School of Computing and Information Systems

INFO90002

Database Systems and Information Modelling

Practice Exam 2 SOLUTIONS

Semester 1 2022

Reading Time: 15 minutes

Writing Time: 120 minutes (2 Hours)

This exam has 8 pages including this page

ATTEMPT ALL QUESTIONS IN ANY ORDER

Authorised Materials:

While you are undertaking this assessment, you are permitted to

- make use of any textbook, lecture slides (including soft copies)
- MySQL Workbench is supported for E.R. modelling questions
- Draw IO (diagrams.net) for Chen conceptual models or any suitable modelling equivalent
- Any lecture notes, books, laptop, PC
- You are free to use the course materials and your laptop/PC in this exam.

While you are undertaking this assessment, you MUST NOT

- make use of any messaging or communication technology
- record, screenshot, stream, upload or in any known format duplicate this document
- record, screenshot, stream, upload or in any known format duplicate your solutions
- make use of any world wide web or internet based resources such as wikipedia, github, stackoverflow, google, Weichat or any known search engine / messaging services
- act in a manner that could be regarded as providing assistance to a student who is undertaking this assessment or in the future will be undertaking this assessment
- seek assistance from any other student who is undertaking this assessment or in the future will be undertaking this assessment

Instructions to Students:

- This exam is in 8 sections. Attempt all questions in all sections.
- We recommend using pencil and paper for modelling questions to save time
- The total for this exam is 100 marks representing 50% of your final assessment
- Attempt **all** questions which are of unequal marks value
- This exam is a timed assessment which must be completed within **150 minutes** of official commencement time
- Questions can be answered in any order (please number your attempts)
- Start a new question on a new page
- **PLEASE DO NOT USE RED font colour**
- You must not communicate with other students whilst taking this exam, e.g. using messaging, chat rooms or email

IMPORTANT

- **Your file upload must be a single PDF document before the elapsed time.**
- **No other document format will be assessed (e.g. Pages, doc, txt, .SQL, etc).**
- **Email submissions will not be assessed.**
- **Every question attempt must be numbered (e.g. Q2C, Q1, Q6) to ensure it is assessed.**
- **The official exam language is English. Sections of the submission in languages other than English will NOT be assessed and will be marked as 0.**
- Before submitting your solution document, check that the diagram(s) is/are readable. It is your responsibility to ensure your answers are readable and make sense to the marker.

The work you submit **must be based on your own knowledge and skills** and without the assistance of any other person. You MUST NOT directly copy work that you have not authored (e.g. slide notes, websites, other student's study notes).

Q1. ER Modelling (30 Marks)

TeaTime

TeaTime is a platform for tea lovers. To use the system, users must install the TeaTime app on their phone and set up a user profile that contains: a login name (their email address), password (encrypted and stored as a string of 64 characters), short screen name e.g. "DonkeyKong64" (which other users will see), profile photo, and a “My Cuppa” entry - their favourite tea type, chosen from a list of standard drink names such as “Peppermint Tea” and “Black Tea with Honey”. We plan to have up to 2 million customers (end users) accessing the system.

While a user’s phone is switched on, the TeaTime app sends the phone’s current location to the server once per minute. We store these, so that the system knows where a given user is “now”, as well as the history of where the user has been.

All locations in this system are recorded as a pair of numbers representing latitude and longitude. Latitudes are between -90 and 90 degrees (south pole to north pole) while longitudes are between -180 and 180 degrees (west or east of the prime meridian in Greenwich). We will use a precision of 4 decimal places, which is about 11 metres at the equator. For example, the Doug McDonell building at UniMelb is at latitude -37.7989, longitude 144.9627.

