Assignement - 2

@ 2019 Fall

1) a) How does UML diagram asist during data modelling.

ans UML diagram assist during data modeling by :-

i) UML diagrams offer a clear & intuitive visual representation of the data model.

ii) UML provides a standardized notation, making it a common language for expressing data models.

iii) UML includes a subset dedicated to data modeling known as Entity - Relationship Diagrams.

iv) Class diagrams in UML are often used for data modeling representing class (entites) attributes and associations (relationship

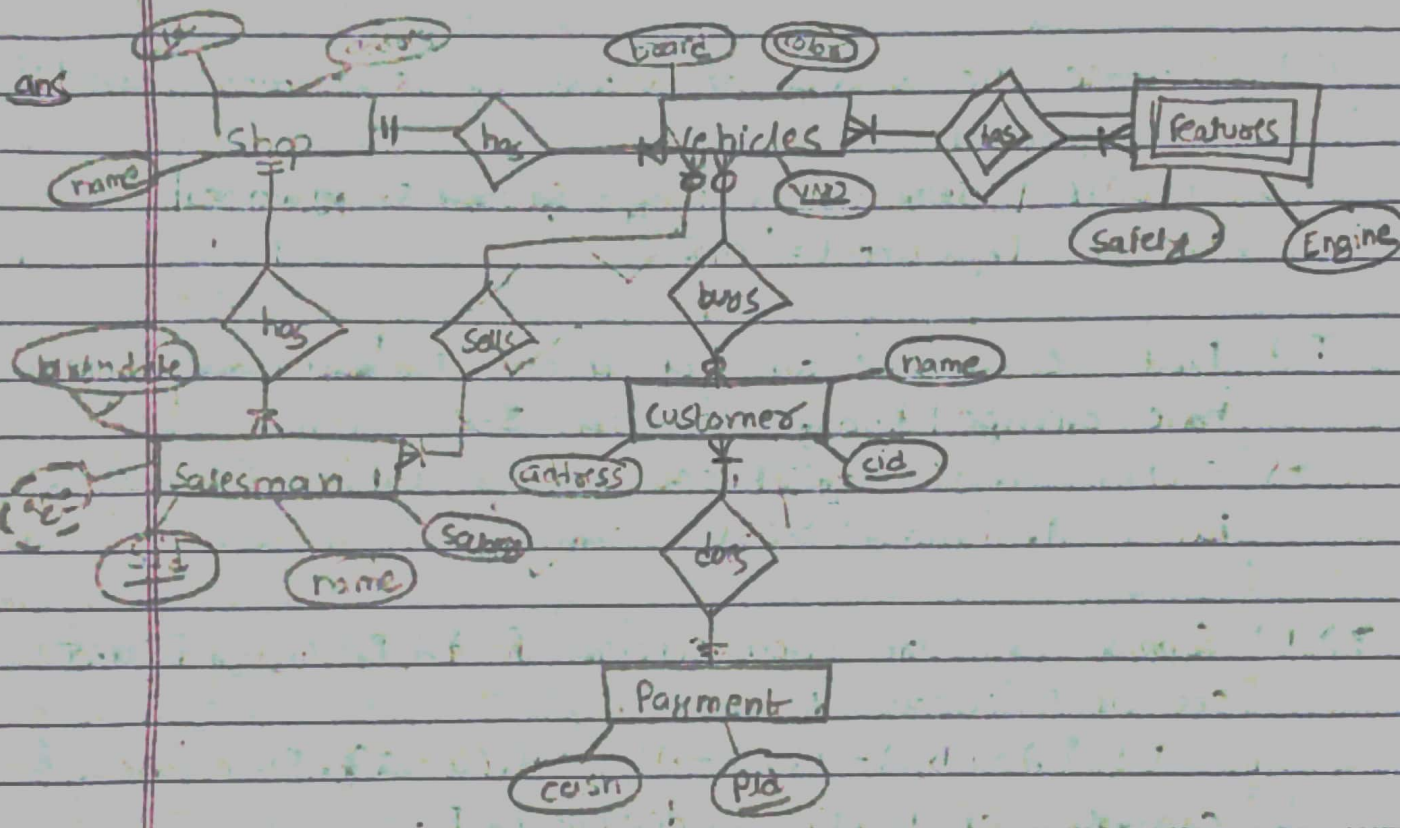
~~Assignment 3~~

Date. _____
Page No. _____

~~Solved~~ Ans:

1) a) Draw an E-R diagram for a Gandaki Auto Vehicle shop System including primary key, weak entity, composite attribute, derived attribute & multivalued attributes in your ER diagram.

ans



2)

b) Doctors (DoctorID, DoctorName, Department, Address, Salary)
Patients (PatientID, Patient Name, Address, Age, Genders)
Hospitals (PatientID, DoctorID, HospitalName, Location)

i) Display ID of patient admitted to hospital at pokhara & whose name ends with 's'.

=> SELECT patientID FROM Patients
INNER JOIN Hospitals
ON Patients.patientID = Hospitals.DoctorID
WHERE location = "pokhara" AND PatientName LIKE '%S'

Date
Page No.

(ii) Delete the records of Doctors whose salary is greater than average salary of doctors

=> DELETE FROM Doctors WHERE salary > (SELECT AVG(salary) FROM Doctors)

(iii) Increase the salary of doctors by 18.5% who works in OPD department.

=> UPDATE Salary SET Salary = Salary + Salary x 0.185
WHERE Department = "OPD"

(iv) Find the average salary of Doctors for each address who have average salary more than 35k.

