School of Computing and Information Technology
University of Wollongong

Session: **Autumn 2017**
Lecturer: **Janusz R. Getta**
Tianbing Xia

CSIT115/CSIT815 Data Management and Security
Assignment 2
27 March 2017

Scope

This assignment consists of the tasks related to the logical design of relational database, implementation of relational tables, and modification of the structures of relational database.

The outcomes of the assignment are due by **Monday, 24 April, 2017, 7.00 pm sharp.**

This assignment contributes to 6% (5% for CSIT815) of the total evaluation in the subject.

A submission procedure is explained at the end of assignment specification.

This assignment consists of 4 tasks and specification of each task starts from a new page.

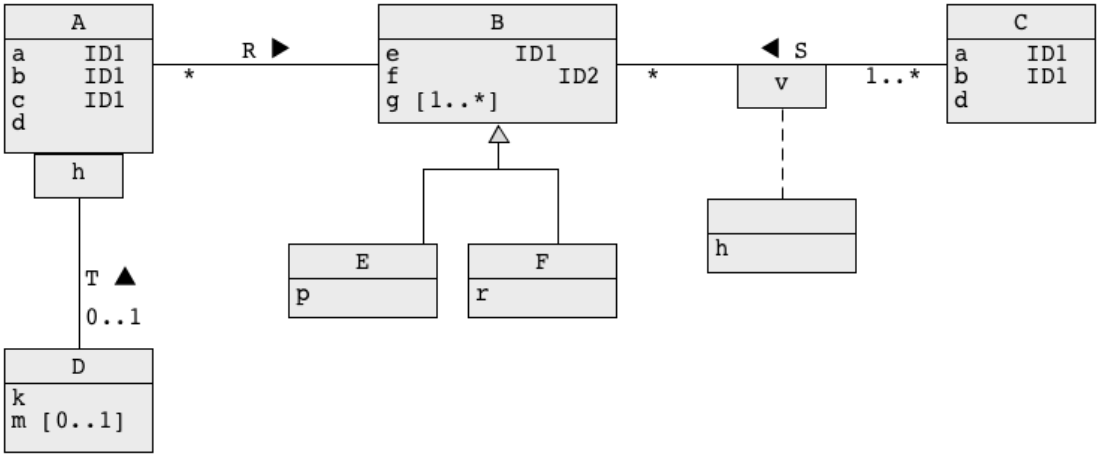
Prologue

Download and unzip a file `assignment2-all-files.zip`. You should get the files `Assignment2.pdf`, `dbcreate3.sql`, `dbdrop3.sql`, `dbcreate4.sql`, and `dbdrop4.sql`. Copy the files to your USB drive or email the files to yourself such that you can access all of them either through command line interface mysql or graphical user interface MySQL Workbench.

Tasks

Task1 (2 marks)

Consider a conceptual schema given below.



Your task is to perform a step of logical database design, i.e. to transform a conceptual schema given above into a collection of relational schemas.

For each relational schema created clearly list the names of attributes, primary key, candidate keys (if any), and foreign keys (if any). Assume, that **partitioning method** must be used to implement a generalization. A way how a conceptual schema can be transformed into a collection of relational schemas is explained in a presentation 06 Logical Design. Listing of the relational schemas in the other format scores no marks.

The relational schemas must be listed in a format presented in the slides 43-44 in a presentation 06 Logical Design. There is no need to show all stages of the transformations. Only the final relational database design is expected.

Deliverables

A file `solution1.pdf` with a list of relational schemas, primary key for each relational schema, candidate keys (if any) for each relational schema, foreign keys (if any) for each relational schema. Submission of a file with a different name and/or different extension and/or different type scores no marks.

