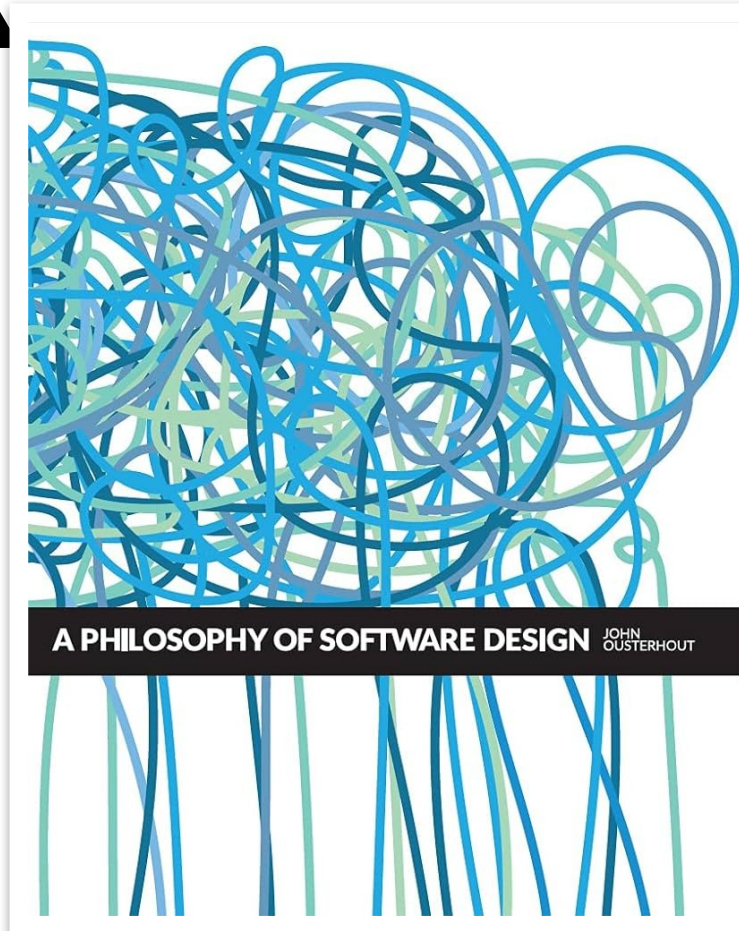
I post announcements in D2L and discuss them at the start of each lecture. If you miss a lecture, check D2L or ask your classmates.

Please do not email me asking me what you missed.

The ONLY two days I require you to attend are for Exams 1 and 2. Check the syllabus for the dates. If you cannot attend on those days, you must have a written doctor's note excusing your absence. Exam dates are

Textbook S

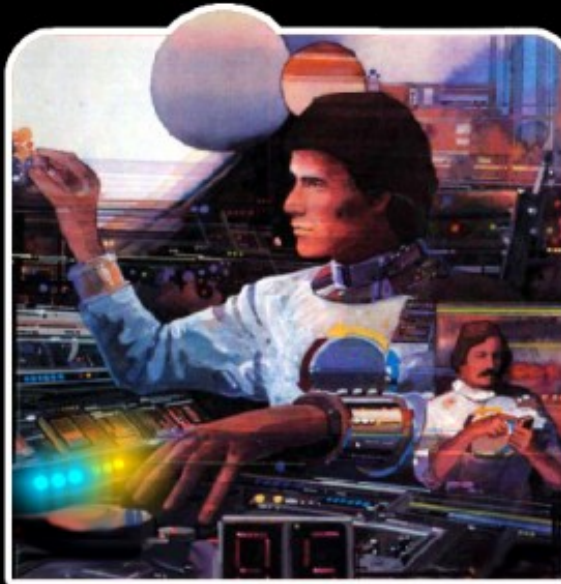
- Readings are assigned.
- Readings are *actually* important.
- Readings will be part of the exam material.



Coding

- We will read and write a lot of code in this course.
- Do the exercises.
- Do the assignments.
- Do the readings.
- Avoid using AI to do your work.
- You cannot master these concepts without writing code.

THE TWO STATES OF EVERY PROGRAMMER



I AM A GOD.



