RMIT UNIVERSITY

School of Computing Technologies

ISYS1055/1057/3412 (Practical) Database Concepts

Assessment 1: Database Design

Assessment type: PDF

Word limit: N/A

Draft Due Date: 26 March at 23:59 (Melbourne time) – Week 4 (otherwise 2 mark deduction)
Final Due Date: 9th April at 23:59 (Melbourne time) – Week 6

Weighting: 20%, 20 Marks

Overview

The objective of this assignment is to measure your understanding of the basic concepts in the relational database model and using entity-relationship model for database design. The assessment is in two parts, split into four tasks which cover Basic ER Modelling and Basic Relational Modelling. The tasks are as follows.

Part A: Entity-Relationship Modelling (12 Marks)

1. Design and plan for the implementation of a database system, diagramming the design to a high standard using UML notation through the diagramming tool Lucidchart.

2. Model the activities of an organisation and present the model as an Entity-Relationship (ER) diagram. Analyse this ER diagram, and possibly modify it, based on additional client requirements.

3. Map an ER diagram into a relational database schema, showing every step of the mapping.

Part B: Relational Database Model (8 Marks)

4. Answer a series of short questions about a Relational Database model.

To complete this assessment, you must be familiar with Lucidchart, which is covered during the Week 1-4 activities.

Assessment criteria

This assessment will measure your ability to:

- Describe various data modelling and database system technologies.
- Explain the main concepts for data modelling and characteristics of database systems.

Course learning outcomes

This assessment is relevant to the following course learning outcomes:

CLO1	Describe the underlying theoretical basis of the relational database model and apply the theories into practice.
CLO2	Explain the main concepts for data modelling and characteristics of database systems.
CLO3	Develop a sound database design using conceptual modelling mechanisms such as entity-relationship diagrams.
CLO4	Develop a database based on a sound database design.

Part A: Entity-Relationship Modelling

Task 1: Designing an Entity-Relationship Model

Exercise Central Study

Exercise Central (EC) is an Australian fitness company. The following are the requirements for managing data about staff, clients and exercise plans for EC.

Exercise Plans

EC is responsible for managing exercise plans for customers in Australia. Each exercise plan has a unique plan number and also has a description such as 'building core strength'. The system keeps track of various general exercises performed as part of an exercise plan, recording the unique name of the exercise, its description and the muscle groups that it targets. For example a 'squat' which involves 'lowering and raising your body by bending your knees on a single spot' which targets your 'legs' and your 'core' muscle groups.

Staff

Staff develop specialised exercise plans that are an instance of an exercise and plan with a specific intensity, for example a 'high' intensity squat (which is part of building core strength). It records the number of repetitions, sets, intervals and number of days per week that the exercise should be performed (for a given intensity level). For example a low intensity squat exercise plan might have lower repetition, sets, intervals and days per week, than a high intensity squat plan.
A staff member can develop many such plans, but only one staff member is responsible for developing the plan. Specialised exercise plans can be managed by many different staff (not necessarily the staff member who developed them), and a staff member can manage many such plans.
The system records the staff members id, name and speciality.

Course

Customers can enrol in many courses, with each course composed of several specific exercise plans (eg a low intensity squat plan and a high intensity bicep curl plan). Plans can be a part of many courses. A course has a course id, a fee and a duration (in weeks).

Customer

Each Customer has a unique number and has a name, address and phoneNo. The date that customers begins each course is recorded.

Based on the given description, model the given business rules, and present your model as an Entity-Relationship (ER) diagram. Carefully state any assumptions that you make. In your ER diagram, you must properly denote all applicable concepts, including weak or strong entities, keys, composite or multi-valued attributes, relationships and their cardinality and participation constraints.

If you cannot represent any of this information in the ER model, clearly explain what limitations in the ER model restrict you from representing your model.

You **must** use UML notation and the diagramming tool Lucidchart to draw your diagram. Your diagram must be drawn to a high standard with minimal clutter. You are **not** required to map the ER model to relational model.

A special note: This is an open-ended question with many different models that can be derived. Your model is assessed based on how accurately it represents business rules described above.

Task 2: Designing an Entity-Relationship Model

Part A: Initial Design

Quick Service Case Study

Quick Service is an Australian vehicle repair centre.
You are asked to design a database for managing the vehicle service details. Requirements for the database are as follows:

- Vehicles are brought in for service. For each vehicle the database records the registration number, make, model and year.
- The date, time, duration and fee of the service are also recorded, along with the details of the mechanic who performed the service (employee id and name).
- During the service a log of issues that were identified are recorded. For each issue, the name, description and outcome are stored.
- For each issue identified, the system records the parts that were installed. The parts have a partNo, supplier, description and cost. Note that the same part may be available from more than one supplier. The same part may be used as part of different issues.