=> SELECT AVG(salary) FROM Doctors GROUP BY address having AVG(salary) > 35k

5) b) Suppose we are given schema $R = \{A, B, C, G, H, I\}$ and set of functional dependencies:
$F = \{A \to B, A \to C, CG \to H, B \to H, CG \to I\}$. Find the closures of functional dependency F.

ans The closures of functional dependency F are:

Since, $A \to B$, $B \to H$
$\Rightarrow A \to H$ [$\because$ Transitivity Rule]

since, $A \to B$, $A \to C$
$\Rightarrow A \to BC$ [$\because$ Union Rule]

Since, $CG \to H$, $CG \to I$
$\Rightarrow CG \to HI$ [$\because$ Union Rule]

Date.
Page No.

Since, $A \to C$, $CG \to H$

$\Rightarrow AG \to H$ [∵ Pseudo Transitivity Rule]

Since, $A \to C$, $CG \to I$

$\Rightarrow AG \to I$ [∵ Pseudo Transitivity Rule]

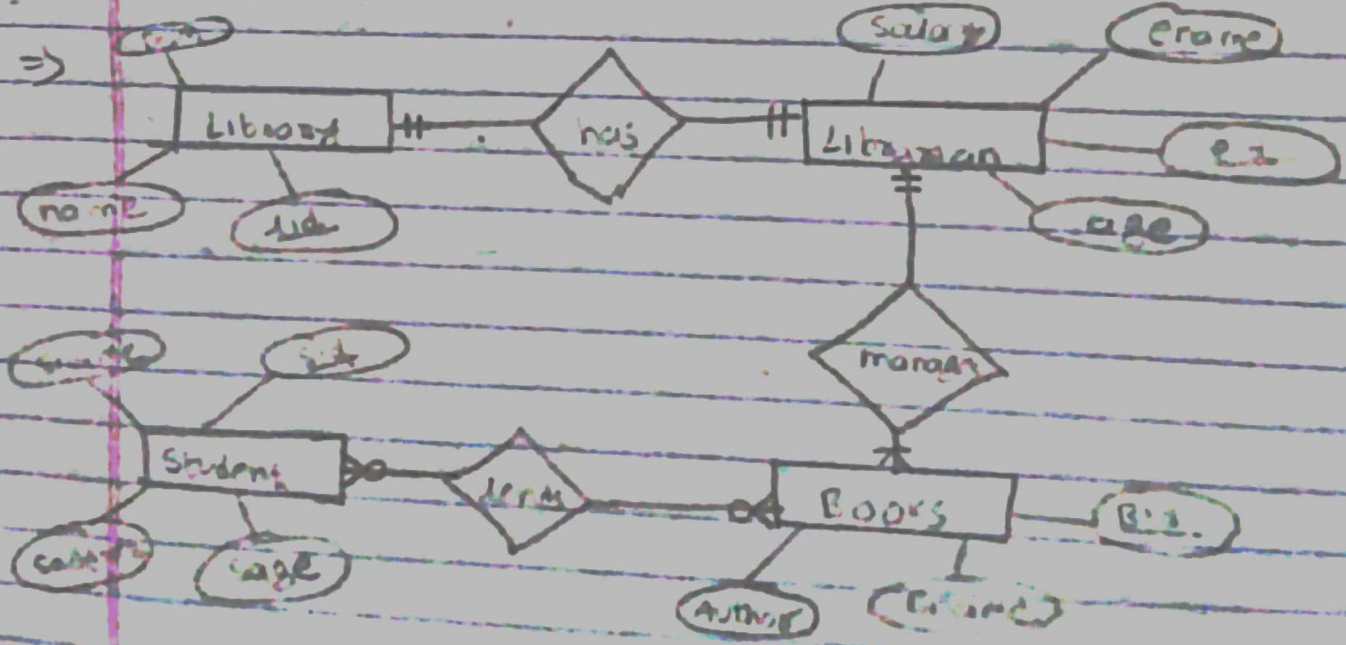
$\therefore F^+ \Rightarrow \{A \to H, A \to BC, CG \to HI, AG \to H, AG \to I\}$

- 2019 Spring

2)

b) Construct an ER diagram for keeping records for Library Management Systems.

=>



Date.
Page No.

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

PROJECT (Project num, ProjectName, ProjectType, Project Manager)

EMPLOYEE (Empnum, Empname)

ASSIGNED-TO (Projectnum, Empnum)

i) Find Employee details working on a project name starts with 'L'

$\Rightarrow \pi_{Empnum, Empname}(\sigma_{projectnum\ LIKE\ "L\%"}(PROJECT \bowtie (ASSIGNED\text{-}TO \bowtie EMPLOYEE)))$

ii) List all the Employee details who are working under project manager "Rohan"

$\Rightarrow \pi_{Empnum, Empname}(\sigma_{projectManage="Rohan"}(PROJECT \bowtie (ASSIGNED\ TO \bowtie EMPLOYEE)$

iii) List the Employees who are still not assigned with any project.

