600.315/415 - DATABASES
Fall 2013 Syllabus

Instructor: Prof. David Yarowsky
Hackerman 324
410-516-5372
yarowsky@gmail.com

TAs: Josh Wheeler
Shaffer 200
301-830-1533
600.415.l0.pdf

Meeting Time: Tu,Th: 3:00-4:15 PM

Classroom: Hackerman B17

Office Hours: Instructor - We 3-4, Tuesday/Thursday after class and by appointment.
TA - TBA, special review sections, and by appointment.

Required Textbook:

- A. Silberschatz, H. Korth and S. Sudarshan, *Database System Concepts*, 6th Edition, McGraw Hill, 2010, ISBN: 978-0-07-352332-3 or 5th edition, 2005, ISBN: 0-07-295886-3.

600.315/415 will explore both formal and practical issues in databases. Hands-on database design and implementation using the MySQL DBMS will be an important component of the course.

Course Requirements: Final grades will be based on the following (subject to change):

Class Participation:	5%
Homeworks (4):	25%
Midterm:	20%
Final Exam:	25%
Final Project:	25%

Lateness Policy:

One homework assignment may be handed in up to 5 days late without penalty. No other late homeworks will be accepted. Final projects handed in late will receive a penalty of 10% for every day late.

600.315 vs. 600.415:

600.315/415 will be share common lectures. They will differ primarily in terms of assignments and grading. Homeworks in 600.415 will include 1 or more additional problems and the final project will include additional component(s) not required for 600.315. Exams will differ somewhat and will be graded on a different scale. Nevertheless, 600.415 should be manageable by advanced undergraduates and upperclass students are encouraged to enroll.

600.315/415 vs. 600.316/416:

Databases (315/415, Fall) and *Database Systems* (316/416, Spring) are complementary courses and make a natural course sequence (see below). 315/415 focuses on how to design and use a database; formal database models, theory and foundations; database programming languages, especially SQL and PL/SQL; object-oriented and XML-based data models and future directions (including data mining and natural language interfaces). The final project will be application-focused (e.g. how to design an implement a database for a novel task) including practical execution of the concepts studied in the class. In contrast, 316/416 will focus on database internals and systems, including query and join processing, indexing, file organization, estimation and optimization, as well as database architectures, streaming and partitioning. The course project(s) will focus on database system internals and their development.

Can I take 316/416 as a stand-alone course without 315 or 415?

Yes, 316/416 does not have 315 or 415 as a formal prerequisite. You should have some database experience before taking 316/416, however, either through prior employment or via a prior course.

Graduate students who have prior database employment experience or have taken a prior course in database systems are normally expected to begin directly with 416.

Anyone with a research focus in the databases area should certainly begin directly with 416.

Can I take 315/316 or 415/416 as a 2-course sequence?

Yes. There will be modest overlap of material (10but taught via different perspectives and emphasis, and will serve as a good refresher.

If you have not taken a prior course in databases and are interested in both the theory/applications and systems sides of the field, then this sequence makes a lot of sense and is encouraged.

The instructors will work to make this a natural 2-course sequence.

However, if you have already had a prior course in databases, or intend to continue in database systems research, then you are strongly encouraged to take 316/416 and then another advanced follow-on course in database systems, transaction processing and/or storage systems taught by Professors Ahmad or Burns.

Can I take 315/416 as a sequence?

Yes, 416 does not require 415 as a prerequisite, but you should have done well in 315 and be prepared to do some background catchup to meet the expectations of the 416 instructor.

Can I take 415/316 as a sequence?

Yes, if you are an undergraduate and would like to continue focusing on database systems and database systems internals but a less difficult level, then this sequence could make sense.

Final Projects:

Students will be able to select final projects of interest to them from a fairly diverse set of options. Details will be provided in class. Students may work in teams of 1 or 2 people. A project proposal will be due in early November, including a detailed system specification and design. The final project submission, including a full database implementation in MySQL, will be due shortly after the end of classes in December. For most projects, students will be required to populate and test their implemented database design with substantial quantities of *real world data* extracted from the world wide web or other online sources.

Computer Science Academic Integrity Code:

Academic honesty is required in all work you submit to be graded. You must solve all homework and programming assignments entirely on your own, unless group work is specified in writing. This means you must not show your program code, problem solutions, or work to other students. However, you may discuss assignment specifications with others in the class to be sure you understand what is required by the assignment. If you use fragments of source code from sources other than your text (such as on-line resources), you must put a reference to that effect in your homework submission. Falsifying program output or results is prohibited. Please see your professor if there are any questions about what is permissible. Students who cheat will suffer a serious course grade penalty in addition to being reported to university officials. You must abide by JHU's Ethics Code, available at http://jhunix.hcf.jhu.edu/~ethicsbd.

Preliminary Class Schedule (subject to change):

Date	Topic	KS 6e	KS 5e	KS 4e	KS 3e
Tu. 9/3	Introduction	–	–	–	–
Th. 9/5	Overview of databases and data modeling	1	1	1	1
Tu. 9/10	Entity-Relationship data model	7	6	2	2
Th. 9/12	Database design principles	2	6	2	2
Tu. 9/17	Relational data model	3	2	3	3
Th. 9/19	Relational algebra	6	2	3	3
Tu. 9/24	Relational algebra and relational calculus	6	2,5	3	3
Th. 9/26	SQL	3	3	4	4
Tu. 10/1	SQL (continued)	4	4	4	4
Th. 10/3	Advanced SQL	5	4	4	4
Tu. 10/8	QBE (Query by Example), Views	C1	5	5	5
Th. 10/10	Relational database administration and additional topics	hnd	hnd	hnd	hnd
Tu. 10/15	Fall Break (no class)				
Th. 10/17	Relational database design, integrity constraints	8	7	7	6
Tu. 10/22	Relational database design, normalization	8	7	7	7
Th. 10/24	MIDTERM (tentative date)				
Tu. 10/29	Query processing and optimization	12-13	14	14	12
Th. 10/31	Embedded SQL; PL-SQL/stored procedures	5	hnd	hnd	hnd
Tu. 11/5	Application design and development	9	8		
Th. 11/7	Transactions and database recovery	14,16	17-19	17-19	15-17
Tu. 11/12	Distributed databases	19	22	19	18
Th. 11/14	Database security	9	8	6	19
Tu. 11/19	Object-oriented databases	22	9	8,9	8,9
Th. 11/21	WWW-based technologies/interfaces; XML data model	21	10	22	21
Tu. 11/26	Special Topics	hnd	hnd	hnd	hnd
Th. 11/28	THANKSGIVING				
Tu. 12/3	Data warehousing, data mining, multimedia databases	25-26	18,24	22,23	21
Th. 12/5	Natural language interfaces and future directions	hnd	hnd	hnd	hnd
Th. 12/12	**Final Examination** [Date to be confirmed] 2-5 PM				

KS = Korth and Silberschatz

Students are responsible for determining if they have an scheduled exam conflict with another course during the officially assigned exam slot for this class. In this very unlikely event, given that this is exclusively the official slot for TuTh3PM classes, students should notify both instructors as soon as possible, and no later than 12/03/13, so the scheduling conflict can be resolved.