Task 2 (1 mark)
Consider the following collection of relational schemas.

```
Staff( Staffno, SName, DOB, SLevel, DName)
Primary key = (Staffno)
Foreign key =(DName) references Department (DName)

Department(DName, MainOffice, Headerno)
Primary key = (DName),
Foreign key = (Headerno) references Staff(Staffno)


Project (Pno, Sponsor, StartDate, EndDate, Budget,
         Managerno)
Primary key = (Pno)
Foreign key = (Managerno) references Staff (Staffno)

Device (ID, System, WiFi, Pno)
Primary key = (ID)
Foreign key = (Pno) references Project(Pno)

Student (Studentno, Name, DOB, Degree, DName)
Primary key = (Studentno)
Foreign key = (DName) references Department (DName)

StaffWorkInDepartment (Staffno, DName, TimePercentage)
Primary key = (Staffno, DName)
Foreign key 1 = (Staffno) references Staff (Staffno)
Foreign key 2 = (DName) references Department (DName)

Assistant (Staffno, Studentno, Pno)
Primary key = (Staffno, Studentno, Pno)
Foreign key 1 = (Staffno) references Staff (Staffno)
Foreign key 2 = (Studentno) references Student (Studentno)
Foreign key 3 = (Pno) REFERENCES Project (Pno)

StaffWorkInProject (Staffno, Pno)
Primary key = (Staffno, Pno)
Foreign key 1 = (Staffno) REFERENCES Staff (Staffno)
Foreign key 2 =(Pno) REFERENCES Project (Pno)
```

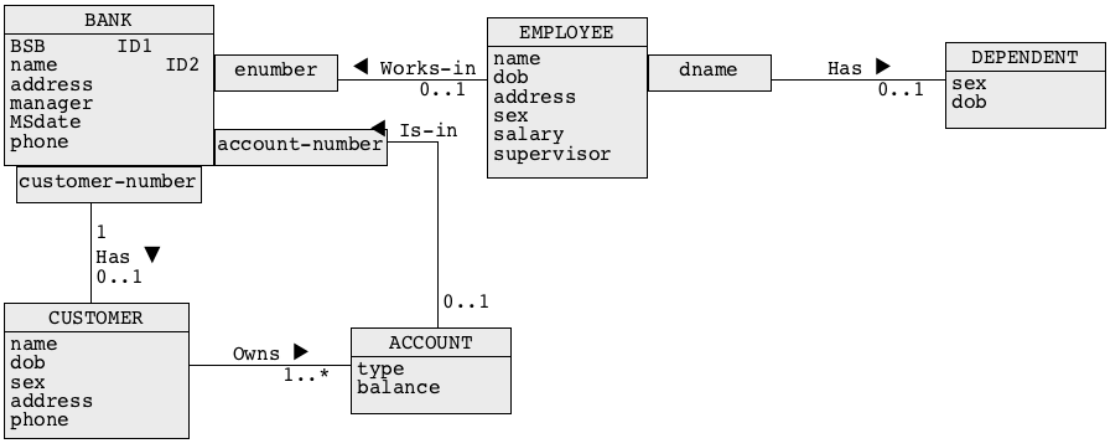
Your task is to perform *reverse database engineering*, i.e. to find a conceptual schema of a database that has a collection of relational schemas given above. Use UMLetlet to draw a conceptual schema found and save your design in a file `solution2.bmp` (you have to Export your design under UMLetlet to a file of type `BMP`).

Deliverables

A file `solution2.bmp` with the *reverse engineered* conceptual schema. Submission of a file with a different name and/or different extension and/or different type scores no marks.

Task 3 (1 mark)

SQL script `dbcreate3.sql` contains `CREATE TABLE` statements that can be used to create a small relational that contains information about banks, employees, customers, accounts and dependants. A conceptual schema of the database is given below.



Analyse, the contents of a script `dbcreate3.sql` to find out how the relational tables are implemented and then execute a script `dbcreate3.sql` to create the relational tables. A script `dbdrop3.sql` can be used to drop all relational tables created by a script `dbcreate3.sql`. Do not drop the tables now !

Your task is to implement SQL script `solution3.sql` with only (!) `ALTER TABLE` statements that implement the following modifications of the original relational tables included in the sample database.

