Name: Maniram chaudhary

Roll No: 28

Subject: Database Management System

2013. Spring

Q. No 2(a) Consider the following relation R & S

R

Sid	SName	Marks(%)
S001	Hari	85
S002	Sita	78
S003	Bidur	85
S005	Vinod	68

S

Sid	SName	Marks (%)
S004	Sarita	76
S003	Bidur	85
S006	Shyam	75
S005	Vinod	68

i) Show the id and name of those students whose marks are less than 80 from relation schema R.

→

$\pi_{sid,\ SName}(\sigma_{Marks<80}(R))$

ii) Write the results

a. R ∪ S

→

Sid	SName	Marks(%)
S001	Hari	85
S002	Sita	78
S003	Bidur	85
S005	Vinod	68
S004	Sarita	76
S006	Shyam	75

b. R − S

→

Sid	SName	Marks(%)
S001	Hari	85
S002	Sita	78

c. $\pi_{SName}(\sigma_{Marks=85}(S))$

SName
Bidur

Q.3 (b) What do you mean by Normalization & why it is necessary? Explain about 3NF and 4NF with suitable example.

→ Normalization is a database design technique that reduces data redundancy and eliminates undesirable characteristics like Insertion, update & deletion anomalies.

Normalization is necessary due to following reasons

- ↳ To minimize redundancy
- ↳ To enhance flexibility.
- ↳ It improves in data consistency
- ↳ enforces the concept of relational integrity.

* 3NF

3NF is based on the concept of transitive dependency. A functional dependency $X \rightarrow Y$ in a relation schema 'R' is a transitive dependency if there exists a set of attributes 'Z' in R that is neither a candidate key nor a subset of any key of R, and both $X \rightarrow Z$ & $Z \rightarrow Y$ hold.

According to Codd's, "A relation shema 'R' is in 3NF if it satisfies 2NF & no nonprime attribute of 'R' is transitively dependent on the primary key".

For example:

Student_Detail

Sid	SName	Address	Fid	FName	Course
001	Manoj	Gulariya	20103	Civil	Irregation
002	Shradha	Attariya	20104	Computer	Web Dev
003	Gauri	Attariya	20105	BCA	Accountancy
004	Manoj	Dhangadhi	20106	Computer	Web Dev

This is not in 3NF because it hold transitive devendency

Here,

Sid → FName → Course

FName → Course

To make it in 3NF we decompose and remove the transitive dependency. So, we convert the given table in 3NF decompos two sub table suchas:-

Student

Sid	SName	Address	Fid	FName
001	Manoj	Gulariya	20103	Civil
002	Shradha	Attariya	20104	Computer
003	Gauri	Attariya	20105	BCA
004	Manoj	Dhangadhi	20106	Computer

Course

FName	Course
Civil	Irregation
Computer	Web Dev
BCA	Accountancy

4NF

In 4NF, there are no non-trival multivalued dependencies other than a candidate key. Mutivalued dependencies occurs when two or more independent multi valued facts about the same attribute occurs within the same relation.

MVD is denoted by A→ B, that means, if for a single value of A, multiple values of B exist then the relation will be multi-valued dependency.

For eg:

Info

Country	Student	Club
Nepal	Money	IT
Nepal	Money	CFC
Nepal	Kaustubh	CFC
Nepal	Kaustubh	IT

In above example, Info table is in 3NF, but the student and club are two independent entity. Hence there is no relationship between student & club.

From table 'Info'

Nepal → → IT

Nepal → → CFC

Now, we can decompose it and it will become 4NF.