$\Rightarrow \pi_{Empname}(EMPLOYEE - (\pi_{Empnum}(ASSIGNED\ TO \bowtie EMPLOYEE)$

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

$\Rightarrow \pi_{Empname}(\underset{projectNum}{S}\ count(Empnum))$

$\Rightarrow \pi_{Empname}(\underset{projectNum}{S}\ count(Empnum) > 1\ (ASSIGNED\text{-}TO \bowtie EMPLOYEE))$

Date. _____
Page No. _____

b) Write the SQL statement for the following queries by reference of Hotel-details relation:

hotel-id	hotel-name	estb-year	hotel-star	hotel-worth
1	Hyatt	2047	Five	15M
2	Hotel Ktm	2043	Three	3M
3	Fulbari	2058	Five	20M
4	Yak & Yeti	2032	Four	11M
5	Hotel Chitwan	2033	Three	7M

1) Create a database named hoteldtable relation

=> CREATE DATABASEE hoteldtable_relation

2) Create a view named Price which shows hotel name & its worth.

=> CREATE VIEW Price AS
SELECT hotel-name, hotel-worth FROM Hotel-details

3) Modify the data so that Hotel chitwan is now 4 star level.

=> MODIFY Hotel-details SET hotel-star = "Four" WHERE hotel-id = "5" hotel-name = "Hotel chitwan".

4) Delete the records of all hotels having worth more than 9M

=> DELETE FROM Hotel-details WHERE hotel-worth > 9M

Date.
Page No.

3)a) What are stored procedures? Explain equi join, natural join, left and right outer join with examples.

=> Stored procedure is a set of SQL statements with an assigned name which are stored in DBMS as a group, so it can be reused & shared by multiple programs.

Syntax:

```
CREATE PROCEDURE procedurename
AS
  SQL statements
  GO
EXEC @procedurename
```

Join operations allows users to combine two relations in a specified way.

i) Equi Join

=> θ is predicate and consists of one of the comparison operato (=, <, >, <=, >=, <>). If θ is "=", then join is equi join

=> The "=" condition is performed by primary & Foreign key

=> Eg:-

$R \bowtie_{R.PK=S.FK} S$

ii) Natural Join

=> Only relates those rows that are equal on their common attributes names.

=> Natural Join can be computed using Fundamental operations: Cartesian product, selection, projection.

=> Eg:-

Employee ⋈ Department

iii) left Outer Join

=> in left outer join of two relation R & S, matchip tuples from both relations as well as tuples from R that do not have matching values in common in S are included in result relation.

=> Eg:-

Employee ⟕ Department.

iv) Right Outer Join.

=> in right outer join of two relations R & S, matching tuples from both relations as well as tuples from S that do not have matching values in common in R are included in result relation.

=> ~~deleted by~~ Eg:-

Employee ⟖ Department

Date.
Page No.

2020 Fall

2)

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

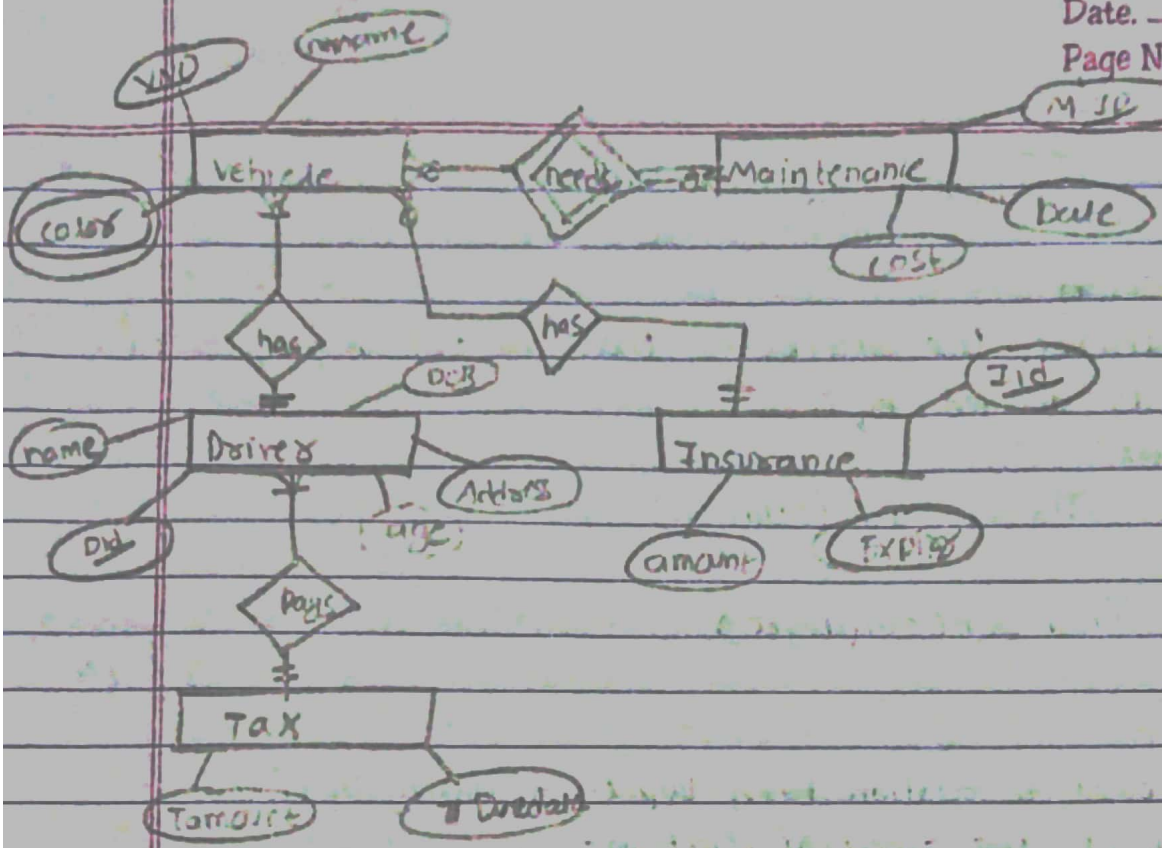
ans: Data model is a collection of conceptual tools for describing data, data relations, semantics etc.
Some of its benefits are:-

i) Data models help to clarify and document business concepts, providing a common understanding of data requirements

ii) Visual representation through data models aids communication between business users, analysts & database engineers designers, ensuring everyone is on the same page.

iii) By identifying relationships & dependencies, data models assist in minimizing data redundancy & improving the overall structural integrity of the database.

iv) Well-designed data models contribute to data quality by defining data constraints, ensuring that the stored information follows specified rules & standards.



