CS 4604: Introduction to Database Management Systems

Spring 2013

From the Undergraduate Catalogue Emphasis on introduction of the basic database models, corresponding logical and physical data structures, comparisons of models, logical data design, and data base usage. Terminology, historical evolution, relationships, implementation, data base personnel, future trends, applications, performance considerations, data integrity. Senior standing required. A grade of C or better required in CS prerequisite 2604.

Instructor's Perspective CS 4604 is intended to be a first course in database systems for advanced undergraduates in computer science. It offers students an introduction to the design and programming of database systems. In particular, we will cover the entity-relationship (ER) approach to data modelling, the relational model of database management systems (DBMSs) and the use of query languages such as SQL. We will also cover relational algebra and the use of SQL in a programming environment. We will briefly touch upon query processing and mention the role of transaction management. We will also devote a class or two to current topics of research such as XML databases, data warehousing and data mining. Topics outside the scope of this course are how to implement a DBMS, active database elements, and object oriented DBMSs.

Meeting Times	9:05-9:55am, Mondays, Wednesdays and Fridays, Lavery Hall 330
Instructor	B. Aditya Prakash badityap AT cs DOT vt DOT edu, http://www.cs.vt.edu/~badityap **Please include "CS 4604" in the subject of all email messages.**
Office Hours	10:00am–11:30am, Mondays and Wednesdays at McBryde Hall 122C
Teaching Assistant	Qianzhou Du, qiand12 AT vt DOT edu
Office Hours	10:00am-12:00noon, Tuesdays and Thursdays at McBryde Hall 106
Listserv	CS4604_91913@listserv.vt.edu
Course Web Page	http://courses.cs.vt.edu/~cs4604

If you are unable to attend scheduled office hours and need to meet with us, please send us email to set up an alternative time. If you need any accommodations because of a disability, if you have emergency medical information to share with the instructor, or if you need special arrangements in case the building must be evacuated, please meet the instructor as soon as possible.

Pre-requisites CS 4604 is intended for senior CS majors and minors only. Force-adds will be very difficult for others. The instructor will rigorously enforce the CS 3114 pre-requisite for all students. On the first day of classes, every student must fill out a pre-requisite form, and must return it to the instructor in order to remain enrolled in this class. Remaining enrolled in this course without having the pre-requisite course is an honour code violation.

Approach The course achieves a balance between establishing a theoretical foundation for DBMS and pragmatic applications of DBMS in a real world environment. The course places particular emphasis on the logical design of relational database systems. A significant semester-long project reinforces lectures. In this project, students will apply concepts presented in class and obtain practical, hands-on experience. Since students are more likely to use rather than implement database systems, this introductory course covers logical aspects of database design and implementation rather than the construction of database management systems.

Textbook and References The required textbook is "*A First Course in Database Systems*," by Jeffrey D. Ullman and Jennifer Widom, Prentice Hall, Third Edition, 2008. **Make sure to get the third edition of the textbook!** The authors maintain a webpage for the book with errata lists and solutions to selected exercises. Your local bookstore has copies for purchase. The instructor and TA will maintain other online resources on the course webpage.

Keeping in Touch We will use Piazza for class discussions—please use it actively for exchanging ideas and doubts. The TA in particular will also check Piazza fairly regularly for any unanswered questions. I will update the homepage with the appropriate Piazza link soon. The listserv mailing-list should be used sparingly. Since it is created automatically by a central university system, any student registered in CS 4604 will be added to the mailing list. If you do not receive a test mail from the instructor by the end of the first week of classes, ensure that your email address is properly recorded in the university system (the system only records @vt.edu addresses, not, for instance, @myfavoriteisp.com). Finally, we will be using the VT Scholar system *only* for raw scores—so that each of you can keep track of your own scores privately.

Evaluation There will be 5–6 homeworks, a midterm exam, a final exam, and a semester-long group project. Both exams are closed book and closed notes. Projects, performed by groups of 2–3 students, will involve designing and programming a real-world database application from scratch. Student groups will do assignments related to their projects at regular intervals and turn in these assignments. The first part of the project consists of modelling a database in relational terms, loading data into a real DBMS, and writing interesting SQL queries. The second part consists of re-designing the database from scratch using E/R diagrams and redundancy-removal techniques, and evaluating the differences between the initial ad-hoc design and the later more formal design. The third part of the project involves embedding SQL queries into host language programmes, and creating feature-rich web-based interfaces to the database. Homeworks account for 30%, the project for 30% (Project Assignments 1-5 are worth 4% each, Project Assignment 6 is worth 10%), the midterm exam for 15%, and the final exam for 25% of the grade.

The homeworks expand on material taught in class, teach you some design and modelling, and will improve your vocabulary. Many of them will be in tune with an appropriate step of your database project. The project will help you to use the concepts learnt in realistic situations and encourages the use of programming skills in a realistic computing environment.