Info_Student

Country	Student
Nepal	Money
Nepal	Kaustubh

Info_Club

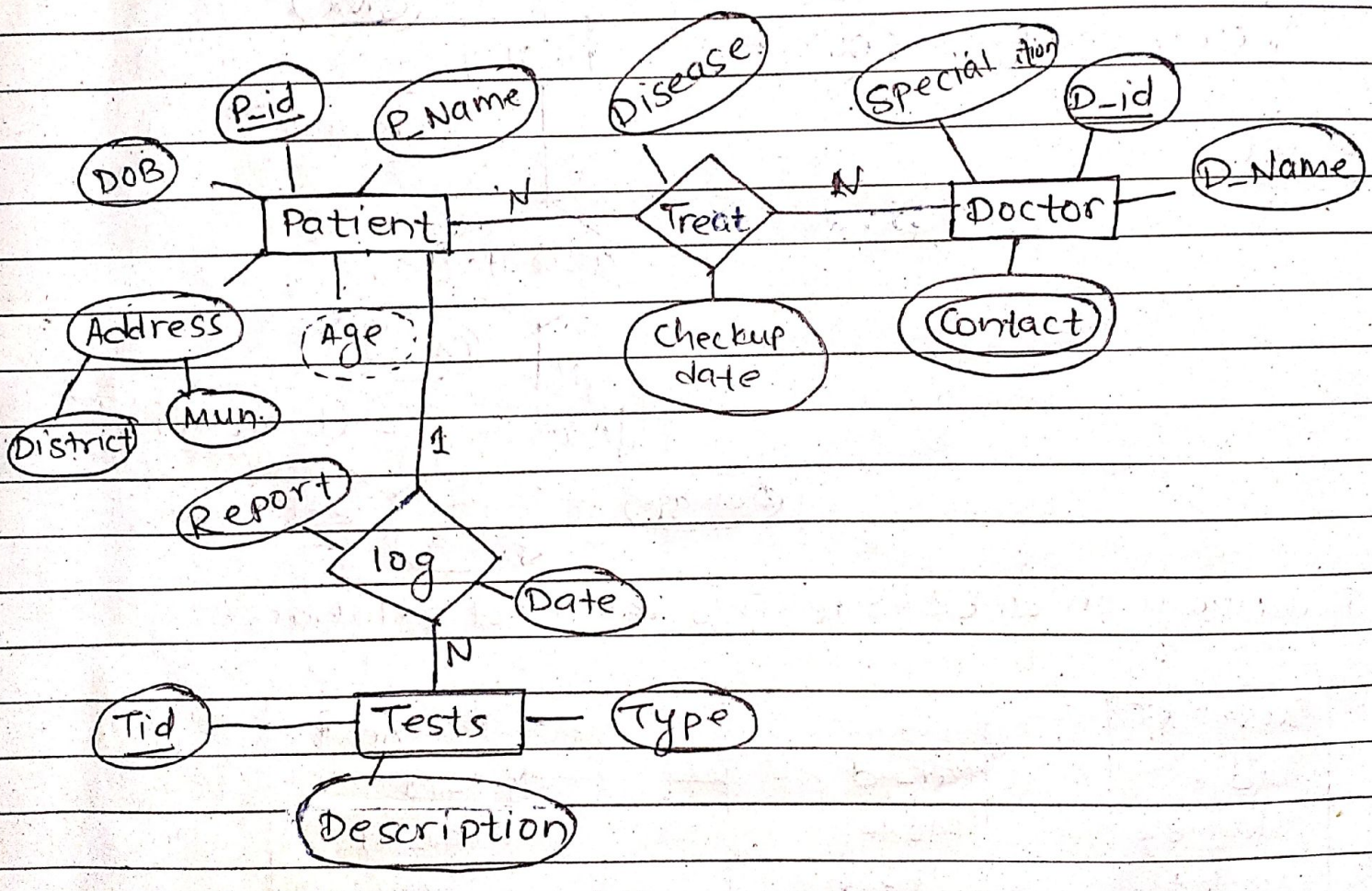
Country	Club
Nepal	IT
Nepal	CFC

2014 Fall

Q. 1 (b) Draw an E-R diagram for the database of a hospital with a set of patients & set of medical doctors. With each patient a log of the various tests conducted is also associated. Make your own assumptions if necessary.

→

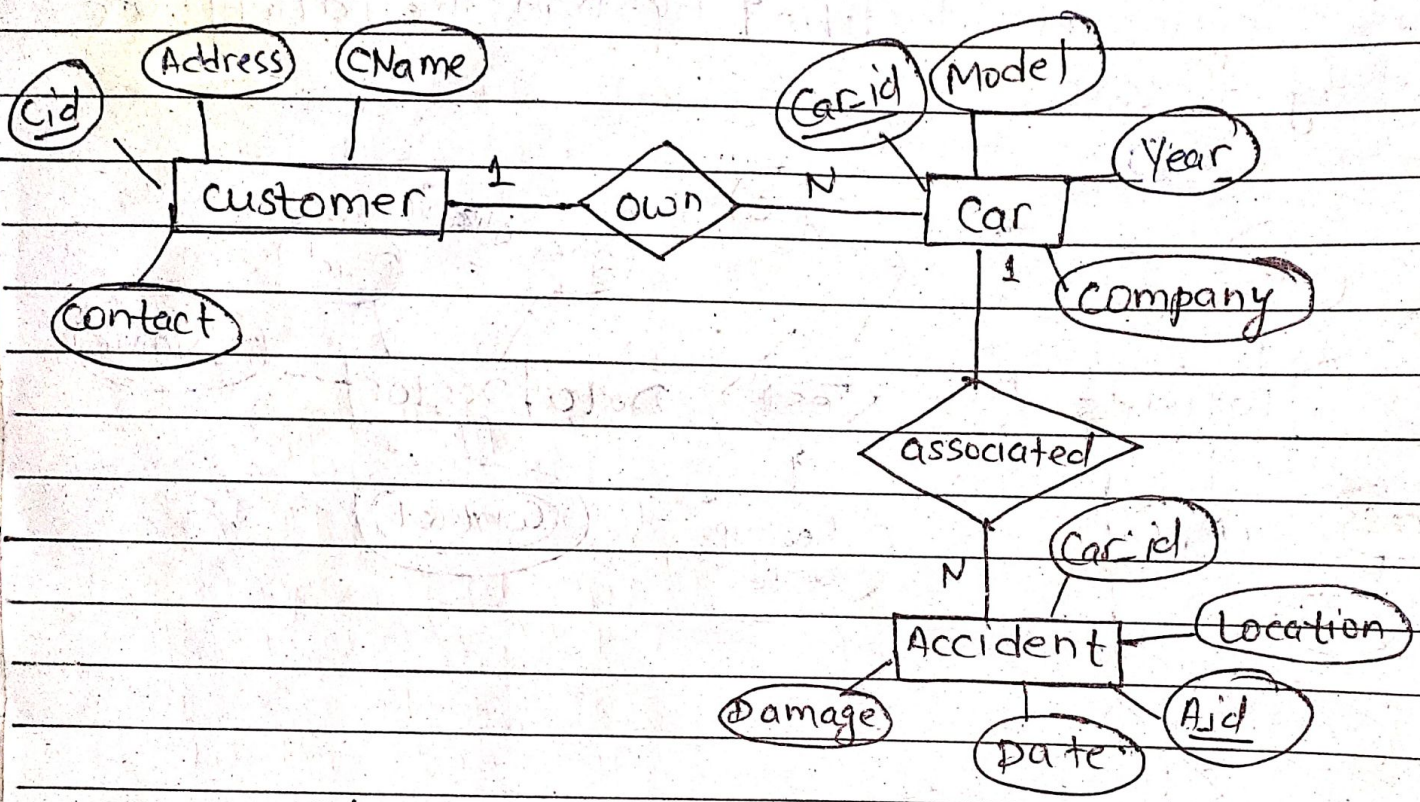
We assume, patients are uniquely identified by P-id and doctors are uniquely identified by D-id. The patients are tested by doctors.



