CBCS SCHEME

USN 2 V D 2 3 C S 0 0 6

BCS403

Fourth Semester B.E./B.Tech. Degree Examination, June/July 2025
Database Management Systems

Time: 3 hrs. Max. Marks: 100

Note: 1. Answer any FIVE full questions, choosing ONE full question from each module.
2. M : Marks , L: Bloom's level , C: Course outcomes.

			M	L	C
		Module – 1			
Q.1	a.	Explain the types of attributes with example.	4	L2	CO1
	b.	Define database. Explain the main characteristics of the database approach.	8	L2	CO1
	c.	Show the ER diagram for an EMPLOYEE database by assuming your own entities (minimum 4) attributes and relationships, mention cardinality ratios wherever appropriate.	8	L3	CO2
		OR			
Q.2	a.	Describe the three schema architecture.	4	L2	CO1
	b.	Explain the component models of DBMS and their interaction with the help of diagram.	8	L2	CO1
	c.	Design ER diagram for a university database by assuming your own entities (4). Mention primary key , constraints and relationships.	8	L3	CO2
		Module – 2			
Q.3	a.	Explain relational model constraints.	6	L2	CO1
	b.	Explain the characteristics of relations with suitable example for each.	6	L2	CO1
	c.	Considering the following schema : Sailors (sid , sname , rating , age) Boats (bid , bname , color) Reserves (sid , bid , day) Write a relational algebra queries for the following : i) Find the names of sailors, who have reserved red and a green boat. ii) Find the names of sailors who have reserved a red boat. iii) Find the names of sailors who have reserved a red or green boat. iv) Find the names of sailors who have reserved all boats.	8	L3	CO1
		OR			
Q.4	a.	Explain the steps to convert the basic ER model to relational Database schema.	6	L2	CO1
	b.	Explain Unary relational operations with example.	6	L2	CO1

Scanned with OKEN Scanner

	c.	Consider the relation schema Employee database. EMPLOYEE (Fname ,Minit , Lname , SSn , Bdates , Address , Sex , Salary Super_SSn , Dno) DEPARTMENT (Dname , Dnumber , Mgr_SSn , Mgr_start_date) PROJECT (Pname , PNumber , Plocation , Dnum) WORKS_ON (Essn , Pno , Hours) DEPENDENT (Essn , Dependent_name , sex, Bdate , Relationship) Write relational algebra queries for the following : i) Retrieve the name and address of all employees who work for the 'Research' department. ii) List the names of all employees with 2 or more dependents. iii) Find the names of employees who work on all the projects controlled by department number 5. iv) List the names of employees who have no dependents.	8	L3	CO3
		Module – 3			
Q.5	a.	What is the need for normalization? Explain second and third normal form with examples.	6	L2	CO4
	b.	Outline constraints in SQL.	6	L2	CO1
	c.	Identify the given Relation R(ABCDE) and its instance, check whether FDS given hold or not. Give reasons. i) A → B ii) B → C iii) D → E iv) CD → E. A B C D E a_1 b_1 c_1 d_1 e_1 a_1 b_2 c_1 d_1 e_1 a_2 b_2 c_1 d_2 e_3 a_2 b_3 c_3 d_2 e_2	8	L3	CO4
		OR			
Q.6	a.	What is Multivalued dependency? Explain 4NF and 5NF with suitable example.	6	L2	CO4
	b.	Outline the informal design guidelines for relational schema.	6	L2	CO4
	c.	Consider relation R with following function dependency : EMPPROJ (SSn , Pnumber , Hours , Ename , Pname , Plocation) SSN , Pnumber → Hours, SSN → Ename Pnumber → Pname , Plocation. Is it 2NF? Verify? If no give reason.	8	L3	CO4

