

Course Syllabus

Course Title

Foundations of Information Processing in LIS

Semester

Summer 2016

Course Number

LIS452LE

Instructor

Kevin Trainor

Catalog Description

Covers the common data and document processing constructs and programming concepts used in library and information science. The history, strengths and weaknesses of the techniques are evaluated in the context of our discipline. These constructs and techniques form the basis of applications in areas such as bibliographic records management, full text management and multimedia. No prior programming background is assumed.

Course Description

This is a first course computer programming using the Java programming Language. The course includes a learning approach that is based on hands-on examples, the use of the Eclipse Integrated Development Environment, the use of the Git source-code version control system, unit testing with JUnit, as well as special emphasis on object-oriented design principles and test-driven development.

Required Text

Reges, S., & Stepp, M. (2017). Building java programs : A back to basics approach (4th Edition). Boston: Pearson. ISBN-13: 9780134322766

Software

Students will be completing coding assignments and working on their projects using the Eclipse integrated development environment, the Git version control system, and the JUnit testing framework. These tools should be installed on each student's computer before our first weekly live session meeting. Further information will be provided separately to help with the installations including details on versions to download and video tutorials for software installations.

Course Topics

- Introduction to Java Programming
- Working With Eclipse
- Working With Git
- Primitive Data and Definite Loops
- Introduction to Parameters and Objects
- Conditional Execution
- Program Logic and Indefinite Loops
- File Processing

- Arrays
- Classes
- Testing With JUnit
- Test-Driven Development
- Inheritance and Interfaces
- Object-Oriented Design Principles
- ArrayLists
- Java Collections Framework

Course Objectives

- Understand the role that programming plays in information processing and management.
- Code object-oriented Java programs at a level consistent with becoming a beginning-level contributor to an open source development project.
- Use the Eclipse IDE development environment competently.
- Use the Git source-code version control system competently.
- Employ object-oriented design techniques.
- Employ test-driven development techniques using JUnit.
- Make a substantial start on a personal programming project that will help the student reach academic and professional goals while providing a practice environment for the future.

Review of Assignment Solutions

Weekly live sessions will typically begin with review of assignments from the prior week. Assignments are due at the start of the weekly live session. We will begin the live session by reviewing solutions to the assignments. One or two student volunteers will be asked to show their solutions each week. I will also show my solution. This part of the class will serve as a debriefing of the assignment exercise and a review of the prior week material. It will be the best opportunity to ask questions and resolve confusion regarding past material.

Lectures

The latter part of each weekly live session will include a lecture on the topic of the week. Questions from students during the lecture are encouraged. My goal is for this part of the class to be as discussion-oriented as is feasible for a technical course. At the end of the weekly live session, I will explain the coding assignments that are due at the beginning of our next class.

Tutorial Examples

I will be recording tutorial videos that illustrate the techniques that you will need to complete coding assignments. During the tutorial video, I will typically be coding a solution to a problem that is similar to the one which you will need to solve in the coding assignment.

Coding Assignments

Coding assignments will give you the opportunity to practice the skills covered in the last weekly live session. Assignments will relate to skills covered during the preceding lecture. Tutorial example videos will be provided to provide support for coding assignments.

The grading rubric for assignments has been set up to encourage all students to make a good faith effort on each exercise and to submit a proper solution before the beginning of the next class. Students who do this can expect a minimum score of 85 on the assignment. Students who submit work late can expect a maximum score of 84. Assignments that are not submitted will receive a zero score.

Office Hours

I have schedule an online office hours session for each week in which we have class. Please check the weekly schedule for dates and times. Office hours will be conducted using the Blackboard Collaborate Drop-In Room. This will serve as a good opportunity to seek help with coding assignments, clear up confusing issues, or just to check in and say hello.

Student Project

Each student will be required to choose and start a programming project during the course. The project should be related to a professional or hobby interest of the student. The goal is to pick a project that can be further developed even after this course is finished, to make some substantial progress in getting that project started during this course, and to articulate a plan for future development on the project. Students will be asked to give a presentation on their project during the final week of classes.

Work Required of Students

Students will be expected to complete all reading assignments before class. All work on coding assignments and the student project must be done independently by each student and submitted by the beginning of the next class. Collaboration among students in discussing alternatives and solving coding problems is encouraged. Nevertheless, the work submitted by each student must be their own work product.

Students will be expected to participate in class actively and share their own experiences (both good and bad) with the class. This will also include interrupting the lecture portion of the class to ask questions or raise issues.

Basis for Determining Grade

The various components of student work will contribute to the final grade based upon the following percentages:

- Coding Assignments 50%
- Student Project: 40%
- Class Participation 10%

Letter grades will be determined as follows:

- A+: 97 - 100%;
- A : 93 - 96%;
- A- : 90 - 92%;
- B+: 87 - 89%;
- B : 83 - 86%;
- B- : 80 - 82%;
- C+: 77 - 79%;
- C : 73 - 76%;
- C- : 70 - 72%;
- D+: 67 - 69%;
- D : 63 - 66%;
- D- : 60 - 62%;
- F : 0 - 59%;

Contact Hours

This course will require approximately 54 contact hours.

Last Revised

2016-05-08