Database Design and Development

GLIS 657

Winter 2017

Quick Links

Instructor: Prof. Charles-Antoine Julien <charles.julien@mcgill.ca >
Teaching Assistant: Jesse David Dinneen < jesse.dinneen@mail.mcgill.ca >
Class times: Wednesdays, 11:30 am - 2:30 pm
Location: EDU 539
Labs : EDU 341; during 2nd half of class
Office hours : Email for appointment or ask during labs.
Duration: Jan. 11th - Apr. 5th, 2017

Description

Theoretical and applied principles of relational database design. Includes relational theory, conceptual design, database normalization, relational database management systems, SQL queries and database management.

Learning Outcomes

- Acquire knowledge of terminology and principles of relational database design
- Understand and apply the concepts of database development
- Design and create a small-scale non-bibliographic database

Course schedule

Week	Date	Title	Readings
1	Jan 11	Introduction Database basics	• Frost (2006), ch. 1 • Riordan (2005), preface
2	Jan 18	Relational Theory • Entity Relationship Diagram (ERD) • Primary and foreign keys • Relationships	• Frost (2006), ch. 2 • Riordan (2005), ch. 1
3	Jan 25	Conceptual Database Design • Multiple table databases • Lab 1: Creating tables and relationships in Access	• Frost (2006), ch. 3 • Riordan (2005), ch. 3
4	Feb 1	Database Design • Problem statement and solutions • Lab 2: Single-table queries in SQL and Access	• Frost (2006), ch. 8, 9 • Riordan (2005), part I, ch. 5
5	Feb 8	**In class quiz (20%)**	• Review of Frost (2006), ch. 1, 2, 3, 8, and 9 • Review of Riordan, ch. 1, 3 and 5.

6	Feb 15	Advanced database design • Recursive relationships • Hierarchies • Lab 3: Implementing advanced design in Access	Frost(2006), chapter 5
7	Feb 22	Database normalization • 1NF, 2NF, 3NF, 4 NF, BCNF, 5NF • Lab 4: Complex queries in Access **Assignment (10%) due at 11:59pm**	• Frost (2006), ch. 4 • Riordan, part I, ch. 2
8	Mar 1	Study Break	
9	Mar 8	Designing the data entry and query interfaces Lab 5: Creating forms, subforms and reports	Riordan, part IV, ch. 15, 16.
10	Mar 15	**In class quiz (20%)**	
11	Mar 22	Database System Interface Design Lab 6: User interface Switch board in Access	
12	Mar 29	Database Maintenance Lab 7: User interface in Access continued	Review of Riordan (2005), 15, 16.
13	Apr 5	**Student Presentations (10%)** **All teams submit slides before 6 pm the day before class**	
14	Apr 12	**Final Project (40%) due at 11:59pm**	

Textbooks

The following textbooks are all freely available from McGill libraries.

- Frost, R., Day, J. and Van Slyke, C. (2006). Database design and development. A visual approach. Pearson Prentice Hall, Upper Saddle River: NJ. (On reserve) http://mcgill.worldcat.org/oclc/60419446
- Riordan, R. (2005). Designing effective database systems. Addison-Wesley. Upper Saddle River, NJ. (Available online at McGill) http://mcgill.worldcat.org/oclc/56672048
- (Other option you might like)
  Hernandez, Michael J. (2013). Database Design for Mere Mortals: A Hands-On Guide to Relational Database Design. (3rd Ed.)

Software

This course requires extensive use of Microsoft Access database development software, which is part of the Microsoft Office Pro for Windows.

We have found no equivalent software that allows database development with little or no coding.

MS Access is freely available to McGill students; however, it only exists on Windows PCs. See http://kb.mcgill.ca/kb/?ArticleId=5172&source=article&c=12&cid=2#tab:homeTab:crumb:7:artId:5172 for installation details.

Unfortunately, there is no Mac version of Microsoft Access.

This should not be a significant issue since over 90% of the world's computers are Windows PCs (see http://en.wikipedia.org/wiki/Usage_share_of_operating_systems#Market_share_by_category) however, it is clear that McGill students' choice of operating systems does not reflect this trend. I encourage Mac users to suggest a database development software that works on their machines.

Assessments

Assignment (10%, in pairs)

Describe your database in terms of the following parts:

1. a clear statement of the purpose, objectives and end-users of the database (this usually requires no more than a page of textual description).

