DATABASE SYSTEMS

HOMEWORKS
CS2102

STUDENT'S GUIDE FOR TUTORIALS, ONLINE ASSIGNMENTS AND LABORATORIES, AND PROJECT

LAST UPDATED 26/12/2013

STÉPHANE BRESSAN

STEPH@NUS.EDU.SG

INTRODUCTION

CONTENT

This document contains several problems, exercises and questions for students following CS2102. If needed, this document will be updated during the semester. Please check that you are always using the latest available version.

TUTORIALS

Every week, during which you have a tutorial, you will be informed of problem and list of questions to prepare (questions from the list in the corresponding Problem section of this document).

As time is limited, the tutorial will focus on a representative subset of the questions. It is very important that you prepare your tutorials in order to be able to indicate to the lecturer the difficulties that you may have encountered.

During the tutorial, some of you will be volunteered to write up and publish onto the forum answers to some of the questions. The volunteers are expected to publish the answers within the next two days.

LABORATORY

SQL laboratories are tutorials conducted in front of a computer. The SQL queries, answers to the tutorial questions, can immediately be tried with a Database Management System.

Every week, during which you have an SQL laboratory, you will be informed of the list of questions to prepare (questions from the list in the Laboratory section of this document).

Check the lesson plan regarding the location of laboratories.

As time is limited, the tutorial will focus on a representative subset of the questions. It is very important that you prepare your laboratories in order to be able to indicate to the lecturer the difficulties that you may have encountered.

GRADIANCE

You will be given a series of online home assignments. We use the Gradiance online system (http://www.newgradiance.com).

These home assignments are either Homework (multiple choice questions) or Lab Projects (English queries to be translated into SQL). Gradiance is able to automatically correct your answers for both Homeworks and Lab Projects. Gradiance homeworks and lab projects can be taken several times while they are open.

If it can, the system gives you some hints on how to improve your SQL query by showing you the result of your query and the results that you should have obtained. It is not possible to 'hard-code' a query because Gradiance uses a secret database to double check your SQL.

PROJECT

Finally, you will be working on a project. In this project you will develop a simple database application. The project will give you an opportunity to use the school's software and infrastructure, to familiarize you with technologies and to apply what you are learning in this module.

The project is done in groups of 5 students. There are no constraints of the composition of groups (can be different tutorial, different option).

LABORATORY

SQL

CASE: NUNSTA ONLINE BOOK EXCHANGE

Students at the National University of Ngendipura (NUN) buy books for their studies. They also lend and borrow books from other students. Your company, Apasaja Pt. Ltd., is commissioned by NUN Students Association (NUNStA) to design and implement an online book exchange system for its students.

Apasaja Pt. Ltd. designs and implements a database application that records information about students, books they own and books they borrow from other students.

The database records the name, faculty, and department and student number of each student. Each student is identified in the system by its email. The database also records the date at which the student joined the university.

The database records the title, authors, publisher, year and edition and the ISBN-10 and ISBN-13 for each book. The International Standard Book Number, ISBN-10 or -13, is an industry standard for the unique identification of books. It is possible that the database records books that are not owned by any students (because the owners of a copy graduated or because the book was advised by a lecturer for a course but not yet purchased by any student.)

The database records the date at which a book copy is borrowed and the date at which it is returned. We will refer to this information as a loan record.

For historical purposes the database records information about the copies and the owners of copies as long as the owners are students or there are loan records concerning the copies.

For historical purposes the database records information about graduated students as long as there are loan records concerning books that they owned.

For historical purposes (in order to keep the loan records for this book) the database records the case of a book that was owned and subsequently sold unless the copy was never borrowed.

LABORATORY QUESTIONS

Without creating the schema using the code in the file NUNStASchema.sql, without populating the database with the code in the files NUMNStACopy.sql, NUMNStALoan.sql and NUMNStAStudent.sql, and without looking at these files, write the following queries in SQL.

Lab 1: Starting up

Without creating the schema using the code in the file NUNStASchema.sql, without populating the database with the code in the files NUMNStACopy.sql, NUMNStALoan.sql and NUMNStAStudent.sql, and without looking at these files, answer the following queries in SQL.

