5DV119 Introduction to Database Management Spring 2015

1 Course Staff

Instructor in charge: (kursansvarig)	Name:	Stephen J. Hegner
	Office:	C444 MIT-huset
	Telephone:	090 786 5760
	E-mail:	`hegner@cs.umu.se`
	URL:	`http://www.cs.umu.se/~hegner`
	Office hours:	after lectures and by appointment
Instructor:	Name:	Thomas Johansson
	Office:	B413 MIT-huset
	Telephone:	090 786 6259
	E-mail:	`thomasj@cs.umu.se`
	URL:	`http://www.cs.umu.se/~thomasj`
	Office hours:	`http://www8.cs.umu.se/~thomasj/schedule.html`
Instructor:	Name:	Johan Mollevik
	Office:	B444 MIT-huset
	Telephone:	090 786 6701
	E-mail:	`mollevik@cs.umu.se`
	URL:	`http://www.cs.umu.se/~mollevik`
	Office hours:	`http://www8.cs.umu.se/~mollevik/schedule.html`

- Both Thomas Johansson and Johan Mollevik will be available to answer general questions about the course material during their scheduled office hours.
- Questions regarding the two obligatory exercises on SQL, and the software exercise on ODBC, should be directed to Johan Mollevik.
- Questions regarding the other obligatory exercises (on the relational algebra and calculus, normalization, and ER design), should be directed to Thomas Johansson.
- Stephen J. Hegner has no assigned consultation hours for this course. Thus, in the first instance, Thomas Johansson and Johan Mollevik should be approached for help with the course material. Nevertheless, Stephen J. Hegner will be available after lectures and by appointment (limited by travel commitments) to answer questions about the lectures and the course material in general. He cannot answer detailed questions about what is or is not acceptable as an answer to an obligatory exercise; such questions must be directed to the instructor in charge of grading the exercise.

2 Course Language

All lectures will be given in English. Questions may be asked during the lectures in either English or Swedish. However, as there are students in the class who do not speak Swedish, questions posed in Swedish will have to be translated by the instructor into English. It is therefore preferable to ask questions in English whenever possible.

Written work may be submitted in either English or Swedish. However, if written work is submitted in Swedish, then *all* members of submitting group must be able to speak Swedish well enough to discuss the solutions in that language.

The questions on final examination will be written in English; answers may be in either English or Swedish. For the final examination, it will be permitted to use an XX-English / English-XX dictionary, where XX is a natural language of the student's choice.

3 Course Literature

The main textbook for this offering of the course is the following.

- Ramez Elmasri and Shamkant B. Navathe, *Database Systems: Models, Languages, Design, and Application Programming*, Sixth Edition, Addison-Wesley, 2011; ISBN: 978-0-13-214498-8.

The sixth edition of this book is also available in hardcover, with the title *Fundamentals of Database Systems*, ISBN: 978-0-13-608620-8. The two books are slightly different. Mostly, chapters (with identical content) are named differently. In a few cases, material is rearranged. Each also contains some advanced material not found in the other, but for all topics covered in this course, there is no substantial difference. Thus, for this course, either will do. The hardcover version is of course much more expensive.

The sixth edition of this book has been re-organized substantially from earlier ones, and since the course will follow the new organization, use of earlier editions is not recommended.

The "Online Access" materials for the "Companion Website" for this book will not be used in the course, so it is fine to buy a used book without an access code for these materials. (New books come with a six-month prepaid subscription.)

The supplementary textbook for the course is the following.

- Alan Beaulieu, *Learning SQL*, Second Edition, O'Reilly, 2009; ISBN: 978-0-596-52083-0 (paper).

This book covers the language SQL much more thoroughly than does the above textbook, and since a substantial part of the course will deal with SQL, the acquisition of this book is recommended.

In addition to the course textbooks, there will be relatively detailed overhead slides. These materials will be available for download on the course Web page.

4 Course Content and Outline

The official kursplan is available on this link. A more offering-specific outline is shown below. The numbers shown in the single rectangular brackets (i.e., [..]) identify chapters and sections in the sixth softcover edition of the textbook. The numbers in angle brackets ⟨..⟩ indicate the approximate number of 45-minute lecture periods which will be devoted to the topic.