**I HAVE NO IDEA
WHAT I'M DOING.**

©georgepelt34r

Computer Science Prerequisites

This course is a senior level programming class, so you are expected to be already familiar with the following. If not, please review these concepts immediately.

- Basic OO concepts such as polymorphism, encapsulation, and inheritance.
- Public, private, and protected scope
- Classes, methods (instance and static), fields (instance and static), and properties
- Primitive, abstract, and concrete types
- Conditional logic – if/then, switch, etc.
- Loops and recursion – for, do/while/until, etc.
- Data structures - arrays, linked lists, queues, stacks, trees, sorting algorithms, etc.
- Displaying and entering data via a console application
- Using an IDE to develop and debug a project with multiple classes
- Computational complexity (Big O).

Exam Dates

- As of January 11, 2026
- Exam dates may change (but it is unlikely)

You must be present in-person for the two written exams.

FEB

6:30 PM

16

Exam 1

MAR

6:30 PM

30

Exam 2

Exam Makeups 😞

- You have to be here two times this semester.
- Please be here on the exam dates.
- Missing an Exam must have very official documentation – not just a photo of your car with the hood up.

Occasionally a student must miss an exam due to illness, death of a family member, etc. You must communicate with me *in advance* to schedule an exam makeup. To receive a makeup exam, you must use the KSU exam proctoring service.

Do not attempt to request a makeup exam by sending me a picture of your car with the hood up, claiming it broke down minutes before the exam. If your car does break down, you must have actual proof from a police officer. Every semester someone performs some version of this. Please do not be that person. On days we have an exam, show up to campus early and be ready to take the exam. This will save both of us a lot of effort and frustration. The exam dates are in the

Semester Project

- Teams of 1 or 2 students
- You can work alone (always my preference)
- You can work with a partner.

You will execute a large design and implementation project.

- C# or Java API, Angular UI, Sqlite data storage, Docker (execution environment)

If you work alone, you are responsible for the entire project.

If you work with a partner, you both receive the same grade. If your partner drops the course, you become a team of one. You can split with your teammate if the relationship is not satisfactory. If that happens, you both are responsible for the entire project.

I do not reduce the requirements or schedule for teams of one.

Late Project Submissions



- There are significant penalties for late project submissions.
- Please do not submit work late. It is unpleasant for everyone.
- Do not wait until the last minute to do your project work. You will greatly improve your chances of failing this course.

Late Project Assignments

No work later than 72 hours will be accepted. Late work submitted within 72 hours of the deadline will be accepted *with penalties*.

- Submit after the due time but within (including) 24 hours - 10% penalty
- Submit after the due time but within (including) 48 hours - 25% penalty
- Submit after the due time but within (including) 72 hours - 50% penalty
- Submit hours after the due time - 100% penalty (zero credit).

Distraction



- Do not distract your classmates.
- Do not distract me.
- If you are distracting your classmates by watching videos, playing games, scrolling Instagram, etc., please leave. Otherwise, I will ask you to leave.

If you want to waste time, do it elsewhere.

If you are watching videos, scrolling Instagram, playing games, etc., do it elsewhere.

I am unlikely to notice, but the people behind you will. And many of them will be distracted. If you are distracting your classmates, I will ask you to leave.

If you have a classmate who is distracting you and you are uncomfortable telling them to stop, tell me after class and I will speak to the student.

You paid to be here. Make the most of your time.

Other students paid to be here. Do not waste their time.

Sleeping



- This annoys me.
- Do not sleep in my classroom.
- Do not put your head down in my classroom.

Do not sleep in my classroom or put your head down on the desk. Just go somewhere else.

I do not object to you leaving or not being present.

But if you are present, I expect you to be aware and engaged.

Arriving Late

- Do not interrupt the lecture.

If you arrive late, respect your classmates and me by being quiet and getting seated without making a lot of racket.

If you routinely arrive late, work on your priorities and stop doing that.

You paid to be here. You paid me to be here.

So be here and be here on time.

AI Usage

- You can use AI.
- I advise against it, other than to perhaps verify your work.
- You are here to learn - not write prompts.
- Grades mean nothing in the real world, so earning a grade via expert AI usage likely will not advance your career much.

AI Use Allowed, but Not Required

In this class, you are welcome to use AI for any purpose outside of written (in-person) exams.