2. produce the normalized Entity-Relationship Diagram (ERD) by following the five steps shown in the class and course material. You can use any diagraming software (e.g., Visio, PowerPoint, etc.) or draw it by hand and scan.

Evaluation criteria

- The assignment must be packaged in a single well formatted document in PDF format. No other formats are acceptable.

- Be sure to follow submitting rules stated below.

- Grade = 100 -
    - up to 10 if content or purpose unclear
    - up to 10 if problem statement not self-contained
    - 5 per missing table
    - 3 per wrong relationship cardinality or missing key
    - 2 per wrong key
    - 1 per wrong or unspecified data type

Two Quizzes (2 × 20%, individual)

Two in–class 90 min quizzes worth 20% each based on the lectures, labs and readings.

- Quiz 1
    - Relational theory
    - Conceptual database design
    - Simple queries

- Quiz 2
    - Complex Queries
    - Normalization
    - Advanced database design (recursions and hierarchies)

All documentation is allowed up to 4 double sided 8.5 by 11 in. paper sheets. You can use both sides of the paper for total of 8 pages of notes. Notes can be in any format (e.g., printed, handwritten, etc.).

MS Access Labs (Deduction if not handed-in)

In order to acquire the MS Access skills required to carry out the final project each student must perform and hand-in at least 6/7 labs (see schedule). Lab requirements and instructions will be posted in myCourses.

Attendance at the labs is not mandatory but **you must submit the labs in myCourses** the night before the following week's class at 11:59pm.

Failure to submit the required number of labs will result in a **25% deduction** for the final project. Note that it is your responsibility to back up your work on the network and/on on external devices (USB key or external hard drive).

The lab exercises will not be graded. Answers will be posted and the TA and instructor will be available for questions.

Note that the labs become gradually more difficult and will require more time: do not get lured into a false sense of security by assuming the 1st lab's difficulty level is representative of the rest.

Final Project (50%, pairs)

The final project is comprised of the following:

- A usable, small-scale non-bibliographic database created in MSAccess (40/50 points). This is usually based on the design proposed in assignment 1.
- **Note:** Usable means that you database can be easily used and appreciated by your users with minimal training.

Requirements

- Approximately 12 tables (including at least 4 associative tables)
- Approximately 30 records in one of the primary tables
- Approximately 20 records in the other primary tables
- Approximately 10 records in the associative tables
- All tables must have a form to enter/edit data.
- At least 2 forms to enter data in a parent and child table (i.e., form/subform)
- At least 2 reports to retrieve data from more than one linked tables (i.e., parent and child tables). These reports must make use of groupings.
- A user manual, if you believe it is necessary. Assume users do not read help or user manuals. As much as possible, the interface should be self-evident and require no additional information to learn how to use.
- Be sure to follow submitting rules stated below.

Evaluation Guidelines

Read and follow these guidelines when developing your final database:

- Do not use wizards. You may think they are helping you but magic is an illusion therefore wizards are charlatans. Don't be fooled!

- Do not use wizards. Use of wizards is easily spotted. We do not provide technical help when the issue resolves around the use of a wizard.
- Resist the urge to forge ahead without stabilizing constituent parts.
  Make sure the following are correct and stable before moving on to the next broad task:
    1. Make tables and stabilize ERD. To test, enter a 2-3 records in each table. You should **never again** enter data directly into the tables.
       Do not assume you are done at this point! This is only about 20% of the total work for the project.
    2. Create minimal forms and subforms. Test the forms by entering a few records. By "minimal" we mean that you should not spend much time on placement and colors.
       Remember that one form should be based on a single table (or single table query), a subform is also based on a single table (or single table query); together a form/subform make it seem like a form uses more than one table but each is based on a single table.
    3. Build a switchboard to provide user access to your forms. Set the switchboard to show automatically when the database is launched. Resist the urge to have multiple buttons leading to the same form. Less is more.
    4. Refine the forms' placements, colors and overall usability. As a final testing, enter the required number of records using the forms. Note that you get no credit for creating records but the exercise serves to thoroughly test the forms; you get points for delievering well functionning forms.
    5. Once all data entry and edition is stable, create your reports. Make sure you develop reports based on multi-table queries that make use of groupings. Provide access to the reports on the switchboard.
    6. At this point you are likely to have secured 80% of the points. Refinements to your forms and reports will secure the last 20%.

Presentation to the class (10/50 points)

Imagine you are presenting your database to the client. Make sure a non-expert in databases can understand the solution, what it will do for them, and how it solves their current issues.

