The University of Melbourne
2017 Semester 1 Examination
School of Computing and Information Systems

INFO90002

DATABASE SYSTEMS & INFORMATION MODELLING

Exam Duration: 3 Hours

Reading time: 15 minutes

This exam paper has 6 pages, including this page.

Authorised Materials:	No materials are authorised.

Instructions to Invigilators:

This exam paper must be handed in at the conclusion of the exam.

Answers are to be written in standard exam script books.

Instructions to Students:

This exam paper has 9 questions, some with multiple parts.

You should answer all questions.

Answers must be written in blue or black ink pen.

Textual answers can be in point form.

The total mark for this exam is 70 marks, representing 70% of your final assessment.

Q1 – Database Design (16 marks)

A university is creating a smartphone app that allows students to form well-matched project groups. It is only available to the 500 students doing the 'Database Design' assignment. Groups must contain either 2 or 3 students.

Each individual student begins by setting up a user profile, which must include their student number, first and last name, a short bio (no longer than 200 characters), and whether the student is full or part time.

After a user registers, they are presented with a sequence of profiles of other users. As each profile is displayed, the student either swipes it to the right to indicate they are interested in grouping with that student, or swipes it to the left to indicate that they are not interested in grouping with that student. Students should not be presented with the same fellow student more than once.

The server periodically runs software which searches for sets of 2 or 3 students who have all "swiped right" on each other. When such a potential group is found, an offer is sent to each prospective member, checking whether they are interested in setting up this group. Each recipient of the offer can respond "accept" or "reject". If anyone responds "reject", the offer is cancelled and the group is not formed. If the members all "accept", the group is registered. Students can then give their group a name. We record groups in the system.

While the group is working on their assignment, the members can send messages to each other. A message contains 140 characters or less, and is sent by one member and seen by all the others. It is important to store these messages for later auditing in case there are disputes among the group.

Eventually, groups submit their work. They may submit several versions, and these must all be stored, though only the most recent will be marked. An assignment consists of an E-R diagram (submitted as a JPEG) and a piece of text up to 3,000 words, both of which are stored in the database, not as files.

After the project is over, each group member can rate each of the other members (1 to 5 stars) and if they wish, can write short testimonials about them (no more than 100 characters). These ratings are voluntary. A student can rate another only once.

Question 1 instructions:

For the above scenario, draw a ***physical*** data model. (You must show the data-types of columns.)

Use crows-foot notation for relationships, and join the lines to the related columns. Show the cardinalities of relationships and whether they are optional or mandatory.