However, you should note that all AI generative tools still tend to make up incorrect facts and fake citations, code generation models tend to produce inaccurate outputs, and image/art generation tools can produce copied work or offensive products.

You will be responsible for any inaccurate, biased, offensive, or otherwise unethical content you submit regardless of whether it originally comes from you or an AI tool.

If you use an AI tool, its contribution must be credited in your submission. **The use of an AI tool without acknowledgement is cheating and constitutes a violation of the KSU Code of Academic Integrity**

D2L

The screenshot shows a D2L course page with the following elements and callouts:

- Navigation Bar:** Course Home, Content, Syllabus, GitHub, Grades, Communication, Student Resources, Course Admin.
- Course Title:** Object-Oriented Development Section 01 Spring Semester 2026
- Content Browser:** Includes Bookmarks and Recently Visited tabs. A message states: "There is no content to display. [Create some content](#)".
- Announcements:** A post titled "Welcome to OO Design!" dated Jan 11, 2026. The text reads: "I am excited about this course. I last taught it in 2018. I have pretty much rewritten the course, so everything we will do this semester is fresh and new. I am Jeff Adkisson, an adjunct in the college. You can call me Jeff or Professor. I have a Masters in CS, but no doctorate, so please do not address me as Doctor Adkisson. In my day job, I am CTO of a company here in Atlanta called HighMatch. We do high volume pre-hire assessment measuring arcane stuff like personality traits, attitudes, skill levels, etc. I have been with the company 15 years. I lead the engineering and..."
- Calendar:** Shows the current date as Sunday, January 11, 2026. Upcoming events include:
 - FEB 16 6:30 PM Exam 1
 - MAR 30 6:30 PM Exam 2
 - MAY 6 All Day Semester Project Due

Callout boxes with arrows pointing to specific features:

- Presentations posted here:** Points to the Content link in the navigation bar.
- Course policies, etc.:** Points to the Syllabus link in the navigation bar.
- Where to find my demos, project details, etc.:** Points to the GitHub link in the navigation bar.
- I post a LOT of announcements:** Points to the Announcements section.
- Stay on top of the calendar:** Points to the Calendar section.

Do not use D2L email! I hate it!

Schedule as of Jan 11

- The schedule may change.
- If it changes, I will announce the change on D2L.
- I typically do not update the syllabus when the schedule changes.

SWE 4743: Object Oriented Design Calendar

Kennesaw State University College of Computing and Software
Engineering Software Engineering

Week 1

Monday, January 12

Syllabus Review

Wednesday, January 14

Lecture 1 – OO Foundations Review


- Polymorphism, encapsulation, inheritance, interfaces
- Access modifiers and visibility boundaries
- Classes, fields, methods, properties
- Primitive vs abstract types and early warning signs of primitive obsession

GitHub

jeff-adkisson

Overview Repositories 15 Projects Packages Stars 22

Pinned

 [swe-4743-spring-2026-oo-design](#)

Public

Software Engineering 4743 Object Oriented Design, Spring 2026, KSU

☆ 1

1,927 contributions in the last year

	Jan	Feb	Mar	Apr	May	Jun	Jul
Mon							
Wed							
Fri							

Our course...

<https://github.com/jeff-adkisson/swe-4743-spring-2026-oo-design/blob/main/README.md>

GitHub

SWE 4743: Object Oriented Design

Kennesaw State University College of Computing and Software Engineering Software Engineering

- [Syllabus](#)
- [Calendar](#)
- [Terminology](#)
- Presentations
 - 01 - Course Introduction, Why Design Matters
 - [PowerPoint](#)
 - [PDF](#)

<https://github.com/jeff-adkisson/swe-4743-spring-2026-oo-design/blob/main/README.md>

Coding Assistance

- I enjoy helping you with your code.
- My code reviews are often done with a Loom video.
- I require your code to be in GitHub.

If you need help with an assignment or your project, always include a link to the code in your personal or team GitHub repository.

Do not email me code!

Do not zip code and post it to D2L!

Share code the way industry professionals share it – via a source control system.

This saves time (I can clone and go), enables me to review the code online or on my phone, etc.

SWE 4743: Object Oriented Design

Jeff Adkisson

Monday, Wednesday at 6:30 PM
Architecture, 175