CSC 261/461 – Database Systems

Lecture #1

Spring 2018
Agenda

• Administrative aspects

• Brief overview of the course

• Introduction to databases and SQL
ADMINISTRATIVE ASPECTS
Teaching Staff

- **Instructor:**
  - Tamal Biswas

- **Graduate TA:**
  - Haosen Wen
  - Zhuojia Shen

- **Undergraduate TAs:**
  - Xuan Tang
  - Liam Heeger
  - Zetian Xiao
  - Qianyi Li
How to contact me?

• **Piazza**

• **Office hour**
  – Tu 12 pm – 1 pm
  – We 11 am – 12 pm
  – Immediately after lecture

• **E-mail:**
  – [tbiswas2@cs.rochester.edu](mailto:tbiswas2@cs.rochester.edu)  (Try to post on Piazza rather than sending email)

• **Appointment (the last resort)**
  • Friday afternoon (after 3 pm)
Work Load

• Heavy!
• Approx. 30 pages of reading per week
• Midterm exam (Mat 07, 2018)
• Final exam (May 06, 2018)
• A semester long project
• 2+ assignments/projects
• Finally, a quiz every week
Grading Policy

• Quiz (20%)
  10+ Quizzes (each worth 2%)
  The lowest quizzes will be dropped

• Exam component (35%)
  – 1 midterm exam (15%)
  – 1 final exam (20%)

• Written and programming assignment component
  – Will not be graded, but a must for quizzes and projects

• Project
  – 3+ projects (45%)

• Class Participation
  – Ask Questions in Class. Take part in Discussion
  – Provide Feedback to improve class
  – A maximum of 3 points will be awarded to top participants (5+ students).
Optional (Extra Credit?)

• **Term Paper**
  – Need to review 1+ conference/journal papers and implement, add features, or survey the related topics.
  – Students need to submit a report based on their results, surveys or findings.
  – Students may need to present posters
  – 8 additional points (making the total grade 108. The score will scaled to 100 for the final letter grade)

**Important:** Term Paper does not guarantee grade improvement.
## Letter Grades

<table>
<thead>
<tr>
<th>Percentage score</th>
<th>Letter grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>90-100</td>
<td>A</td>
</tr>
<tr>
<td>85-89</td>
<td>A-</td>
</tr>
<tr>
<td>80-84</td>
<td>B+</td>
</tr>
<tr>
<td>75-79</td>
<td>B</td>
</tr>
<tr>
<td>70-74</td>
<td>B-</td>
</tr>
<tr>
<td>65-69</td>
<td>C+</td>
</tr>
<tr>
<td>60-64</td>
<td>C</td>
</tr>
<tr>
<td>55-59</td>
<td>C-</td>
</tr>
<tr>
<td>50-54</td>
<td>D</td>
</tr>
<tr>
<td>0-49</td>
<td>F</td>
</tr>
</tbody>
</table>
• Q: I am an undergraduate student. Do I need to worry about the graduate students?
• A: No. The final letter grading would be done separately and will be curved (if required) differently. You are not competing with none except yourself.

• Q: I am not CS major. Do I need to worry?
• A: If you have good hold of CSC 172 (Data Structures), you should be fine.
Q & A

• Q: I am not good with SQL or PHP. Also, more practice sessions would be better.
  
  • A: This semester, TAs will conduct workshops where you ask questions and do more practice. I will provide you problem sets.

• Q: I do not like to read textbook. Is that Ok?
  
  • A: Unfortunately, No. You must study the textbook. I can’t cover everything in lecture and there is no alternative to studying.
Policy

• **Late submission:**
  – No late submission allowed

• **Incompletes & Make-up exams**
  – Not given except in provably extraordinary cases! (see syllabus)
teachers call it cheating, we call it teamwork
Academic Honesty

• On plagiarism:
  – Please don’t cheat
  – I will report the cheating:
    • Instructor Resolution Warning Letter
    • Instructor Resolution with Penalty
    • Board Resolution

• Group study/discussion is encouraged, but the submission must be your own work!

• Programming:
  – Discussions of ideas are welcomed, but no exchange of codes
  – If you use a piece of code from Mr. Google, say so!

• Please refer
  [https://www.rochester.edu/college/honesty/policy.html](https://www.rochester.edu/college/honesty/policy.html)
No Lame Excuses, Please

• I want to go home early, can I take the final early?

