Introduction to Databases

Syllabus

Web Page

http://www.eecs.northwestern.edu/~pdinda/db

Instructor

Peter A. Dinda
Technological Institute L463
847-467-7859
pdinda@northwestern.edu
Office hours: Thursdays, 2-4pm, or by appointment

Lack of teaching assistant

Due to a lack of TA support for this course, some content has been removed. In particular, the third project and the homeworks will be handed out, but they will not be graded or supported. I strongly encourage you to study these materials on your own since I think they are very useful for understanding important elements of database theory (the homeworks) and database systems (the project) at a deeper level.

The intended TA-led recitation section and office hours will not be held.

Eliminated elements of the course are highlighted by ~~strikethrough~~ in this document.

Location and Time

Lecture: Technological Institute, M1152, 12-12:50am, MWF

~~Optional TA-led Recitation: Thursdays, 6pm, room, Tech L158~~
~~*Recommended, especially for asking questions to clarify lecture and for help with projects*~~

Prerequisites

Required	EECS 311 or equivalent data structures course
Required	EECS 213 or equivalent computer systems course
Recommended	Familiarity with concepts from discrete math such as set theory (EECS 310 might help)
Recommended	Some familiarity with Perl or other scripting language

Textbook and other readings

Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer D. Widom, *Database Systems: The Complete Book*, 2nd Edition, Prentice Hall, 2009. (Textbook - Required)

- An in-depth introduction to databases and database implementation

Phillip Greenspun, *SQL for Web Nerds*, http://philip.greenspun.com/sql/. (Required, but available for free on the web)

- A great introduction to RDBMS systems from the perspective of a web application developer.

Joe Celko, *SQL for Smarties: Advanced SQL Programming*, 2nd edition, Morgan Kaufman, 1999. (Useful)

- A collection of wisdom on how working developers get useful things done in SQL.

Jim Gray, Andreas Reuter, *Transaction Processing: Concepts and Techniques*, Morgan Kaufman, 1993. (Related)

- Definitive book on transactions, a very important component of any modern database system.

Larry Wall, Tom Christiansen, Jon Orwant, *Programming Perl*, 3rd Edition, O'Reilly and Associates, 2000. (Useful)

- A detailed introduction to the Perl language. Your web-oriented projects in this class will be based on Perl CGI. You will need to know (or learn) only limited amounts of Perl.

Objectives, framework, philosophy, and caveats

This course introduces the underlying concepts behind data modeling and database systems using relational database management systems (RDBMS), the structured query language (SQL), and web applications (Perl DBI in CGI) as examples.

You will learn:

- How to model your data using the entity-relationship model
- How to design a normalized schema in the relational data model
- How to implement your schema using SQL
- How to keep your data consistent and safe with your schema using the ACID properties that a modern RDBMS gives you
- How to query your data using SQL
- How to interface to a modern RDBMS from a modern programming language.
- How such interfaces are used to create web applications
- How an RDBMS provides quick access to your data using indices, and how indices are implemented.
- How an RDBMS manages the storage hierarchy.
- How an RDBMS optimizes and execute your queries using the relational algebra, the theoretical underpinning of database systems.

- The history of database systems, including old ideas, like hierarchical databases, that are seeing a resurgence of interest today in the context of XML.
- About issues in database security, including access control and SQL injection attacks.

The textbook I have chosen is actually a combination of two books, an introduction to the concepts and use of databases and an introduction to the implementation of RDBMS systems. We will cover mostly the former. However, this is a very useful and essentially timeless book to have on your bookshelf for both elements.

This is a learn-by-doing kind of class. You will dive right in and modify a small database-based web application, a web log. Next, you will design and implement your own database-based web application for finance. Finally, you'll implement a B+Tree index data structure. The majority of the programming in this class will be from scratch. We will use SQL, Perl, and C++ on Linux systems.

Projects

At the beginning of the course, I will provide you with a simple web application, a tiny web log ("blog"). Microblog is based on an Oracle database and provides a web interface using a CGI application written in Perl that talks to the database via DBI. This is a very common form of web application. You will spend three weeks learning how Microblog works and extending it in several simple ways. The goal is to immediately introduce you to SQL right away and bring you up to speed on the programming elements of the course that you'll need for the second project.

The second project is focused on developing a simple financial portfolio manager that track's a user's investments, and allows the user to "mine" historical financial data in several ways. I will give you a set of requirements and access to about 10 years of stock price data, and you will design and implement a database-backed web-based system. This project will take four weeks.

~~The third project is to build a B+Tree data structure. B+Trees are common on-disk (as opposed to in-memory) data structures used in relational database systems. I will provide you with a framework, starter code, and a test harness.~~

~~Homework~~

~~There will be three (maybe four) homework problems sets that will be periodically assigned to help you improve your understanding of the material.~~

Exams

There will be a midterm exam and a final exam. The midterm exam will take place in the evening outside of class. The final exam will not be cumulative.

Grading

13 and 1/3 % ~~10 %~~ Dry-run project (Microblog)
33 and 1/3 % ~~25 %~~ Portfolio manager project
~~15 % Implementation project (B+Tree index)~~
26 and 2/3 % ~~20 %~~ Midterm
26 and 2/3 % ~~20 %~~ Final
~~10 % Homework~~

Final grades will be computed in the following way. A final score from 0 to 100 will be computed as a weighted sum of each of the projects, the homeworks, and the exams. Scores greater than 90 or greater than 90th percentile will be assigned As, scores greater than 80 or greater than 80th percentile will be assigned Bs, scores greater than 70 or greater than 70th percentile will be assigned Cs, scores greater than 60 or greater than 60th percentile will be assigned Ds, and the remainder will be assigned Fs. Notice that this means that if everyone works hard and gets >90, everyone gets an A. Please choose wisely where you put your time.