Based on the given description, model the business rules of Quick Service, and present your model as an Entity-Relationship (ER) diagram. Carefully state any assumptions that you make. In your ER diagram, you must properly denote all applicable concepts, including weak or strong entities, keys, composite or multi-valued attributes, relationships and their cardinality and participation constraints.

If you cannot represent any of this information in the ER model, clearly explain what limitations in the ER model restrict you from representing your model. Avoid introducing unnecessary artificial keys.

You **must** use UML notation and the diagramming tool Lucidchart to draw your diagram. Your diagram must be drawn to a high standard with minimal clutter. You are **not** required to map the ER model to relational model.

A special note: This is an open-ended question with many different models that can be derived. Your model is assessed based on how accurately it represents business rules described above.

Part B: Client Adjustments

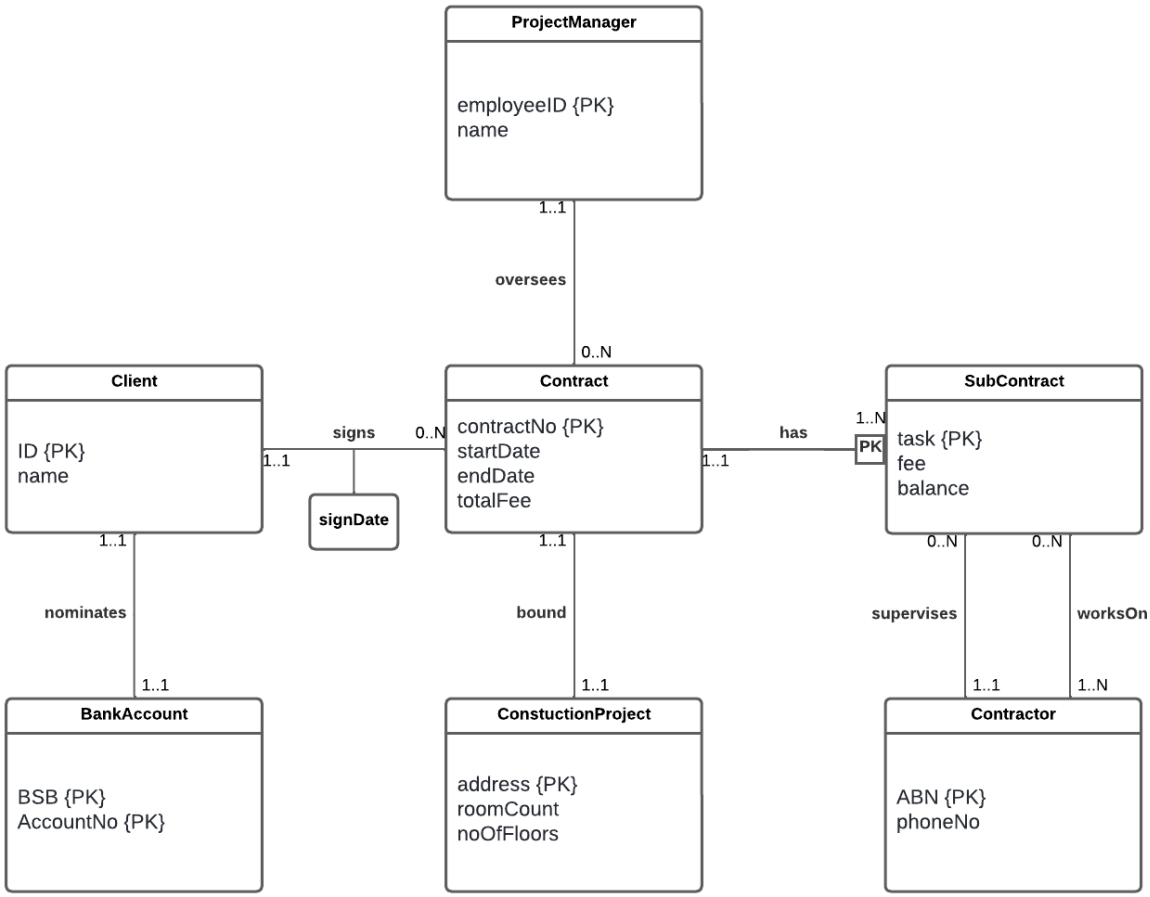
After presenting your ER model to Quick Service management, you are asked if it can be used to perform the following additional tasks.