Scanned with OKEN Scanner

		Module – 4			
Q.7	**a.**	Consider the following schema for a company database : Employee (FName , LName , SSn , Adderss , Sex , Salary , Dno , Super_SSn) Department (Dname , Dnumber , mgr_SSn, mgr_st_date) Project (Pname , Pnumber , Plocation , Dnum) WORKS_on (Essn , Pno , Hours) DEPENDENT (Essn , Dependent name , Sex , Bdate, relationship) Write the SQL queries for the following : i) List the names of managers who have atleast one dependent (use correlated nested). ii) Retrieve the name of each employee who has a dependent with the same first name and is the same sex as the employee. iii) For each project retrieve the project number , project name and the number of employees who work on that project. iv) Retrieve the SSN of all employees who work on project number 1, 2 or 3. (Use IN). v) Find the sum of the salaries of all employees of the 'Research' department as well as maximum salary , minimum salary , average salary in this department.	**10**	**L3**	**CO3**
	b.	Why concurrency control is needed? Demonstrate with an example.	**10**	**L2**	**CO5**
		OR			
Q.8	**a.**	Consider the following schedule. The actions are listed in the order they are scheduled and prefixed with the transaction name. S1 : T1 : R(X) , T2 : R(X) T1 : W(Y) , T2 : W(Y) , T1 : R(Y) , T2 : R(Y) S2 : T3 : W(X) , T1 : R(X) , T1 : W(Y) , T2 : R(Z) , T2 : W(Z) , T3 : R(Z) For each schedule answer the following : i) What is the precedence graph for the schedule? ii) Is the schedule conflict_serializable? If so what are all the conflicts equivalent serial schedules? iii) Is the schedule view serializable? If so what are all the view equivalent serial schedules?	**10**	**L3**	**CO5**
	b.	Explain triggers with example write a trigger in SQL to call a procedure "Inform_Supervisor" whenever an employees salary is greater than the salary of his or her direct supervisor in the COMPANY database.	**10**	**L3**	**CO5**
		Module – 5			
Q.9	**a.**	Describe the two – phase locking protocol for concurrency control provide example to illustrate how it ensures serializability in transaction schedule.	**10**	**L2**	**CO5**
	b.	Explain the characteristics of NOSQL system.	**10**	**L2**	**CO6**
		OR			
Q.10	**a.**	Explain binary locks and shared lock with algorithm.	**10**	**L2**	**CO5**
	b.	Explain MongoDB data model, CRUD operations and distributed system characteristics.	**10**	**L2**	**CO6**

* * * * *

Scanned with OKEN Scanner

Fourth Semester B.E Examination
June / July - 2025

Data Base Management Systems [BCS403]

Q.1
a) Explain the types of attributes with Example.

Soln: Types of Attributes :

(1) Simple Attributes : are attributes that are drawn from an atomic values or attributes that are not divisible.
Ex: Apartment_Number, Street,

(2) Composite attribute : are those which can be divided into smaller subparts, which represents more basic attributes with independent meaning.
Ex: Address
- Street_Address → Number, Street, Apartment_Number
- City
- State
- Zip

(3) Single valued Attributes : which can take only one value for a given entity from an entity set.
Ex: Age.

(4) Multivalued Attributes : which can take more than one value for a given entity from an entity set.
Ex: A person who can speak is more than one language.

(5) Derived Attributes : which can be derived from other attributes.
Ex: Age of student is derived from date of birth.

(6) Key Attributes : Which can identify an entity uniquely in an set.
Ex: Roll no. of a student.

KLS's V.D. Institute Of Technology CSE HALIYAL

b) Define database. Explain the main characteristics of the database approach. (1)