• I had a fight with my girlfriend

• I’ve studied very hard, I understood the stuff very well, but I got a C-, please consider giving an A-
How to do well in the course?
How to do well in the course?

study
(verb)

The act of texting, eating and watching TV with an open textbook nearby.
How to do well in the course?
How to do well in the course?

• Participate: discuss, answer, ask questions
  “the only stupid question is the question you don’t ask”

• Give suggestions, please! I’ll take them seriously
• Study every week
• Visit Office Hours
• Start early!
COURSE OVERVIEW
Course website

- or

- Other components:
  - Piazza
  - Blackboard
  - Course Calendar
What you will learn from this course

• Database Systems
  – Efficiently store and retrieve data
  – Relational Databases and SQL
  – Entity-relationship modeling and database design
  – Relational Algebra
  – Web database programming (using PHP)
  – Distributed databases and NOSQL
  – Data Mining and Big Data (Spark)
  – Bonus Learning:
    • Unix: The MySQL server runs on a Unix Machine (betaweb)
    • We will create account and send you the credentials soon.
What you will not learn from this course

• A lot!

• Why?
  – Database Systems is a huge topic
  – Big Data
  – https://www.youtube.com/watch?v=Tzmjbl-i4Y
• **Fundamentals of Database Systems, 7th Edition**
  • (ISBN: 0133970779) by
  • Ramez Elmasri & Shamkant B. Navathe
INTRODUCTION TO DATABASES AND SQL
Databases and Database Users

• Types of Databases and Database Applications
• Basic Definitions
• Typical DBMS Functionality
• Example of a Database
• Main Characteristics of the Database Approach
• Advantages of Using the Database Approach
• Current Database Technology
• Extending Database Capabilities
Types of Databases and Database Applications

• Traditional Applications:
  – Numeric and Textual Databases

• More Recent Applications:
  – Multimedia Databases
  – Geographic Information Systems (GIS)
  – Biological and Genome Databases
  – Data Warehouses
  – Mobile databases
  – Real-time and Active Databases
Recent Developments

- Social Networks
  - Facebook
  - Twitter
  - Linked-In
- Search Engines
  - Google, Bing, Yahoo
- Big Data storage
- NOSQL (Not Only SQL) systems
Basic Definitions

• Database:
  – A collection of related data.

• Data:
  – Known facts that can be recorded and have an implicit meaning.

• Mini-world:
  – Some part of the real world about which data is stored in a database. For example, student grades and transcripts at a university.

• Database Management System (DBMS):
  – A software package/system to facilitate the creation and maintenance of a computerized database.

• Database System:
  – The DBMS software together with the data itself. Sometimes, the applications are also included.
Figure 1.1
A simplified database system environment.
Typical DBMS Functionality

• **Define** a particular database in terms of its data types, structures, and constraints

• **Construct** or Load the initial database contents on a secondary storage medium

• **Manipulating** the database:
  – Retrieval: Querying, generating reports
  – Modification: Insertions, deletions and updates to its content
  – Accessing the database through Web applications

• **Processing and Sharing** by a set of concurrent users and application programs – yet, keeping all data valid and consistent
Example of a Database

• Mini-world for the example:
  – Part of a UNIVERSITY environment.

• Some mini-world *entities*:
  – STUDENTs
  – COURSEs
  – SECTIONs (of COURSEs)
  – (academic) DEPARTMENTs
  – INSTRUCTORs
• Some mini-world *relationships*:
  – SECTIONs *are of specific* COURSEs
  – STUDENTs *take* SECTIONs
  – COURSEs *have prerequisite* COURSEs
  – INSTRUCTORs *teach* SECTIONs
  – COURSEs *are offered by* DEPARTMENTs
  – STUDENTs *major in* DEPARTMENTs

• expressed in a conceptual data model, such as the ENTITY-RELATIONSHIP (ER) data model
## Course Information

### COURSE

<table>
<thead>
<tr>
<th>Course_name</th>
<th>Course_number</th>
<th>Credit_hours</th>
<th>Department</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intro to Computer Science</td>
<td>CS1310</td>
<td>4</td>
<td>CS</td>
</tr>
<tr>
<td>Data Structures</td>
<td>CS3320</td>
<td>4</td>
<td>CS</td>
</tr>
<tr>
<td>Discrete Mathematics</td>
<td>MATH2410</td>
<td>3</td>
<td>MATH</td>
</tr>
<tr>
<td>Database</td>
<td>CS3380</td>
<td>3</td>
<td>CS</td>
</tr>
</tbody>
</table>

