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 the entity relational model. It is worth 15% of your final grade. The assignment is marked out of 100.

The assignment is due on **Friday, 26 March, 23:59 pm**. Please submit your assignment in **pdf** via the submission system.

Question 1 [5 marks]

a) [2 marks] Give your own example of a relation schema with at least three attributes to illustrate what a relation schema is.

Answer: STUDENT({ID: String, Fname: String, Lname: String, Course: String}, {ID})

with primary ID

b) [3 marks] Give a relation (represented as a table) with at least four tuples over your relation schema in part **a)**. Explain how your relation satisfies the properties of schema keys.

Answer:

Student			
ID	**Fname**	**Lname**	**Course**
300000	Dave	Daveson	COMP
300100	Gorge	Smith	SWEN
300010	Elizabeth	Daveson	SWEN
300110	Steve	Cook	COMP

1. No duplicate values of the key.
2. No unique proper subsets of the key
3. No null values of the key.

Question 2 [15 marks]

Suppose you are the manager of an IT company in Wellington, and you are using a relational database to manage your business data. The following table shows an instance of the STAFF relation schema that stores basic data on the employees working in your company.

STAFF

Name	DoB	EmpNo	JobTitle
Andy	22/01/1988	88-11	Developer
Mickey	11/02/1996	96-02	Project Manager
Jerry	22/02/1989	89-08	Developer
Andy	15/05/1990	90-01	Business Analyst
Alice	22/02/1989	89-06	Developer
Mary	12/07/1990	90-04	Architect
Mary	25/11/1996	96-22	Tester

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

Answer: {EmpNo} is unique, minimal, and not null so any larger subset containing it is not minimal and can be ruled out.

Name, DoB and JobTitle have duplicate values so cannot be keys individually. Additionally, an employee with the same Name, DoB and JobTitle seems plausible so combinations of these three can be ruled out

b) [4 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.

Answer: EmpNo seems like a suitable key, it is presumably uniquely assigned to employees on hiring so there should be no duplicates or null values and as a single attribute it is minimal. However the EmpNo seems to be the last two digits of the birth year and a two digit number so there may be issues if > 99 employees were born in one year.

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

Answer: EmpNo

d) [1 mark] Add a tuple with your own data into the STAFF relation. How would you check that the primary key identified in part **c)** is still valid?

Answer: (Sam, 04/ 09/ 1991, 91-05, Tester). simply check if the EmpNo already exists on the STAFF table

Question 3 [10 marks]

Suppose your software company has developed a relational database for the grocery store "Fruits and more". The underlying database schema contains the following relation schemas:

- COMPANY (Cid: STRING, Name: STRING, Location: STRING) with primary key {Cid}
- FRUITS (Fid: STRING, Name: STRING, Cid: STRING, InStock: INTEGER, Price: INTEGER) with primary key {Fid, Cid} and foreign key Cid $\subseteq$ COMPANY[Cid]

Below you find instances of these two relation schemas:

FRUITS

Fid	Name	Cid	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

COMPANY

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

Your tasks are as follows.

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

1. Insert tuple ('XYZ4', 'Wellington', 'Yummy') into COMPANY

Answer: No. This is adding the location as the name and the name as the location, 'Wellington' is potentially a valid name but there may be constraints ensuring the Location is a real city.

2. Insert tuple (*null*, 'Tasty', 'Wellington') into COMPANY

Answer: No. Cid is the primary key so *null* values cannot be entered

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

Answer: No. The Cid: 'FVT35' already exists.

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

Answer: Yes. The tuple is identified by the key value (Cid) so the tuple will be deleted even though the location is not *null*.

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

Answer: Yes. This is a valid tuple in COMPANY

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

1. Insert tuple (‘55555’, *null*, ‘F15A’, 2, 99) into FRUITS

Answer: Yes. The primary key and foreign key are valid

2. Insert tuple (‘54556’, ‘Lemon’, ‘FV35’, 20, 43) into FRUITS

Answer: No. ‘FV35’ is not an instance of the foreign key Cid

3. Insert tuple (‘53557’, ‘Apple’, ‘5AB32’, 500, 1) into FRUITS

Answer: Yes. The primary key and foreign key are valid

4. Delete tuple (‘36773’, ‘Kiwi’, ‘23XY’, 50, 21) from FRUITS

Answer: Yes. The primary key and foreign key are valid

5. Delete tuple (‘46557’, ‘Apple’, ‘23XY’, 1, 21) from FRUITS

Answer: No. (‘46557’, ‘23XY’) is not an instance of the primary key {Fid, Cid}

Question 4 [25 marks]

Suppose your software company is planning to build a relational database for a new event booking system. The following relation schemas are part of the underlying database schema.