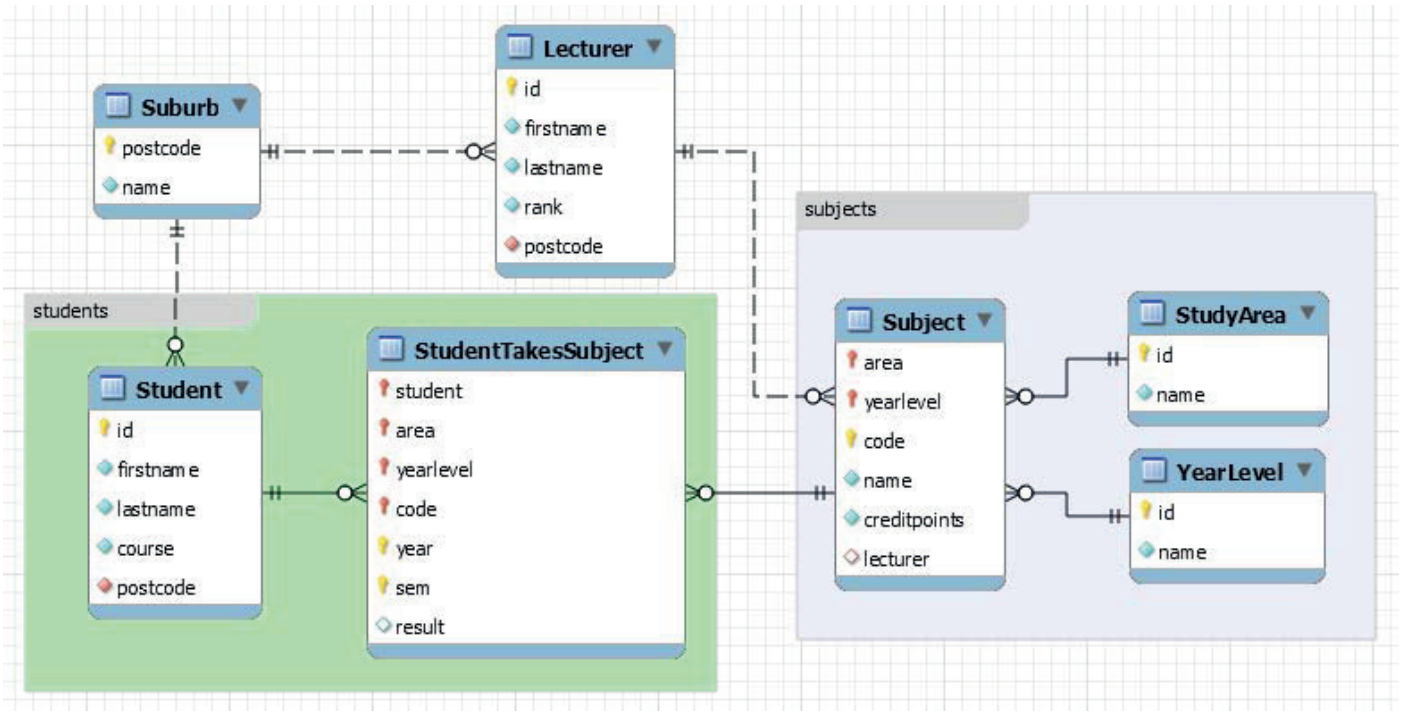
There is no need to add names to relationships. You do not need to write a data dictionary.

If you wish, you may explain the reasoning behind any design decisions or assumptions you made.

Note that marks are largely based on a workable model that enforces all the constraints stated in the case. Marks may be lost for incorrect entities, relationships, cardinalities, attributes, data types or notation, lack of detail or internal contradictions.

Q2 – SQL (17 marks)

Consider the following data model for a university teaching system. Students study subjects which are taught by lecturers. We record which suburbs students and lecturers live in. Subjects are identified by a subject code, parts of which defines the study area (e.g. “INFO”) and the year level (e.g. “9” for postgrad). Sample data (maximum 10 rows per table) are shown below.



Student, Suburb tables

id	firstname	lastname	course	postcode
123001	Britteny	Abston	MC-IT	3161
123002	Margy	Alter	MC-IT	3018
123003	Freida	Amaral	MC-IT	3072
123004	Kera	Basham	MC-IT	3145
123005	Barbra	Batchelor	MC-IT	3046
123006	Lon	Belew	MC-IT	3195
123007	Heide	Bergen	MC-IT	3170
123008	Susan	Binion	MC-IT	3110
123009	Ouida	Bisceglia	MC-IT	3133
123010	Wai	Bruton	MC-IT	3161

postcode	name
3000	MELBOURNE
3002	EAST MELBOURNE
3003	WEST MELBOURNE
3004	MELBOURNE
3005	WORLD TRADE CENTRE
3006	SOUTHBANK
3008	DOCKLANDS
3010	UNIVERSITY OF MELBOURNE
3011	FOOTSCRAY
3012	BROOKLYN

Subject table

area	yearlevel	code	name	creditpoints	lecturer
COMP	9	0007	Internet Technologies	12.5	111111
COMP	9	0038	Algorithms and Complexity	12.5	444444
COMP	9	0041	Programming and Software Development	12.5	222222
INFO	9	0002	Database Systems & Information Modelling	12.5	222222
ISYS	9	0026	Fundamentals of Information Systems	12.5	111111
ISYS	9	0049	Business Analysis Modelling and Design	12.5	444444
ISYS	9	0081	Organisational Processes	12.5	111111

Lecturer table

id	firstname	lastname	rank	postcode
111111	Alan	Turing	B	3161
222222	Ada	Lovelace	B	3062
333333	John	Von Neumann	C	3151
444444	Grace	Hopper	C	3053