- Reasonably detailed overhead slides will be available for many topics. The authoritative source for relevant (i.e., possible examination) material is the course lectures and these slides. In many cases, material not covered in the textbook may nonetheless be covered in lecture presentations.
- The number of 45-minute lecture "hours" to be devoted to each topic is approximate, and in particular is rounded to the nearest integer. Adjustments will be made as the course progresses, and so the table below should not be used a definitive guide to which topics will be covered on which days.
- The lectures will consist of a mixture of formal presentations using slides, demonstrations of the use of software, and solution of exercises from the book and other sources. The slides are *not* a self-contained written record of all that will be covered during the class meetings. Students are responsible for all material which is covered in the course, regardless of whether or not it is found in the slides.

1. Introduction [1, 2] ⟨1⟩
2. The Relational Model and Its Query Languages
   - 2.1 Overview of the Relational Model [3] ⟨2⟩
   - 2.2 A Comprehensive Introduction to SQL [4,5] (+ Beaulieu) ⟨6⟩
   - 2.3 The Relational Algebra and Calculus [6] ⟨4⟩
3. Normalization of Relational Schemata
   - 3.1 Principles of Normalization [14.1-14.5] ⟨2⟩
   - 3.2 Formal Properties and Algorithms for Normalization [15.1-15.4] ⟨4⟩
4. Conceptual Modelling and Database Design
   - 4.1 Entity-Relationship (ER) Modelling [7] ⟨2⟩
   - 4.2 Relational Schema Design using ER modelling [8.1] ⟨1⟩
5. Database Access

5.1 An Overview of Programming-Language-Based Access to Databases [12.1-12.2] ⟨2⟩

5.2 Programming-Language-Based Database Access Using ODBC [12.3] ⟨2⟩

6. Database Security [25.1-25.5] ⟨2⟩
7. Review ⟨2⟩

5 Course Materials Outline

5.1 Textbook Materials Outline for the Softcover Edition

The following is a list of those chapters and sections of the main textbook which will be covered in the course. For each chapter or section, a symbol is given which indicates the nature of coverage in the course. The meaning of these symbols is provided in the table below.

✔	Material will be covered in the course.
✣	Material will be covered partially or selectively.

Note the following:

- If an entire chapter is covered, no section-by-section breakdown is given.
- Entries have not been provided for sections entitled "Summary" or the like.
- In general, omitted items will not be covered. However, the possibility that some covered material may appear in an omitted chapter or section remains. In all cases, the lectures and course notes should be taken to be the definitive statement for the course material.

1. Introduction to Databases ✔
2. Overview of Database Languages and Architectures ✔
3. The Basic (Flat) Relational Model Constraints ✔
4. SQL: Data Definition, Constraints, and Basic Queries and Updates ✔
5. SQL: Advanced Queries, Assertions, Triggers, and Views ✔
6. Formal Relational Languages: The Algebra and Calculus ✔
7. Conceptual Modeling Using Entities and Relationships

   7.1 Using High-Level Conceptual Data Modeling for Database Design ✔

   7.2 A Sample Database Application ✔

- 7.3 Entity Types, Entity Sets, Attributes, and Keys ✔
- 7.4 Relationship Types, Relationship Sets, Roles, and Structural Constraints ✔
- 7.5 Weak Entity Types ✔
- 7.6 Refining the ER Design for the COMPANY Database ✔
- 7.7 ER Diagrams, Naming Conventions, and Design Issues ✔
- 7.8 Relationship Types of Degree Higher than Two ✔

8. Mapping a Conceptual Design into a Logical Design

- 8.1 Relational Database Design Using ER-to-Relational Mapping ✔

12. SQL Application Programming Using C and Java

- 12.1 Database Programming: Issues and Techniques ✔
- 12.2 Embedded SQL, Dynamic SQL, and SQLJ ✣
- 12.3 Database Programming with Function Calls: SQL/CLI and JDBC ✣(SQL/CLI only)

14. Database Design Theory: Introduction to Normalization Using Functional and Multivalued Dependencies

- 14.1 Informal Design Guidelines for Relation Schemas ✔
- 14.2 Functional Dependencies ✔
- 14.3 Normal Forms Based Upon Primary Keys ✔
- 14.4 General Definitions of Second and Third Normal Forms ✔
- 14.5 Boyce-Codd Normal form ✔

15. Database Design Theory: Normalization Algorithms Dependencies

- 15.1 Further Topics in Functional Dependencies: Inference Rules, Equivalence, and Minimal Cover ✔
- 15.2 Properties of Relational Decompositions ✔
- 15.3 Algorithms for Relational Database Schema Design ✔
- 15.4 About Nulls, Dangling Tuples, and Alternative Relational Designs ✔