Question 1. Create a table book that contains information about books (title, format, and number of pages, authors, publisher, year, edition, ISBN-10 and 13). Find one book on the Web, for instance on amazon.com, to see some examples of the values of these attributes. Search the Web for available SQL domains (types).

Question 2. Delete the table book.

Question 3. Create a table book that contains information about books (title, format (paperback or hardcover), number of pages, authors, publisher, year, edition, ISBN-10 and ISBN-13). Choose a primary key. Forbid NULL values for the title and ISBN-10 and 13, make sure the format is either "paperback" or "hardcover".

Question 4. Insert one book called "Introduction to Database Systems". Go to the Web to find actual details.

Question 5. Insert half a dozen books with title containing "Introduction to Database Systems" or authored by C.J. Date. Go to the Web, for instance amazon.com, to find the details of such books.

Question 6. Find all the information about all books.

Question 7. Modify all books authored by C.J. Date to mention the author's first name (find the author's first name from the Web.)

Question 8. Delete all books authored by C.J. Date.

Question 9. Find the title, format, number of pages, authors, publisher, year, edition, ISBN-10 and ISBN-13 of the books.

Question 10. Find the titles of the books.

Question 11. Find the authors of the books called "Introduction to Database Systems".

Question 12. Add a language attribute to all books. Set the default language to English.

Question 13. Delete the table book. Create the tables for the remainder of this tutorial and populate them using the SQL code provided to you for the SQL tutorials. Create the tables using the file NUNStASchema.sql. Populate the tables using the files NUMNStABook.sql, NUMNStACopy.sql, NUMNStALoan.sql and NUMNStAStudent.sql. You can clean up data and tables anytime using the file NUNStAClean.sql. The files are available in IVLE Workbin.

Lab 2: queries, duplicates, referential integrity, multiple relations and conditions and complex conditions, UNION.

Always ask yourself the question about duplicates potentially created by the query with or without DISTINCT. Only use DISTINCT when strictly necessary.

Question 1. Find the emails of students.
Question 2. Find the different emails of students.
Question 3. Print the names of students in descending alphabetical order.
Question 4. Are there students with the same name?
Question 5. Find the different names of students. Is the result sorted? Look at the execution plan.
Question 6. Find the names of students who owned a copy of book '978-0262033848'.
Question 7. Find the names of students who owned a copy of book with more than 100 pages whose title contains the word 'Computer'.
Question 8. Find the number of A4 pages needed to photocopy the two books with ISBN-13 '978-0262033848' and '978-0321295354' (2 pages of a book can be copied on one A4 page).
Question 9. Find the different names of students who owned a copy of a book other than of '978-0262033848'.
Question 10. Find the names of students who borrowed a copy of book '978-0262033848'.
Question 11. Find the names of students who owned or borrowed a copy of book '978-0262033848'. Use UNION.
Question 12. Find the names of students who owned or borrowed a copy of book '978-0262033848'. USE OR.
Question 13. Delete all the data in table loan.
Question 14. Try again the last two queries.

Lab 3: Aggregate functions, nested queries, views and triggers.

Question 1. Find the total number of copies

Question 2. Find, for each book, the corresponding number of copies. [Print the primary key of the book and the number of copies.]

Question 3. Find the available books with the largest number of copies.

Question 4. Find the names of the students who have borrowed some book by 'Charles Dickens.

Question 5. Find the number of different books by 'Charles Dickens.

Question 6. Find the names of the different students who have borrowed all the books by 'Charles Dickens'. Use aggregate functions.

Question 7. Find the names of students who owned a copy of a book with more than 100 pages whose title contains the word 'Computer'. Use nested queries. This is not the preferred answer.

Question 8. Find the different names of students who never owned a copy of a book other than of book '978-0262033848'.

Question 9. Find the names of the different students who have borrowed all the books by 'Charles Dickens'. Use NOT EXISTS. (You may also try with NOT IN and EXCEPT.)

Question 10. Find the names of the different students who have borrowed all the books by 'Amelie Nothomb'.

Question 11. Create and query views for the copies and loans for which the owner is a Computer Science student.

Question 12. We could not cascade the update and deletion of book borrowers to the loan table. Create a trigger that propagates the update of a student's email to the loan.