2016 Fall

1(b) Construct E-R model for a car insurance company whose customer own one or more cars each. Each car has associated with it zero to any number of recorded accidents. Also, design a relational database corresponding to the E-R diagram.

→



* Decomposition of ER model into relational database:

Customer
Cid
CName
Contact
Address

Car
Car_id
Model
Year
Company

Accident
A-id
Car_id
location
Damage
Date

Q.2(a) Consider a student registration database comprising of the below given schema.

Student (CRN, Name, Gender, Address, Telephone)
Course (CourseID, CourseName, Hour, TeacherID)
Teacher (TeacherID, TeacherName, Office)
Registration (CRN, CourseID, Date)

Write relational algebraic expression for the following tasks;

i) Count the number of student registered subject in year 2015 gender wise.

→

$$R_1 \leftarrow \left(\left(\sigma_{Date=2015}(Registration)\right) \bowtie Student\right)$$

$$G_{count(Gender)}(R_1)$$

ii) Show student details taught by teacher Rohit Shrestha.

→

$$R_1 \leftarrow \left(\sigma_{TeacherName="Rohit\ Shrestha"}(Teacher \bowtie course)\right)$$

$$R_2 \leftarrow Registration \bowtie R_1$$

$R_1 \leftarrow \sigma_{Teacher.TeacherID = course.TeacherID}(Course \bowtie Teacher)$

$\Rightarrow \sigma_{TeacherName="Rohit Shrestha"}(Student \bowtie Registration \bowtie R_1)$

iii) Delete student information taught by teacher N. Mathema.

→ $R_1 \leftarrow Course \bowtie Teacher$

$Student1 \leftarrow Student - \sigma_{TeacherName="N. Mathema"}(Student \bowtie Registration \bowtie R_1)$

2014 Fall

Q.3 (a) Explain BCNF

→ Boyce-Codd Normal Form (BCNF) is based on functional dependencies that take into account all candidate keys in a relation. It is advance version of 3NF and stricter than 3NF.

A table is in BCNF if every functional dependency $X \rightarrow Y$, X is the superkey of the table. For BCNF, the table should be in 3NF and for every Functional dependencies left hand side is superkey.

For eg:

EmpId	Country	Emp_dept	deptType	emp_DeptNo.
264	Nepal	Designing	D394	283
264	Nepal	Testing	D394	300
364	Ching	Store	D283	232
364	China	Developing	D283	549

Functional dependencies are as follows:

EmpId → Country

Emp_dept → {deptType, emp_DeptNo}

Candidate key: {EmpId, Emp_dept}

The table is not in the BCNF because neither EmpId nor Emp_dept alone are keys to convert the given table into BCNF. We decompose it into 3 tables.

Emp-Country

EmpId	Country
264	Nepal
364	China

Emp_dept

Emp_dept	deptType	EmpDeptNo
Designing	D394	283
Testing	D394	300
Store	D283	232
Developing	D283	549

Emp-dept map

EmpId	Emp_dept
D394	283
D394	300
D283	232
D283	549

Now, Functional dependencies:-

EmpId → Country

Emp_dept → {deptType, empDept No}

Candidate keys:

For 1st table: EmpId

2nd table: Emp_dept

3rd table: {EmpId, Emp_dept}

Now, this is in BCNF because left side part of both the functional dependencies are keys.

2015 Spring

Q. 1(b) Suppose you are given the following requirements for a simple database for the National Cricket League (NCL):

* NCL has many teams.
* each team has a name, a city, a coach, a captain, & a set of players.
* each player belongs to only one team.
* each player has a name, a type (such as batsman or bowler) a skill level, and a set of records.
* team captain is also a player.
* each player is sponsored by at least one brand.
* a brand has its name, established date, property, multiple contact no.

Construct a clean & concise ER diagram for the NCL database. List your assumptions and clearly indicate the cardinality mapping as well as any role indicators in your ER diagram.