2)a) Explain Relational Algebra. What are the relational algebra operations that can be performed? Give an example of all.

ans Relational Algebra is a procedural query language which takes instances of relations as input & gives instances of relation as output. It helps to understand query exception and optimization in RDBMS.

The relational algebra operations that can be performed are :-

i) SELECTION

=> It selects a subset of rows from relation that satisfies some condition.

=> It is denoted by σ

Syntax :-

$\sigma_{condition}(relation)$

Date
Page No

e.g. $\sigma_{salary>10000}(employee)$

ii) Projection

=> it deletes the attributes that are not in project list.

=> It is denoted by Π

=> Syntax;

$\Pi_{attribute\ list}(relation)$

E.g:-

$\Pi_{name,job}(employee)$

iii) Union

=> It builds a relation from tuples appearing in either or both of the specified relation.

=> It is denoted by $\cup$.

=> Syntax:-

$\Pi_{attribute\ list}(relation) \cup \Pi_{attribute\ list}(relation)$

Eg:- Employee $\cup$ Project

iv) Intersection.

=> Results tuples that appear in both relation.

=> It is denoted by $\cap$

=> Syntax:-

$\Pi_{attribute\ list}(relation) \cap \Pi_{attribute\ list}(relation)$

Eg:- Employee $\cap$ Project

v) Set Difference

=> It results tuples present in 1st relation but not in 2nd relation

=> It is denoted by -

=> Syntax:-

$\Pi_{attribute\ list}(relation) - \Pi_{attribute\ list}(relation)$

=> Eg:-

Employee - Project.

Date. _____
Page No. _____

vi) Cartesian Products:
- It builds relation with concatenation of every row in first relation with every row in second relation.
- is denoted by X
- Syntax:
  R X S
- Eg: Employee X Project:

b) Write SQL statements for following:-

i) Create a table named Automotor with chasis_number as primary key & following attributes:
veh-brand, veh-name, veh-model, veh-year, veh-cost, veh-color, veh-weight.

```
CREATE TABLE Automotor (
Chasis-number int PRIMARY KEY,
veh-brand VARCHAR(200),
veh-name, VARCHAR(200),
veh-model INT,
veh-year INT,
veh-cost INT,
veh-color VARCHAR(200),
veh-weight INT
)
```

ii) Enter a full detailed information of an automotor:

```
INSERT INTO Automotor VALUES (1, "Ford", 'Figo', 'I5', 2015
50000, 'black", 100)
```

iii) Change any Automotor's year to 2019

```
UPDATE Automox SET veh-year = 2019 WHERE Chasis_numb
= 1
```

Date.
Page No.

iv) Remove all Automotor records whose model contains character 'i' in last position.

=> DELETE FROM Automotor WHERE veh-model like "%i"

v) Display the total cost of all vehicles of the table Automotor

=> SELECT SUM(veh-cost) FROM Automotor

vi) Create a view from above table having vehicles only red color.

=> CREATE VIEW red AS

=> SELECT * FROM Automotor WHERE veh-color = "red"

vii) Display details of Automotor ordering on descending manner by brand name & by ascending on model when brand matches.

=> SELECT * FROM Automotor

ORDER BY ~~BRANDDESC,Mod~~ Veh-brand DESC, Veh-Model ASC

viii) Change data type of color so that it only takes one character

=> ALTER TABLE Automotor

ALTER COLUMN veh-color VARCHAR(1)

Date. _____
Page No. _____

3)

a) Differentiate between Join and Sub query. Explain different SQL Joins with examples.

ans A Join is used to combine rows from two or more tables based on a related colum, creating a result set with columns from both tables. It's a way to retrive information from multiple tables simultaneously. A subquery, on the other hand, is a query nested inside another query. It's used to retrive data that will be used in the main query condition.

Different SQL Joins are :-

i) Inner Join

→ Selects all rows from both tables as long as there is a match between the columns in both table.

→ Eg:-

```
SELECT * FROM Employee
    INNER JOIN Department
    ON Employee.eid = Department.eid
```

ii) Left Join

→ Returns all rows from the left table with matching rows in the right table.

→ The result is null in right side where there is no match

→ Eg:-

```
SEECT * FROM Employee
LEFT JOIN Department
ON Employee.eid = Department.eid
```

Date
Page No

iii) Right Join

→ Returns all rows from the right table with matching rows in the left table.

- the result is null in left side where there is no match.

- Eg:-

```
SELECT * FROM Employee
RIGHT JOIN Department
ON Employee.eid = Department.eid
```

iv) Full Outer Join

⇒ Returns all rows from the left table & right table.

