

San José State University
College of Science/Department of Computer Science
CS151, Object-Oriented Design, Section 5, Spring, 2019

Course and Contact Information

Instructor:	Dr. Katarzyna Tarnowska
Office Location:	MacQuarrie Hall (MH) 215
Telephone:	(408) 9245076
Email:	Katarzyna.tarnowska@sjsu.edu
Office Hours:	Monday, 3PM – 4PM, Wednesday 10:30AM – 11:30AM
Class Days/Time:	Monday & Wednesday, 1:30PM – 2:45AM
Classroom:	Duncan Hall 450
Prerequisites:	Math 42, CS46B, and CS 49J (or equivalent knowledge of Java) with a grade of C- or better in each or instructor consent.

Course Description

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: MATH 42, CS 46B, and CS 49J (or equivalent knowledge of Java) (with a grade of "C-" or better in each); Computer Science, Applied and Computational Math or Software Engineering majors only; or instructor consent.

Course Goals

- OO Design:
 - Introduce core UML concepts
 - Introduce a simplified OO analysis and design methodology
 - Present the concept of design pattern
 - Present the concept of a software framework
- Java Language:
 - Make students proficient in the use and creation of interfaces and inheritance hierarchies
 - Make students proficient in the Java type system
 - Introduce threads and thread safety
- GUI Programming:
 - Introduce a GUI toolkit, including basic widgets and the event handling mechanism

Course Learning Outcomes (CLO)

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

1. OO Design
 - Interpret and produce UML class diagrams and UML sequence diagrams
 - Develop simple use cases, perform noun-verb analysis, interpret and produce CRC cards
 - Appropriately select and apply the following design patterns in the construction of a software application: Composite, Decorator, Iterator, Strategy, Template method, and Observer
 - Be able to follow a systematic OO design methodology

2. Java language
 - Create a class hierarchy involving existing and new interfaces and classes, including inner classes.
 - Design, implement, test, and debug programs in an object-oriented language, involving the creation of at least 10 classes and interfaces
 - Implement correctly the equals, hashCode, clone, toString methods
 - Use serialization, reflection, and generics
 - Throw, propagate and catch exceptions
3. GUI Programming
 - Use a GUI toolkit to create a graphical user interface involving frames, buttons, text components, panels, menus, and simple geometric shapes

Required Texts/Readings

Textbook

C. Horstmann, Object-Oriented Design & Patterns, 3rd edition.

Other technology requirements / equipment / material

- Software
 - Programming Language: [Java Platform SE 8](#)
 - [Download](http://www.oracle.com/technetwork/java/javase/downloads/index.html) at <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
 - IDE:
 - [Eclipse](http://eclipse.org/) at <http://eclipse.org/>
 - Version control (for group projects)
 - SVN at <https://svnkit.com>
 - SVN for Eclipse at <https://www.eclipse.org/subversive/>
 - Git at <https://git-scm.com/>
 - Git for Eclipse at <https://www.eclipse.org/egit/>
 - UML design
 - Lucidchart
 - Violet

Course Requirements and Assignments

- [University Policy S16-9](#): *“Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally three hours per unit per week) for instruction, preparation/studying, or course related activities, including but not limited to internships, labs, and clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus.”*
- Team project – design and implement an application (topics will be given later in the semester). Apply object-oriented design process (CRC and UML), design patterns, GUI programming and concurrency (detailed requirements will be presented at the lecture). Documentation and collaboration tools required.

Final Examination or Evaluation

Final exam with multiple choice questions and short open-ended questions.

Grading Information

Determination of Grades

- Total points for the course will be weighted by:

CS151 / Object-Oriented Design, Section 5, Spring 2019, Course Schedule

Course Schedule

The schedule is subject to change with fair notice.

Week	Date	Topics, Readings, Assignments, Deadlines
1	1/28	Introduction to the course, A Crash Course in Java, Chapter 1
1	1/30	A Crash Course in Java, Chapter 1
2	2/4	A Crash Course in Java, Chapter 1
2	2/6	The Object-Oriented Design Process, Chapter 2
3	2/11	The Object-Oriented Design Process, Chapter 2
3	2/13	The Object-Oriented Design Process, Chapter 2
4	2/18	Guidelines for Class Design, Chapter 3
4	2/20	Guidelines for Class Design, Chapter 3
5	2/25	Interfaces and Polymorphism, Chapter 4
5	2/27	Interfaces and Polymorphism, Chapter 4
6	3/4	Patterns and GUI Programming, Chapter 5
6	3/6	Patterns and GUI Programming, Chapter 5
7	3/11	Patterns and GUI Programming, Chapter 5
7	3/13	Inheritance and Abstract Classes, Chapter 6
8	3/18	Inheritance and Abstract Classes, Chapter 6
8	3/20	Inheritance and Abstract Classes, Chapter 6
9	3/25	The Java Object Model, Chapter 7
9	3/27	The Java Object Model, Chapter 7
10	4/1	Cesar Chavez Day – Campus Closed
10	4/3	Spring Recess – Campus Closed
11	4/8	Frameworks, Chapter 8
11	4/10	Frameworks, Chapter 8
12	4/15	Concurrent Programming, Chapter 9
12	4/17	Concurrent Programming, Chapter 9
13	4/22	More Design Patterns, Chapter 10
13	4/24	More Design Patterns, Chapter 10

Week	Date	Topics, Readings, Assignments, Deadlines
14	4/29	More Design Patterns, Chapter 10
14	5/1	Project Presentations
15	5/6	Project Presentations
15	5/8	Project Presentations
16	5/13	Final review
16	5/16 (tentative)	(tentative) Final exam DH 450, 12:15-14:30