### SECTION

<table>
<thead>
<tr>
<th>Section_identifier</th>
<th>Course_number</th>
<th>Semester</th>
<th>Year</th>
<th>Instructor</th>
</tr>
</thead>
<tbody>
<tr>
<td>85</td>
<td>MATH2410</td>
<td>Fall</td>
<td>04</td>
<td>King</td>
</tr>
<tr>
<td>92</td>
<td>CS1310</td>
<td>Fall</td>
<td>04</td>
<td>Anderson</td>
</tr>
<tr>
<td>102</td>
<td>CS3320</td>
<td>Spring</td>
<td>05</td>
<td>Knuth</td>
</tr>
<tr>
<td>112</td>
<td>MATH2410</td>
<td>Fall</td>
<td>05</td>
<td>Chang</td>
</tr>
<tr>
<td>119</td>
<td>CS1310</td>
<td>Fall</td>
<td>05</td>
<td>Anderson</td>
</tr>
<tr>
<td>135</td>
<td>CS3380</td>
<td>Fall</td>
<td>05</td>
<td>Stone</td>
</tr>
</tbody>
</table>

### GRADE REPORT

<table>
<thead>
<tr>
<th>Student_number</th>
<th>Section_identifier</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>17</td>
<td>112</td>
<td>B</td>
</tr>
<tr>
<td>17</td>
<td>119</td>
<td>C</td>
</tr>
<tr>
<td>8</td>
<td>85</td>
<td>A</td>
</tr>
<tr>
<td>8</td>
<td>92</td>
<td>A</td>
</tr>
<tr>
<td>8</td>
<td>102</td>
<td>B</td>
</tr>
<tr>
<td>8</td>
<td>135</td>
<td>A</td>
</tr>
</tbody>
</table>

### Prerequisite

<table>
<thead>
<tr>
<th>Course_number</th>
<th>Prerequisite_number</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS3380</td>
<td>CS3320</td>
</tr>
<tr>
<td>CS3380</td>
<td>MATH2410</td>
</tr>
<tr>
<td>CS3320</td>
<td>CS1310</td>
</tr>
</tbody>
</table>

---

**Figure 1.2**

A database that stores student and course information.
Data on the Web and E-commerce Applications

– Web contains data in HTML (Hypertext markup language) with links among pages.

– This has given rise to a new set of applications and E-commerce is using new standards like XML (eXtended Markup Language) and JSON (JavaScript Object Notation).

– Script programming languages such as PHP and JavaScript allow generation of dynamic Web pages that are partially generated from a database.
Big Data Technologies and NOSQL databases

• **Big Data:**
  – New data storage, management and analysis technology was necessary to deal with the onslaught of data in petabytes a day ($10^{15}$ bytes or 1000 terabytes) in some applications.
  – This started being commonly called as “Big Data”.

• **Hadoop and MapReduce:**
  – Hadoop (which originated from Yahoo) and MapReduce Programming approach to distributed data processing (which originated from Google) as well as the Google file system have given rise to Big Data technologies.
    • We will learn Spark

• **NOSQL:**
  – NOSQL (Not Only SQL- where SQL is the de facto standard language for relational DBMSs) systems have been designed for rapid search and retrieval from documents, processing of huge graphs occurring on social networks, and other forms of unstructured data with flexible models of transaction processing
    • We will learn MongoDB
WHY DATABASE?
Questions to ponder

• Why not Lists? Why database?
• How related tables avoid problems associated with lists?
Problems with Lists

• Multiple Concepts or Themes:
  – Microsoft Excel vs Microsoft Access
• Redundancy
• Anomalies:
  – Deletion anomalies
  – Update anomalies
  – Insertion anomalies

http://www.w3schools.com/sql/trysql.asp?filename=trysql_select_all
List vs Database

- Lists do not provide information about relations!

- Break lists into tables

- Facilitates:
  - Insert
  - Delete
  - Update

- Input and Output interface (Forms and Reports)
- Query!
Structured Query Language (SQL)

- [http://www.w3schools.com/sql/trysql.asp?filename=trysql_select_all](http://www.w3schools.com/sql/trysql.asp?filename=trysql_select_all)
Again, Why database?

• To store data
• To provide structure
• Mechanism for querying, creating, modifying and deleting data.
• CRED (Create, Read, Update, Delete)
• Store information and relationships
Last Words before we end today

• You’ll learn as much from me as I will learn from you
• Give me feedback:
  – Let me know what you like and what you don’t

• Welcome, again!
Acknowledgement

• Some of the slides in this presentation are taken from the slides provided by the authors.
• Thanks to YouTube for providing many useful videos.