25. Introduction to Database Security

- 25.1 Introduction to Database Security Issues ✔
- 25.2 Discretionary Access Control Based Upon Granting and Revoking Privileges ✔

March 05, 2015

25.3 Mandatory Access Control and Role-Based Access Control for Multilevel Security ✔

25.4 SQL Injection ✔

25.5 Introduction to Statistical Database Security Relational Designs ✔

5.2 Textbook Materials Outline for the Hardcover Edition

The hardcover edition is organized a bit differently, but it includes all of the material covered in the course. Specific information is given below, using the same notation as for the description above for the softcover edition.

1. Databases and Database Users ✔
2. Database System Concepts and Architecture ✔
3. The Relational Data Model and Relational Database Constraints ✔
4. Basic SQL ✔
5. More SQL: Complex Queries, Triggers, Views, and Schema Modification ✔
6. The Relational Algebra and Relational Calculus ✔
7. Data Modeling Using the Entity-Relationship (ER) Model ✔
8. Relational Database Design by ER- and EER-to-Relational Mapping

   8.1 Relational Database Design Using ER-to-Relational Mapping ✔

13 Introduction to SQL Programming Techniques

   13.1 Database Programming: Issues and Techniques ✔

   13.2 Embedded SQL, Dynamic SQL, and SQLJ ✣

   13.3 Database Programming with Function Calls: SQL/CLI and JDBC ✣(SQL/CLI only)

15. Database Design Theory and Normalization

   15.1 Informal Design Guidelines for Relation Schemas ✔

   15.2 Functional Dependencies ✔

   15.3 Normal Forms Based Upon Primary Keys ✔

   15.4 General Definitions of Second and Third Normal Forms ✔

15.5 Boyce-Codd Normal form ✔

16. Relational Database Design Algorithms and Further Dependencies

    16.1 Further Topics in Functional Dependencies: Inference Rules, Equivalence, and Minimal Cover ✔

    16.2 Properties of Relational Decompositions ✔

    16.3 Algorithms for Relational Database Schema Design ✔

    16.4 About Nulls, Dangling Tuples, and Alternative Relational Designs ✔

24. Database Security

    24.1 Introduction to Database Security Issues ✔

    24.2 Discretionary Access Control Based Upon Granting and Revoking Privileges ✔

    24.3 Mandatory Access Control and Role-Based Access Control for Multilevel Security ✔

    24.4 SQL Injection ✔

    24.5 Introduction to Statistical Database Security Relational Designs ✔

5.3 Online Materials

The Web site for the course is located at

`http://www.cs.umu.se/kurser/5DV119/V15/index.html`.

The following materials may be found on these pages.

1. This syllabus, in both PDF and HTML.
2. The lecture slides for the course.
3. Descriptions of the obligatory exercises.
4. Information on the database systems *PostgreSQL*, *MySQL*, and *Leap*.
5. Information on using *Python*.
6. Sample programs and other aids for *ODBC*.
7. Miscellaneous links to database-related things.
8. Some official documents required by the Department of Computing Science.

6 Laboratory Schedule and Computer Resources

There is no official laboratory booking for the course, nor any in-laboratory instruction. In general, when not reserved by a course, the computer laboratories of the department are open for use by students for their coursework.

The relational database systems *PostgreSQL* and *MySQL* will be used in this course, both as stand-alone query processors and via interfaces to the programming languages *C* and/or *Python* via *ODBC*. In addition, the relational-algebra query system *Leap* may be used in some assignments.

Time permitting, access to a relational database via a Web-based interface using *PHP* will also be considered.

7 Course Schedule

The table below identifies the course meeting times and places, together with the nature of the meeting. The key "L" denotes a lecture, "R" a review session, and "E" an examination booking.

For each lecture, the topics to be covered are identified via the outline header number of Section 4 of this syllabus. So, for example, on January 26 the topics of 2.2 (SQL) will be covered. This is only an approximate assignment of meeting times to topics, and it may be altered as the course progresses.

Room BiA201 is located in the Biologihuset (the Biology Building). Rooms whose name begins with S are located in Samhällsvetarhuset (the Social Sciences Building), Rooms whose name begin with H are located in Humanisthuset (the Humanities Building). "Hörsal" means "lecture hall".