Questions and Requests

- **Clarification questions** about assignments should go to the TA, or on Piazza.
- **Regrading requests** for any assignments or exams should done in writing, via e-mail to the TA and the instructor, at the latest 96 hours after graded exams or assignments are distributed by the instructor or the TA.

Late Policy

- The assignment due date is the one posted on the web. All assignments are due at the beginning of the lecture, on the due date.

- **Slip Days:** To accommodate for coinciding deadlines you may have from other courses, or personal unforeseen events such as sickness, each person has an automatic extension of **4 calendar days**—no questions asked. You can use the extension on any assignment(s) remaining during the semester. For instance, you can hand in one assignment 4 days late, or each of four assignments 1 day late.

  - Late days are rounded up to the nearest integer. For example, a submission that is 4 hours late will count as one day late.
  - When you hand in a late assignment, you must identify at the top of the assignment, (i) how late this assignment is, and (ii) how much of the total slip time you have left.
  - After you have used up your slip time, any assignment handed in late will be marked off 25% per day. That is, after 4 days, the grade will be zero.

- **Extreme** circumstances, like medical emergencies, etc.: In such cases, additional, no-penalty extensions will be granted. Contact the instructor, with written documentation, like doctor's note.

Workload The course will appear deceptively simple but history shows that it is possible to get a variety of grades. The course also moves at a fast pace and the workload is generally high! The instructor assumes that you will relish a design-oriented course and have the initiative to learn about various aspects of database design and programming. If you would like a career involving database technology (i.e., how to implement databases, not just use them effectively), you should also take a more advanced course in database system implementation or pursue a graduate program in this area. As mentioned earlier, this course does not cover these aspects.

Syllabus Here is a weekly breakdown of the schedule for the course. The schedule on the class website lists required reading for each class. This schedule is subject to change.

- Week 1: Course logistics, Introduction to Database Systems (Chapter 1)

- Weeks 2–5: SQL, Relational Algebra, Data Definition (Chapters 2.1–2.5, 5.1–5.2, 6 and 7)

- Weeks 6–7: Data Modelling using E/R diagrams (Chapter 4.1–4.4)

- Week 8: Midterm review, Midterm examination

- Weeks 9–12: Design theory for the Relational Model (Chapter 4.5–4.6, 3)

- Week 13: Programming with SQL

- Week 14: Students' choice (XML, Data mining, etc.)

- Week 15: Final review, Final examination

Electronic Accounts You are expected to have an account in the undergraduate Linux lab in the first floor of McBryde Hall that will give you access to a PostgreSQL server. The server will run on the machine `cs4604.cs.vt.edu`. You will be able to access the server from any machine in the Linux lab. If you require Linux lab accounts, please contact the lab administrators. We will provide accounts for you on the database server at the appropriate time in the semester. You may use your personal computer for project assignments and for testing homework solutions. However, we cannot guarantee that we will be able to grade work (especially the first few steps and the last step of the project) done at home (on other versions of software or because of file transfer problems).

Programming Languages We assume that you are familiar with programming languages such as C, C++, and Java and scripting languages such as Perl and PHP. We may cover rudiments of PHP in class. The course does not teach these languages but you will be required to use them for some of your project assignments (particularly toward the end of the course). If you have never used these languages before, we expect that you would either (i) obtain the necessary expertise outside the classroom or (ii) come and talk to the instructor about it (in which case, you will most probably be advised to drop this course). We strongly believe this background is essential to fully appreciate this course and that you should have acquired such a background by the end of the junior year.

Honour Code The honour code is in effect for every aspect of this class. You are expected to do your own work. No one may give you answers to homeworks or exams. You are allowed to work on project assignments with the members of your group. In other words, students are encouraged to communicate about general principles of the course, but *all assigned homework must be done on an individual basis.* The instructor and the TA are available to provide any assistance that you may need. You may not exchange any code or solutions, either in pieces or in entirety, by any electronic means or hard copy.

Things to do on the First Day Sign the pre-requisite sheet and return it to the instructor. This step is compulsory and IMPORTANT!

More information about CS 4604

- CS 4604 is not easy. There is typically no correlation between the number of years of experience with ORACLE/MySQL/substitute-your-favourite-db-here and your grade in CS 4604.
- Homework problems are often tricky and difficult. Many of them will not involve straightforward applications of concepts taught in class but will require you to apply these concepts in creative ways.
- For most of the homework problems, there is more than one correct solution. Therefore, solution sketches posted by the instructor cannot cover all possible answers. Keep in mind that many questions are designed to address sources of learning/design difficulties.
- Since you are students at or above the senior level, you are expected to demonstrate initiative in learning the various software systems needed for this course. A one-semester course such as this cannot cover all nuances, nor should you as computer science students get bogged down by the details of different implementations.