Sol:-

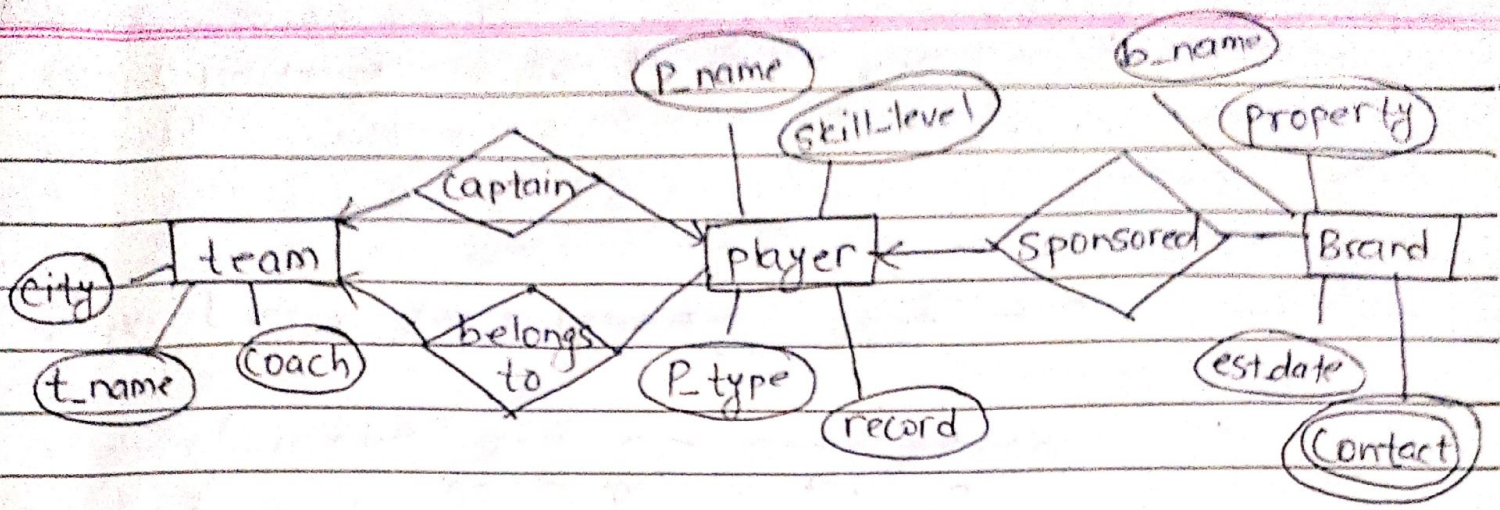


Fig: E-R diagram

Q. 2 (a) Consider the relational database of figure below, where the primary key are underlined. Give an expression in the relational algebra to express each of the following queries:

employee (person_name, street, city)
works (person_name, bank_name, salary)
bank (bank_name , city)
manages (person_name, manager_name)

i) Find the names of employees in this database who live in the same city as the company for which they work.

$$\Rightarrow$$

$$\pi_{\text{person\_name}}\left(\sigma_{\text{works.city}=\text{bank.city}}\left((\text{employee} \bowtie \text{works}) \bowtie \text{bank}\right)\right)$$

ii) Give all employees of first Bank Corporation a 10 percent salary raise.

⇒

$R_1 \leftarrow (works - \sigma_{bank.name = "First\ Bank\ Corporation"}(works))$

$R_2 \leftarrow (\sigma_{bank.name = "First\ Bank\ Corporation"}(works))$

$works \leftarrow (\pi_{person\_name, bank\_name, 1.1 * salary}(R_1 \cup R_2))$

iii) Modify the database so that Harish now lives in Biratnagar.

