Course Description

Database processing and architecture; conceptual data modeling and data design; logical
data models; relational models, operations and algebra; query languages and SQL;
normalization and schema refinement; basic indexing techniques (hashing and B+trees).
PREREQUISITE: COMP 2150, or permission of instructor.

This course will expose students to a wide variety of data models used in database design and
implementation, as well as tools for building a database on an existing system. The students are
further expected to develop practical skills and experience performing various tasks within
database systems. Examples, include database design, database development, basic query
processing, and security.

Topics to be covered include:

i.  **Database Systems Concepts**: Data abstraction; schema; data independence; data definition
    and data manipulation languages

ii. **Data Models**: Entity-relationship, and relational models: definition and operations; query
    languages under each model; SQL

iii. **Database design**: Functional dependencies; normal forms; dependency preservations;
    information loss

iv. **Storage strategies**: B-trees, hashing and relational operation implementation

v. **Database Security**

vi. **Introduction of Transactions and Concurrency Control**

Professional Conduct:

Students are expected to conduct themselves in a professional manner at all times. This
expectation includes but is not limited to the showing respect to the instructor and students during
class times, staying focused during class time, preparing yourself to learn, submitting assignments
in a timely manner, asking questions, and attending class.

Each student is further held accountable to The University of Memphis’s code of conduct.
**Limited Collaboration Policy:**

Students are permitted and encouraged (but not required) to discuss the ideas and concepts of any classroom topic or assignment. Unless otherwise specified, the product of each assignment and test is expected to be sole, individual work each student. Specifically, students can discuss ideas and concepts but one student is not permitted to write code or prose for another student. All help must be documented and credited appropriately including help provided by peers, the course text, the Internet, the instructor, and graduate assistants.

Plagiarism or cheating behavior in any form is unethical and detrimental to proper education and will not be tolerated. All work submitted by a student (projects, programming assignments, lab assignments, quizzes, tests, etc.) is expected to be a student's own original work. Plagiarism is incurred when any part of work submitted for evaluation is presented as your own such that the reader is led to believe it is the result your own effort (i.e. no proper credit is listed to the sources in your own work). You can be penalized for plagiarism but there is no penalty by showing proper respect to others by crediting their contribution. Students are allowed and encouraged to discuss with each other and look up resources in the literature (including the Internet) on their assignments, but appropriate references must be included for the materials consulted, and appropriate citations made when the material is taken verbatim.

If plagiarism or cheating occurs, the student will receive a failing grade on the assignment and (at the instructor’s discretion) a failing grade in the course. The course instructor may also decide to forward the incident to the University Judicial Affairs Office for further disciplinary action. For further information on U of M code of student conduct and academic discipline procedures, please refer to: [http://www.people.memphis.edu/~jaffairs/](http://www.people.memphis.edu/~jaffairs/)

**Warning 1:** Each student should accept help with care. It is very easy to mislead yourself into believing that you understand a concept when others are providing aid or assisting. In a crunch (such as an exam), this error can prove fatal.

**Warning 2:** Please give help with care. Collaboration is intended to improve the classes understanding of a concept. If too much help is given, students may be enabled to fail!

**Warning 3:** When it comes to learning basic concepts such as definitions and explanations, the Internet can be a two edge sword; while the Internet can often provide accurate data, it can also provide accurate data in the wrong context as well as incorrect data, or accurate data in misleading language. Be aware.

**Grading:**

Mastery of this courses material will be evaluated as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Three (3) exams</td>
<td>60%</td>
</tr>
<tr>
<td>[Final Exam will replace the lowest of these test scores!]</td>
<td></td>
</tr>
<tr>
<td>Class project</td>
<td>17%</td>
</tr>
<tr>
<td>Class participation</td>
<td>3%</td>
</tr>
<tr>
<td>Homework</td>
<td>20%</td>
</tr>
</tbody>
</table>

1 Each student is expected to prepare in advance for each in-class discussions. Class participation is evaluated by the student responses to instructor prompted questions.

2 Homework and project submissions should include examples (screen captures) demonstrating the functionality of the code.
Course Resources:
As part of this course, you are expected to demonstrate mastery of database concepts and skills (not mastery of a particular database tool). In this light, most assignments will require that your submissions be compatibility with MySQL running on Linux platform. In this light, I provide a test environment for you. You must access the test environment via SSH. Therefore, an SSH client is required for this course.

A special thanks and acknowledgement to Amazon Web Services for the grant which has provided a small grant that is used to fund the server and database resources made available to this class.

Course Outline:
1. Introduction
2. Introduction to the Relational Model
3. Introduction to SQL
4. Intermediate SQL
5. Formal Relational Query Language
6. Database Design and the E-R Model
7. Relational Database Design
8. Storage and File Structure
9. Indexing and Hashing
10. Query Processing
11. Query Optimization
12. Transactions
13. Concurrency Control
14. Recovery System
15. Security