PROBLEM I

MODELING WITH THE ENTITY-RELATIONSHIP MODEL AND THE RELATIONAL MODEL

CASE

The Varsity International Network of Oenology (VINO) whishes to computerize the management of the information about its members as well as to record the information they gather about various wines. The organization is big enough so that there are several members with the same name. A card with a unique number is issued to identify each drinker. The contact address of each member is also recorded for the mailing of announcements and calls for meetings.

At most once a week, VINO organizes a tasting session. At each session, the attending members taste several bottles. Each member records for each bottle his or her evaluation of the quality (very good, good, average, mediocre, bad, very bad) of each wine that she or he tastes. The evaluation may differ for the same wine from one drinker to another. Actual quality and therefore evaluation also varies from one to another bottle of a given wine. Every bottle that is opened during the tasting session is finished during that session.

Each wine is identified by its name ("Parade D'Amour"), appellation ("Bordeaux") and vintage (1990). Other information of interest about the wine is the degree of alcohol (11.5), where and by whom it has been bottled ("Mis en Bouteille par Amblard-Larolphie Negociant-Eleveur a Saint Andrede Cubzac (Gironde) - France"), the certification of its *appellation* if available ("Appellation Bordeaux Controlee"), and the country it comes from (Produce of "France").

Generally, there are or have been several bottles of the same wine in the cellar. For each wine, the bottles in the wine cellar of VINO are numbered. For instance the cellar has 20 bottles numbered 1 to 20 of a Semillon from 1996 named Rumbalara. For documentation purposes VINO may also want to record wines for which it does not own bottles.

The bottles are either available in the cellar, or they have been tasted (and drunk...).

TUTORIAL QUESTIONS

1. Design an entity-relationship schema that most correctly and most completely captures the constraints expressed in the above description of the VINO application. Draw the corresponding entity-relationship diagram. Do not omit to declare the necessary integrity constraints. Indicate in English the constraints that cannot be captured, if any.
   Follow the following steps. At each step, justify your choice by quoting the sentences in the text that support it.
   a. Identify entity sets .
   b. Identify relationship sets and link them to the entity sets they relate.
   c. Indicate attributes of entity and relationship sets.
   d. Indicate the combination of attributes that form keys.
   e. Indicate the participation constraints.
2. Translate your entity-relationship diagram into a relational schema. Give the SQL DDL statements to create this schema. Do not omit to declare the necessary integrity constraints. Indicate in English the constraints that cannot be captured, if any.

PROBLEM II

QUERY LANGUAGES: CALCULUS, ALGEBRA AND SQL

CASE

Let us consider a database application for a pizza company. The company manages several stores in different parts of Singapore.

Pizzas have a code identifying them, a name, and a size in inches. A store has a name. The database records the location of the store (area) and the telephone number of the store. Different stores may sell the same pizzas at different prices.

The schema of the database is as follow:

PIZZA (code, name, size)
STORE (name, area, phone)
SELLS (store_name, code , price)

TUTORIAL QUESTIONS

Write the following queries in SQL, relational algebra, t-uple relational calculus and domain relational calculus. For each of the following queries find as many equivalent queries in in SQL, relational algebra, t-uple relational calculus and domain relational calculus. In particular find all the possible SQL queries for question 7.

1. Find the names of pizzas that come in a 10 inch size.
2. Find the names of pizzas that come in a 10 inch or a 12 inch size.
3. Find the names of pizzas that come in both a 10 inch and a 12 inch size.
4. Find the pairs of different codes of pizzas with the same name and same size (is there any?)
5. Find the names and phone numbers of the stores in "College Park" or "Greenbelt" that sell a 10 inch pizza named "pepperoni" for less than $8.
6. Find the codes of the most expensive pizzas – assume that the schema of the database is reduced to a relation pizza(code, price) for the sake of simplicity.
7. Find the names of the stores that sell all pizzas.

PROBLEM III

FUNCTIONAL DEPENDENCIES AND NORMALIZATION

TUTORIAL QUESTIONS

1. Is the following rule correct?

$\forall X \in R\ \forall Y \in R$ (if $X \rightarrow Y$ then $Y \subseteq X$)

2. The following rule is called Pseudo-transitivity.

$\forall X \in R\ \forall Y \in R\ \forall Z \in R\ \forall V \in R$ (if $X \rightarrow Y$ and $Z \rightarrow V$ and $Z \subseteq Y$, then $X \rightarrow V$)

   a. Prove it using the Armstrong axioms.
   b. Argue that if we replace transitivity with pseudo-transitivity in the Armstrong's axioms we still have a set of axioms that is complete.