⇒ Combines the result of both lef & right join

Eg:-

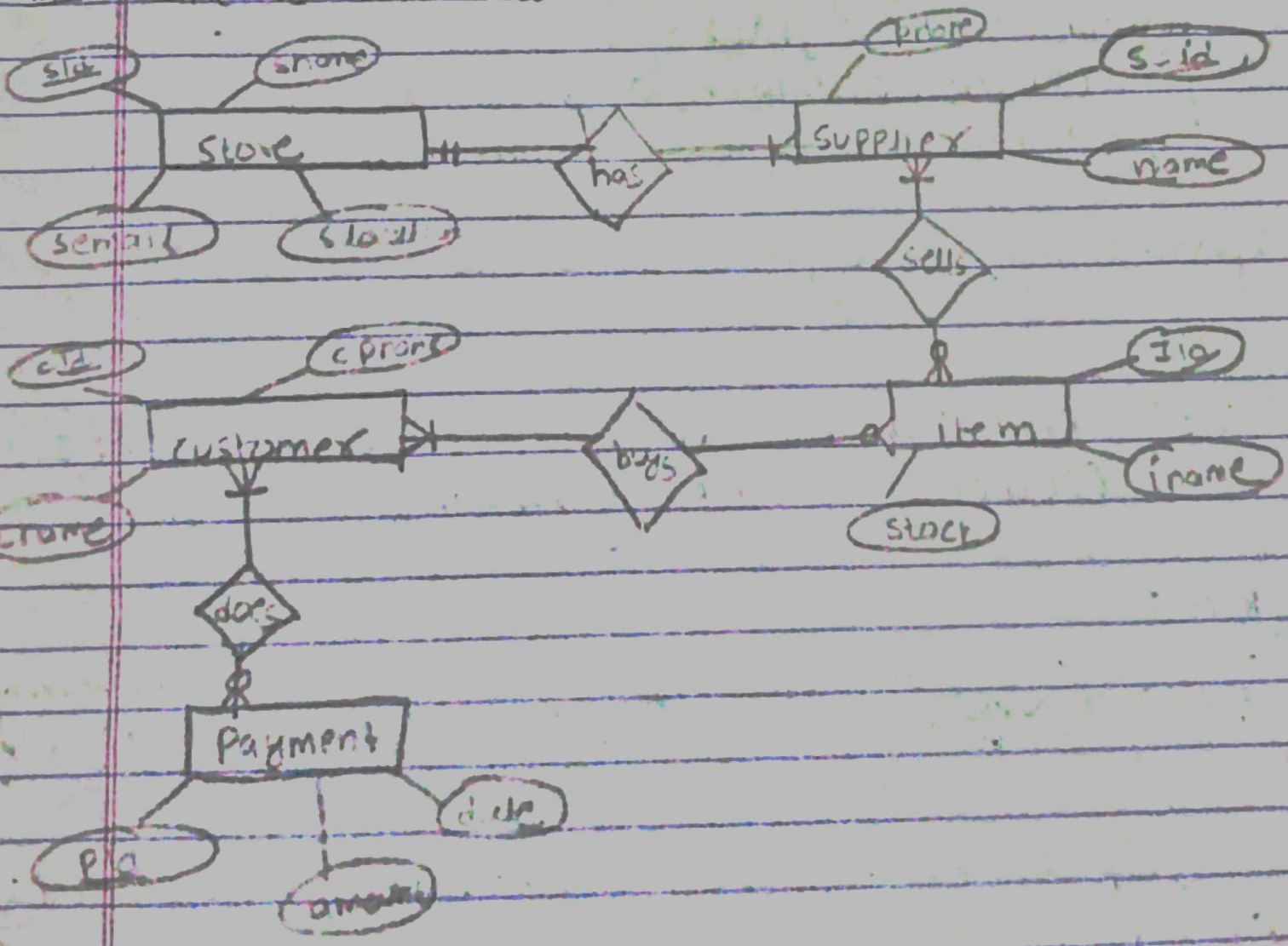
```
SELECT * FROM Employee
FULL OUTER JOIN Department
ON Employee.eid = Department.eid
```

Date. _____
Page No. _____

2020 spring:

2) Explain the importance of data modeling? Draw E-R diagram for a departmental store that keeps the informati about customer, supplier & item.

ans Same as 2020 fall



Date
Page No

6) Write both relational algebra & SQL statement for the following schema

Employee (EmpNO, Name, Address)
Project (PNO, Pname)
Workon (EMPNO, PNO)
part (Partno, part-name, Q by-on-hand)
Use (EmpNO, PNO, Partno, Number)

a) Listing all the employee details who are not working yet.

SQL: SELECT * FROM employee WHERE empNO NOT IN (SELECT Emp-NO FROM Workon)

R.A:

$$\Pi_{EmpNo,name,Address}\left(Employee - \Pi_{empNO}\left(\sigma_{Employee \bowtie Workon}\right)\right)$$

b) Listing partname & Quantity on hand those were used in DBMS project

=) SQL:

```
SELECT part_name, Q by-on-hand From part
INNER JOIN use
ON part.partno = use.partno
INNER JOIN Project
ON project.PNO = use.PNO
WHERE Pname = "DBMS"
```

R.A:

$$\Pi_{partname, Q by-on-hand}\left(\sigma_{Pname = "DBMS"}\left(project \bowtie \left(use \bowtie part\right)\right)\right)$$

c) List the name of the projects that are used by employee from London

=> SQL:

```
SELECT Pname FROM Project
INNER JOIN use
ON Project.PNO = use.PNO
INNER JOIN Employee
ON Employee.EmpNO = use.EmpNO
WHERE Address = 'London';
```

R.A:

$\Pi_{pname}(\sigma_{Address='London'}(project \bowtie (use \bowtie Employee)))$

d) Modify the database so that Jones now lives in USA.

