

COURSE DESCRIPTION

| | | | |
|-------------------------------------|---|---------------------------|---------------|
| Department and Course Number | CMPS 360 | Course Coordinator | Frank Ducrest |
| Course Title | Programming in Java | Total Credits | 3 |
| URL | General CMPS 360 Resources Page: http://fidelio.cacs.louisiana.edu/360/ | Semester | Spring |
| | | hours | 3 |

Current Bulletin Description

Programming in Java. Java syntax and semantics, use of interfaces, packages, threads, I/O, and collections. Creation of GUI applications, server pages, servlets, Jars, remote methods and database applications. Prereq: CMPS 250 or CMPS 260 with a grade of C or better.

Textbook

Just Java 2 (6th Edition, 2004); by Peter van der Linden, Publisher – Prentice Hall; ISBN - 0131482114

References

JAVA Backpack Reference Guide; by Peter J DePasquale, Publisher – Addison-Wesley; ISBN - 0321304276

Java Programming Language (4th Edition); by Arnold, Gosling, Holmes, Publisher - Addison Wesley; ISBN - 0321349806

Sun SE Online Tutorials: <http://java.sun.com/javase/reference/tutorials.jsp>

Sun EE Online Tutorials: <http://java.sun.com/javaee/reference/tutorials/>

Course Goals

Students are to be exposed to and become familiar with

- Java syntax and semantics
- use of packages, threads, interfaces, collections
- use of various Java Application Programming Interfaces (APIs) to create Java programs including GUI applications, socket servers, Java Server Pages, servlets and database clients
- use of Jars and other Java distribution mechanisms
- use of programming environment

Course Outcomes

Each student passing CMPS 360 should:

- be capable of working in the Java programming language, compilers and development tools in order to develop and distribute software independent of platform
- be sufficiently knowledgeable of and experienced with the underlying structure of Java Applications Programming Interfaces and Frameworks in order to quickly find, learn and use unfamiliar APIs and Frameworks when developing Java software
- be familiar with the issues involved in creating GUI, network and web based applications and services

Prerequisites by Topic

- experience with programming in an object oriented programming language such as C++ through introductory data structures
- basic data types and strings
- user and file I/O
- control structures
- arrays
- functions
- concepts of scope
- pointers or reference variables
- lists, stacks, queues as ADTs and implementations
- automatic and dynamic variables
- dynamic data structures
- structures / records
- classes and objects
- inheritance
- generic / template classes
- exceptions
- recursion
- comparison of algorithms, Big-O notation

Major Topics Covered in the Course

1. Introduction, distribution of software and installation instruction (1 hours 15 minutes)
2. Conversion of previous programming experience to Java (4 hours 15 minutes)
3. Extended Topics in Object Orientation (4 hours 15 minutes)
 - inheritance, polymorphism, abstraction
 - exceptions
 - interfaces
 - nested classes
4. Inclass Practice with Techniques and Use of Javadocs (1 hour 15 minutes)
5. Threads (2 hours 30 minutes)
6. Generics and Collections (1 hour 15 minutes)
7. I/O (2 hours 30 minutes)
8. Swing and applications with GUIs (3 hours 45 minutes)
9. Socket Clients and Servers (1 hour 15 minutes)
10. Java Enterprise Edition Overview (1 hour 15 minutes)
11. Databases (5 hours)
 - basic relational databases
 - introduction to SQL
 - connecting to a local DBMS
 - connecting to a DBMS running as a socket server
12. Servlets and Java Server Pages (5 hours)
 - Basic HTML and HTML forms
 - Servlets
 - Java Server Pages
13. additional topics (3 hours 45 minutes)
14. Exams (includes final) (3 hours 45 minutes)

Oral and Written Communications

Every student is required to submit at least 0 written reports (not including exams, tests, quizzes, or commented programs) of typically 0 pages and to make 0 oral presentations of typically 0 minutes duration. Include only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

Social and Ethical Issues

Please list the topics that address the social and ethical implications of computing covered in all course sections. Estimate the class time spent on each topic. In what ways are the students in this course graded on their understanding of these topics (e.g., test questions, essays, oral presentations, and so forth)?

N.A.

Theoretical Content

Please list the types of theoretical material covered, and estimate the time devoted to such coverage.

Concepts of Object Orientation:

(2 hours)

Problem Analysis

Please describe the analysis experiences common to all course sections.

N.A.

Solution Design

Please describe the design experiences common to all course sections.

Students are required to design object oriented solutions to assigned problems.