INF1343: Data Modeling and Database Design

Course Code: INF1343H

Course Section:

Semester Offered: Winter 2011

Course Meeting Time: Mondays, 9 am – noon

Course Location: Bissel Rm. 417

Credit Value: 0.5

Pre-requisites: INF1003. *Students who have not taken INF1003 should discuss their situation with the instructor at the earliest opportunity.*

Instructor

Professor:	Yuri Takhteyev
Email:	yuri.takhteyev@utoronto.ca **Please write from your utoronto email account and put "inf1343" in the subject of the message.** Expect two business days turn around time. If you do not receive a reply within this period, please resend your message.
Phone:	415-946-3809
Office Hours:	2-3 pm on Mondays or by appointment, Rm. 328 in iSouth (45 Wilcocks St.) *Students are encouraged to make use of office hours.*
Course Website:	http://takhteyev.org/courses/11W/inf1343/

Course Description

The purpose of this course is to provide an introduction to databases by analyzing their structure, content and measurement and by applying principles governing data modeling, database design and production with an emphasis on modeling, design and representation of content, decisions and tradeoffs involved in modeling, design and creation, and issues of standardization, security and emerging trends.

Goals and Learning Objectives

Students will learn the main principles behind relational databases and information storage systems more broadly. In particular, they will acquire the following skills:

- analysis of domain and user requirements
- database design in the form of ER diagrams
- using SQL to implement and query a database
- implementing a web interface to a database

Teaching Methods

The course will consist of twelve three-hour classes, with the thirteenth session used for a final exam. Class time will be primarily used for lectures, which will explain and expand on the material presented in the readings and will include live demonstrations of how to use database software. Some of the class time will be used for in-class exercises and tutorials. Students are strongly encouraged to do all the required readings *before* the class for which they are assigned and to come to class ready to ask questions about those readings. Some of the material

covered in the assigned readings may not be covered in class; the students are none-the-less responsible for that material. Assignments will generally stress practical skills in database design and implementation.

Required Materials

Required readings will be primarily from two books, which are both available in the University book store. Students are encouraged to purchase paper copies. However, the University of Toronto library also provides access to electronic versions of both books.

Jan L. Harrington (2010) *SQL: Clearly Explained*, Third Edition. Morgan Kaufmann.

→ http://www.sciencedirect.com.myaccess.library.utoronto.ca/science/book/9780123756978

Jan L. Harrington (2009) *Relational Database Design and Implementation: Clearly Explained,* Third Edition. Morgan Kaufmann.

→ http://www.sciencedirect.com.myaccess.library.utoronto.ca/science/book/9780123747303

Students who choose to use the electronic copies are advised to download and open the required chapters ahead of time. Unavailability of the electronic resources at the necessary time or software problems related to their use will not be accepted as valid reasons for late assignments.

Some of the required readings will come from other sources. Those are all available over the Internet, either openly or through the University Library's MyAccess service.

Recommended Materials

The students may find the following additional resources useful:

MySQL

We will be using a MySQL 5.5 database for examples, assignments and the final project. MySQL's implementation of SQL is sometimes different from from the official standard described by Harrington. MySQL Reference manual does a good job describing MySQL implementation. It is also sercheable.

MySQL 5.5 Reference Manual

→ http://dev.mysql.com/doc/refman/5.5/en/

SQL in a Nutshell

→ http://proquestcombo.safaribooksonline.com.myaccess.library.utoronto.ca/9780596155322

Unix / Linux tools

We will be using a Linux database server in this course. Some familiarity with Unix commands will make your life easier.

Introduction to the Command Line

→ http://en.flossmanuals.net/CommandLineIntro/Introduction

Introduction to Command Line Linux

→ http://www.physics.ubc.ca/mbelab/computer/linux-intro/html/

Python

We will be doing some very basic Python programming. You can use the following resources to supplement materials provided in class.

Think Python: How to Think Like a Computer Scientist

→ http://greenteapress.com/thinkpython/html/index.html

Learning Python, Second Edition

→ http://proquestcombo.safaribooksonline.com.myaccess.library.utoronto.ca/0596002815

Assessment and Grading

Your grade will consist of one individual assignment, one group assignment, a group project and a final exam.