⇒

$R_1 \leftarrow (employee - \sigma_{person\_name = "Harish"}(employee))$

$R_2 \leftarrow (\sigma_{person\_name = "Harish"}(employee))$

$employee \leftarrow (\pi_{person\_name, street, "Biratnagar"}(R_1 \cup R_2))$

iv) Delete all tuples in works relation for employee of first Bank Corporation

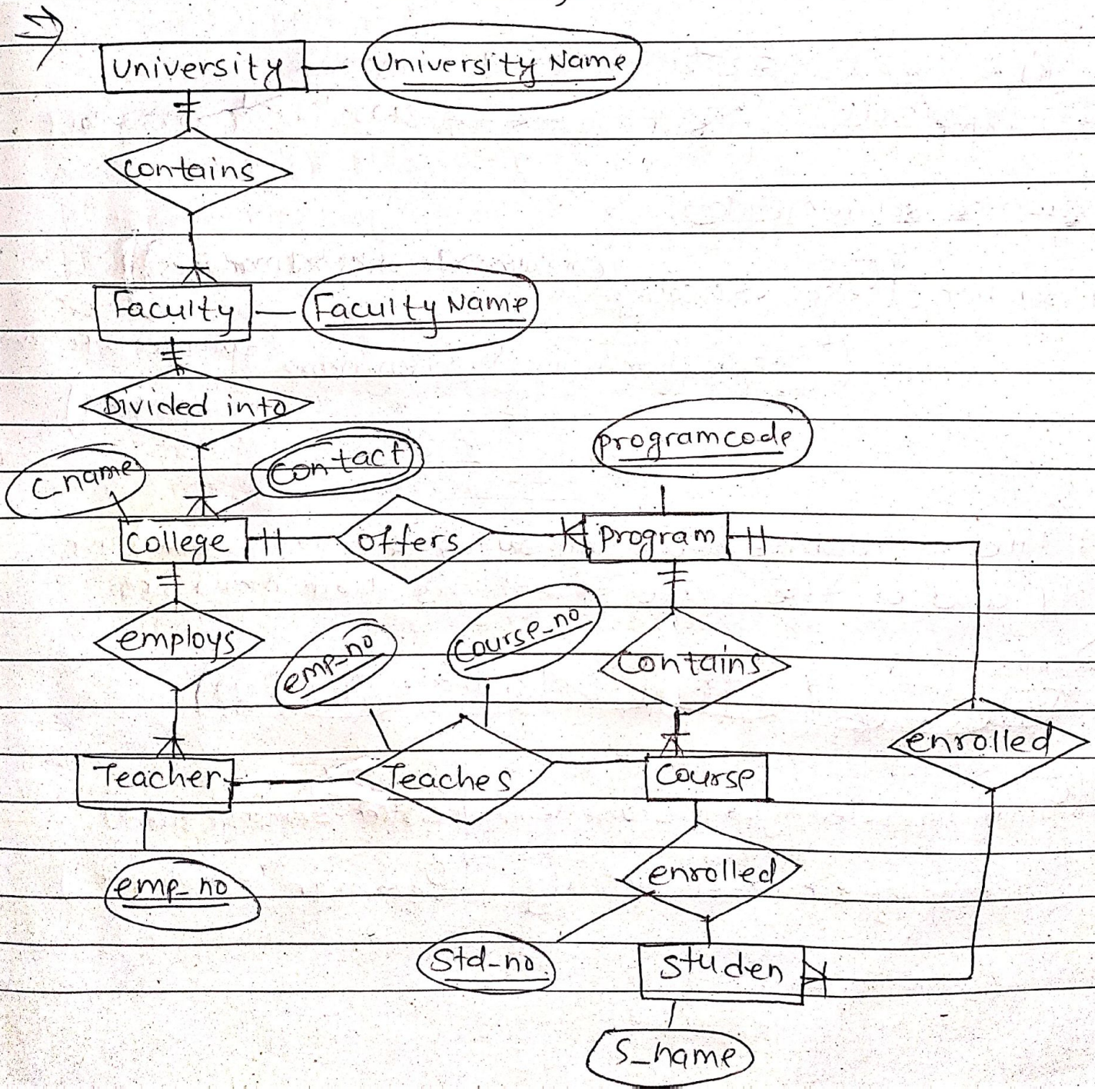
⇒

$works \leftarrow (works - (\sigma_{bank\_name = "First\ Bank\ Corporation"}(works)))$

2017 Fall

Q. 1(b) Draw an ER diagram for following scenario:

A university contains many faculties. The faculties in turn are divided into several colleges. Each college offers numerous programs and each program contains many courses. Teachers can teach many different courses & even the same course numerous times. Courses can also be taught by many teachers. A student is enrolled in only one program but a program can contain many students. Students can be enrolled in many courses at the same time & the courses have many students enrolled.



Q.2 (a) Consider the following schema:

employee (person_name, street, city)
works (person_name, company_name, salary)
company (company_name, city)
manages (person_name, manager_name)

Give an expression in relational algebra to express each of the following queries:

a) Find the names of all employees who earn more than their managers

⇒ ~~R ← employee ⋈ works ⋈ manages~~

~~$\Pi_{person\_name}(works) - \{\Pi_{works.person\_name}(works \bowtie (works.salary < works2.salary \wedge works2.$~~

$\Pi_{E.person\_name}(\sigma_{E.salary > M.salary}(employee \bowtie works \bowtie manages) \bowtie_{E.manager\_name = M.person\_name}(employee \bowtie works \bowtie manages))$

b) Find the names of all employees who live in the same city and on the same street as their managers.

⇒

~~$\Pi_{person\_name}(employee \bowtie manages) \bowtie R1$~~

$R1 \leftarrow (\sigma_{manager\_name = empmgr.person\_name \wedge employee.street = empmgr.street \wedge employee.city = empmgr.city}(\rho_{empmgr}(employee)))$

$\Pi_{person\_name}(employee \bowtie manages) \bowtie R_1$

c) Find the names of all employees within the database that do not work for "NBL company"

$\Rightarrow$

$$\left(\Pi_{person\_name}(works)\right) - \left(\sigma_{company\_name="NBL\ Company"}(works)\right)$$

d) Find the names of all employees in the database who earn more than top earner at "NBL company" in the database.