StudentTakesSubject, StudyArea, YearLevel tables

student	area	yearlevel	code	year	sem	result
123010	COMP	9	0007	2016	1	82
123010	COMP	9	0038	2016	2	77
123010	COMP	9	0041	2016	1	45
123010	COMP	9	0041	2017	1	NULL
123010	INFO	9	0002	2015	1	77
123010	ISYS	9	0026	2015	2	82
123010	ISYS	9	0049	2016	1	88
123010	ISYS	9	0081	2015	2	65
123011	INFO	9	0002	2015	1	70
123011	ISYS	9	0026	2015	2	88

id	name
COMP	Computer Science
INFO	Informatics
ISYS	Information Systems
NULL	NULL

id	name
1	first
2	second
3	third
4	fourth
5	fifth
6	sixth
9	postgrad

Tasks:

Questions 2A-E require you to write one single SQL statement per question. Do not use Views or temporary tables. Format code for ease of reading. Ensure user-friendly output by ordering and renaming columns where appropriate.

A) List any students who have passed more than 30 credit points of subjects. "Pass" means a result of 50 or more. Show the students' first and last names and total credit points passed. (3 marks)

B) Which lecturers have taught or are teaching students who live in the same suburb as them? Show the lecturers' first and last names. (There is no need to list the students.) (4 marks)

C) How many different students has each lecturer taught? Show the lecturers' first and last names, with a count of students. Include students who have not completed their current subjects. (3 marks)

D) Print a list of students, showing studentid and lastname, the number of subjects the student has completed, and their average result. Only include students who have enrolled in at least one subject. (3 marks)

E) Which subject has the best average result, over all students who studied it? Print the subject code in one column and the average result in the second column. (4 marks)

Q3 – Normalisation (3 marks)

We are modelling a database to store data from supermarket checkouts.

Customers take their goods to the checkout and purchase them: we record this as one purchase, consisting of quantities of different products. We must record each item scanned at the checkout, and the method used to pay for the purchase. We record which store the goods were purchased at. We do not identify our customers.

Our modeller has arrived at the following relation. But this relation is not in third normal form.

SCAN (storeId, storeAddress, purchaseId, whenPurchased, paymentMethod,
(productId, productName, quantityPurchased)
)

Your job is to convert the relation to 3rd normal form.
Mark your primary keys with a solid underline, and your foreign keys with a dotted underline.
(Any attributes that are both primary and foreign keys should get both underlines.)

You don't need to show intermediate normal forms – just the 3rd normal form you end up with.

Q4 – Physical Design (6 marks)

A) For each of the following columns in the "university teaching" database shown in **question 2**, write the column's correct MySQL data type, include the width of the column where appropriate.

The following sub-questions of 4A are worth 1/2 mark. (4 marks)

i) Suburb.postcode

ii) Student.id
iii) Student.lastname
iv) Student.postcode
v) Student.course

vi) YearLevel.id

vii) Subject.creditpoints

viii) StudentTakesSubject.result

B) For the same "university teaching" database shown in **question 2**, it has been found that the most common queries being run against the Student table are the following:

```
select id from Student where lastname = @input;

select id, course from Student where postcode = @input;
```

Based on this evidence, which columns in the Student table should we index? (2 marks)

Q5 – Applications (6 marks)

A) Explain the two (2) major reasons why database systems usually require application software. (4 marks)

B) Draw a diagram of a 3-tier application architecture. Label each component. (2 marks)

Q6 – Distributed Databases (6 marks)

A) Describe two (2) major advantages that distributed databases offer. (4 marks)

B) Explain the CAP theorem. (2 marks)

Q7 – Transactions (6 marks)

A) Briefly describe the two (2) main reasons for using database transactions. (2 marks)

B) Explain, using a timeline diagram and text, and using the example of booking rooms at a hotel, how 2 processes simultaneously accessing the same data can create the ***Lost Update*** problem. (4 marks)

Q8 – Database Administration (5 marks)

A) List four (4) different types of failure that can occur in a database. (2 marks)

B) How does an 'SQL Injection' attack work, and how can we protect against it? (2 marks)

C) What is the main reason for having offsite backups? (1 mark)

Q9 – NoSQL (5 marks)

A) Explain three (3) ways in which NoSQL databases are different from relational databases. (3 marks)

B) Explain the difference between a document store and a graph database. (2 marks)

END OF EXAM