Late Policy

For each calendar day after the due date for a homework or a lab, 10% is lost. After 1 day, the maximum score is 90%, after 2 days, 80%, etc, for a maximum of 10 days.

Cheating

Since cheaters are mostly hurting themselves, we do not have the time or energy to hunt them down. We much prefer that you act collegially and help each other to learn the material and to solve development problems than to have you live in fear of our wrath and not talk to each other. Nonetheless, if we detect blatant cheating, we will deal with the cheaters as per Northwestern guidelines.

Schedule

Note that the schedule is subject to change. I will announce schedule and due-date changes via email. If you do not receive a welcome email from me, please let me know.

Lecture	Date	Topics	Readings	Homework and Project
1	9/22	Class mechanics Introductory material, Web applications, client/server, and three-tier	GUW Intro, 9.1, 9.3.1,9.3.2; PG preface + 1	Project A (Microblog) out

If you're unfamiliar with Unix, now would be a good time to view the Unix introduction video available from the course web site.				
2	9/24	More introductory material: why a database is different from a filesystem and what it helps you with. Data modeling, transactions/ACID, queries, abstracting storage+indices, some history lessons (Hierarchical, Network, Relational, Object, Object Relational, Hierarchical again). Hot stuff: P2P, MapReduce	GUW 1; PG preface + 1	
3	9/27	How web applications work. Apache, CGI, Perl, DBI, RDBMS, SQL in a nutshell; some discussion of AJAX	PG 1-7, Perl HO, Oracle HO, GUW 9.3.9	
4	9/29	SQL in a nutshell, Walk through Microblog (SQL)	PG 1-7, Perl HO, Oracle HO	Note: you might find PG 10 useful reading
5	10/1	Database security topics or catchup on Microblog SQL or Perl	GUW 10.1 (although lecture will focus elsewhere)	*Optional SQL Injection Attack Challenge*
6	10/4	(Lecture Canceled; University Meeting)		
7	10/6	Perl in a nutshell	PG 1-7, Perl HO, Oracle HO	
8	10/8	More Perl	PG 1-7, Perl HO, Oracle HO	
9	10/11	Walk through Microblog (Perl)	PG 1-7, Perl HO, Oracle HO	
10	10/13	Walk through Microblog (Perl)	PG 1-7, Perl HO, Oracle HO	HW 1 out (no handin)
11	10/15	Walk through Microblog (Perl). SQL injection attacks	PG 1-7, Perl HO, Oracle HO	
12	10/18	Data models and Data modeling: Why? Start Entity-Relationship: Entity sets, attributes, relationships, ER diagrams, instances, multiplicity, roles, multiway	GUW 2.1, 4.1-4.4	Project A (Microblog) in. Project B out

13	10/20	Entity-Relationship Model: conversion to binary relationships, subclassing, design principles	GUW 4.1-4.4	
14	10/22	(Lecture Canceled; Conference)		
15	10/25	Entity-Relationship Model: constraints, weak entity sets	GUW 4.1-4.4	
16	10/27	Relational Data Model: basics, translating from ER to relational	GUW 2.2, 2.3, 4.5	HW 2 out (no handin)
17	10/29	Relational Data Model: subclasses, functional dependencies	GUW 4.6, 3.1-3.2	
18	11/1	Relational Data Model: Schema design and normal forms	GUW 3.3-3.5, 3.6.6	
Midterm Exam Review: Monday, 11/1, 6pm, in Tech M164 *Midterm Exam: Tuesday, 11/2, 6pm in Tech L251* *Midterm Exam will cover Lectures 1-18 (Note that this is the same as before, there are simply 2 lectures that were canceled)*				
19	11/3	Relational Data Model: Multivalued dependencies	GUW 3.6	HW 3 out (no handin)
20	11/5	Relational Algebra: Sets: union, intersection, difference, selection, projection, Cartesian product, and cross, inner, outer, left, right joins	GUW 2.4, 5.1-5.2	
21	11/8	Relational Algebra: Bags, equivalent expressions, some extended operators	GUW 5.1-5.2	
22	11/10	Relational Algebra: grouping, constraints, data-mining	GUW 5.1-5.2, 2.5	
23	11/12	SQL: strings, regular expressions, date/time, nulls, 3-valued logic, explain plan, subqueries in/exists/>all/>any, correlation	GUW 6	
24	11/15	SQL: insert/update/delete, multi-statement transactions using PL/SQL; create schemas: bit-fields, decimal, blob; drop, alter; indexes; views	GUW 6, 7, 8	Project C out (no handin)
25	11/17	SQL: Constraints, Triggers, systems aspects.	GUW 6, 7, 8	

26	11/19	(Lecture Canceled; NSF)		
27	11/22	Implementation: Storage and Representing Data	GUW 13	Project B in
Thanksgiving Break				
28	11/29	Implementation: Indexes, Btrees	GUW 14.1, 14.2	
29	12/1	Implementation: Indexes, Hashes	GUW 14.3	
30	12/3	Implementation: Indexes, Bitmaps	GUW 14.7	
(dropped due to lack of time)		Implementation: Transactions (Logging, Locking)	GUW 17.1-17.4, 18.1-18.3	
Final Exam, Thursday, 12/9, 12-2pm, in our classroom. Covers Lectures 19-30				

PG = Phillip Greenspun, *SQL for Web Nerds*
GUW = Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer D. Widom, *Database Systems: The Complete Book*