Java II - Object-Oriented Programming

This is an intermediate course that builds upon the elementary concepts of Java I. It focuses on object-oriented programming principles, including inheritance, polymorphism, and encapsulation. Students will learn how to create and use classes, interfaces, and abstract classes, as well as how to use design patterns such as the Singleton and Observer patterns. The course also covers advanced topics such as multithreading, exception handling, and file I/O. By the end of the course, students will have the skills to build more complex applications with Java, including graphical user interfaces and networking applications.

Vocabulary Terms - Java II

Unit 1: Object-Oriented Programming

- Introduction to Java classes and objects
- Encapsulation: access modifiers and getters/setters
- Inheritance: extending classes, overriding methods, super() function
- Polymorphism: using interfaces and abstract classes
- Packages and modules

Unit 2: Advanced OOP Principles

- Multithreading: creating and managing threads
- Generics: creating classes and methods with type parameters
- Lambda expressions: creating functional interfaces and using lambda syntax
- Stream API: processing collections with streams
- Reflection: examining and modifying classes at runtime

Unit 3: Advanced Java Concepts

- 13 Advanced (but useful) Git Techniques and Shortcuts
- Graphical User Interfaces: creating GUIs with Swing and JavaFX
- Networking: creating and communicating with sockets, working with HTTP requests

- Database Connectivity: using JDBC to connect to and manipulate databases
- Web Development: creating dynamic web pages with servlets and JSP
- Maven & Gradle
- Final Project: Multi-threaded chat app? GUI game?