The examination rooms (skrivsalarna) are in the building known as Östra paviljongen.

Week	Day	Type	Date	Time	Room	Topics	Lecturer
04	Mo	L	Jan 19	0815-0955	BiA201	1, 2.1	Hegner
04	We	L	Jan 21	0815-0955	S Hörsal C	2.1,2,2	Hegner
04	Th	L	Jan 22	0815-0955	S Hörsal C	2.2	Hegner
05	Mo	L	Jan 26	0815-0955	H Hörsal F	2.2	Hegner
05	We	L	Jan 28	0815-0955	H Hörsal E	2.2,2.3	Hegner
05	Th	L	Jan 29	0815-0955	S Hörsal C	2.3	Hegner
06	Mo	L	Feb 02	0815-0955	S Hörsal C	2.3,3,1	Hegner
06	Th	L	Feb 05	0815-0955	S Hörsal C	3.1,3.2	Hegner
07	Mo	L	Feb 09	0815-0955	H Hörsal F	3.2	Hegner
07	Th	L	Feb 12	0815-0955	H Hörsal F	3.2,4.1	Hegner
08	Mo	L	Feb 16	0815-0955	S Hörsal C	4.1,4.2	Hegner
08	Th	L	Feb 19	0815-0955	H Hörsal F	5.1	Hegner
09	Mo	L	Feb 23	0815-0955	S Hörsal C	5.2	Hegner
09	Th	L	Feb 26	0815-0955	H Hörsal F	6	Hegner
11	Th	R	Mar 12	0815-0955	H Hörsal F	7	TBA
12	We	E	Mar 18	0900-1500	Skrivsal 3	Final examination	
24	Mo	E	Jun 08	0900-1500	Skrivsal 5	Final examination	
35	Tu	E	Aug 25	0900-1500	Skrivsal 7	Final examination	

The lectures will run 0815-0900 and 0910-0955. The reason for the five-minute shift and shorter break than is traditional is to avoid collisions with courses from other faculties, which often begin on the hour (hh00).

8 Prerequisites

The formal requirements are listed in the course plan, which may be found at the following link. They include the following.

1. A knowledge of programming in one of the languages *C* or *Python* in the *Unix/Linux* environment.

2. A thorough knowledge of data structures and algorithms, as presented in the course *Datastrukturer och algoritmer* (Data Structures and Algorithms).

3. A knowledge of mathematics. Formally, any combination of mathematics courses worth at least 22.5 ECTS is acceptable, but some knowledge of propositional logic would prove particularly helpful.

9 Grading System

This course has two parts (*moment* in Swedish), a conceptual part (*teoridelen* in Swedish) and an exercise part (*laborationsmoment* in Swedish).

The only possible grades for the exercise part are S (Satisfactory; G=*Godkänd* in Swedish) and U (Unsatisfactory, *Underkänd* in Swedish). The grade on this part will be determined entirely by five smaller obligatory exercises and one larger obligatory project exercise. Each will be graded as S (Satisfactory) or U (Unsatisfactory). To obtain the grade of S for the exercise part of the course, it is both necessary and sufficient to obtain the grade of S on all six obligatory submissions. In addition, for each obligatory exercise, it will be possible to earn a maximum of 30 quality points, and for the obligatory project, it will be possible to earn a maximum of 50 quality points. These points will be assigned based upon the overall quality and correctness of the work. Thus, a maximum of 200 points may be earned on the six obligatory submissions.

The examination will have a total of 1000 points.

The final point total F for the course is computed as

$$F = \max(E, \quad 0.8 \times E + L)$$

with E the number points earned on the examination and L the number of points earned on the obligatory exercises. The final grade on the conceptual part of the course is computed as follows.

Number of points	Grade
$F \geq 800$	5 (med beröm godkänd – excellent)
$650 \leq F < 800$	4 (icke utan beröm godkänd – very good)
$500 \leq F < 650$	3 (godkänd – satisfactory)
$F < 500$	U (underkänd – unsatisfactory)

In addition, to pass the course, a minimum of 500 points on the examination is necessary, regardless of how many points are earned on the exercises. Thus, exercise points can only be used to increase the grade from 3 to 4, or from 4 to 5. They cannot be used to rescue a performance of less than 50% on the examination.

10 Obligatory Work

10.1 Weekly Obligatory Work

The course includes five obligatory written and short computer exercises. The rules governing these submissions are as follows.

