Final Project - Due May 13

Implement a program to query and update the text database for a library using a client/server model allowing multiple clients safe access to the database, and using a producer-consumer model to control the inventory of each item.

The project will be graded according the criteria for the final project - see below.

Elaboration:

1. Multiple instances of a client class should be supported.
2. Use threads and locks to provide safe access to the database.
3. Add a field to each record in the database representing the maximum number of that book the library will hold. This number should be given a default value if the field is not present in the text file.
4. Functions should include:
    - Returning a book to the library, blocking if the maximum number of that book is reached until a book is borrowed.
    - Borrowing a book, blocking if the desired book is not in the library until a book is returned.
    - Allow a client to cancel a request that has been blocked.

Deliverables

1. Java source code files
2. any configuration files used
3. a well-written Word document describing:
    a. your overall design, including a UML class diagram showing the type of the class relationships
    b. description of how to set up your application
    c. your test plan, including test data and results, with screen snapshots of each of your test cases
    d. your approach, lessons learned, design strengths and limitations, and suggestions for future improvement and alternative approaches

Your project is due by midnight, EST, on the date posted in the class schedule. Your instructor's policy on late projects applies to this project.

Submitted projects that show evidence of plagiarism will be handled in accordance with UMUC Policy 150.25 — Academic Dishonesty and Plagiarism.

Format

Documentation format and length. The documentation describing and reflecting on your design and approach should be written using Microsoft Word, and should be no more than five pages in length and no less than two pages. The font size should be 12 point. The page margins should be

one inch. The paragraphs should be double spaced. All figures, tables, equations, and references should be properly labeled and formatted using APA style.

- Code format:
  - header comment block
  - appropriate comments within the code
  - appropriate variable and function names
  - correct indentation

Grading

This activity is awarded 20 percent of the total grade in the course. In the grade book, the total number of points will be set to 100. The project elements will be assessed as follows:

Attributes	Value
Project design	20 points
Project functionality	40 points
Test data	20 points
Approach documentation	15 points
Approach documentation grammar and spelling	5 points
Total	100 points