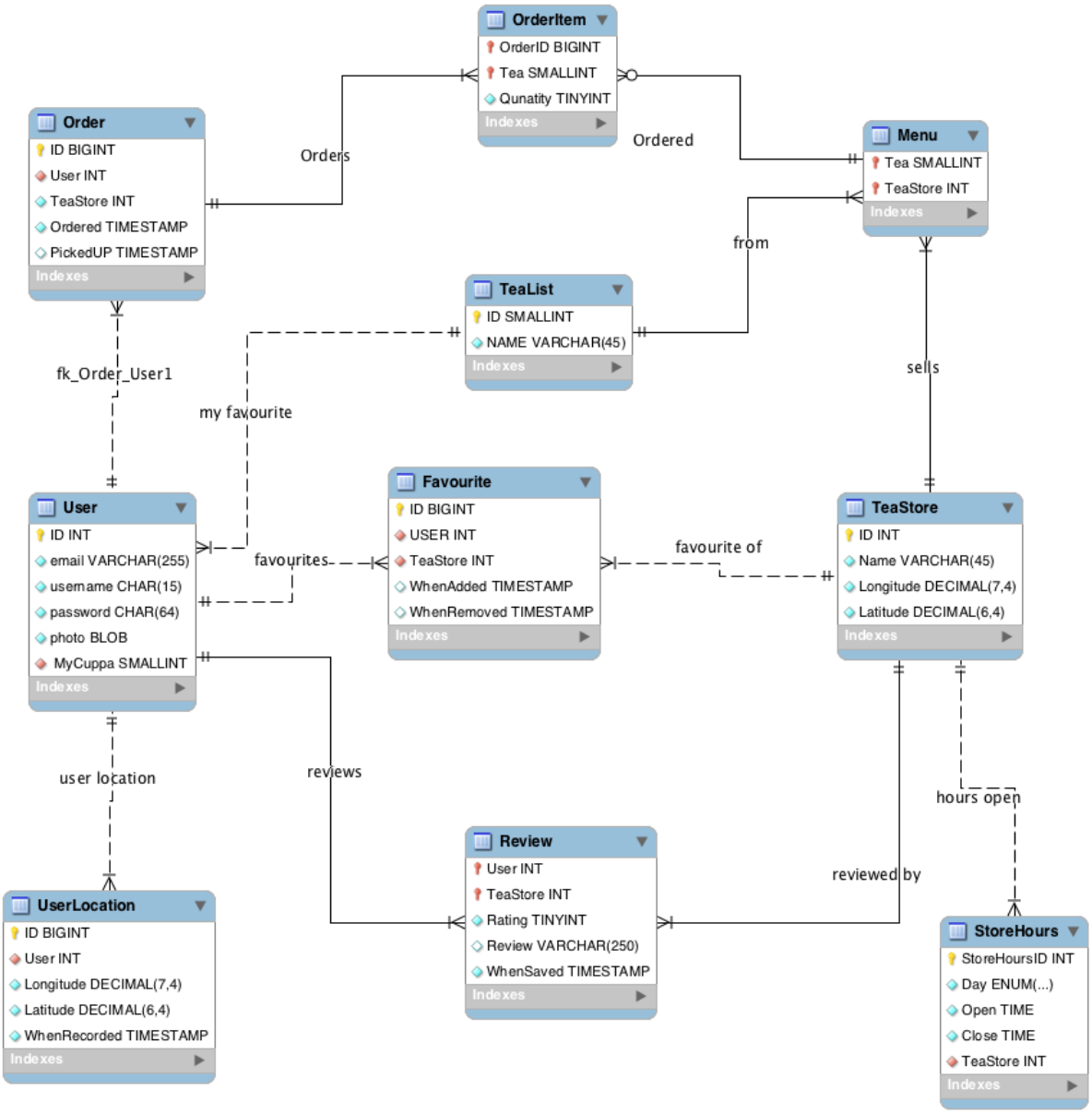
The names, locations and opening hours of about 600 tea stores are available via the app. Users can browse stores either in an alphabetical list or via a map. When viewing a store's profile, users can see reviews and photos uploaded by other users, and the store's average rating. The store's opening hours are recorded as an opening and closing time for each day of the week (for example, a café might open Mondays 8am to 6pm, Tuesdays 9:30am to 7pm, and so on). Each café stores a menu of the teas it sells: these must be drawn from our standard list mentioned above.