- Points will be assigned to the submitted solutions for each exercises. These points will be based upon the quality of the solutions, as well as their timeliness.

- Exercises will furthermore be graded as satisfactory or unsatisfactory. To receive a passing grade in the course, the student must have submitted a satisfactory solution to each exercise.
- Each exercise will have a due date. For each working day or fraction thereof that the submission is late, five points will be subtracted from the grade. (The grade may never be less than zero, of course.)
- Exercises marked as unsatisfactory may be resubmitted, in order that a satisfactory evaluation be obtained. However, resubmitted work will never receive any additional points.
- Each exercise will be worth 30 points.

10.2 Obligatory Programming Project

In addition to the weekly obligatory exercises, there will be one larger programming project, involving interfacing to a relational database using CLI/ODBC (Call-Level Interface / Open Database Connectivity) and either the C or else the Python programming language. The rules governing submission are as follows.

- The project will be given up to 50 quality points, as well as marked as satisfactory or unsatisfactory. To receive a passing grade in the course, a student must have submitted a satisfactory solution for the project.
- Submissions marked as unsatisfactory may be resubmitted, in order that a satisfactory evaluation be obtained. However, resubmitted work will never receive any additional points.
- For each working day or fraction thereof that the submission is late, five points will be subtracted from the grade. (The grade may never be less than zero, of course.)

10.3 General Remarks on the Obligatory Work

- The obligatory exercises and project may be completed in groups, and collaboration is permitted on the software exercises, roughly as described in the documents *Riktlinjer vid labgenomförande (Policy for Obligatory Exercises)* and *Hederskodex (Honor Code)*. More details will be provided later, when the descriptions of these exercises are distributed.
- The written exercises, as well as the programming project, may be submitted individually or in groups. However, once a solution is submitted, only those named on the submission will receive credit for it. Partners in solution may not be added after the initial submission

10.4 Deadlines for Submission of Obligatory Work

- The deadlines for exercise submissions are as follows.

Exercise	Deadline	Deadline Type
Exercise 1	February 02, 2015 at 08:00	Initial
Exercise 2	February 09, 2015 at 08:00	Initial
Exercise 3	February 16, 2015 at 08:00	Initial
Exercise 4	February 23, 2015 at 08:00	Initial
Exercise 5	March 02, 2015 at 08:00	Initial
ODBC Project	March 09, 2015 at 08:00	Initial
All	May 04, 2015 at 08:00	Second
All	June 08, 2015 at 08:00	Final

- The initial deadlline for each exercise is the time at which points begin to be deducted for lateness.
- An exercise solution which is submitted in advance of its initial deadline will be graded in a timely fashion, and students will be given the opportunity to improve unsatisfactory submissions and to resubmit them for an additional evaluation.
- For each obligatory exercise, there will be a resubmission deadline, defined to be five working days after the graded initial submission is returned. The resubmissions will be graded within ten working days of the resubmission deadline. This applies only to resubmissions of initial submissions which were received no more than four days after the deadline. For initial submissions which are more than four days late, there are no guarantees.
- Re-grading of submissions, other than for first resubmissions which are received no more than four working days after the initial deadline, is entirely at the discretion of the grader. The only exception is that all remaining submission will be graded within ten working days of the resubmission dates of May 4 and June 8.
- With the exception of extenuating circumstances (which will generally require written documentation), no submissions will be accepted after June 08, 2015.

10.5 Obligatory Work Completed in Previous Years

- Credit for individual exercises may not be carried over from previous years. A student who does not already have a satisfactory grade recorded for the exercise part of the course must complete all requirements for that part as defined by this offering of the course.
- Quality point from previous years may not be carried over to the current year. Students who have already completed the laboratory exercises for previous years may however re-submit solutions for the current year in order to obtain quality points.

11 The Course Mailing List

Important information about the course will be sent by the instructors to students via the course mailing list. While this list is available via a Web interface (link on the course home page; see Sec. 5.3), the individual messages are also sent to all students, at their `xxx@cs.umu.se` accounts. Since some information may be timely, it is critical that students read e-mail at that account. If the `xxx@cs.umu.se` account is not checked regularly, then forwarding to another account should be enabled.

This mailing list is configured automatically. Neither the course staff nor the support group can arrange for a another e-mail address to be used instead. Students must arrange forwarding if it is preferred to receive these messages at another e-mail address.