- Several mechanics may be involved in the service of a vehicle. The system need to be able to identify which mechanic installed each part and resolved each issue during the service.
- The system needs to record not just the cost of parts but also the installation fee.
- Some mechanics have higher seniority and may supervise other mechanics. Mechanics cannot have more than one supervisor. The date when they began supervising others is recorded.
- The same issue could come up with a different description or outcome during various services of vehicles.

For each one of the tasks specified above, explain how your ER diagram is supporting it. If it is not possible to achieve any of the tasks above given your current design, state why, modify the model, provide the modified ER diagram (in addition to your original ER diagram), and explain how the new model achieves the missing requirements.

Task 3: Mapping an ER Model to a Relational Database Schema

Consider the following ER diagram, which shows aspects of a construction project management system.



You are requested to map the above ER diagram into a relational database schema. Show every step of the mapping. **No marks are awarded to the final schema if you do not show the partially built schema at the end of each step.** Clearly indicate the primary key (underlined) and foreign keys (with an asterisk) in each relation.

Part B

Task 4: Relational Database Model

This section contains the schema and a database instance for the Employee database that stores employee data for an organisation. The data includes items such as personal info (e.g., name, phone, salary), departments of the organisation (e.g., name and location of each department, who the manager is), jobs (e.g., titles, salary range), and a history for past contracts with each employee. A database instance is shown in Figure 2 followed by the database schema.

Employees

employee_id	first_name	last_name	phone_number	hire_date	empjob_id	salary	department_id
50	Adam	Smith	1234	26/10/2009	22	$66,000	2
66	Tom	Moosa	1235	10/12/2016	10	$140,000	2
10	Jonny	Deans	1236	21/08/2002	33	$70,000	1
12	Adam	Jones	1247	8/08/2009	10	$138,000	1
18	Joseph	Ryan	1277	5/05/2020	10	$150,000	3

Departments

department_id	department_name	manager_id	location_id
1	IT Services	12	10
2	Accounting	66	20
3	Human Resource	18	30

Jobs

job_id	job_title	min_salary	max_salary
10	Dep Manager	$120,000	$150,000
22	Accountant	$60,000	$80,000
33	Programmer	$60,000	$80,000
45	Senior Programmer	$70,000	$120,000

Locations

location_id	street_address	postal_code	city	state_province	country_id
10	123 Collins St	3000	Melbourne	VIC	1
20	222 Bourke St	3000	Melbourne	VIC	1
30	555 Swanston St	3000	Melbourne	VIC	1

Countries

country_id	country_name
1	Australia
2	Vietnam
3	Spain

Job History

employee_id	start_date	end_date	job_id	department_id
10	1/01/2001	10/04/2002	33	1
10	11/04/2002	20/08/2002	33	1
12	1/01/1998	5/10/2003	33	1
12	6/10/2003	6/10/2004	33	1
12	7/10/2004	7/08/2009	33	1

Figure 2: Employee Database Instance

The database schema is shown below, and the meaning of most attributes is self-explanatory. “Job History” is simplified as “JobHistory”. Primary keys are underlined, and foreign keys are annotated with a *.

Employees(employee_id, first_name, last_name, phone_number, hire_date, empjob_id, salary, department_id)

Departments(department_id, department_name, manager_id*, location_id*)

Jobs(job_id, job_title, min_salary, max_salary)

Locations(location_id, street_address, postal_code, city, state_province, country_id*)

Countries(country_id, country_name)

JobHistory(employee_id*, start_date, end_date, job_id*, department_id*)

The following table further clarifies the connection between the keys across multiple tables.

Foreign Key		Primary Key
Job History.employee_id	-------->	Employees.employee_id
Departments.manager_id	-------->	Employees.employee_id
Job History.department_id	-------->	Departments.department_id
Job History.job_id	-------->	Jobs.job_id
Locations.country_id	-------->	Countries.country_id
Departments.location_id	-------->	Locations.location_id

Figure 3: Keys

The following questions must be answered based on the given database schema and instance. Where explanation is required, each answer should be a SHORT passage of at most several lines.

QUESTION 4.1: Does the database schema ensure that there is a manager associated with each department? Explain your answer.