Users can rate stores. A rating consists of a whole number between 0 and 10, along with an optional piece of text (up to about 30 words). A given user can only rate a given tea store once. Users can mark particular stores as favourites. These can be viewed in a list. Users can later “unfavourite” the tea store if they wish, and yet later “favourite” it again. We keep a history of these favourites and unfavourites. Customers can use the app to order drinks from a tea store. To do this a user first selects a tea store, then chooses how many of each tea type(s) they want from the store's menu. We keep track of when orders are placed and when customers later pick up the order.

Q.1. Draw a physical model in Crow’s Foot notation for this case study. Be sure to write down any assumptions you make.

(30 marks)

SOLUTION:



Q2. SQL (20 marks)

Given the schema in Figure 2, write a single SQL statement to correctly answer each of the following questions (3A – 3D). DO NOT USE VIEWS to answer questions.

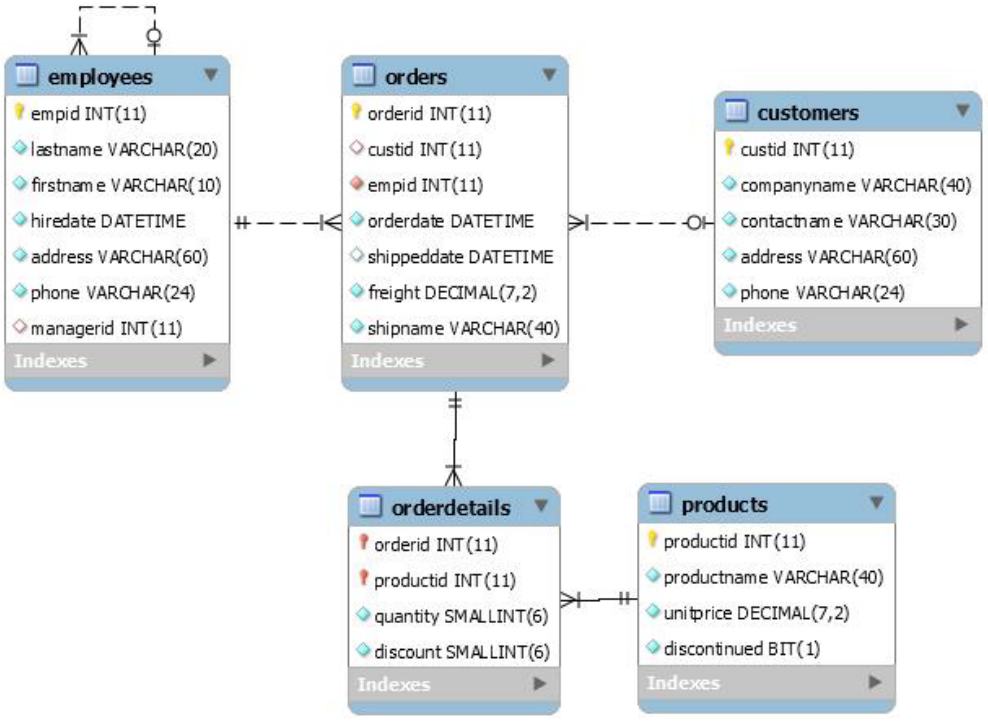


Figure 1 Company Schema

Q.2A. Write a query that returns customers (company names) and the details of their orders (orderid and orderdate), including customers who placed no orders.

(3 marks)

```
SELECT C.companyname, O.orderid, O.orderdate
FROM Customers AS C
LEFT OUTER JOIN Orders AS O
ON O.custid = C.custid;
```

Q.2B. Write a query that returns the first name and last name of employees whose manager was hired prior to 01/01/2002.