=> SQL:

```
UPDATE Employee SET Address = "USA" WHERE name = "Jones"
```

R.A:

$Employee \leftarrow [\Pi_{EmpNo, Name, Address = "USA"}(\sigma_{Name = "Jones"}(Employee))] \cup (\sigma_{Name \neq "Jones"}(Employee))$

e) Update address of an employee 'Japan' to USA"

=> SQL:

```
UPDATE Employee SET Address = "USA" WHERE Address = "Japan"
```

R.A:

$Employee \leftarrow [\Pi_{EmpNO, Name, Address = 'USA'}(\sigma_{Address = "Japan}(Employee))] \cup (\sigma_{Address \neq Japan}(Employee))$

Date
Page No.

2021 Fall.

7)

b) Differentiate between Data model and E-R model. Draw an E-R diagram for a Library Management System including primary key, weak entity, composite attribute, derived attribute & multivalued attributes in your ER diagram.

ans(The difference between Data model & E-R model is :-

i) Data model encompasses various models & details whereas ER model focuses specifically on conceptual relationships between entities within a database.

ii) Data model serves as a comprehensive framework for designing database whereas ER model is primarily used during the initial stages of database design.

iii) Data model deals with both high-level architectural considerations and low-level implementation details whereas ER model provides a high level, visual abstraction of the relationships & dependencies between entities.

Same as 2019 spring

Date. _____
Page No. _____

2) a) Give an expression in the relation algebra to express each of the following queries:

Doctor (SSN, FirstName, LastName, Speciality, YearsOfExperience, phoneNum)
Patient (SSN, FirstName, lastName, Address, DOB, PrimaryDoctor-SSN)
Medicine (TradeName, unitPrice, GenericFlag)
Prescription (Id, Date, Doctor-SSN, patient-SSN)
Prescription_Medicine (Prescription Id, TradeName, NumOfUnits)

i) List the trade name of generic medicine with unit price less than $50.

=> $\Pi_{TradeName}(\sigma_{unitprice \geq 50}(Medicine))$

ii) List the first and last name of patients whose primary doctor is named 'John Smith'

=> $\Pi_{patient.FirstName, patient.lastName}(\sigma_{Doctor.primaryDoctor\text{-}SSN = SSN \text{ AND } Doctor.FirstName = 'John' \text{ AND } Doctor.lastName = 'smith'}(Doctor \bowtie Patient))$

iii) List the first & last last name of doctors who are not primary doctors to any patient.

=> $\Pi_{FirstName, lastName}(\sigma_{SSN \notin primaryDoctor\text{-}SSN}(Patient))$

iv) List the SNN of distinct patients who have 'Aspirin' prescribed to them by doctor named 'John Smith'

=> $\Pi_{SSN}(\Pi_{patien\text{-}SSN}(\sigma_{Doctor.FirstName='John', Doctor.LastName='Smith'}(Doctor \bowtie (Medicine \bowtie (prescription \bowtie prescription\text{-}Medicine)))))$

Date
Page No.

b) Write SQL statement for following:

i) Create a table named chef with chef-license as primary key and following attributes:
Chef-license, C-fname, C-lname, C-dob, C-gender, C-experience-hours, C-photograph

=> CREATE TABLE chef(
chef-license int PRIMARY KEY
C-fname VARCHAR(255),
C-lname VARCHAR(255),
C-dob INT,
C-gender VARCHAR(10),
C-experience-hours INT,
C-photograph BLOB)

ii) Enter a full detailed information of a chef.
=> INSERT INTO chef VALUES (505, "Kuber", "Pathak", 2052, "male", 23, "img.png")

iii) Change chef's experience hours by any value.
=> MODIFY chef //experience-hours
MODIFY chef SET C-experience-hours = "24"

iv) Remove all chef records whose name contains character 'r' in second position in his first name.
=> DELETE FROM chef WHERE C-fname LIKE "_r%"

v) Display the total experience hours of all chef.
=> SELECT SUM(C-experience-hours) FROM chef.

Date. _____
Page No. _____

vi) Ans Create a view from above table-

```
CREATE VIEW display AS
SELECT * FROM chef.
```

vii) Display details of chef ordering on descending manner in last name and by first name when last name matches.

Ans

```
SELECT * FROM chef ORDER BY L-lname DESC, C-fname ASC
```

Page No.

3) a) Explain Data Constraints & its types with examples.

ans Data constraints, also known as data integrity constraints, are rules that define the permissible value and relationships within a database.

Types of Data constraints:

i) Primary key Constraint:
- It ensures uniqueness and identifies each record uniquely within a table.

Eg:-

```
CREATE TABLE Test(
ID INT PRIMARY KEY
);
```

ii) Foreign key constraint:
- Establishes a link between 2 tables, ensuring referential integrity.

Eg:-

```
CREATE TABLE Test(
ProductID INT,
FOREIGN KEY (PRODUCTID) REFERENCES Products(ProductID)
);
```

iii) Unique Constraints:
- Ensures that all the values in a column are UNIQUE

Eg:

```
CREATE TABLE Test(
ID INT UNIQUE
);
```

Date
Page No.

iv) CHECK constraint

- Enforces a condition that must be satisfied for data to be entered.
- Eg: CREATE TABLE Test(
  Price DECIMAL(10,2),
  CHECK (Price > 0)
  );

v) NOT NULL constraint

-> Requires a column to have a value.

