

Course Syllabus

Course Title

Introduction to Databases

Semester

Fall 2016

Course Number

LIS490DBL

Instructor

Kevin Trainor

trainor1@illinois.edu

847-650-9706

Online Office Hours Sessions

Every Monday evening, I will be holding Online Office Hours using the Blackboard Collaborate platform. You are invited to join us to discuss questions and issues concerning the class. Meet in the Drop In Room of the Blackboard Collaborate platform. Join the Kevin Trainor Office Hours breakout room. Please use a headset.

Description

The course provides students with both theoretical and practical training in good database design. By the end of the course students will create a conceptual data model using entity-relationship diagrams, understand the importance of referential integrity and how to enforce data integrity constraints when creating a database. Students will be proficient in writing basic queries in the structured query language (SQL) and have a general understanding of relational database theory including normalization.

Required Texts

Murach, J. (2015). Murach's MySQL (2nd Edition). Fresno, CA: Mike Murach & Associates, Inc. ISBN 978-1-890774-82-0. Digital edition available directly from the publisher at <https://www.murach.com/shop/murach-s-mysql-2nd-edition-detail>

Oppel, Andrew J. (2010) Data Modeling : A Beginner's Guide New York : McGraw-Hill, ISBN-10: 0071623981, ISBN-13: 978-0071623988

Software

The coding assignments for this course will require you to install MySQL software on your computer. This includes:

- MySQL Community Server
- MySQL Workbench

You should accomplish this task before the end of Week 1.

The Murach textbook is a great source of information on where to locate the MySQL software that you will need and how to install the software on your computer. Appendix A

of the Murach text addresses installing the software on a Windows computer. Appendix B addresses installing the software on an OS X computer. While it is possible to install the software on a Linux computer, doing so will require that you rely on documentation that you must find at the Oracle Web site. So, I recommend using Windows or OS X if possible.

The Murach text was written while using a specific version each MySQL software product:

- MySQL Community Server version 5.6
- MySQL Workbench version 6.2.3

The most current version of each of these software products is now a few releases beyond those used in creating the Murach text. The functionality of the software will be essentially the same whether you choose to install the same version that was used in Murach or whether you choose to install the latest version. If you choose to install the latest version, you may need to follow the slightly different installation instructions that will be available on the Oracle Web site. If you choose to install the earlier version, you will have to hunt around on the Oracle Web site to find the download for a prior product version. Neither of these approaches is foolproof. You just need to choose the approach that you find most appealing and persevere until you have reached the goal.

I intend to use a more recent version of the MySQL software. I have created tutorial demo videos showing me installing the software on both Windows and OS X. These videos were created for the Fall 2015 Semester. Nevertheless, they should serve as a good tutorial on installing the MySQL software.

I have created a MySQL Software Forum on our Moodle page. This will be a place that you can post messages regarding your challenges and successes with the MySQL Software. I will monitor the forum and provide support. I invite you to do the same. I only ask that you don't post full working solutions to exercise problems. Instead, I prefer that you offer hints as to how to fix broken code.

Course Topics

- An introduction to relational databases and SQL
- How to use MySQL Workbench and other development tools
- How to retrieve data from a single table
- How to retrieve data from two or more tables
- How to code summary queries
- How to code subqueries
- How to insert, update, and delete data
- How to work with data types
- How to use functions
- Introduction to Data Modeling
- How to design a database
- How to create databases, tables, and indexes
- How to create views
- How to back up and restore a database
- How to use transactions and locking
- Relational Model Components

- Conceptual Data Modeling
- Logical Database Design Using Normalization

Course Objectives

- Identify the advantages of relational database management systems over conventional files and spreadsheets.
- Code SQL queries that extract data from a single table.
- Code SQL queries that extract data from two or more tables.
- Code SQL summary queries.
- Code SQL subqueries.
- Code SQL statements that implement insert, update, and delete functions.
- Identify the standard data types available in relational databases and their typical use.
- Code SQL functions.
- Create databases, tables, and indexes using the Data Definition Language (DDL).
- Create views.
- Backup and restore a database
- Create a non-trivial conceptual data model.
- Create a non-trivial logical data model.

Work Required of Students

General Issues

1. Because our class sessions will include a significant amount of discussion, you will be expected to speak often during class using your headset. You will also be expected to use the application sharing, whiteboard, and other features of Blackboard Collaborate in order to fully participate in class. If you need to practice your online platform skills, I encourage you to attend one of the *Introduction to Blackboard Collaborate* sessions being held by the iSchool ITD (see Moodle front page for details).
2. You will be expected to complete all reading assignments before class. Each class session will include a lecture / discussion portion based upon the readings for that class.
3. Written and coding assignments should be completed before the class in which they will be discussed. Substantial assignment credit will be associated with handing in work that reflects a good faith effort before class begins. My discussion of Coding Assignments (below) will explain why this timeliness is so important.
4. Written and coding work that you submit must be your own. So copying of the work of others (current students, past students, me, or others) is prohibited. Nevertheless, I encourage you to consult with others or with me prior to handing in your work. Feel free to show the current state of your work to anyone and ask for feedback, suggestions or encouragement. I also encourage you to review the

work of others that is not a direct solution to the assignment at hand (examples in textbooks, examples on the Internet, or examples that a group of people might work through on a white board). Collaborate as much as required in order to fully understand the techniques needed to do your assignment

Class Contribution

My goal for this class is that it will be very interactive. To achieve this, you will need to speak during class, use other Blackboard Collaborate features, and post to the Moodle forums in order to share your questions and insights.