(4 marks)

```
SELECT E.firstname, E.lastname
FROM Employees AS E
INNER JOIN Employees AS MNGR
ON E.managerid = MNGR.empid
WHERE MNGR.hiredate < '2002-01-01';
```

Q.2C. Write a query that returns customers (customer ID) whose company name is 'Google', and for each customer return the total number of orders and total quantities for all products that were not discontinued ('1' means discontinued, '0' not discontinued).

(5 marks)

```
SELECT C.custid, COUNT(O.orderid) AS numorders, SUM(OD.quantity) AS
totalqty
FROM Customers AS C
INNER JOIN Orders AS O
ON O.custid = C.custid
INNER JOIN OrderDetails AS OD
ON OD.orderid = O.orderid
INNER JOIN Products AS P
ON OD.productid = P.productid
WHERE C.company = 'Google' AND P.discontinued = '0'
GROUP BY C.custid;
```

Q.2D. Write a query that returns the ID and company name of customers who placed orders in 2007 but not in 2008.

(8 marks)

```
SELECT custid, companyname
FROM Customers
WHERE custid IN
  (SELECT custid
   FROM Orders
   WHERE orderdate >= '2007-01-01' AND orderdate < '2008-01-01')
  AND custid NOT IN
  (SELECT custid
   FROM Orders
   WHERE orderdate >= '2008-01-01' AND orderdate < '2009-01-01');

--    BETTER option

SELECT custid, companyname
FROM Customers INNER JOIN ORDERS O1
ON Customers.custid = O1.custID
WHERE YEAR(orderdate) = 2007
AND NOT EXISTS
      (SELECT *
       FROM Customers
       INNER JOIN ORDERS O2
       ON Customers.custid = O2.custID
       WHERE YEAR(orderdate) = 2008);
```

Q3. Normalisation (16 marks)

Q.3A. The table shown below is part of an office inventory database. Identify the design problems and draw a revised table structure in 3rd Normal Form (3NF) that corrects those problems. For each step explicitly identify and discuss which normal form is violated.

(Key: PK = Bold FK = Italic PFK = Bold + Italic)

Inventory (**ItemID**, Description, Qty, Cost/Unit, Dept, Dept Name, Dept Head)

ItemID is the candidate key for this table.

The following functional dependencies hold:

Dept ⟶ Dept Name and Dept Head

Description ⟶ Cost/Unit

Qty, Cost/Unit ⟶ Inventory Value

ItemID	Description	Dept	Dept Name	Dept Head	Qty	Cost/Unit	Inventory Value
4011	1.4m Desk	MK	Marketing	Jane Thompson	5	200	1000
4020	Filing Cabinet	MK	Marketing	Jane Thompson	10	75	750
4005	Executive chair	MK	Marketing	Jane Thompson	5	100	500
4036	1.2m Desk	ENG	Engineering	Ahmad Rashere	7	200	1400

Table 1. The Inventory table

(10 marks)

1NF: All data is atomic – there are no repeating groups.
2NF: The table is in 1NF and there are no partial functional dependencies, so the table is in 2NF.
3NF: The table is in 2NF however there is a transitive functional dependency:
Dept → Dept Name, Dept Head. Dept is not a key attribute. This transitive FD violates 3NF.

NB: Quan, Cost/Unit → Inventory Value
Because there is derived value between these columns (Quan × Cost/Unit = Inventory Value), this FD can be resolved by removing the redundant Inventory Value column.

3NF
Inventory2(**Item ID**, Description, *Dept*, Quan, Cost/Unit)
Department (**Dept**, Dept Name, Dept Head)

KEY:
BOLD = PK
ITALIC = FK
BOLD + ITALIC = PFK

Q.3B. Given the following relation (Inventory), and its functional dependencies - is it possible to demonstrate Armstrong's Axioms of Reflexivity, Augmentation and Transitivity?

