SJSU SAN JOSÉ STATE UNIVERSITY

College of Science · Computer Science

# Object-Oriented Design

## CS 151

Fall 2026 Section 03 In Person 3 Unit(s) 08/19/2026 to 12/07/2026 Modified 08/18/2026

## Contact Information

Instructor: Eric Chen
Email: eric.chen02@sjsu.edu
Lecture: MW 12:00PM - 1:15PM, MacQuarrie Hall 223
Office hour: Friday 1-2 PM (ZOOM Link (TBD), or in person by appointment. If you like to meet in person, please send me request one day before, or after class.

## 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.
**Grading:** Letter Graded

## Classroom Protocols

- This course is an in-person class. Do your best to be on time. If you miss a class, contact your classmates and review the lecture slides to not fall behind since most weeks build on previous weeks.
- Attendance is encouraged to succeed in this course. While slides will be published for study, they are NOT meant to be a complete resource for succeeding on exams. There will be topics and problems on the exams not directly referenced in the slides. Anything verbally discussed during class, including topics or details not included on the slides, will be tested on exams. If you miss a lecture, you can always reach out to your teaching staff to learn what other topics you might have missed.
- Students are prohibited from recording class activities including lectures, office hours, advising sessions, etc. Materials created by the instructor for this course are copyrighted by the instructor. This university policy (S12-7) is in place to protect the privacy of students in the course, as well as to maintain academic integrity through reducing instances of cheating. Students who record, distribute, or

post these materials will be referred to the Student Conduct and Ethical Development office.Unauthorized recording may violate university and state law. It is the responsibility of students that require special accommodations or assistive technology due to a disability to notify the instructor.
- Information in the syllabus is subject to change. If this happens, it will be announced both in class and on Canvas.

# 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 Goals

Upon successful completion of this course, students will 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.

### 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 generics
Implement exception handling.
Implement threads and thread-safe data structures.
Java GUI Programming for desktop applications.
Java Reflections
Java Performance Analysis, garbage collections.

# Course Materials

This course does not have any required textbook. Exams are designed based on the contents of in-person lecture.

Supplemental Reading:Cay Horstmann, "Object-Oriented Design & Patterns," 3rd edition
https://horstmann.com/oodp3

# Course Requirements and Assignments

### Requirement

This class requires students complete their assignments in Java. Even though Java is not a prerequisite, but we encourage students get familiar with Java basics and have a comfortable programming environment that can develop and debug java program locally.

### Workload

- Quizzes and assignments
- 1 midterm and 1 final exam
- 1 term project (recommended done as a group project with 2 - 3 students, but no more than 3 students)

# Grading Information

### Grade

| From | Grade |
|---|---|
| 97 | A+ |
| 93 | A |
| 90 | A- |
| 87 | B+ |
| 83 | B |
| 80 | B- |
| 77 | C+ |
| 73 | C |
| 70 | C- |
| 67 | D+ |
| 63 | D |
| 60 | D- |
| 0 | F |

Break down

| | |
|---|---|
| Assignment | 30% (5% each) |
| Quizzes | 20% (2% each) |
| Project | 15% |
| Midterm | 15% |
| Final | 20 % |
| **Total** | **100 %** |

# University Policies

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

Tentative course schedule:

| | | | | | |
|---|---|---|---|---|---|
| Date | Lec | week | Topic | Quiz | Assignment / Project |
| 8/19 | 0 | 0 | Course Introduction | | |
| 8/24 | 1 | 1 | Java OOP Basics | 0 | |
| 8/26 | 2 | 1 | Java OOP Basics | | A1 out |
| 8/31 | 3 | 2 | Java OOP Basics | 1 | |
| 9/2 | 4 | 2 | Java OOP Basics | | A1 due, A2 out |
| 9/7 | | 3 | No class | | |
| 9/9 | 5 | 3 | OOP Basics UML | 2 | |

| | | | | | |
|---|---|---|---|---|---|
| 9/14 | 6 | 4 | OO Design Patterns | | |
| 9/16 | 7 | 4 | OO Design Patterns | | A2 due, A3 out |
| 9/21 | 8 | 5 | OO Design Patterns | 3 | |
| 9/23 | 9 | 5 | OO Design Patterns | | A3 Due A4 out |
| 9/28 | 10 | 6 | OO Design Patterns | 4 | |
| 9/30 | 11 | 6 | OO Design Patterns | | |
| 10/5 | 12 | 7 | Project Start Discussion | 5 | A4 Due |
| 10/7 | 13 | 7 | Project Start Discussion | | |
| 10/12 | | 8 | Study Guide, Q & A | | |
| 10/14 | | 8 | Mid Term | | |
| 10/19 | 14 | 9 | Advanced Java Topics (Exceptions) | | |
| 10/21 | 15 | 9 | Advanced Java Topics (Generics) | | A5 Out |
| 10/26 | 16 | 10 | Advanced Design Patterns (Comparator / Sorting) | 6 | |
| 10/28 | 17 | 10 | Advanced Design Patterns (Iterator / Java built-in data structures) | | |
| 11/2 | | 11 | Project Checkpoint | 7 | |
| 11/4 | 18 | 11 | Advanced Java Topics (Multithreading) | | A5 Due, A6 out |
| 11/9 | 19 | 12 | Advanced Java Topics (Serializations, JSON,8 Protobuf) | | |
| 11/11 | | 12 | No class | | |
| 11/16 | 20 | 13 | Advanced Java Topics (Reflections) | 8 | |
| 11/18 | | 13 | Project Final Checkpoint | | A6 Due |
| 11/23 | 21 | 14 | Java Performance Optimizations | | |
| 11/25 | | | No class | | |
| 11/30 | 22 | 15 | Java Unit Test | | |

| | | | | | |
|---|---|---|---|---|---|
| 12/2 | 23 | 15 | Java Unit Test | | |
| 12/7 | | 16 | Review, Q & A | 9 | Project Due |