Please note that the Class contribution component of your grade is substantial (10% of your final grade). So, please take these activities seriously. Activities considered in this portion of your grade will include:

- Attendance during live sessions
- Speaking (using your microphone) during class. This includes asking questions, providing answers, or simply offering your observations.
- Presenting your coding assignment work (see below) during class.
- Posting to Moodle forums.

Assignment Descriptions

As indicated in the schedule, there are one or more required assignments for each week of class. It is important that you check the schedule each week and complete the assignments on time in order to receive credit for your work. The following are general descriptions of each type of assignment:

1. **Coding Assignments:** Our textbook has a companion set of exercises named *My Guitar Shop*. For the most part, one set of these exercises has been assigned for each chapter. While starter files have been provided for these exercises, the solutions have not been provided. Generally, this set of exercises is organized in a manner that is parallel to the exercises that are published at the end of each chapter in the textbook. If you should get stuck on a *My Guitar Shop* exercise, you might look to the textbook exercises and their solutions for inspiration.

Coding Assignments will be due by the start of the online class session in which we will discuss them. Solutions to Coding Assignments will be published on our Moodle page immediately after the due date. Also, we will be reviewing your solutions (and mine) to the exercises during class. **The published solutions and my discussion of those solutions during class will serve as your primary feedback on Coding Assignments.**

As secondary feedback, your Coding Assignment submissions will be graded and commented upon. Feedback will be published to the Moodle assignment submission activity. Due to the size of the class, you can expect to get grading and comment feedback several weeks after the due date for the assignment.

To encourage you to work on and submit your Coding Assignments on time, I have worked out a grading system that rewards good faith effort, timely submission, and attention to detail. In order to demonstrate good faith effort, you

must turn in work that shows that you tried to do the exercise. In order for your submission to be timely, it must be submitted before the assignment deadline. In order to demonstrate attention to detail, you must submit the proper number of files in the proper file format using the proper file naming conventions.

Submissions that meet all of the requirements will earn a minimum score of 85. Submissions that do not meet all of the requirements will earn a maximum score of 84.

2. Final Project: The final project will require you to design and implement a MySQL database solution to a business problem. You will be given documentation on an existing crude information system solution that is implemented with spreadsheets. You will be responsible for extracting the business requirements from the existing solution, designing a database, populating a sample database, and writing the SQL statements required to produce the same functionality using this more robust set of tools. More detailed instructions for this project will be published separately.

Basis for Determining Grade

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

- | | |
|-------------------------------|-----|
| • Class participation | 10% |
| • Required coding assignments | 45% |
| • Final project | 45% |

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%; |

Academic Integrity

Please review and reflect on the academic integrity policy of the University of Illinois, http://admin.illinois.edu/policy/code/article1_part4_1-401.html to which we subscribe.

By turning in materials for review, you certify that all work presented is your own and has been done by you independently, or as a member of a designated group for group assignments.

When describing assignments in the syllabus or in other documents that provide the detailed requirements for one of your assignments, I have tried to be quite specific about the degree to which collaboration is encouraged and the degree to which you are expected to submit an original work of which you are the author. If you have any questions regarding the policy regarding a particular assignment, please contact me directly for advice.

Please be aware that the consequences of academic dishonesty will be severe. Students who violate university standards of academic integrity are subject to disciplinary action, including a reduced grade, failure in the course, and suspension or dismissal from the University.

Statement of Inclusion

<http://www.inclusiveillinois.illinois.edu/mission.html> As the state's premier public university, the University of Illinois at Urbana-Champaign's core mission is to serve the interests of the diverse people of the state of Illinois and beyond. The institution thus values inclusion and a pluralistic learning and research environment, one which we respect the varied perspectives and lived experiences of a diverse community and global workforce. We support diversity of worldviews, histories, and cultural knowledge across a range of social groups including race, ethnicity, gender identity, sexual orientation, abilities, economic class, religion, and their intersections.

Accessibly Statement

To obtain accessibility-related academic adjustments and/or auxiliary aids, students with disabilities must contact the course instructor and the Disability Resources and Educational Services (DRES) as soon as possible. To contact DRES you may visit 1207 S. Oak St., Champaign, call 333-4603 (V/TTY), or e-mail a message to disability@uiuc.edu.

Contact Hours

This course will require approximately 45 contact hours.

Last Revised

2016-08-21