ID:300476924 Last Name: Yan First Name: Zichu

Assignment 1 swen304_19

School of Engineering and Computer Science

SWEN 304 **Database System Engineering**

Assignment 1

The objective of this assignment is to test your understanding of database foundations, basic terms, and the relational data model. It is worth 10% of your final grade. The assignment is marked out of 100.

The assignment is due on **Friday, 9 August, 23:59 pm**. Please submit your assignment in **pdf** with your **student ID and Name** via the submission system. Submissions without ID and name will incur **3 marks** deduction from the total marks.

Question 1 [10 marks]

a) **[2 marks]** What is a relation schema and what is a relation?

- A Relation Schema is denoted by $N(A_1, A_2, ..., An)$
- N is the name of the relation
- $A_1, A_2, ..., An$ are the attributes of the relation

Relation schema $N(R, C)$

- N is the name, R is the set of attributes, C is the set of constraints.

A relation is an instance of the relation schema $N(A_1, ..., An)$, denoted by $r(N)$, or simply r if it satisfies all constraints of N

b) **[3 marks]** What is a candidate key? List the properties of a candidate key. What is a foreign key? List the properties of a foreign key.

A candidate key is a combination of attributes that can be uniquely used to identify a database record without referring to any other data.

properties :All candidate keys have some common properties. One of the properties is that for the lifetime of the candidate key, the attribute used for identification must remain the same. Another is that the value cannot be null. Lastly, the candidate key must be unique.

A foreign key is a column or group of columns in a relational database table that provides a link between data in two tables. It acts as a cross-reference between tables because it references the primary key of another table, thereby establishing a link between them.

properties :1, we can have multiple of foreign keys

2, we may insert null attribute in foreign key

3, A foreign key can not automatically create an index, clustered or non-clustered.

c) **[2 marks]** What is a database and what are its main features?

· A database (DB) is a collection of related data that is well structured and stored permanently.

Essential database characteristics are: Represents an aspect of the real world, called miniworld or the universe of discourse (UoD),

Reflects (or should reflect) current state of the UoD,

We shall suppose it is well structured (even has a strict regular structure),

Has users and applications, and

Stored in a permanent (persistent) computer memory,

Managed by a Database Management System (DBMS)

d) **[3 marks]** What is a database management system (DBMS)? What are the tasks of a DBMS?

a database management system (DBMS) is a general-purpose software system that facilitates the process of defining, constructing, manipulating, and sharing databases among various users and applications.

Define a particular database in terms of its data types, structures, and constraints

Construct or load the initial database contents on a secondary storage medium

Manipulating the database:

Retrieval: querying, generating reports

Modification: insertions, deletions and updates to its content

Processing and Sharing by a set of concurrent users and application programs keeping all data valid and consistent

Question 2 [15 marks]

Consider the instance of the STAFF relation shown in the following table. It stores basic data for restaurant staff.

STAFF

Employee	DoB	StaffNo	JobTitle
Tom	22/01/1985	85-11	Waiter
Mickey	11/02/1993	96-02	Chef
Jerry	22/02/1985	85-08	Waiter
Tom	15/05/1990	90-01	Trainee
Minnie	12/07/1990	90-04	Manager
Minnie	25/11/1996	96-22	Waitress

a) **[8 marks]** For every set of attributes (that is, for every subset of {Employee, DoB, StaffNo, JobTitle}) decide whether you can deduce that it is *not* a candidate key, assuming the instance is legal. Justify your answer.

{Employee }and {JobTitle} are not suitable candidate keys. Because It's possible someone have same name (e.g. Minnie)and same job title(e.g. Waiter).These two kinds of values of properties can be duplicate respectively.

{Employee} and {DoB} these two attributes can be a candidate key, because Employee can be duplicate but when we combine Employee with DoB it won't cause DoB number is unique.

{Employee} and {StaffNo} these two attributes can be a candidate key, because Employee can be duplicate but when we combine Employee with StuffNo it won't cause stuff number is unique.

Although {Employee, DoB} and {Employee ,StaffNo} can be candidate keys but DoB and StuffNo are minimal keys ,It means it's unnecessary to combine them ,we can use {StuffNo} and {DoB} directly.

b) **[5 marks]** For every remaining set of attributes (that is, every set not ruled out as a candidate key in part **a)**), discuss whether you consider it a suitable candidate key? Justify your answer.

