School of Computing and Information Technology
University of Wollongong

Session: **Summer 2020**
Lecturer: **Janusz R. Getta**

CSCI835 Database Systems
Assignment 1
20 June 2020

Scope

This assignment includes the tasks related to normalization of relational databases.

The outcomes of the assignment are due by **Saturday, 27 June, 2020, 9.00 pm (sharp).**

Please read very carefully information listed below.

This laboratory contributes to 13% of the total evaluation in a subject CSCI835 Database Systems.

A submission procedure is explained at the end of specification.

This assignment consists of 3 tasks and specification of each task starts from a new page.

It is recommended to solve the problems before attending the laboratory classes in order to efficiently use supervised laboratory time.

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

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

A submission of compressed files (zipped, gzipped, rared, tared, 7-zipped, lhzed, … etc) is not allowed. The compressed files will not be evaluated.

All files left on Moodle in a state "`Draft(not submitted)`" will not be evaluated.

An implementation that does not compile due to one or more syntactical errors scores no marks.

It is expected that all tasks included within **Assignment 1** 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 the laboratory classes. Plagiarism will result in a **FAIL** grade being recorded for the assessment task.

Tasks

Task 1 (4 marks)

Analyze a collection of incorrectly designed relational schemas listed below.

To find what is wrong with the relational schemas listed below use a method of row insertions explained in a presentation 01 Database Design Quality. Insert into the relational tables with the schemas (headers) listed below from 3 to 5 rows that demonstrate the redundancies.

Include into a file `solution1.pdf` the drawings of relational tables with redundancies and briefly explain the reasons behind each redundancy. The scanned neat hand drawings are acceptable.

`STUDENT(snumber, first-name, last-name, ccode)`

A relational table `STUDENT` contains information about the students and the courses enrolled by the students. A course (`ccode`) is enrolled by more than one students (`snumber`) and each student enrols several course. Student number (`snumber`) uniquely identifies each students and course code (`ccode`) uniquely identifies each course. The first (`first-name`) and the last (`last-name`) names describe the students.

`HOTEL(name, city, capacity, enumber, salary)`

A relational table `HOTEL` contains information about the hotels and employees working in the hotels. A hotel is identified by a pair of attributes (`name, city`) and it is also described by the total number of rooms available (`capacity`). Each employee is identified by employee number (`enumber`) and it is described by a salary (`salary`).

`BUILDING(bnumber, bname, rnumber, area, enumber)`

A relational table `BUILDING` contains information about a building number (`bnumber`) and building name (`bname`). A building number (`bnumber`) uniquely identifies each building. The rooms are located in a building. Each room has a unique number within a building it is located in. Additionally, each room is described by an area (`area`). An employee is identified by employee number (`enumber`) and he/she is located in a room. More than one employee can be located in one room.

`TEAM(tname, player, supporter)`

A relational table `TEAM` contains information about football teams, football players who belong to the teams and supporters of the teams. Each football team is described a unique name (`tname`). Players and supporters are described by unique names (`player`) nd (`supporter`). A team has many players and many supporters.

Deliverables

A file `solution1.pdf` with the drawings of relational tables with redundancies and the brief explanations of the reasons behind each redundancy

---

Task 2 (4 marks)
Perform the following steps and save the outcomes in a file `solution2.pdf`.

(1) Consider a relational schema R(A, B, C, D, E) and the following set of functional dependencies valid in the schema,

$\{A \rightarrow B, E \rightarrow CD\}$

List all minimal keys valid in the schema. List all derivations of function dependencies that lead to the identification of minimal keys. Note, that a schema can have more than one minimal key.

(2) Consider a relational schema R(A, B, C, D, E) and the following set of functional dependencies valid in the schema,

$\{AB \rightarrow DE, D \rightarrow ABC\}$

List all minimal keys valid in the schema. List all derivations of function dependencies that lead to the identification of minimal keys. Note, that a schema can have more than one minimal key.

(3) Consider a relational schema R(A, B, C, D, E) and the following set of functional dependencies valid in the schema,

$\{A \rightarrow CE, CE \rightarrow BD\}$

List all minimal keys valid in the schema. List all derivations of function dependencies that lead to the identification of minimal keys. Note, that a schema can have more than one minimal key.

(4) Consider a relational schema R(A, B, C, D, E) and the following set of functional dependencies valid in the schema,

$\{A \rightarrow B, B \rightarrow A\}$

List all minimal keys valid in the schema. List all derivations of function dependencies that lead to the identification of minimal keys. Note, that a schema can have more than one minimal key.

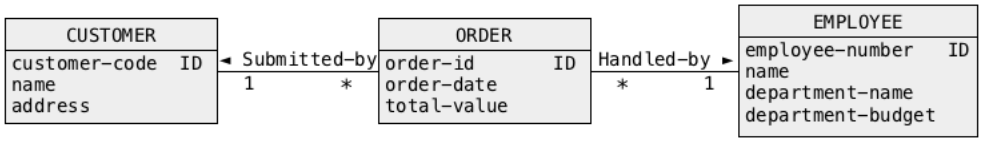
Deliverables
A file `solution2.pdf` with the lists of minimal keys and the derivations of functional dependencies that lead to the identification of minimal keys in the steps listed above.

Note, that "educated guesses" of minimal keys score no marks. You must provide the derivations for each one of the minimal keys found,

---

Task 3 (5 marks)

Consider the following conceptual schema of a sample database domain where the customers submit the orders that contain the items and the orders are handled by the employees who belong to the departments.



Unfortunately, during the conceptual modelling stage and later on during the transformation of the conceptual schema into the relational schemas a database designer made few mistakes. To make a situation even worse, a database designer added to the design an attribute without modification of the original conceptual schema.

A collection of the relational schemas created by a database designer is listed below.

```
CUSTOMER(customer-code, name, address, order-id) primary key =
(customer-code)

EMPLOYEE(employee-number, name, department-name, department-budget)
primary key = (employee-number)

ORDER(order-id, order-date, total-value, customer-code, employee-
number, employee-name)
primary key = (order-id)
foreign key 1 = (customer-code) references CUSTOMER(customer-code)
foreign key 2 = (employee-number) references EMPLOYEE(employee-number)
```

Your task is to find the highest normal form valid for each one of the relational schema listed above and if a relational schema is not in BCNF decompose it into the relational schemas in BCNF. To do so perform the following sequence of steps for each of the relational schemas listed above:
- list functional dependencies valid in each relational schema,
- find minimal keys in each relational schema,
- find the highest normal valid for each schema,
- when necessary transform a schema to BCNF.

Deliverables
A file `solution3.pdf` with the lists of functional dependencies, minimal keys and determination of the highest valid normal form for each relational schema and potential decompositions.
Note, that "educated guesses" of the minimal keys score no marks. You must provide the derivations for each one of the minimal keys found,

---

Submission

Submit the files **solution1.pdf**, **solution2.pdf**, and **solution3.pdf** 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 **CSCI835 (JI20) Database Systems**
(4) Scroll down to a section **SUBMISSIONS**
(5) Click at a link **In this place you can submit the outcomes of Assignment 1**
(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 a step (7) for the files **solution2.pdf**, and **solution3.pdf**.
(9) Click at a button **Save changes**
(10) Click at a button **Submit assignment**
(11) Click at the checkbox with a text attached: **By checking this box, I confirm that this submission is my own work,** ... in order to confirm the authorship of your submission.
(12) Click at a button **Continue**

End of specification