Eg:-

CREATE TABLE Test(
  Name VARCHAR(30) NOT NULL
);

Date
Page No.

2021 Spring

2)a) Define schema & views. Considering the following schemas:

Sailors(sid, sname, rating, age)

Boats(bid, bname, color)

Reserves(sid, bid, day)

Write relational algebra expressions.

i) Find the records of sailors who have reserved boat number 103 (bid = 103).

⇒ $\Pi_{sid, sname, rating, age}(\sigma_{bid=103}(Sailors \bowtie Reserves))$

ii) Update the color of the boat, where bid is 104, into green.

⇒ $Boats \leftarrow [\Pi_{bid, bname, color = 'green'}(\sigma_{bid=104}(Boats))] \cup [\sigma_{bid \neq 104}(Boats)]$

iii) Find the names of sailors who have reserved a red or green boat.

⇒ $\Pi_{sname}(\sigma_{color='red' \text{ OR } color='green'}(Sailors \bowtie (Boats \bowtie Reserves)))$

iv) Find the names of sailors who have reserved boat number 103 on day 5

⇒ $\Pi_{sname}(\sigma_{bid=103 \text{ AND } day=5}(Sailors \bowtie (Reserves \bowtie \ldots)))$

v) Find the name of sailors whose name is not 'Ram'

⇒ $\Pi_{sname}(\sigma_{sname \neq 'Ram'}(Sailors))$

Date. ______
Page No. ______

vi) Find the name of all boats

=> ~~Thename (~~ ~~Third~~ π_{bname} (Boats)

b) What are DDL & DML queries in SQL? Consider the relation in 2(a) & write the SQL Statements for the queries in 2(a)

ans The queries used to define data structure and modify data are called DDL queries.
The queries used to edit, delete and query row-level data are called DML queries.

i) =>
```
SELECT * FROM sailors
INNER JOIN Reserves
ON sailor.sid = Reserves.sid
WHERE bid=103
```

ii) =>
```
UPDATE Boats set color="green" WHERE bid=104
```

iii) =>
```
SELECT sname FROM sailors
INNER JOIN Reserves
ON sailors.sid=Reserves.sid
INNER JOIN Boats
ON Boats.bid= Reserve.bid
WHERE color="red OR color="green.
```

iv) =>
```
SELECT name FROM Sailor
INNER JOIN Reserves
ON Sailor.sid= Reserves.sid
WHERE bid=103 AND day=5
```

V) => SELECT sname FROM sailors WHERE sname != 'Ram'

VI) => SELECT bname FROM Boats

2022 Fall

I) b) Why do you need E-R diagram? Draw an E-R diagram for online shop management system. Assume relevant entities and attributes for the given system.

=> We need E-R diagram because :-

i) It provides a clear & visual representation of the database structure

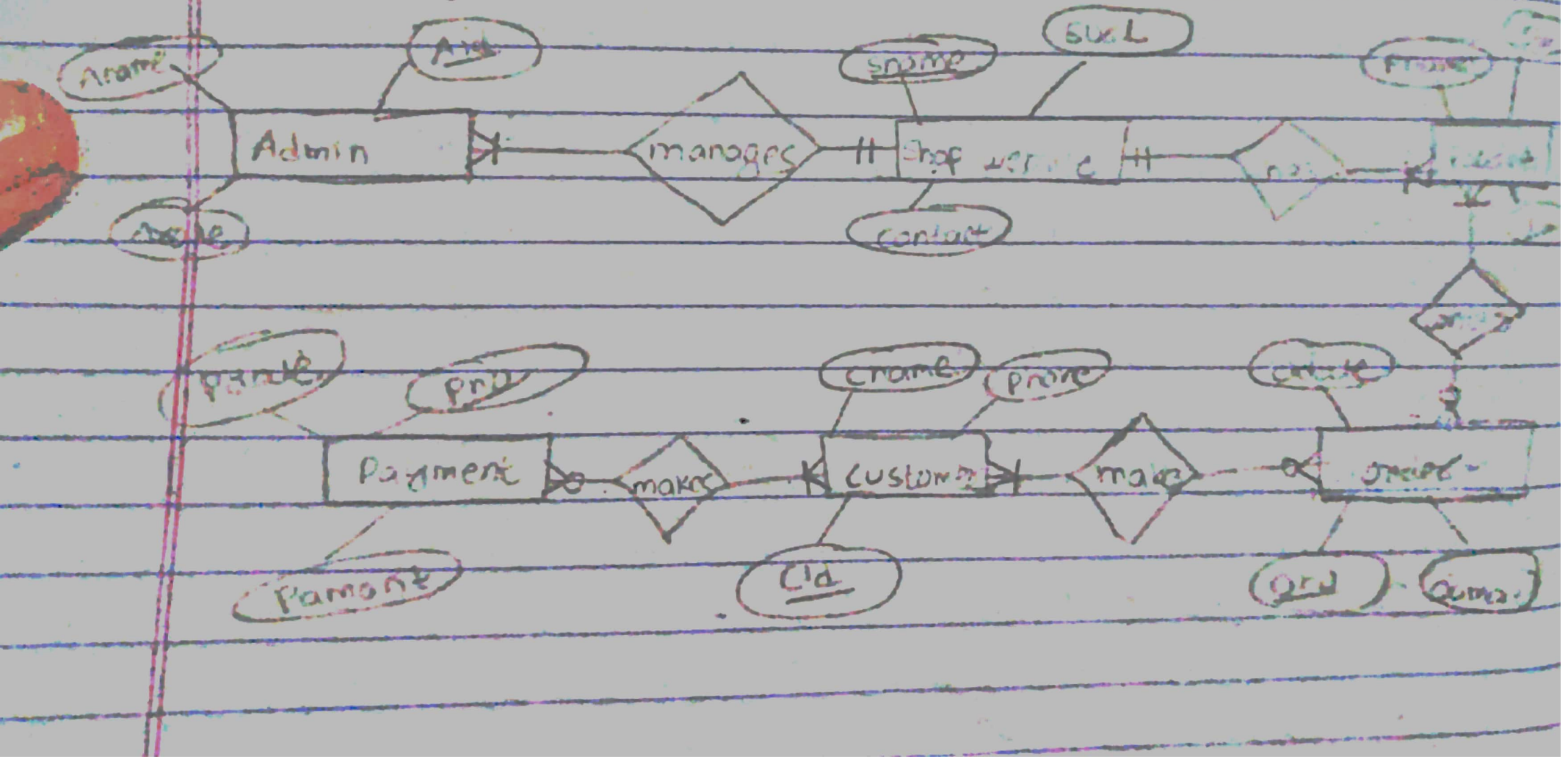
ii) It serves as a communication tool between stakeholders involved in development process.