Yes.Staff number and Dob are suitable candidate keys which are unique for every staff in this table an don't have .

c) **[2 marks]** Which of the candidate keys identified in part **b)** would you choose as the primary key?

StaffNo(Staff number),because it's possible that two staff have same Dob .However,StaffNo can be controlled by people(database system manager).It is the most efficient way to distinguish staff and avoid duplicate.

Question 3 [10 marks]

Your supermarket is using a relational database to manage its data on products and their suppliers. Suppose the underlying database schema includes the following relation schemas:

- SUPPLIER (SID: STRING, Name: STRING, Location: STRING)
  with primary key {SID}
- PRODUCT (PID: STRING, Name: STRING, SID: STRING, InStock: INTEGER, Price: INTEGER)
  with primary key {PID, SID} and foreign key SID ⊆ SUPPLIER[SID]

Below you find instances of these two relation schemas:

PRODUCT

PID	Name	SID	InStock	Price
557	Apple	23XY	50	21
85520	Pear	A15F	0	78
63311	Pear	FVT35	211	49
36773	Kiwi	23XY	50	21
36773	Kiwi	FVT35	29	22

SUPPLIER

SID	Name	Location
23XY	GreatFruits	Wellington
FVT35	Yummy	Wellington
F15A	GreatFruits	Levin
A15F	BetterFruits	Lower Hutt
5AB32	NiceFruits	*null*

Your tasks are as follows. **Justify your answers**!

a) **[5 marks]** Decide which of the following tuples can be added or removed, respectively.

1. Insert tuple (*null*, 'Tasty', 'Wellington') into SUPPLIER. False

SID cannot be null cause it is the primary key of SUPPLIER Schema.

2. Insert tuple ('XYZ4', 'Wellington', 'Yummy') into SUPPLIER. True

It have the essential criteria ,primary key is unique ,although the location and name make no sense .

3. Insert tuple ('FVT35', 'SweetFruits', 'Porirua') into SUPPLIER. False

No, cause there already have FVT35 primary key in the table which cannot be duplicate.

4. Delete tuple ('A15F', 'BetterFruits', *null*) from SUPPLIER. False

There doesn't exist a tuple ('A15F', 'BetterFruits', *null*) in the table .And it couldn't be deleted cause A15F is a foreign key in PRODUCT table, if it disappear the PRODUCT table gonna be meaningless

5. Delete tuple ('23XY', 'GreatFruits', 'Wellington') from SUPPLIER. False

Although there is a valid tuple ('23XY', 'GreatFruits', 'Wellington') exist but we shouldn't delete it because 23XY is a foreign key in PRODUCT table.

b) **[5 marks]** Decide which of the following tuples can be added or removed, respectively.

1. Insert tuple ('55555', *null*, 'F15A', 2, 99) into PRODUCT. True

Yes, it can be Insert because the primary key is unique.

2. Insert tuple ('54556', 'Lemon', 'FV35', 20, 43) into PRODUCT. False

No, cause there is no FV35 primary key in SUPPLIER table which should be the foreign key of PRODUCT table.

3. Insert tuple ('53557', 'Apple', '5AB32', 500, 1) into PRODUCT. True

Yes, it can be Insert because the primary key is unique.

4. Delete tuple ('36773', 'Kiwi', '23XY', 50, 21) from PRODUCT. True

Yes, there is a valid tuple ('36773', 'Kiwi', '23XY', 50, 21) exist.

5. Delete tuple ('46557', 'Apple', '23XY', 1, 21) from PRODUCT. False

No, there is no such primary key is the table .

Question 4 [25 marks]

A travel agency is using a relational database to manage its business data. Suppose the following relation schemas are part of the underlying database schema.

- CUSTOMER (emailaddress, name, dob, phone) with primary key {emailaddress}
- AGENT (staffId, name) with primary key {staffId}
- BOOKING (staffId, tourId, emailaddress, dateFrom, noOfDays, noOfTravellers) with primary key {tourId, emailaddress, dateFrom}
- TOUR (tourId, destination)

The following additional constraints are known:

1. Every customer may only use a single emailaddress.
2. An agent may also be a customer, but may not book a tour for her/himself.
3. For every booking, noOfTravellers must be specified, while noOfDays may be left blank.
4. A tour may visit up to five destinations.