Questions 4.2: What foreign key constraints are missing from Figure 3? Write down the missing constraints in the format shown in figure 3. Show the updated schema for the corresponding relations.

QUESTION 4.3: The IT Services department has recently changed to have two sub-departments (i.e., IT Support, and Software Devlopment). Now, each sub-department is supposed to have a separate manager. Additionally, the managers of all IT Services sub-departments now must report to a single Director (i.e., "Director of Support Services"). Temporarily and until the new managers are hired, Adam Jones has been appointed to the management of both sub-departments as well to the single role of Director of Support Services. His salary range is between $130,000 and 160,000.

The following SQL statements are intended to record all the changes required in the database instance. Will they work? If they are sufficient to achieve the requirements specified above, explicitly mention they are sufficient. If there are any shortcomings, identify them and briefly justify your answer.

```
UPDATE Departments SET department_name='IT Support' WHERE department_id=1;

INSERT INTO Departments VALUES(4, 'Software Development', 12, 10);
```

QUESTION 4.4: The employee named Jonny Deans changed his job to become a Senior Programmer on 6/06/2020. The following SQL statement intends to make the required changes to reflect Jonny's promotion.

```
UPDATE Employees SET empjob_id=45, hire_date=6/06/2020 WHERE empjob_id=33;
```

Explain if there any issues with the outcome of the update and how it should be fixed?. After running the above query, consider the request "find all the past contracts that Jonny Deans used to have". Can this request be completed using the given database schema and after the above statement is run? If yes, explain how the request can be answered. If no, explain what is missing and how it should be fixed.

QUESTION 4.5: Explain what the result of executing the following SQL statement on the database instance will be.

```
DELETE FROM LOCATIONS WHERE location_id=10;
```

Identify all changes that must be completed to allow this to successfully execute.

QUESTION 4.6: Write an SQL statement to create the JobHistory table including all the constraints, assuming all the tables that JobHistory depends on already exist in the database. Make reasonable assumptions for the data type associated with each field. Your SQL statement must be valid for SQLite Studio environment and free of any errors.

QUESTION 4.7: Write an SQL statement to create the Jobs table including all the constraints, assuming all the tables that Jobs depends on already exist in the database. Make reasonable assumptions for the data type associated with each field. Your SQL statement must be valid for SQLite Studio environment and free of any errors.

QUESTION 4.8: On '01/01/2021' Adam Smith is rehired with a new salary of $90,000. You are asked to update the given database instance so that it includes ALL relevant changes required to store this consistently across all realations. Your SQL statement must be valid for SQLite Studio environment, free of any errors, and compatible/consistent with existing data in the instance in Figure 2.

Submission format

You should submit one PDF document with all answers together. Do not submit Word files.

You must use Lucidchart to work on Part 1 of your assignment. You may use Word or any other word processor to compile your submission. Use section titles to indicate which question you are answering. At the end, convert your answer sheet into PDF format. Microsoft Word has the option of saving your document in PDF format. If the conversion option is not available on your system or word processor, there are free PDF converters online you can utilise (e.g., http://convertonlinefree.com/).

Academic integrity and plagiarism

Academic integrity is about honest presentation of your academic work. It means acknowledging the work of others while developing your own insights, knowledge, and ideas.

You should take extreme care that you have:

- Acknowledged words, data, diagrams, models, frameworks and/or ideas of others you have quoted (i.e., directly copied), summarised, paraphrased, discussed, or mentioned in your assessment through the appropriate referencing methods.
- Provided a reference list of the publication details so your reader can locate the source if necessary. This includes material taken from Internet sites.

If you do not acknowledge the sources of your material, you may be accused of plagiarism because you have passed off the work and ideas of another person without appropriate referencing, as if they were your own.

RMIT University treats plagiarism as a very serious offence constituting misconduct.

Plagiarism covers a variety of inappropriate behaviours, including:

- Failure to properly document a source
- Copyright material from the internet or databases
- Collusion between students

For further information on our policies and procedures, please refer to the University website.

Referencing guidelines

Where referencing is required, use RMIT Harvard referencing style for this assessment.

Refer to the RMIT Easy Cite referencing tool to see examples and tips on how to reference in the appropriated style. You can also refer to the library referencing page for more tools such as EndNote, referencing tutorials and referencing guides for printing.

Penalties for late submissions

Assignments received late and without prior extension approval or special consideration will be penalised by a deduction of 10% of the total score possible per calendar day late for that assessment.

Assessment declaration

When you submit work electronically, you agree to the assessment declaration.