Assignment	Weight	Due Date	Type
Assignment 1: SQL	15%	Mon, January 24, 2011	individual
Assignment 2: Database Design	20%	Mon, February 14, 2011	individual or pairs
Initial Project Design	5%	Mon, March 7, 2011	group
Project Report	25%	Mon, March 28, 2011	group
Final Exam	35%	Mon, April 4, 2011	individual

Assignments

For the first assignment, worth 15% of the grade, you will use SQL to perform queries on existing databases, create simple tables and import data into them. You will also be asked to answer questions related to the materials from the first several weeks. This assignment must be done individually.

For the second assignment, worth 20% of the grade, you will do exercises related to designing and implementing a database and answer questions related to the material from the corresponding weeks. This assignment can be done individually or in pairs.

The Project

For the final project you will design and implement an information system with a MySQL database back-end and a simple web interface. The overall purpose and functionality of the system will be specified in a project handout that will be distributed later in the course. Students who want to develop a system serving a different purpose can do so **with instructor's permission, but must discuss this with the instructor early in the semester.**

The final project will consist of two submissions, which are jointly worth 30% of the grade. The first submission is an initial design, worth 5% of the grade. The second submission is the actual implementation and a report, worth 25% of the grade.

The projects can be done individually, in pairs, or in teams of three. The students who want to work in groups are responsible for forming the groups. **Students who worked together on Assignment 2 cannot work on the same final project team.**

The Final Exam

The final, worth 35% of the grade, will test your understanding of the material from the whole semester. You will be asked to write SQL statements, draw ER diagrams, and answer conceptual questions about database design.

Each assignment submission must adhere strictly to requirements specified in the assignment handout and must be submitted on time. **There will be a penalty for lateness of 5% deducted for each full or partial 24 hours of lateness.** That is, an assignment due at 9:10 am on Monday loses 5% if not submitted by 9:10 am on Monday, 10% if not submitted by 9:10 am on Tuesday, etc. **Additionally, no assignment will be accepted more than 7 days late.** 7 days means 168 hours.

Deadline extensions will be granted only for a compelling reason and with appropriate documentation and students should contact the instructor immediately, and no later than the due date, if a deadline cannot be met. Students are required to **declare their absence on ROSI,** in order to receive academic accommodation for any course work such as missed tests, late assignments, and exams.

The students should familiarize themselves with the University's policy towards academic integrity:

→ http://www.utoronto.ca/academicintegrity/

Students will be asked to sign an academic integrity pledge before submitting any assignments for grading and should expect **zero tolerance** towards academic violations.

Weekly Class Schedule and Readings

Every attempt will be made to follow this schedule, but its contents are subject to change.

Abbreviations for the two books that we will be primarily using:

- □ **SQL:** Jan L. Harrington (2010) *SQL: Clearly Explained*, Third Edition. Morgan Kaufmann.
- ■ **RDD:** Jan L. Harrington (2009) *Relational Database Design and Implementation: Clearly Explained,* Third Edition. Morgan Kaufmann.

Week 1 | January 3
Introduction

Databases in the context of modern information systems. Types of databases. Connecting to a remote server using SSH. Basic UNIX commands. Using a command-line MySQL client. Very basic SQL. Executing SQL statements from a file. Editing SQL files locally and remotely. Copying files using SCP.

- □ SQL ch. 3, "Introduction to SQL."
- □ SQL ch. 1, "The Relational Data Model" (optional, but recommended).
- ● *Also, have a look at the UNIX resources listed in under "Recommended Materials" above. Those are not required, but they might make your life easier.*

Week 2 | January 10
Relational Databases and SQL

- ► Before this class make sure that you can successfully connect to the database server and execute basic SQL queries.

The relational data model. One-table SELECT. Aggregation. Modifying data with INSERT and UPDATE.

- □ SQL ch. 4, "Simple SQL Retrieval."
- □ SQL ch. 7, "Working with Groups of Rows", pages 161–177.
- □ SQL ch. 8, "Data Modification."

Week 3 | January 17
Joining Tables

Queries using multiple tables. JOIN ON and JOIN USING. "Traditional" JOINs. Subqueries.

- □ SQL ch. 2, "Relational Algebra," pages 36–63.
- □ SQL ch. 5, "Retrieving Data from More than One Table."

Week 4 | January 24
Database Design

- ► **Assignment # 1 due before the beginning of the class.**

Requirements analysis and data modeling. The ER model. ER diagrams. Cardinality and optionality. Weak entities and belonging.