Your tasks are as follows:

a) **[3 marks]** For the relation schema TOUR, identify all suitable candidate keys (if there are any). Explain your answer.

I think tourId is more suitable as a candidate key than destination because a tour may visit to five destination which is possible to be duplicate .

b) **[5 marks]** For each of the relation schemas, identify all suitable foreign keys (if there are any). Explain your answer.

BOOKING: Email address and Tour ID are Foreign keys which are relative other tables as a primary key respectively.

TOUR:Email address is primary key in customer table and Tour ID is primary key in Tour. Those two key work as a foreign key in Booking table.

c) **[3 marks]** Is it possible to add a booking to the database in the emailaddress of a customer who is not listed in the CUSTOMER relation? Explain your answer.

No , it gonna cause the data inconsistencies, because the Email address is primary key in customer table and booking table using email address as a foreign key, so they have relation but if we add email address in booking table, it will cause referential integrity as booking table email address referencing the customer table email address.

d) **[4 marks]** What would be the consequence of removing the attribute tourId from the primary key of BOOKING? Explain your answer.

It will cause the issue that we will be hard to know whether it is a different tour or not. e.g. people can book more than one tour in a day but different tour, and using {Email address}{DateFrom} these two attributes which cannot ensure they are different tours or not.

e) **[5 marks]** Suppose, a customer ('tom@vuw.ac.nz', 'Tom', 22/01/1985, '381-1230') in the CUSTOMER relation has made several bookings stored in the relation BOOKING. When deleting the record of this customer from CUSTOMER, all his bookings should be deleted, too. How would you ensure this requirement? Explain your answer.

I would like to say that we need to delete that customer booking information in booking table first otherwise, if we delete information in customer table first , it will cause referential integrity as customer email address referenced booking table, so delete from booking table first and delete in customer table is the right order.

f) **[5 marks]** Suppose, an agent (007, 'James') in the AGENT relation quits his job at the travel agency. When deleting the record of this agent from AGENT, all the bookings he

made for customer should not be lost. How would you ensure this requirement? Explain your answer.

In order to ensure this requirement, I'd like to say that we need to make all the booking James made in the booking table, and changed setting staff Id as a null or default value then delete it from agent. After that the booking will remaining in booking table but staff id will be set null.

Question 5 [25 marks]

You are asked to design an ONLINE_AUCTION database for an online auction site whose members (buyers and sellers) participate in the sale of items. The data requirements for this database are summarized as follows:

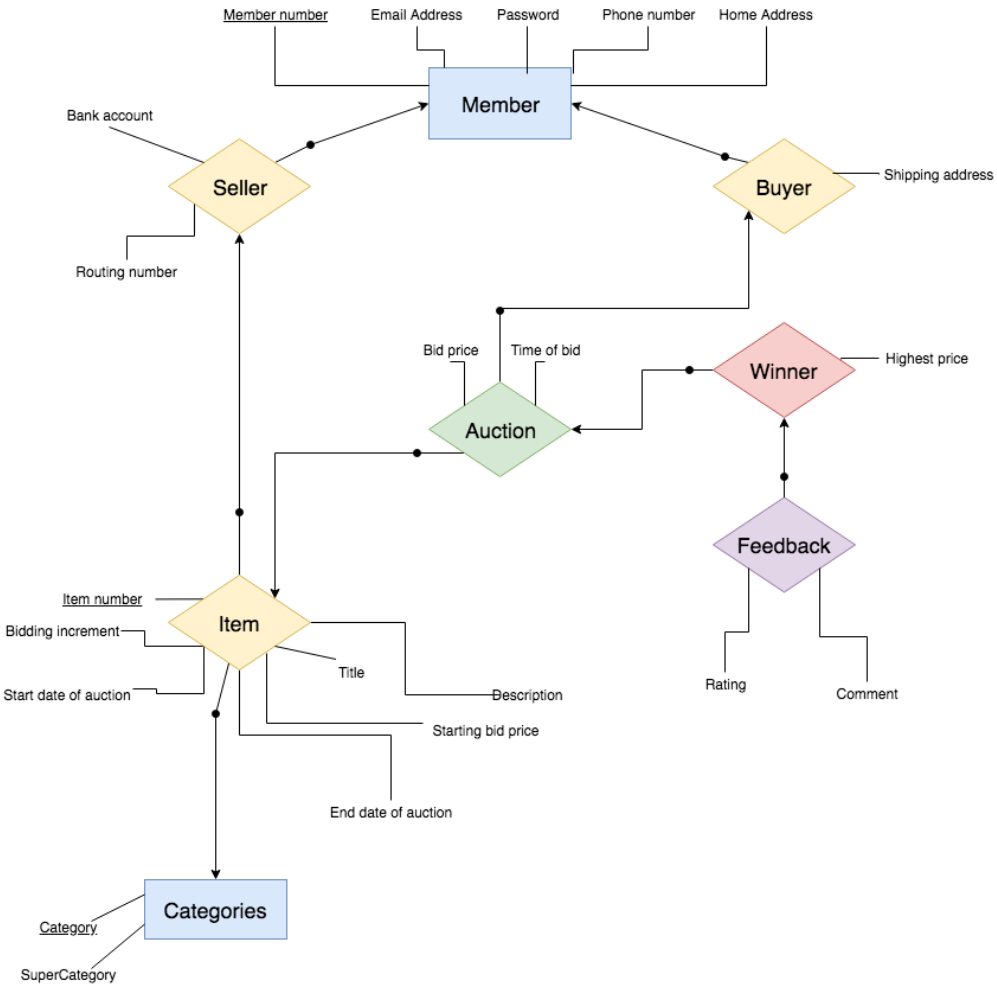
- The online auction site has members, each of whom is identified by a unique member number. Members are described by a name, an email address, a password, a home address, and a phone number.
- A member may be a seller and/or a buyer. Each seller has a bank account number and a routing number. Each buyer has a shipping address.
- Items are placed by a seller for sale. Each item is identified by a unique item number that is assigned by the system. Items are further described by an item title, a description, a starting bid price, a bidding increment, the start date of the auction, and the end date of the auction.
- Items are assigned a category by the seller. The categories are predefined by the online auction site and form a classification hierarchy, i.e., categories may be subcategories of other categories. (For example, the category HARDWARE is a subcategory of the category COMPUTER).
- During an auction, buyers make bids for an item that they are interested in. For each bid, the bid price and the time of bid are recorded in the database. At the end of the auction, the bidder with the highest bid price is declared the winner of the auction. Afterwards the transaction between the seller and the winner may proceed.
- The buyer and the seller may record feedback regarding their completed transactions. Feedback includes a rating of the other party participating in the transaction (a value in the range 1-10) and a comment.

a) **[20 marks]** Draw an extended ER diagram for the database above. Write down the corresponding extended ER schema, including declarations of all the entity types (showing attributes and keys) and relationship types (showing components, attributes and keys). **The diagram I made in the next page.**

b) **[5 marks]** Validate your extended ER diagram against the problem description above. Are there any information, requirements or integrity constraints that you are not able to represent in your diagram? If so, give concrete examples.

The below my diagram shows , the item connected with the auction but there is not transaction in my digram between seller and buyer ,so this is my diagram unable to show. Feedback: the rank of rate what seller and buyer made can not reflect in the diagram. The categories are predefined by the online auction site and form a classification hierarchy but the website I am unable to represent in my diagram.

Remark: Whenever you feel that information is missing in the problem description above, add an assumption and make your assumption explicit. In practice you would consult the domain experts or potential users for clarification.



