Question One

1. **What is a relation schema and what is a relation?**

A schema is the logical representation of the data. A relation schema is table of related attributes; a set of tuples. A relation is an instance of a relation schema.

1. **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 key is a normal attribute of a table which is used to uniquely identify any two tuples in a table. Candidate key is an attribute/are attributes which can uniquely identity a specific tuple without referring to anything else in the table. Therefore, they must remain unique. From the set of candidate keys, one primary key is chosen which is the key most used for identification and as primary key values cannot be null, consequently candidate keys cannot be null and cannot contain a null value.

Foreign keys are used to connect relation schemas to each other by (primary key, foreign key) pairs. While they can contain a null value , they have to maintain referential integrity. Referential integrity is where the base table values should not be changed in any way.

1. **What is a database and what are its main features?**

A database is a collection of related data that is well structured and stored permanently. The main features of a database are:

* It should maintain an aspect of the real world, called miniworld or the universe of discourse (UoD).
* Reflects the current state of the UoD
* Is well structured
* Has users and applications
* Is stored in permanent computer memory
* Managed by a Database Management System (DBMS)

1. **What is a database management system (DBMS)? What are the tasks of DBMS?**

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

Question Two

**E = Employee**

**D = DoB**

**S = StaffNo**

**J = JobTitle**

**({E}, {D}, {S}, {J}, {E,D}, {E,S}, {E,S}, {E,J}, {D,S}, {D,J}, {S,J}, {E,D,S}, {E,D,J}, {E,S,J}, {D,S,J}, {E,D,S,J})**

1. **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 candidate keys as they cannot be used to uniquely identify tuples without referring to anything else in the table. Employees can have the same name, birthday and job title, and without referring to some other column, you cannot defer who it is.

1. **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.**

StaffNo and DoB are suitable candidate keys, as they are unique identifier as no one person can has the same staff number or date of birth in that specific schema and consequently, a user does not have to look at another column to pin point the specific tuple. Therefore, the values involving S and D from the power set above are al possible candidate keys.

1. **Which of the candidate keys identified in part b) would you choose as the primary key?**

The StaffNo would be the primary key as it is the one attribute that can be commonly used to differentiate between employees on a system. If another employee is added to the table, there is a possibility of them having the same birth date as another employee, but they would never have the same staff number unless there has been an error.

Question Three

1. **Decide which of the following tuples can be added or removed, respectively.**

1. No, because SID is a unique identifier and therefore, cannot have a null value.

2. Yes, as it still maintains referential integrity as it does not already exist.

3. No, as it does not maintain referential integrity as FVT35 is already in SUPPLIER.

4. No, as it does not exist

5. No, as if it is deleted from SUPPLIER, the values in PRODUCT will no longer be able to be accessed as the foreign key has been deleted.

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

1. Yes, as it still maintains referential integrity.

2. No as DV35 is not in SUPPLIER SID

3. Yes, as it still maintains referential integrity.

4. Yes, as there is still another foreign key, 23XY.

5. No, as it does not exist.

Question Four

1. **For the relation schema TOUR, identify all suitable candidate key (if there are any). Explain your answer.**

TourID, as it is a unique identifier, and a destination can be in multiple different tours as tours can go to up to five different places.

1. **For each of the relation schemas, identify all suitable foreign keys (if there are any). Explain your answer.**

BOOKING[emailaddress] ⊆ CUSTOMER[emailaddress]

BOOKING[staffID] ⊆ AGENT[staffID]

BOOKING[tourID] ⊆ TOUR[tourID]

Therefore, emailaddress, staffID and tourID are foreign keys, as a BOOKING cannot be created without the specific tour, customer contact, as well as the staff who made the booking.

1. **Is it possible to add a booking to the database in the email address of a customer who is not listed in CUSTOMER relation?**

No, because it’s a primary key of booking and customer and they cannot be null. Also, it is a foreign key which means that it must exist in the parent schema of CUSTOMER to be added to BOOKING.

1. **What would be the consequence of removing the attribute tourId from the primary key of BOOKING? Explain your answer.**

If it were not a primary key, there is a chance of it being a null value and therefore an error in the booking as there will be no connected tour in the booking. As there is now just emailaddress and dateFrom as primary keys of booking, a specific email address can only book one tour on the same day. Consequently, if they a customer wanted to go to two half a day tours, on the same day, in the same city, they would not be able to.

1. **Suppose a customer (‘tom@vuw.ac.nz’, ‘Tom’, 22/01/1985, ‘381-120’) 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.**