Inventory (**ItemID**, Description, Qty, Cost/Unit, Dept, Dept Name, Dept Head)

ItemID is the candidate key for this table.

The following functional dependencies hold:

Dept ⟶ Dept Name and Dept Head

Quan, Cost/Unit ⟶ Inventory Value

Reflexivity: Pick any two sets of attributes where one is a subset of the other.

E.g. {Dept} ⊆ {Dept, Qty} which then means

Dept, Qty ⟶ Dept

Augmentation: Take an existing FD and stick the same attribute to both sides. So from Description ⟶ Cost/Unit we can get:

Description, Qty ⟶ Cost/Unit, Qty

Transitivity: If we take (2) above together with the original 3rd dependency we can do:

Description, Qty ⟶ Cost/Unit, Qty AND Cost/Unit, Qty ⟶ Inventory Value gives us

Description, Qty ⟶Inventory Value

(6 marks)

Q4. Data Warehousing (10 marks)

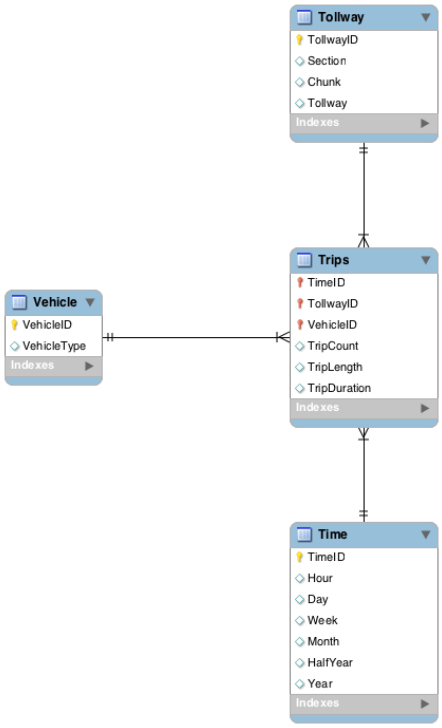
Q4A. Transurban operates toll roads in Australia and the United States of America. Traffic data analysis helps determine road maintenance, RFID reader maintenance, gantry maintenance and capacity management to reduce inefficiencies in its tollway network.

Each part of the tollway is identified by a section (E3), and multiple sections (E1, E2, E3) make up a chunk, (C3) and multiple Chunks make up an entire roadway ("Eastlink"). Many different vehicles on toll roads every day. Each vehicle falls into one category: motor cycles, passenger vehicles, 4WD's, vans, trucks, prime movers, buses, trailers, recreational and miscellaneous machinery (e.g. tractors, cranes, street sweepers, back hoes).

Transurban's management wants to understand the vehicle trips on its road network. They need to understand the number of trips, trip length, trip duration,

Draw a *star schema* to support the design of this data warehouse, showing the attributes in each table. You do not need to select data types. Clearly display the legend for Primary Key, Foreign Key and Primary Foreign Key.

(8 marks)



Q.4B. Why are star schemas preferred over relational database designs to support decision making?

(2 marks)

Star schemas support Data Warehouses which are organized around facts (business measures) and dimensions that help with managerial decision making. Star schemas are denormalised, making it faster to aggregate and query data. Faster aggregates and query data is required due to the large volume of data stored in data warehouses.

Q5. Security and Backups (10 marks)

Q5A. One of the technical safeguards of database systems is access control. Which type of access is based on the "need to know" principle? Explain how this principle works with an example.

(4 marks)

The "need to know" principle is the core of mandatory access control (MAC); it works on user clearance levels which restricts (or allows) user access to objects. Users can only access objects cleared at their levels or lower access levels. The access to objects is centrally controlled by the policy. Users cannot transfer access to owned objects to other users.