$\Rightarrow$

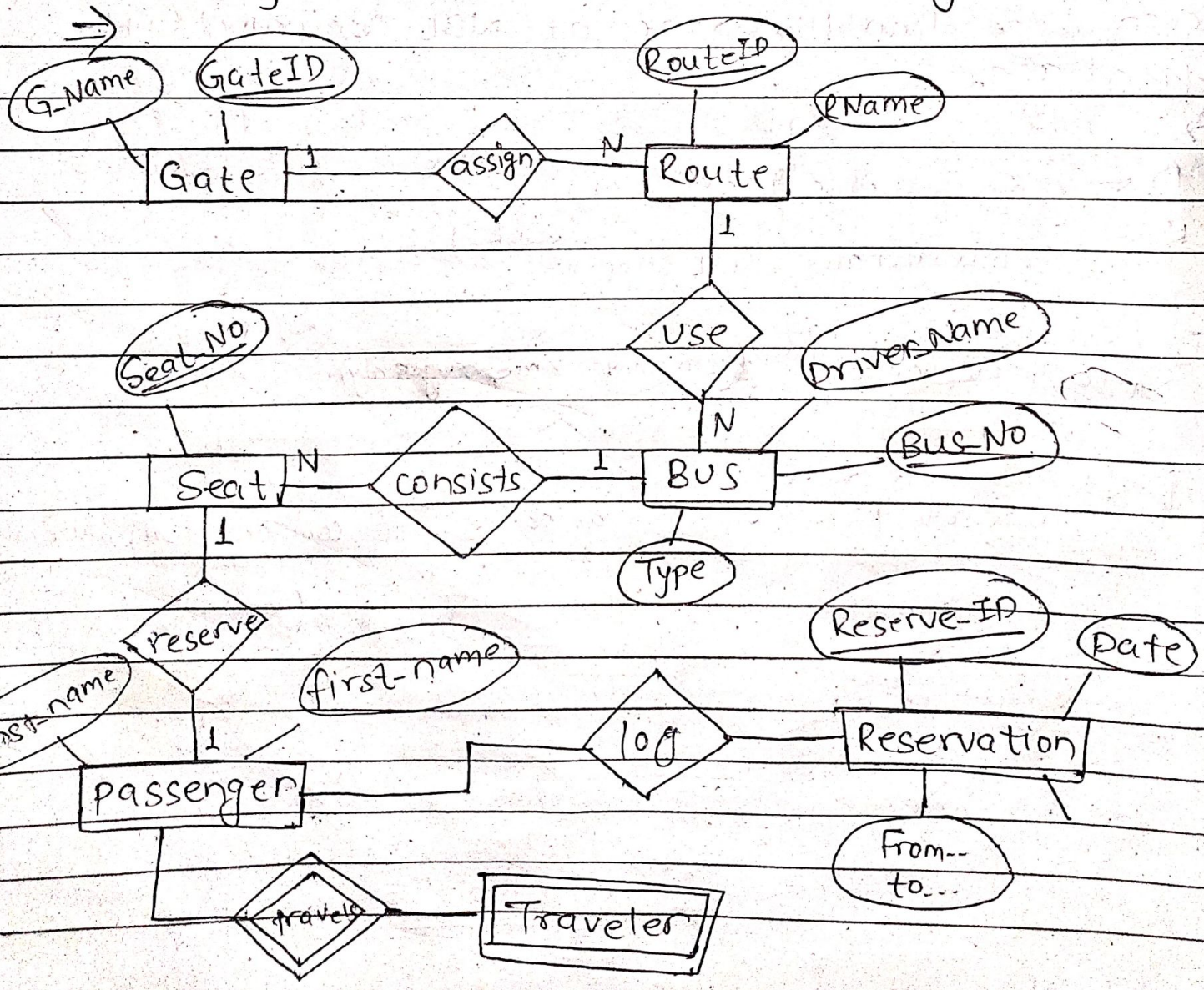
$R_1 \leftarrow {}_{company\_name}\mathcal{G}_{max.Salary}(works)$

$R_2 \leftarrow \sigma_{company\_name="NBL\ Company"}(R_1)$

$$\Pi_{R_3.company\_name}\left(\left(\rho_{R_3}(company\_name, max\_salary)^{(R_1)}\right) \bowtie_{t_3.max\_salary > NBLcompany.max\_salary} \left(\rho_{NBL\ company}(company\_name\ max\_salary^{(R_2)})\right)\right)$$

2018 Spring

Q.1(b) Construct an ER diagram for Metropolitan Bus Park. There are many gates for entering bus park. Different gates are assigned to different routes. A route use different buses. Bus consists of different seats which are assigned to different passengers. Frequent travelers are also in passenger. Associate a log of reservation date while reserving seats. The passenger name must have two attributes first_name & last_name. Each of the entities must have primary key attribute as far as possible. The cardinality mapping should be explained properly.



Q.2(a) Consider the relational database model:

Users(uid, cname, city)

Items(itemid, itemname, city, quantity, price)

Manager(mid, aname, city)

Query(queryno, uid, mid, itemid, query_datails, hitratio)

Write the relational algebraic expression for the following tasks:

i) Find all (queryno, uid) pairs for query with a hitratio value greater than 500.

$$\Rightarrow \pi_{\text{queryno, uid}} \left(\sigma_{\text{hitratio} > 500} (\text{Query}) \right)$$

ii) Find all item names of items in Pokhara ordered with query-datails as pokhara_details.

$$\Rightarrow \pi_{\text{itemname}} \left(\sigma_{\text{city='Pokhara'} \wedge \text{query\_details='Pokhara\_details}} (\text{Items} \bowtie \text{Query} \bowtie \text{Users}) \right)$$

iii) Find itemids of items ordered through manager 35 but not through manager 27.

$$\Rightarrow \pi_{\text{itemid}} \left(\left(\sigma_{\text{mid}=35} (\text{Query}) - \left(\sigma_{\text{mid}=27} (\text{Query}) \right) \right) \bowtie \text{items} \right)$$

2018 Fall

Q.1(b) Suppose you are given the following requirements for a simple database for the Employee management System.

i) An employee may work in up to two departments or may not be assigned to any department.

ii) Each department must have one & may have up to three phone numbers.

iii) Each department can have anywhere between 1 & 30 employees.