- EVENT (eventId, venue)
- CLIENT (emailAddress, name, dob, phone) with primary key {emailAddress}
- MANAGER (managerId, name) with primary key {managerId}
- BOOKING (managerId, eventId, emailAddress, date, noOfTickets, promoCode) with primary key {eventId, emailAddress, date}

The following additional constraints are known:

1. Each client may only use a single emailAddress.
2. Managers may also be clients, but may not book an event for themselves.
3. For each event booking, the noOfTickets must be specified, while promoCode may be left blank (if not available).
4. An event may have up to four venues.

Your tasks are as follows:

a) [3 marks] For the relation schema EVENT, identify all suitable candidate keys. Explain your answer.

Answer: eventId is the only suitable candidate key. There may be multiple events in the same venue, but the eventId is presumably unique to each event.

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

Answer:

BOOKING: managerId, eventId and emailAddress all suitable foreign keys. These are all keys in other relation schema. The other three relation schemas have no suitable foreign keys.

c) [3 marks] Is it possible to add an event booking to the database with the emailAddress of a client who is not listed in the CLIENT relation? Explain your answer.

Answer: Yes, emailAddress has not been specified as a foreign key of BOOKING from CLIENT.

d) **[4 marks]** If the attribute eventId would not be part of the primary key of BOOKING, what would be the consequence? Explain your answer.

Answer: A single CLIENT could potentially book the same event multiple times on the same date

e) **[5 marks]** Suppose, a client ('paula@vuw.ac.nz', 'Paula', 22/01/2000, '381-1230') in the CLIENT relation has made several bookings stored in the relation BOOKING. When deleting the record of this client from the CLIENT relation, all her bookings should be deleted, too. How would you ensure this requirement? Explain your answer.

Answer:

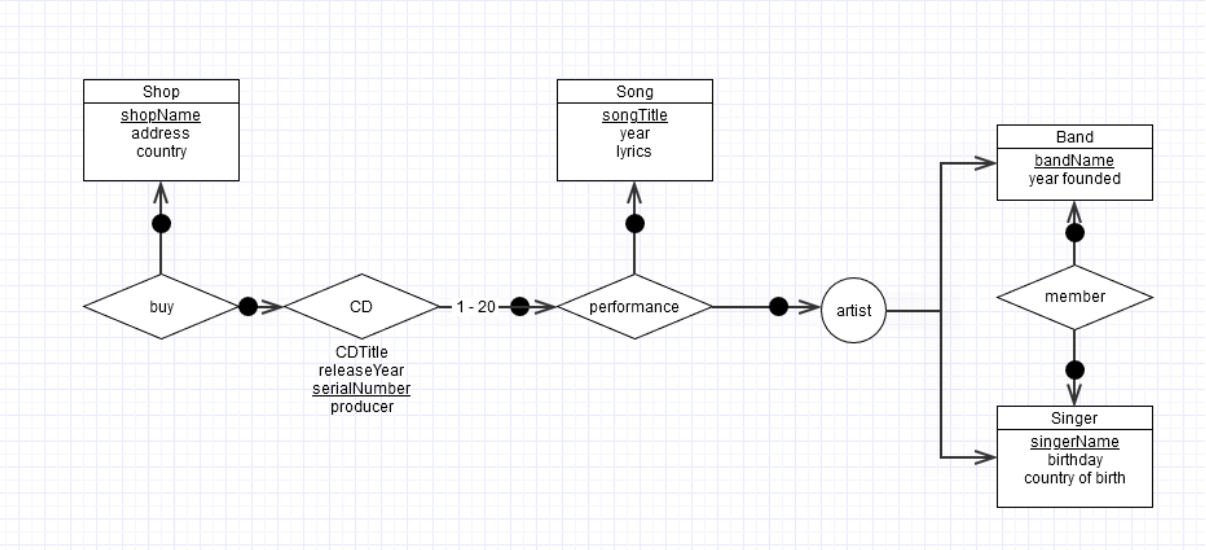
f) **[5 marks]** Suppose, a person (555, 'Mary') in the MANAGER relation is no longer working for the event booking system. When deleting the record of this manager from the MANAGER relation, all the bookings she made for clients should not be lost. How would you ensure this requirement? Explain your answer.

Answer:

Question 5 [30 marks]

You are asked to design a CD_COLLECTION database for your grandma's CD collection. A CD has a title, a release year, a unique serial number and was produced by a certain producer. A CD contains performances of songs by an artist. There are at most 20 songs on a CD. A song has a title, the year in which it was written, and the song lyrics. An artist is a singer or a band. A singer has a name (first name and last name), a birthday and a country of birth. A band has a band name and the year in which it was founded. A singer can be a member of a band. Furthermore, your grandma buys CDs at certain shops, which have a name, an address, and a country.

a) [24 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).