(1) We would like to be able to add information about the total number of employees working at a particular bank for each bank.

(2) We would like to increase the maximum length of bank name to 50 characters;

(3) We would like to define a supervisor of each employee as one of the employees in a bank.

(4) We would like to increase maximum balance of an account up to $99999999.99.

(5) We would like to allow for automatic deletion of a row from `Dependent` table when a row from `Employee` table is deleted.

You can find a lot of information about application of `ALTER TABLE` statements in a presentation 09 SQL - Data Definition Language (DDL) and in Cookbook, How to use data definition and basic data manipulation statements of SQL, Recipe 4.1 How to create and how to alter the relational tables?

When a file `solution3.sql` is ready connect to MySQL either through command line client mysql or graphical user interface MySQL Workbench and execute your script file file.

It is recommended to use a script `dbdrop3.sql` to drop all relational tables modified during execution of a script `solution3.sql` and it is recommend to execute `dbdrop3.sql` after each execution of `solution3.sql` (you can also put `DROP TABLE` statements at the end of a script `solution3.sql`) and later re-create the original database with a script `dbcreate3.sql`. In such a way your script always operates on the original structures of the sample database.

When execution of your script returns no errors then connect to MySQL server using command based interface mysql and create a report from processing of the script `solution3.sql`. Save your report in a file `solution3.rpt`. You can find more information about creating reports from processing of SQL scripts in Cookbook, Recipe 3.1 How to use "mysql? Command based interface to MySQL database server? Step 4 How to save the results of SQL processing in a file?"

Your report must contain a listing of all SQL statements processed.

You can find more information on how to display SQL statements while a script is processed in Cookbook, Recipe 3.1 How to use "mysql? Command based interface to MySQL database server? Step 3 How to process SQL script ?"

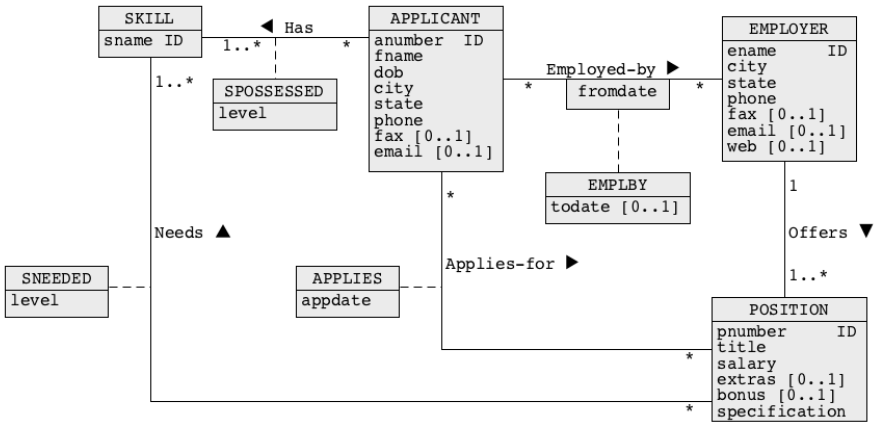
A report that contains no listing of executed SQL statements scores no marks and report that contains errors of any kind also scores no marks!

Deliverables

A file `solution3.rpt` with a report from processing of SQL script `solution3.sql`. The report MUST have no errors and the report MUST list all SQL statements processed. Submission of a file with a different name and/or different extension and/or different type scores no marks.

Task 4 (2 marks)

SQL script `dbcreate4.sql` contains `CREATE TABLE` statements of database that contains information about the skills, applicants, employers, positions, and applications for positions. A conceptual schema of the database is given below.



Analyse, the contents of a script `dbcreate4.sql` to find out how the relational tables are implemented and then execute a script `dbcreate4.sql` to create the relational tables. A script `dbdrop4.sql` can be used to drop all relational tables created by a script `dbcreate4.sql`. Do not drop the tables now !