In no more than 8 minutes, all group members will present the result of their project (i.e., each person on the team has to present). This should include:

1. Tell the audience who they are; in other words, describe the client or users that are meant to use this database.

2. The problem statement

3. the entity-relationship diagram, including recursive relationships and generalization/specialization hierarchies if appropriate to model your problem statement. Careful when showing your ERD; make sure the audience can adequately read your diagram. A slide is of no use if the audience only sees fussy boxes.

4. the fully-normalized database using the interface for entering and retrieving data from the database with sample data (at least one report and form)

5. Clearly state how your solution will add value (e.g., time saved, reduced errors, etc.). How does your solution address the issues of the current situation? What were the issues, show the improvement with your solution

The presentation must include a **PowerPoint slide deck** that has to be submitted to myCourses the day before. For the sake of efficiency, *no other type of slide deck format is accepted*.

You will present the submitted slide deck. No changes or additional files will be permitted between the submission and the presentation.

Be sure to follow submitting rules stated below.

Evaluation

The exact time limit of the presentations will depend on the number of projects: it is announced in the weeks prior to the presentation. Aim for 8±2 minutes.

Exceeding the given time limit will result in an automatic 10% deduction.

The following grid will guide the evaluation of the presentation and database:

	Excellent (A)	Good (B)	Not Satisfactory (F)
Organization of ideas	The presentation is clearly and logically structured. Ideas are well articulated and communicated. Reader can follow line of reasoning. Details are minimized so that main points stand out.	The presentation is clearly and logically structured. Ideas are articulated. A few minor points may be confusing.	Organization is not evident. Lack of connection between ideas. Arguments are not clear. Reader can follow only with effort.
Writing and visuals	Language is coherent and precise. Slides are clear, uncluttered, and easy to read. Bullets are used effectively to convey the message. Aesthetically slides are pleasing.	Language is clear and precise. Slides are a little cluttered with too much writing, but fairly easy to understand. Aesthetically slides are simple and plain.	Language is vague. Slides are cluttered, difficult to follow, containing whole paragraphs. The use of colours and fonts are distracting and do not enhance the presentation.
Database design	The problem statement is motivated and clearly modeled. All tables and relationships are well justified. All tables contain records. The database is fully normalized. The user interface is usable. At least one form and one report work.	The problem statement is well defined. With a few exceptions, tables and relationships are well justified. Most tables contain records. The user interface is usable. Few violations of normal forms.	The problem statement is poorly defined. Missing associative tables. Violations of normal forms. Not meeting the requirements in terms of number of table. The user interface is not usable.

The following presentation guidelines should be followed:

- The instructor will sit at the back of the class to evaluate the presenters. Make sure those at the back can read your slides and understand your voice.
- You must practice your presentation orally as a group to ensure timing and transitions.
- You must make short bullet points that summarize an idea you will talk about.
- You cannot strictly paste a screen shot of a complex diagram into your slides as this will make it illegible and overload the slide. Make smaller zoom-in

images to talk about portions.

- If you show it, make sure we can see/read it. Careful with small font or diagrams.
- If you show it, talk about it. No point showing content that is not presented orally.
- Present an outline at the start of the presentation to set the expectations of the audience.
- Project your voice with energy! Talk to the audience as opposed to your slides or paper.
- If you show your actual design you can receive early feedback.

Submitting your assignments

- All assignments must be submitted electronically using the Assignments tool in myCourses. **Email submissions are not accepted.**
- Submissions that contain **multiple files must be packaged in a single zip file**. No other formats are acceptable (e.g., RAR, multiple files).
  Multi-file submissions not packaged in a single zip file will be deducted by 20%.
  Single file submissions can be submitted as is without zipping.
- **All diagrams, tables, etc. must be inserted in your document**: they cannot be submitted as separate files (.xls, .pdf, etc).
- Late assignments will not be accepted, and will receive a grade of zero (0). Extensions are only granted in the most exceptional of circumstances, and they must be requested at least one week before the assignment is due.

General Information

Academic Integrity

McGill University values [academic integrity](#). Therefore, all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Council and Disciplinary Procedures.

Academic Policies

Additional policies governing academic issues which affect students can be found in the [McGill Charter of Students' Rights.](#)

Language of Instruction

All lectures, discussions, and course materials are in English. However, students have the right to submit their work in French.

Students with disabilities

If you have a disability please consult the [Office for Students with Disabilities](#)

Aknowledgements

This syllabus is largely based on the work of Prof. Catherine Guastavino.