- ■ RDD ch. 2, "Systems Analysis and Database Requirements."
- ■ RDD ch. 3, "Why Good Design Matters."
- ■ RDD ch. 4, "Entities and Relationships."

Week 5 | January 31
Converting a Database Design into a Relational Form

Translating an ER diagram into relational form. Normalization.

- RDD ch. 5, "The Relational Data Model."
- RDD ch. 6, "Normalization."

Week 6 | February 7
Implementing a Database with SQL

Implementing a database with SQL. CREATE DATABASE, CREATE TABLE, and ALTER TABLE. Revisiting INSERT. Importing data with LOAD.

- SQL ch. 9, "Schemas and Tables."
- SQL ch. 11, "Keeping the Design Up to Date."
- RDD ch. 10, "Using Case Tools for Database Design."
- RDD ch. 11, "Database Design Case Study 1: Mighty Mite Motors."

Week 7 | February 14
Embedded SQL

► **Assignment # 2 due before the beginning of the class.**

Using an SQL database from within a program. SQL with Python. SQL in a CGI web application.

- Garshol, L. (1999). "How the web works: HTTP and CGI explained"

  → http://www.garshol.priv.no/download/text/http-tut.html
- Python Library Reference: CGI

  → http://docs.python.org/release/2.5.2/lib/module-cgi.html
- *Also, have a look at the Python resources listed in under "Recommended Materials" above. Those are not required, but they might make your life easier.*

Reading Week | February 21 — No class

Week 8 | February 28
Relational Databases and Documents

Full-text search. Markup languages. XML, YAML, JSON. Markup languages as input or output format. XML support in databases. Exporting and importing XML with --xml and LOAD XML.

- SQL ch. 17, "XML Support."
- Wilde, E. & R. Glushko (2008). "Document design matters." Communications of the ACM, 51(10).

  → http://portal.acm.org.myaccess.library.utoronto.ca/citation.cfm?id=1400195
- *Additional readings TBA.*

Week 9 | March 7
Relational Databases and Objects

► **Initial design for the final project due before the beginning of the class.**

Object-oriented programming. Object-relational mapping. Objects in databases.

- SQL ch. 18, "The Object-Relational Data Model."
- *Think Python*, ch. 15, "Classes and Objects" → http://www.greenteapress.com/thinkpython/html/book016.html
- *The Django Book, ch. 5* → http://www.djangobook.com/en/1.0/chapter05/

Week 10 | March 14
Database Security

Users and access rights. Views. Firewalls. SQL injection attacks. Denial of service attacks.

- □ SQL ch. 12, "Users and Access Rights."
- □ SQL ch. 10, "Views, Temporary Tables, CTEs, and Indexes," pages 237 – 240.
- ■ RDD ch. 15, "Database Security."

Week 11 | March 21
Storage, Structure, and Performance

Representing tables. Indexing. Storage Engines. Transactions. Backup. RAID.

- Lentz, A. (2004). "MySQL Storage Engine Architecture," parts 1, 2, and 3

  → http://dev.mysql.com/tech-resources/articles/storage-engine/part_1.html

  → http://dev.mysql.com/tech-resources/articles/storage-engine/part_2.html

  → http://dev.mysql.com/tech-resources/articles/storage-engine/part_3.html
- SQL ch. 10, "Views, Temporary Tables, CTEs, and Indexes," pages 248 – 253.
- SQL ch. 13, "Users, Sessions, and Transaction Control"

Week 12 | March 28
Scaling Data Management

► **A paper copy of the project report and the digital submission due before the beginning of the class.**

Challenges in large-scale data management. "NoSQL" data management systems. Data management and distributed computing. MapReduce and Hadoop. Cloud computing.

- *Readings TBA.*

Week 13 | April 4 — In-class final exam.

Religious Observance

Students whose religious observances may conflict with the course schedule should consult University's Policy on Scheduling of Classes and Examinations and Other Accommodations for Religious Observances.

→ http://www.governingcouncil.utoronto.ca/policies/religious.htm

Accessability

Students who need accommodations due to a disability or health consideration are asked to approach the instructor and/or the Accessibility Services office as soon as possible. The Accessibility Services staff are available by appointment to assess specific needs, provide referrals and arrange appropriate accommodations. The sooner you let them and us know about your needs, the quicker we can assist you in achieving your learning goals in this course.

→ http://www.accessibility.utoronto.ca/