

Object-Oriented Design Section 61

CS 151

Summer 2024 3 Unit(s) 06/03/2024 to 08/09/2024 Modified 06/04/2024

Contact Information

Instructor(s):	Dr. Chung-Wen (Albert) Tsao
Office Location:	MH411
Email:	chung-wen.tsao@sjsu.edu (Once the class starts, use Canvas Inbox)
Class Days/Time:	MW 3:00PM - 5PM
Classroom:	MacQuarrie Hall 422
Office Hours:	T/Th/F 10:30 – 11:30am at MH411 (or on ZOOM at https://sjsu.zoom.us/j/84753088946). (https://sjsu.zoom.us/j/84753088946).

Course Description and Requisites

Design of classes and interfaces. Object-oriented design methodologies and notations. Design patterns. Generics and reflection. Exception handling. Concurrent programming. Graphical user interface programming. Software engineering concepts and tools. Required team-based programming assignment.

Prerequisite(s): MATH 42, CS 46B, and [(CS 48 or CS 49J) if CS 46B was not in Java], each with a grade of "C-" or better; Allowed Declared Majors: Computer Science, Applied and Computational Math, Software Engineering, or Data Science; or instructor consent.

Letter Graded

* Classroom Protocols

- Instructor may drop students (by the Instructor Drop Deadline) who
 - are absent for 1st day of class without informing you before 2nd day of class, or
 - have no proof of the prerequisite fulfillments.
- Do not ask for special treatment. The rules for this course apply to everyone equally.
- Cheating will not be tolerable; a ZERO will be given to any cheated assignment/exams, and it will be reported to the Department and the University.
- Do NOT share/post online any course materials, PPT slides, or homework solutions.
- Use of electronic devices during exams is NOT allowed unless stated otherwise.
- You are required to check Canvas for reading/assignments.
- The information on this syllabus is subject to change; changes, if any, will be clearly explained in class, and it is your responsibility to become aware of them.
- Once the class starts, use Canvas Inbox to email me for a faster response. I check the Canvas Inbox emails much more often than my school emails.

≡ Program Information

Diversity Statement - At SJSU, it is important to create a safe learning environment where we can explore, learn, and grow together. We strive to build a diverse, equitable, inclusive culture that values, encourages, and supports students from all backgrounds and experiences.

▮ Course Learning Outcomes (CLOs)

Upon successful completion of this course, students would be able to:

- Object-Oriented Design
 - Follow a systematic object-oriented design methodology.
 - Develop use cases, perform noun/verb analysis, interpret, and produce CRC cards.
 - Interpret and produce UML diagrams.
 - Understand object-oriented concepts.
 - Use design patterns.
 - Practice SOLID design principles.
- Advanced Java Language
 - Implement Java fundamental concepts of OOP.
 - Implement Java constructs such as: Interfaces, Abstract classes, Nested classes, ...
 - Implement Java standard Object methods.
 - Implement Java type system, lambda expression, serialization, Java generics, ...
 - Implement exception handling.
 - Implement threads and thread-safe data structures.
- GUI Programming

- Use JavaFX to create graphical user interface (GUI) for desktop applications.

Course Materials

(Optional) Head First Object-Oriented Analysis and Design

- Author: Brett McLaughlin, Gary Pollice, David West
- Publisher: O'Reilly Media, Inc.
- ISBN: 0596008678
- Availability:
 - <https://learning.oreilly.com/library/view/head-first-object-oriented/0596008678/>

(Optional) Head First Design Patterns

- Author: Eric Freeman, Elisabeth Robson
- Publisher: O'Reilly Media, Inc.
- Edition: 2nd Edition
- ISBN: 9781492077992
- Availability:
 - <https://learning.oreilly.com/library/view/head-first-design/9781492077992/>

(Optional) "Object-Oriented Design & Patterns,"

- Author; Cay Horstmann,
- Edition: 3rd edition
- Availability:
 - <http://horstmann.com/oodp3/>

(Optional) "Object-Oriented Design in Java,"

- Authors: Stephen Gilbert and Bill McCarty,
- Publisher: Sams
- ISBN-13: 978-1571691347

Course Requirements and Assignments

Midterm Exams:

- Midterms will only be given during class time.
- Makeup midterm exams will only be given in cases of verifiable emergency.
- Midterm exam dates in this syllabus are approximate and are subject to change.

Final Exam:

- The final exam will be cumulative and will only be given during class time.
- Makeup exams are only given if there is a verifiable emergency or illness OR if a student has more than two final exams within a 24 hour period and notifies the instructor 2 weeks before the last class meeting.

✓ Grading Information

- Final grades will not be adjusted in any way - so an 89.99% is still a B+.
- No incomplete grades will be given.
- No late submission of assignments will be accepted.
- However, everyone has two passes in the last week of semester to waive the penalty for
- any two submissions that are each turned in within 24 hours after the due date, or
- any one submission of that are turned in within 48 hours after the due date.

Grade Percentage

- Assignments: 30%
- Quizzes: 15%
- Project 15%
- Midterm 20%
- Final 20%

Criteria

The grading scale is as follows:

Grading Scale					
A+	97%	A	93%	A-	90%
B+	87%	B	83%	B-	80%
C+	77%	C	73%	C-	70%
D+	67%	D	63%	D-	60%
F	below 60.0%				

Per [University Policy S16-9 \(PDF\)](http://www.sjsu.edu/senate/docs/S16-9.pdf), relevant university policy concerning all courses, such as student responsibilities, academic integrity, accommodations, dropping and adding, consent for recording of class, etc. and available student services (e.g. learning assistance, counseling, and other resources) are listed on the [Syllabus Information](https://www.sjsu.edu/curriculum/courses/syllabus-info.php) web page. Make sure to visit this page to review and be aware of these university policies and resources.

Course Schedule

Week	Topic	Notes
1	Introduction, Java Recap - Functions, Packages, Classes and Objects	
2	OOP Fundamentals Part	
3	UML Diagrams, Design Documentation	
4	Exception Handling, Testing	
5	Midterm/JavaFX GUI	
6	Design Patterns	
7	Software Lifecycle and Testing Part	
8	Generics, Reflection	
9	Advanced Java	
10	Concurrent Programming/Final Exam	