iii) It helps in identifying entities and attributes which aids in system modeling

iv) It haps in ensuring data integrity, maintaining the accuracy & consistency of data.

aids in System modeling

iv) It helps in ensuring data integrity, maintaining the accuracy & consistency of data.



Date. _____
Page No. _____

2) a) Suppose we have following relation:
Employee (Peason-name, street, city)
Works (Peason-name, C-name, salary)
Company (C-name, city)
Write R.A expressions:-

i) Find name of all employess who live in 'Butwal' and whose salary is less than RS. 50,000.

=> $\Pi_{person\text{-}name}(\sigma_{Employee.city = \text{"Butwal"}\ AND\ salary < 50,000}(Works \bowtie (Company \bowtie Employee)$

ii) Find the name of all employees who work for "Nabil Bank"

=> $\Pi_{person\text{-}name}(\sigma_{c\text{-}name = \text{"Nabil Bank"}}(Employee \bowtie Company))$

iii) Find the names and cities of residence of all employees who work for "Global bank"

=> $\Pi_{person\text{-}name, city}(\sigma_{c\text{-}name = \text{"Global bank"}}(Employee \bowtie Works))$

iv) Update the salary of all employees by 10%.

=> $Works \leftarrow \Pi_{person\text{-}name, c\text{-}name, salary = salary + salary * 0.1}(Works)$

b) Define stored procedure. What is the advantages of the stored procedure? Explain how stored procedures are created in SQL.

=> Stored procedure is a set of SQL statements with an assigned name which are stored in DBMS as a group, so it can be reused & shared by multiple programs.
Syntax:

-> CREATE PROCEDURE name
AS
SQL Statements
GO
-> EXEC name

Date
Page No.

The advantages of stored procedures are:-

1) They are precompiled and stored in database, which leads to faster execution time.
2) Procedures can be reused across multiple parts of an application, reducing redundancy & promoting maintainability.
3) Access to data can be controlled through stored procedures, allowing for better security management by limiting direct table access.
4) Debugging is often easier with stored procedures and maintenance becomes simpler.
5) As the entire procedure is sent to the database server in a single call it reduces network traffic.
6) They support transaction control, allowing you to group multiple SQL statement into a single transaction.

To create a procedure in SQL:

```
CREATE PROCEDURE procedure_name
parameters (optional)
AS
SQL statements
GO
EXEC procedure_name;
```

E.g:-

```
CREATE PROCEDURE empInfo
AS
SELECT * FROM employee
GO
```

=> EXEC empInfo

Date. _____
Page No. _____

3)a) Consider the relation Actress-Details & write the SQL statements:-

players-id	Actress-name	Debut-year	Recent-release	Fee
1	Renu	2010	Samay	400000
2	Sita	2022	Radha	30000
3	Geeta	2001	Mato	60000
4	Amita	1990	Man	70000
5	Karishma	1989	Prem	100000

i) Create the table Actress-details relation

=> CREATE TABLE Actress-details (
players-id INT PRIMARY KEY,
Actress-name VARCHAR(50),
Debut-year INT,
Recent-release VARCHAR(50),
Fee INT)

ii) Delete the data of actress whose recent release is Prem

=> DELETE FROM Actress-Details WHERE Recent-release = "Prem"

iii) Modify the database so that Renu's new relase is "Win the Race" Film.

=> MODIFY Actress-Details SET Recent-release = "Win the Race" WHERE Actress-name = "Renu"

iv) Insert a new record in the above table.

=> INSERT INTO Actress-Details VALUES (6, 'Gita", 2000, "Hero", 50000)

Date
Page No

2017 spring

2)

a) Define relation schema and views. Consider the following relations for a database that keeps track of student enrollment in courses and the books adopted for each course:

STUDENT (SSN, Name, Major, Bdate)
COURSE (course#, cname, Dept)
ENROLL (SSN, course#, Quarter, Grade)
BOOK-ADOPTION (course#, quarter, Book-ISBN)
TEXT (Book-ISBN, Book-Title, Publisher, Author)

Draw a relational schema diagram specifying the foreign keys for this schema.

ans A relation schema defines the structure or blueprint of a relation database

Views in a relational database is a virtual table that is based on the result of a SELECT query.