MAC model is often enforced in government systems or at NASA with labels assigned to objects, e.g. "confidential", "secret", "top secret". When a user tries to access a "top secret" object, the clearance level is checked before allowing or denying access.

Q5B. Learning Management Systems (LMS) use Role-based Access Control (RBAC). Explain this statement.

(2marks)

In LMS, e.g. in Canvas, when a user account is created, that user is allocated a role, e.g. Student, Instructor, Tutor, Marker. Students have read only access to Modules whereas Instructors have read and write access. A user in the Student role can only see his/her marks whereas users who are Instructors, Tutors, Markers can read and write marks for all students.

Q5C Illustrate the concept of an SQL injection. Describe a scenario and a string a hacker could enter that would jeopardise database security.

(4 marks)

SQL injections are used by hackers to insert their own commands into the developers' SQL statements. If user input becomes part of the pre-existing statement it is possible to edit that statement so that it does what the hacker wants (e.g. shows the database structure, such as table names, or deletes or modifies data). For example, let's say a developer has a statement with the direct input of user name and password

```
SELECT *
FROM User
WHERE username = ' @name '
and password = ' @pw ';
```

A hacker can enter username as **' or 1=1; --**

Or 1=1 – will evaluate the expression to true telling the system that username is valid
-- will discard any SQL following them including password validation so that access is granted and only hacker's commands are executed.

Q6. Transactions (10 marks)

Q6A. What is the inconsistent retrieval problem? Describe the problem and use your own example to demonstrate the answer.

(4 marks)

The inconsistent retrieval problem is where data in one long running read session (SQL SELECT) is reading data that has been modified in other sessions. At the end of the read session it may have read unmodified and modified data from other sessions.

In the table below Session 1 is calculating the interest payment for credit card payments for all credit card accounts at a bank. Session 2, Alice is paying off her entire credit card balance (a series of update statements). In Session 3 Bob is adding more credit to his credit card balance - another update.

Timestamp	**Session 1** Query Calculating credit card interest	**Session 2** Alice	**Session 3** Bob
1	SELECT begins	Balance 673.50	Balance 99.50
23	Reads Alice's balance		
34		Alice pays off her entire balance	
47			Bob adds $49.50 to his credit
55	Reads Bob's balance		
60	Finishes running query		

Table 6. Transaction log demonstrating inconsistent retrieval problem.

In this scenario, the query is not including Alice's payment, but is including Bob's purchase. This is an inconsistent retrieval of data.

Q6B. A sales company decided to increase salaries by 2% across all jobs. Write a transaction to increase annual salary of the job with JobCode 9.

The Jobs table was created using DDL:

CREATE TABLE Jobs (JobCode int, JobTitle char(15), AnnualSalary decimal(10,2), primary key(JobCode));

Hint: it is a better practice to use variables instead of hardcoded values.
Syntax example: SET @name='John';

(6 marks)

```
START TRANSACTION;
SELECT * FROM Jobs; -- optional
```

```
SET @increase = 0.02;
UPDATE Jobs SET AnnualSalary = AnnualSalary * (1 + @increase);
COMMIT;
```

Q7. NoSQL (4 marks)

Q7. Domain integrity can be violated in NoSQL databases whereas Relational databases will report a domain integrity violation. Discuss the benefits and risks of violating domain integrity.

(4 marks)

Benefits of domain integrity – data stored in the database must comply with the rules of that domain (e.g. Date datatype must be stored as a date).
Disadvantage of domain integrity – is that we are limited by the databases capability to what types of domains we can store (e.g. spatial data)

Benefit of violating domain integrity – we can store many different types of data that is thematically related on a topic e.g. novel, movie, reviews, critical essays, pictures they do not confirm to any specific domain integrity and can be stored. We can store data as we collect it without having to modify it (schema on read).

Disadvantage of violating domain integrity
Data accuracy, data integrity and data reliability are all compromised, resulting in poor data quality.

END OF EXAM