Your task is to implement SQL script `solution4.sql` with `CREATE TABLE`, `ALTER TABLE`, and `DROP TABLE` statements that implement the following modifications of the original sample database. Note that your script must operate on the already existing database and it must not create a new database from very beginning.

(1) It should be possible to store information about the total number of positions needed by each employer in a table `EMPLOYER` and about the total number of applications submitted by each applicant in a table `APPLICANT` so far.

(2) The following columns:

```
extra             VARCHAR(50),
specification     VARCHAR(2000) NOT NULL,
```

included in a relational table `POSITION` must be moved to another relational table in order to decrease an average length of rows in the relational table `POSITION`. Such modification may have a positive impact on the performance of query processing on the table. A name of another relational table is up to you.

(3) The relational tables `SPOSSESSED` and `SNEEDED` must have two foreign keys each. Discover the foreign keys and add the missing foreign keys to both tables.

(4) Add the domain constraints that restrict the values in both columns `slevel` in the relational tables `SPOSSESSED` and `SNEEDED` to the positive integer numbers in a range from 1 to 10 inclusive.

(5) Information about `gender` of applicant is needed. The information `fax` of employer is no longer needed in the database.

Note, that you must enforce appropriate consistency constraints after all modifications!

When processing of your script returns no errors then connect to MySQL server using command based interface mysql and create a report from processing of the script `solution4.sql`. Save your report in a file `solution4.rpt`. You can find more information about creating reports from processing of SQL scripts in Cookbook, Recipe 3.1 How to use "mysql? Command based interface to MySQL database server? Step 4 How to save the results of SQL processing in a file?"

Your report must contain a listing of SQL statements processed.

You can find more information on how to display SQL statements while a script is processed in Cookbook, Recipe 3.1 How to use "mysql? Command based interface to MySQL database server? Step 3 How to process SQL script?

A report that contains no listing of executed SQL statements scores no marks and report that contains errors of any kind also scores no marks!

Deliverables

A file `solution4.rpt` with a report from processing of SQL script `solution4.sql`. The report MUST have no errors and the report MUST list all SQL statements processed. Submission of a file with a different name and/or different extension and/or different type scores no marks.

Submission

Note, that you have only one submission. So, make it absolutely sure that you submit the correct files with the correct contents. No other submission is possible!

Submit the files **`solution1.pdf`**, **`solution2.bmp`**, **`solution3.rpt`**, and **`solution4.rpt`** through Moodle in the following way:

(1) Access Moodle at **`http://moodle.uowplatform.edu.au/`**
(2) To login use a **`Login`** link located in the right upper corner the Web page or in the middle of the bottom of the Web page
(3) When logged select a site **`CSIT815/CSIT115 (S116) Data Management and Security`**
(4) Scroll down to a section assignment **`Submissions`**
(5) Click at a link **`in this place you can submit the outcomes of Assignment 2`**
(6) Click at a button **`Add Submission`**
(7) Move a file **`solution1.pdf`** into an area **`You can drag and drop files here to add them`**. You can also use a link **`Add`**...
(8) Repeat step (7) for the files **`solution2.bmp`**, **`solution3.rpt`**, and **`solution4.rpt`**.
(8) Click at a button **`Save changes`**
(9) Click at a button **`Submit assignment`**
(10) Click at a button **`Continue`**

A policy regarding late submissions is included in the subject outline.

Only one submission of the second assignment is allowed and only one submission per student is accepted.

A submission marked by Moodle as "late" is always treated as a late submission no matter how many seconds it is late.

A submission that contains an incorrect file attached is treated as a correct submission with all consequences coming from the evaluation of the file attached.

It is expected that all its tasks included in **Assignment 2** will be solved **individually without any cooperation** with the other students. If you have any doubts, questions, etc. please consult your lecturer or tutor during lab classes or office hours. Plagiarism will result in a **FAIL** grade being recorded for that assessment task.

The evaluated outcomes of will be electronically returned to the students before 7.00 pm on Monday, 15 May, 2017.

End of specification