Level 0:

Band = ({bandName, yearFounded}, {bandName})

Song = ({songTitle, year, lyrics}, {songTitle})

Singer = ({singerName, birthday, countryOfBirth}, {singerName})

Shop = ({shopName, address, country}, {shopName})

Cluster of level 1:

Artist = Band⊕Singer

Level 1:

Member = ({Band, Singer}, {bandName, singerName}, {bandName, singerName})

Level 2:

Performance = ({Song, Artist}, {}, {Song, Artist})

Level 3:

CD = ({Performance}, {CDTitle, releaseYear, serialNumber, producer}, {Performance, serialNumber})

Level 4:

Buy = ({CD, Shop}, {}, {CD, Shop})

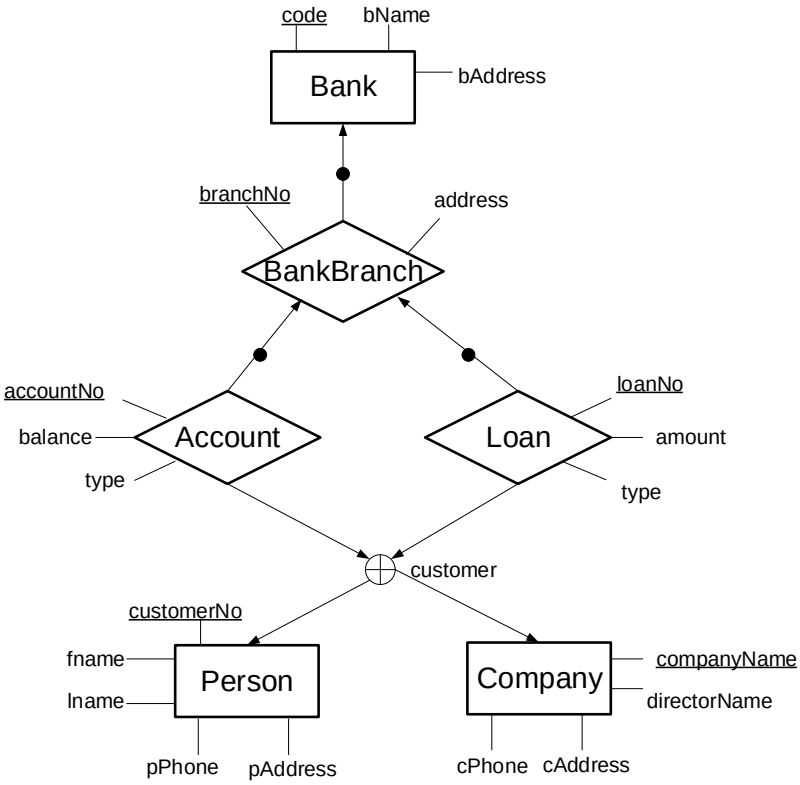
b) [6 marks] There may be information, requirements or integrity constraints that you are not able to represent in your diagram. Give three examples of integrity constraints that have not been represented in your diagram.

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.

Answer:

Question 6 [15 marks]

Consider the extended ER diagram below.



a) [5 marks] Present the extended ER schema of the extended ER diagram above.

Level 0:

Bank = ({code, bName, bAddress}, {code})
Company = ({companyName, directorName, cPhone, cAddress}, {companyName})
Person = ({customerNo, fname, lname, pPhone, pAddress}, {customerNo})

Cluster of Level 1:

Customer = Company⊕Person

Level 1:

BankBranch = ({Bank}, {branchNo, address}, {Bank, branchNo})

Level 2:

Account = ({BankBranch, Customer}, {accountNo, balance, type}, {BankBranch, accountNo})
Loan = ({BankBranch, Customer}, {loanNo, amount, type}, {BankBranch, loanNo})

b) [10 marks] Transform your extended ER schema into a relational database schema. In particular, 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.

Answer:

Bank = {code, bName, bAddress}
minimal key {code}

Company = {companyName, directorName, cPhone, cAddress}
minimal key {companyName}

Person = {customerNo, fname, lname, pPhone, pAddress}
minimal key {customerNo}

BankBranch = {branchNo, address, code}
minimal key {branchNo, code}
foreign key [code] ⊆ Bank[code]

Person_Account = {accountNo, balance, type, customerNo, branchNo}
minimal key {accountNo, customerNo}
foreign key [customerNo] ⊆ Person[customerNo] and
[branchNo] ⊆ BankBranch[branchNo]

Company_Account = {accountNo, balance, type, companyName, branchNo}
minimal key {accountNo, companyName}
foreign key [companyName] ⊆ Company[companyName] and
[branchNo] ⊆ BankBranch[branchNo]

Person_Loan = {loanNo, amount, type, customerNo, branchNo}
minimal key {loanNo, customerNo}
foreign key [customerNo] ⊆ Person[customerNo] and
[branchNo] ⊆ BankBranch[branchNo]

Company_Loan = {loanNo, amount, type, companyName, branchNo}
minimal key {loanNo, companyName}
foreign key [companyName] ⊆ Company[companyName] and
[branchNo] ⊆ BankBranch[branchNo]

****