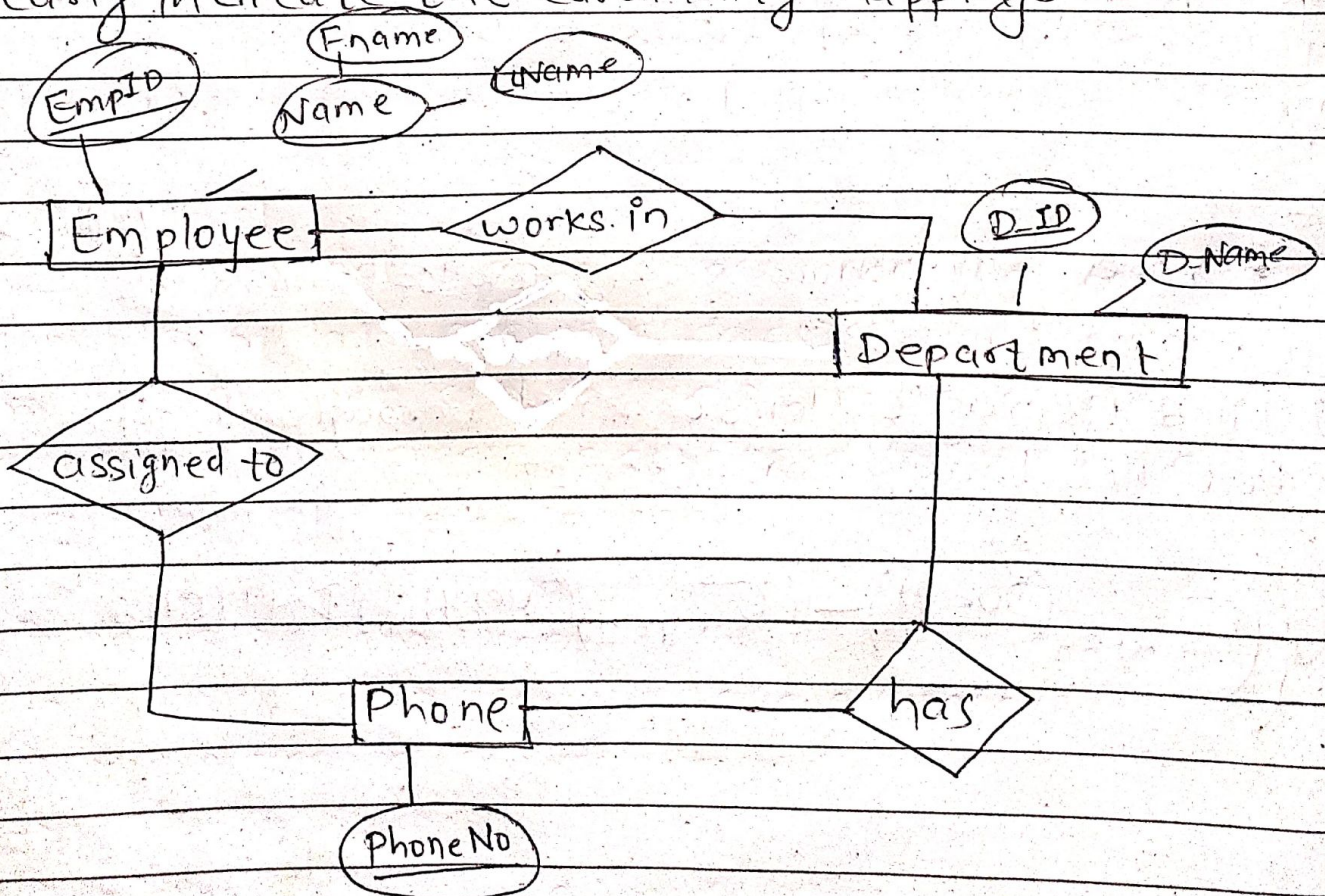
iv) Each phone is used by one and only one department.

v) Each phone is assigned to at least one, & may be assigned up to 30 employees.

vi) Each employee is assigned at least one, but no more than 5 phones.

Construct a clean & concise ER diagram for the database. Clearly indicate the cardinality mappings.

⇒



Q. 2(a) Consider the following relational schema :

* Department (DepartmentID, DepartmentName)
* Designation (DesignationID, DesignationName, Salary)
* Employee (EmpID, EmpName, Gender, DesignationID, DepartmentID)
* Allowance (AllowanceID, AllowanceName)
* Allowance Details (DetailID, EmpID, AllowanceID, Amount)

Write the relational algebraic expression for the following tasks:

i) Find the number of employees department-wise.

⇒

$$R_1 \leftarrow \text{Employee} \bowtie_{\text{Department.DepartmentID} = \text{Employee.DepartmentID}} \text{Department}$$

$$\left(\pi_{\text{DepartmentID, DepartmentName, count(EmpID)}} (R_1) \right)$$

ii) List the employee details whose total salary is above Rs. 50,000.

⇒

$$R_1 \leftarrow \text{Employee} \bowtie_{\text{Employee.DesignationID} = \text{Designation.DesignationID}} \text{Designation}$$

$$\pi_{\text{EmpID, EmpName, Gender, DesignationID, DepartmentID}} \left(\sigma_{\text{Salary} > 50000} (R_1) \right)$$

iii) List the employee those who are getting house allowance.

⇒

$$R_1 \leftarrow \left(\sigma_{\text{AllowanceName} = \text{'HouseAllowance'}} \left(\text{Employee} \bowtie \text{AllowanceDetails}\right)\right)$$

$$\Pi_{\text{EmpID, EmpName, Gender, DesignationID, DepartmentID}} (R_1)$$

2020 Fall

Q. 1 (b) Define & explain benefits of data model. Draw an ER-diagram for a Vehicle Management System including primary key, weak entity, composite attribute, derived attribute & multivalued attributes in your ER diagram.