3. Consider the set of functional dependencies
   $F=\{ \{A\} \rightarrow \{B\}, \{C\} \rightarrow \{D\}, \{B,D\} \rightarrow \{E\}, \{D\} \rightarrow \{A,D\}, \{A,C\} \rightarrow \{E,B\} \}$ on the relation scheme $R= \{A,B,C,D,E\}$.

   a. Give an example instance of R that complies with the functional dependencies.
   b. Give an example instance of R that violates the functional dependencies.
   c. Compute F+ the closure of F.
   d. Give an example of a trivial functional dependency in F+
   e. Give an example of a completely non trivial functional dependency in F+
   f. Give an example of a non completely non trivial and non trivial functional dependency in F+
   g. Compute $\{C\}+$ the closure of the set of attributes $\{C\}$.
   h. Compute a minimal cover of F

4. Invent three examples consisting of a relation scheme R and a set of integrity constraints on R such that, respectively,

   a. R is not in 2NF;
   b. R is not in 3NF but in 2NF;
   c. R is not in BCNF but in 3NF.

5. Using the algorithms we have learned, decompose your examples in the previous question

   a. in BCNF;
   b. is the BCNF dependency preserving;
   c. or, if not possible, it in 3NF .

PROJECT

WEB DATABASE APPLICATION

OBJECTIVES

The objective of this project is to familiarize you with database technologies, to give you an opportunity to use the school's available software and infrastructure, and to apply the concepts and techniques learned in class for the design and programming of a database application.

The evaluation of the project will consider the scope of concepts and techniques used and their relevance. For instance, you should try and appropriately use the simple and advanced SQL constructs that you have learned: simple queries, aggregate queries, integrity constraints, views, etc. Feel free to extend the application requirements and add features in order to demonstrate interesting use of the technology learned. Consequently, the amount of data in the database should be sufficient for a complete and realistic demonstration of the system.

TECHNOLOGY

Figure 1 is an overview of the architecture of your application. The architecture consists of a Web server, a server page language and, of course, a relational database management system.

The DBMS and possibly the Web Server can be located on the Cloud.

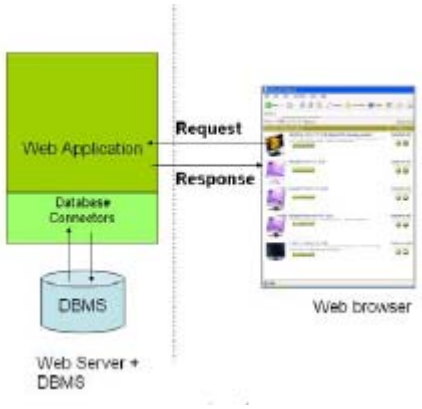

Figure 1: General Architecture

DELIVERABLES

The project deliverables are:

1. A demonstration of your software
2. A brief Report (with all group members' names and matric numbers on the front page) containing:
    a. Indication of the web server, server page language and database management system used (indicate whether you use the zone or else)
    b. The ER diagram for the application
    c. The Relational Schema (in SQL DDL code)
    d. Chosen sample and representative SQL code (indicate the SQL code and the function of the service it helps to implement)
    e. 2 or 3 representative screen dumps of the Web interface

TOPICS

The manager of your company, ThinkCan Pte Ltd, a Singaporean software house, asked your group to design and implement the prototype of an online (Web based) system. The prototype should be realistic in order to convince a major customer to commission your company to develop the system, but should also illustrate the use of relational database technology in order to serve as an in-house showcase application for engineers in your company.
It is left to your creativity to design a realistic model for the description of the elements and ancillary information in the system. The design can be kept simple but should be sufficiently rich to allow the meaningful demonstration of SQL and DBMS features. Similarly, you should also populate the database with sufficiently enough data to both make the demonstration realistic and to illustrate the use of interesting SQL and DBMS features.