Level 0: Member =({Member number, Email address, password, Phone number, home address},{Member number})

Categories = ({Category, SuperCategory},{Category})

Level 1: Item =({Categories},{Item number, bidding increment ,start date of auction} {Category, Item number})

Seller =({Member}{Bank account, Rating number, Member number}{Member number})

Buyer =({Member}{Shipping address, Member number}{Member number}

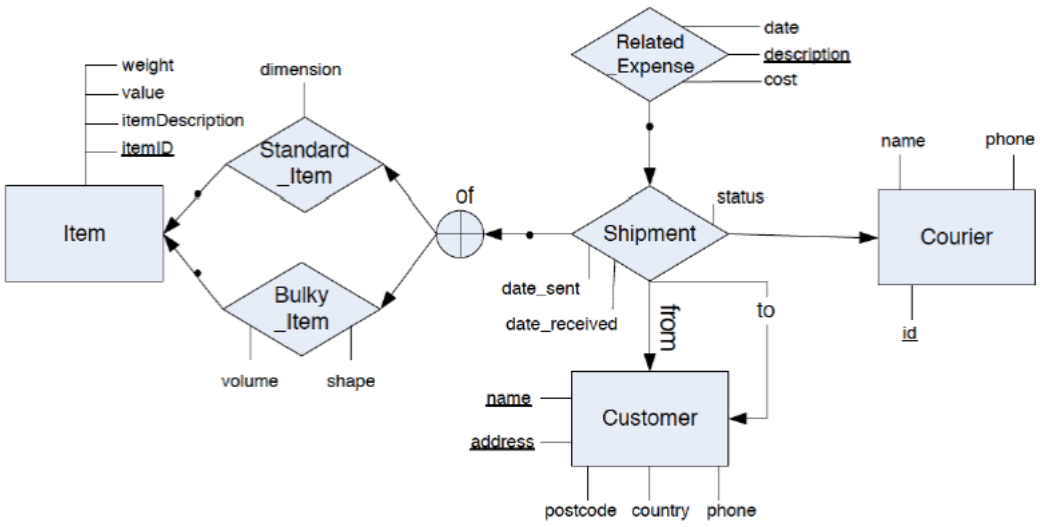
Level 2: Action =({Item, Buyer}{Bid price, Time of bid, Item number, Member number} {Member number, Item number})

Level 3: Winner=({Auction}{Highest price, Member number}{Member number})

Level 4: Feedback = ({Member}{Rating, Comment,}{Member number})

Question 6 [15 marks]

Consider the extended ER diagram below. Transform the corresponding extended ER schema into a relational database schema.



List all the relation schemas in your relational database schema. For each relation schema, list all attributes, the primary key, the NOT NULL constraints, and the foreign keys.

Item = {itemID, itemDescription, value, weight}
Minimal key: {itemID}
Customer={name, address, postcode, country, phone}
Minimal key: {name, address}
Courier={id, name ,phone}
Minimal key: {id}
Standard_Item = {itemID, dimension}
Minimal key:{itemID}
Foreign key: Standard_Item[itemID] ⊆ Item[Item_ID]

Bulky_Item={itemID, volume, shape}
Minimal key:{itemID}
Foreign key:Bulky_Item[itemID] ⊆ Item[item_ID]

Shipment_Standard_Item={,itemID, date_sent , date_received, status}
Minimal key:{itemID}
Foreign key:[itemID] ⊆ Standard_Item[Item_ID]
Standard_Item[Item_ID] ⊆ Item[Item_ID]
}
Shipment_Bulky_Item={itemID, date_sent , date_received, status}
Minimal key:{itemID}

Foreign key:[itemID] ⊆ Bulky_Item[Item_ID]
Bulky_Item[Item_ID] ⊆ Item[Item_ID]

Shipment_from_Customer={name, address, date_sent , date_received, status}
Minimal key: {name ,address}
Foreign key : [name ,address] ⊆ Customer[name, address]

Shipment_to_Customer={name, address, date_sent , date_received, status}
Minimal key: {name ,address}
Foreign key : [name ,address] ⊆ Customer[name, address]

Courier_Shipment={id, date_sent , date_received, status}
Minimal key: {id}
Foreign key : [id] ⊆ Courier[id]

Related_Expense_Shipment_of_Standard_Item={date, description, cost, itemID}
Minimal key:{description}
Foreign key:[itemID] ⊆ Standard_Item[Item_ID]
Standard_Item[Item_ID] ⊆ Item[Item_ID]

Related_Expense_Shipment_of_Bulky_Item={date, description, cost, itemID}
Minimal key:{description}
Foreign key:[itemID] ⊆ Bulky_Item[Item_ID]
Bulky_Item[Item_ID] ⊆ Item[Item_ID]