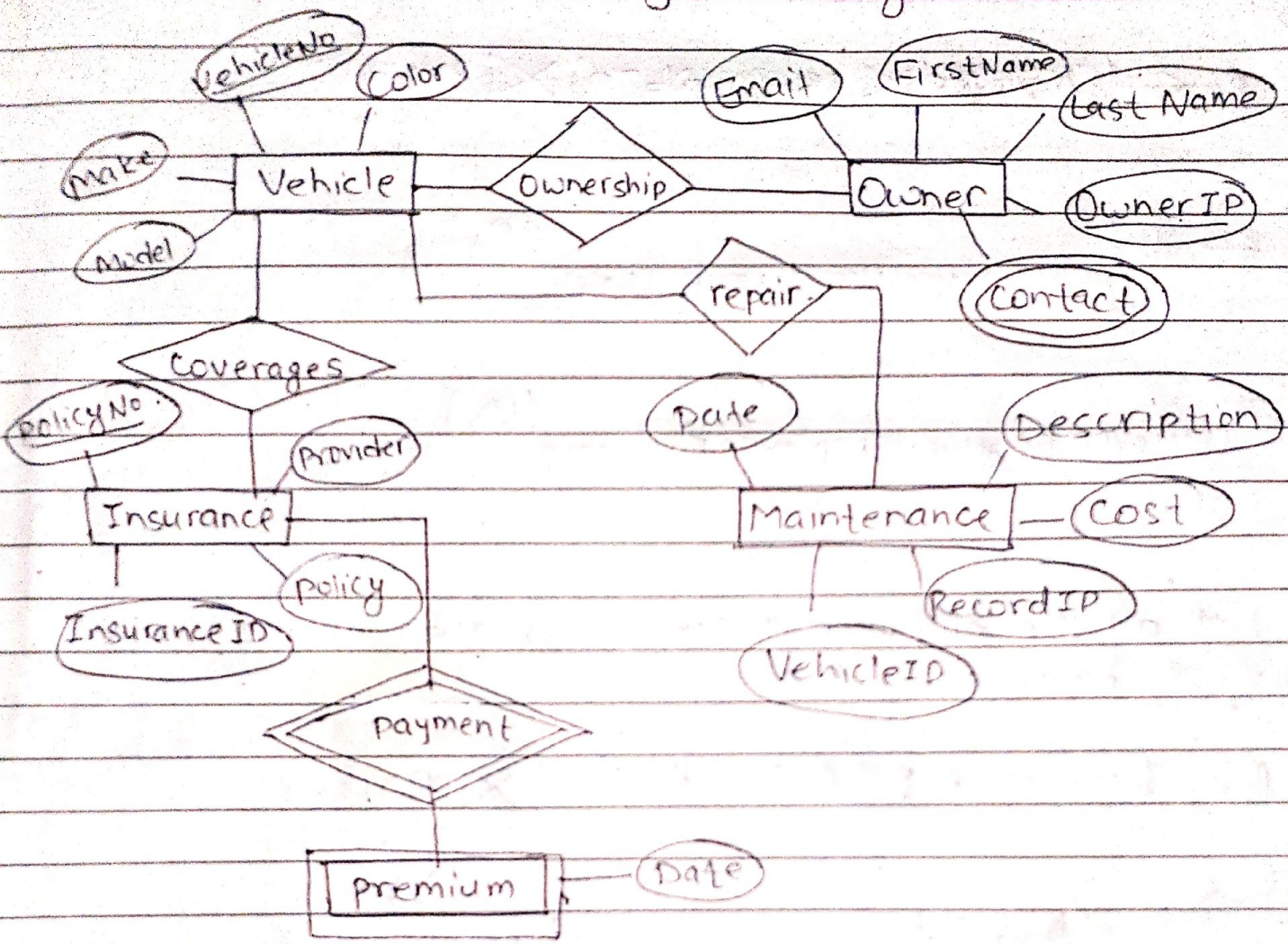
⇒

Data Model

Data model is an abstract model that organizes elements of data and standardizes how they relate to one another and to the properties of real-world entities. This concept is developed to summarize the description of the database.

* Benefits of Data Model
  - → Help us in representing data accurately.
  - → Minimizes data redundancy.
  - → provides security
  - → defines relationship among entities.

Vehicle Management System



2019 Spring

2(a) Using the following schema represent the following queries using relational algebra.

PROJECT (projectnum, projectName, ProjectType, ProjectManager)
EMPLOYEE (EmpNum, EmpName)
ASSIGNED_TO (ProjectNum, EmpNum)

(i) Find Employee detail working on a projectname starts with 'L'.

⇒

$R_1 \leftarrow (EMPLOYEE \bowtie ASSIGNED\_TO) \bowtie PROJECT$

$R_2 \leftarrow \sigma_{PROJECT.ProjectName\ LIKE\ 'L\%'}(R_1)$

$\pi_{EMPLOYEE.EmpNum,\ EMPLOYEE.EmpName}(R_2)$

(ii) List all the employee details who are working under project manager "Rohal"

⇒

$R_1 \leftarrow (EMPLOYEE \bowtie ASSIGNED\_TO) \bowtie PROJECT$

$R_2 \leftarrow \sigma_{PROJECT.ProjectManager = 'Rohan'}(R_1)$

$\pi_{EMPLOYEE.EmpNum,\ EMPLOYEE.EmpName}(R_2)$

(iii) List all the employees who are still not assigned with any project.

⇒

$R_1 \leftarrow \left(\pi_{EMPLOYEE.EmpNum,\ EMPLOYEE.EmpName}(ASSIGNED\_TO \bowtie EMPLOYEE)\right)$

$R_2 \leftarrow EMPLOYEE - R_1$

$$\Pi_{EMPLOYEE.EmpNum,\ EMPLOYEE.EmpName}\ (R_2)$$

(iv) List all the employees who are working in more than one project.

$$\Rightarrow \Pi_{EmpNum,\ EmpName}\left(\sigma_{count(ProjectNum)>1}\left(EMPLOYEE \bowtie ASSIGNED\_TO \bowtie PROJECT\right)\right)$$

Q. 4(a) Define 3NF. Convert the following 2NF relation to 3NF (consider Name as primary key)