The three standard topics are:
Topic A, Social network: the system is a catalog of user profiles and interest groups. Users can search the catalog. They can create new groups and join or leave existing groups. They can create, modify and delete their own entries (see http://www.friendster.com/ , http://groups.yahoo.com , etc. for examples and data).

Topic B, Music catalogue: the system is a catalogue of albums and songs. Users can search the catalog by title, composer, artist, genre etc. Registered users can buy albums and songs. Administrators can create, modify and delete all entries. (see iTunes for examples and data.)

Topic C, Online classified: the system is an online classified portal. Users can search the classified by content, category, dates etc. Registered users can post new classified. Administrators can create, modify and delete all entries. (see http://www.stclassifieds.sg for examples and data).

In order to meet the minimum requirements, the system should (1) allow browsing and searching of entries (basic and advanced search features) and ancillary data, (2) should allow creation, deletion, and modification of entries and ancillary data. Simple login and sessions may be required by the topic but advanced access control or shopping basket is not required. Extra credit will be gained by appropriate usage and demonstration of advanced SQL and DBMS features (advanced SQL queries such as aggregates, nested queries, views, non standard integrity constraints, triggers, and stored procedures).

You may propose your own topic by submitting a one page description of the application to Stephane Bressan (steph@nus.edu.sg) for approval.

CS210

ACADEMIC YEAR 2013-2014 SEMESTER II

OVERVIEW

The dates and figures in this section are given tentatively for your information. They may be updated or modified during the semester. The latest information is available in IVLE (announcement and lesson plan).

The table below indicates the weightage of each component of the assessment.

CS2102	
Final Exam	60%
Home assignments	9%
Midterm	20%
Project	11%

MIDTERM TEST

The midterm test is one hour long. It covers the introduction, the design of database applications with the entity-relationship model and with the relational model, the mapping of entity-relationship diagrams to relational schemas, and SQL DDL (creation of tables with constraints, views, etc.) and DML (insertion, deletion, queries and updates). It is a mixture of multiple choice and essay questions. It is closed book. Exact date, time and location will be announced in the lesson plan and announcements.

TUTORIALS

Tutorial attendance is taken. If you miss or are late to tutorial two times, one point will be deducted from your general mark per subsequent absence or late arrival.

If you are officially excused (official or medical reason), bring the certificate to the undergraduate office and inform your tutor as soon as possible.

You may occasionally attend a different tutorial than the one you are registered to. If you do so, inform your tutor as early as possible.

GRADIANCE

To allow more flexibility, to help you manage your time and effort, and to help you with your revisions, we use the Gradiance system for online Homework and Lab Projects (http://www.newgradiance.com/cguw). The exact opening dates and deadlines will be announced in the lesson plan and announcements.

Registration and Login

You need to sign-up to Gradiance at http://www.newgradiance.com/cguw with:

- **your NUSNET id as user-id** (if you use another user-id, we will not be able to account for your registration and for your home assignments),
- and a password of ten or more characters and digits.
- Also enter your first name, last name, and SoC email address to facilitate identification and communication.

ATTENTION: Invalid accounts will be dropped from the system and owners will fail all online assignments.

Access and Documentation

You access the Homework and Lab Projects of the module using the Class Token **31EC6F19.**

You will find the documentation of the Gradiance system at http://www.newgradiance.com under Services>User Guides/Manuals.

Contact Mr Tang Ruiming <tangruiming@nus.edu.sg>, if you encounter technical problems with the system.

You will be informed of the opening dates and deadlines for online assignments with Gradiance (Homeworks and Lab Projects) in IVLE.

Lab Projects

After selecting the Class in your Current Class Portfolio, click on the Laboratory Projects menu (left frame) and start by clicking on the assignment you have be given. You will be asked to write queries in SQL.

Your answers are executed against the laboratory database as well as against a secret database. The system can automatically determine if your answer is correct or not. Sometimes, but not always, it can give you feedback on a wrong answer in order to help you get the correct answer.

You may come back and retake the homework as many times as you want until the deadline.

Your score for the assignment will be the last score obtained at the last try submitted. The system keeps previous correct answers.

Homeworks

After selecting the Class in your Current Class Portfolio, click on the Homework menu (left frame) and start by clicking on the assignment you have be given. You will be asked to answer several questions.