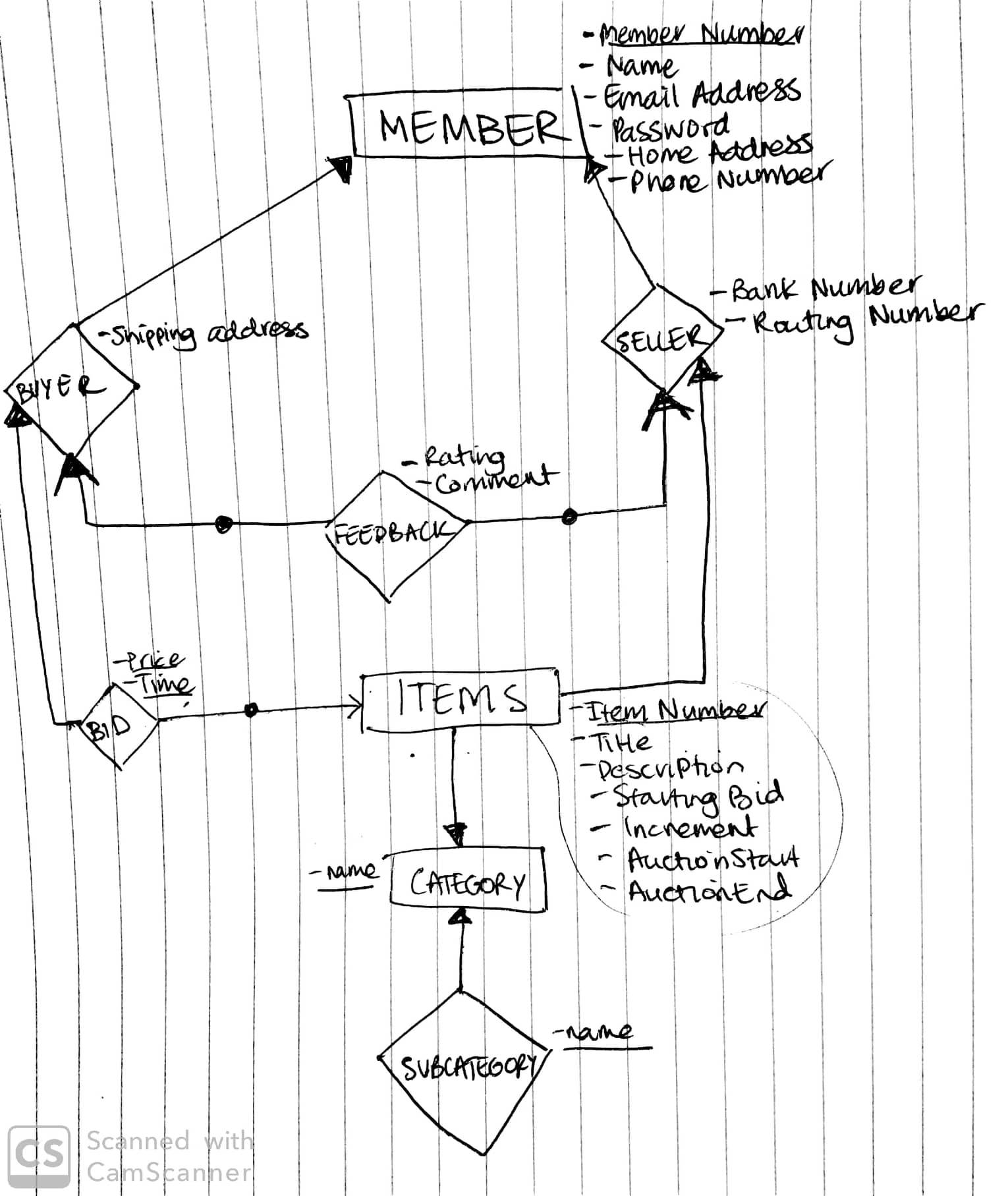
I would use ON DELETE CASCADE. This allows us everything referencing the ‘parent’ – in this case; CUSTOMER, to be deleted. Therefore, all booking associated with the CUSTOMER will go.

1. **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.**

I would use ON DELETE SET DEFAUL NULL. Doing so will allow the staffID to have a value of 0 when deleted and therefore, all the staffID’s associated with him in the bookings would turn to null and not cascade delete.

Question Five

1. **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 the relationship types (showing components, attributes and keys).**





Level 0:

**MEMBER** = {memberNumber, name, emailAddress, password, homeAddress, phoneNumber}, {memberNumber}

**CATEGORY** = {name}, {name}

Level 1:

**BUYER** = ({MEMBER}, {shippingAddress},{shippingAddress})

**SELLER**=({MEMBER},{routingNumber, bankNumber}, {routingNumber})

**ITEMS** = ({SELLER},{itemNumber, title, description, startingBid, increment, auctionStart, auctionEnd}, {itemNumber})

**BID** = ({BUYER}, {price, time}, {price, time})

SUBCATEGORY=({CATEGORY},{name}, {name})

Level 2:

**FEEDBACK** = ({SELLER, MEMBER}, {rating, comment})

1. **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.**

I assumed the auction was the system so did not represent it within this diagram. Potentially, I could have made the subcategory a foreign key connection to the main category, but chose not to do so. I assumed that there could be only one subcategory for each category. I did not account for multiple bids, and showed more of a static approach to an auction which is another limitation.

Question Six

**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.**

*Not null in italics and underlined*

**ITEM** = {*weight, value,* itemDescription, *itemID*}

Primary key: {itemID}

**CUSTOMER** = {*name, address, postcode, country, phone*}

Primary key: {name, address}

**COURIER** = {*name, phone, id*}

Primary key: {id}

**STANDARD\_ITEM** = {*dimension, weight, value,* itemDescription, *itemID*}

Primary key: {itemID}

Foreign key: [itemID] ⊆ Item[itemID]

**BULKY\_ITEM** = {*volume, shape, weight, value*, itemDescription*, itemID*}

Primary key: {itemID}

Foreign key: [itemID] ⊆ Item[itemID]

**SHIPMENT** = {*status, date\_sent, date\_recieved, name, phone, id, name, address, postcode, country, phone*}

**RELATED\_EXPENSE** = {*description, date, cost*}

Primary Key: {description}

Foreign key: [date] ⊆ Shipment[date\_sent]

**SHIPMENT\_OF\_STANDARD\_ITEMS** = {*status, date\_sent, date\_recieved, dimension, itemID,* *value*, itemDescription, *weight*}

Foreign key: [itemID] ⊆ item[itemID]

**SHIPMENT\_OF\_BULKY\_ITEMS** = {*status, date\_sent, date\_recieved, shape, volume, itemID,* *value*, itemDescription, *weight*}

Foreign key: [itemID] ⊆ Item[itemID]