You may come back and retake the homework as many times as you want until the deadline. However, you need to wait 1 minute between two attempts.

Attention: every time you retake the homework, you need to answer all questions. Although the questions don't change, the choices for the answer may change!

Your score for each assignment is the last score obtained at the last trial submitted. The system does not keep previous correct answers and scores.

The marking scheme for each Gradiance assignment is as follows.

Assignment not submitted or not correct: 0

Assignment partially (less than half of the questions) completed and correct: 1

Assignment partially (more than half of the questions) completed and correct: 2

Assignment completed and fully correct: 3

There is strictly no deadline extension for Gradiance assignments.

Training

One lab Project (Kings) and one Homework (Homewok 0) are kept for training purpose only (they do not count for continuous assessment mark).

LABORATORY

Tutorials that are conducted in the laboratory (SQL tutorials in week 3, 4 and 5) take place in Programming Lab.

Check IVLE and CORS for exact date, time and location.

ORACLE

You will receive your userid and password from your tutor or by email.
You can find information about using Oracle at http://www.comp.nus.edu.sg/~oradoc
You can find some notes on the differences between Oracle SQL and standard SQL at http://infolab.stanford.edu/~ullman/fcdb/oracle/or-nonstandard.html
You can download and install on your personal computer Oracle 11g Express and Oracle SQL Developer after registering at http://www.oracle.com/technetwork/products/express-edition/downloads/index.html

PROJECT

Project groups are formed of 5 members.

Check IVLE Lesson plan for the following deadlines:

- group registration in IVLE (Project tool – choose any empty group and register your members. DO NOT REGISTER A TEAM WITH ONLY ONE, TWO or THREE MEMBERS!). After the deadline the remaining students will be randomly assigned to new groups or to groups with less than

5 members. There is no other constraint on the composition of groups: group members need not belong to the same tutorial group.

- Project start. At this date your group will have been assigned a topic.

- Project report submission deadline. Put a soft copy in PDF of the report in the IVLE Project Report Workbin (it will open approximately one week before the submission deadline). The file should be called GroupXX.pdf, where XX is the group number. If you miss the deadline, put a soft copy in PDF of the report in the IVLE Project Report Late Submission Workbin.

The Project evaluation (demonstration) will take place during tutorial hours in week 13. Each group is given 10 minutes to demonstrate the main features of its system and to answer questions. Your group will need to register to one of the available timeslots (how to register will be announced later). It is not necessary that the entire group be present for the demonstration.

In case of outstanding issues within the group (e.g. member not participating) contact the lecturer as soon as possible to find a consensus. If the issue persists, use the Group Member Evaluation (inform the lecturer). Contacting the lecturer for such a case in the last two weeks is too late.

Groups may also use their SoC zone account (a Unix account with web facilities). Do apply for your own zone account. Do not indicate the course lecturer as supervisor in your application. In the zone you can use Apache Tomcat with php or JSP and Oracle (recommended), Sybase, MYSQL, or DB2.

If you use the zone, teaching assistants and lecturers will provide basic technical supports concerning the database management system and active server page language. However, it is expected that you start by looking at the many tutorials and examples that are readily available on the World Wide Web.

Zone setup, Tomcat installation and basic examples of php code to access Oracle can be found at http://www.comp.nus.edu.sg/~oradoc/common/

If you prefer not to use the zone, feel free to approach the teaching assistant in charge and propose to use another combination of available technology on any machine you have access to. It could be any combination of Apache or Microsoft IIS with php, ASP, ASP.net or JSP (of, course it must be supported by the web server that you have chosen), and SQL Azure, DB2, Sybase, Oracle, MySQL or SQLite. We cannot provide technical support if you do not use the zone with the recommended technology.

The project marking scheme is as follows.

Project report: 4% (criteria: architecture, ER and DDL design, SQL samples, and screen shots).

Project demonstration: 6% (criteria: basic application and interface, SQL, look and feel).

Bonus for outstanding projects: 1%.

FINAL EXAM

The final exam is two hours long. It covers all topics in the syllabus. It is a mixture of multiple choice and essay questions. It is closed book. Exact date, time and location will be announced by the School.