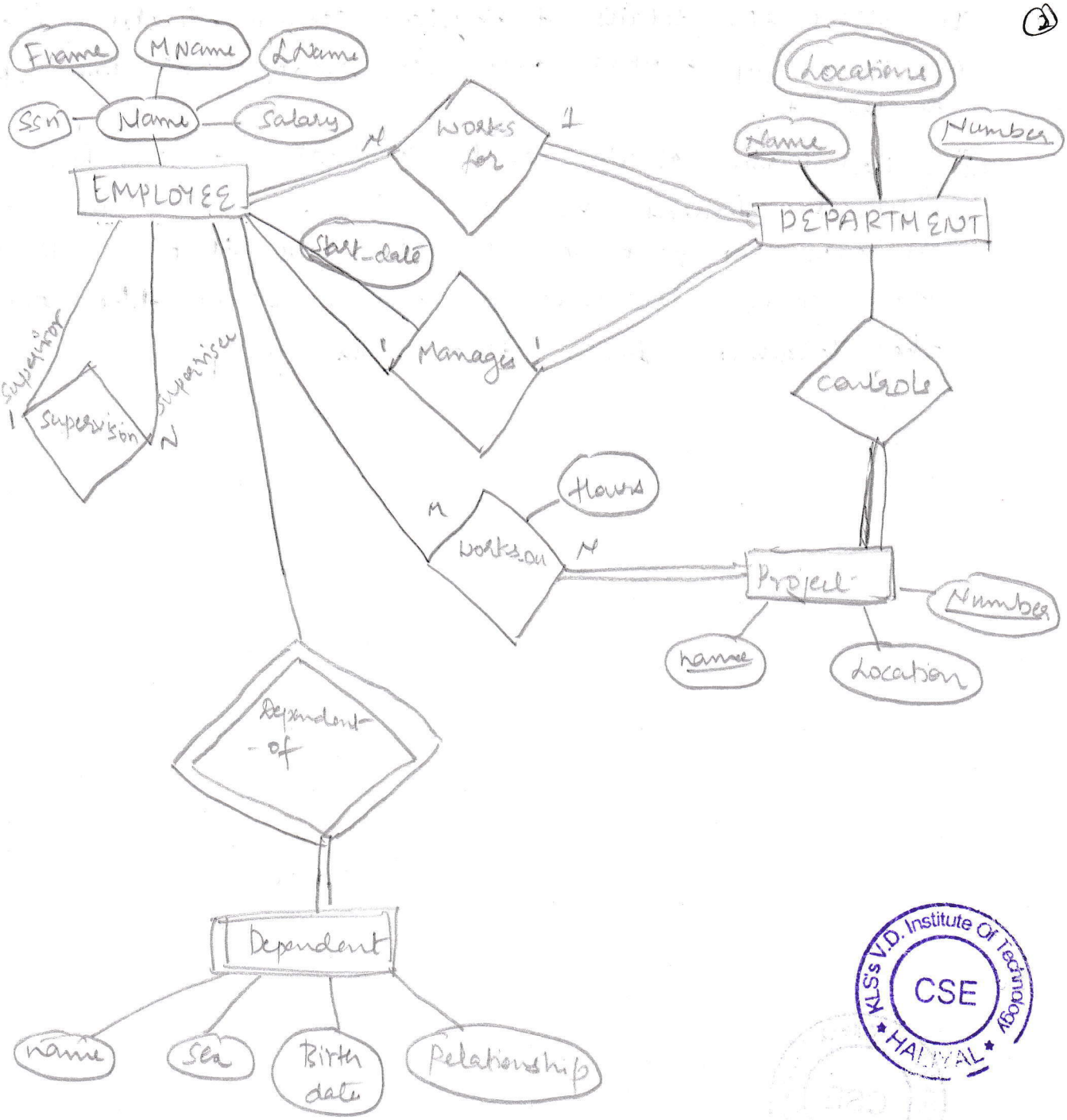
Soln: A database is collection of related data.

The main characteristics of the database application,

* Self Describing nature of the database system:
→ Database system stores both data & its complete definition.
→ DBMS catalog contain: structure of each file, Name of the data item, data types, storage formats, size & position of each data item, constraints on the data item.
* Insulation between Programs & Data & Data abstraction:
→ DBMS based systems store the structure of data files seperately in the DBMS catalog.
→ This seperation between programs & data definations is called program — data independence.
* Support for Multiple views of the data:
→ A database system typically stores Multiple types of users with different roles & requirements.
→ Each user may need a different perspective of the same databases.
→ A view is a customization representation of the database tailored to a specific user or application.
* Sharing of Data & Multiuser Transaction Processing:
→ Multiple users can access the same database simultaneously.
→ A DBMS includes concurrency control mechanism to manage simultaneous access to a shared data.

c) Show the ER diagram for an EMPLOYEE database by assuming own entities, attributes, relationships, mention cardiality ratios, wherever appropriate.

KLS's V.D. Institute Of Technology CSE HALIYAL



KLS's V.D. Institute Of Technology
CSE
HALIYAL

ER diagram of Employee database

Q.2

a) Describe the three schema architecture.

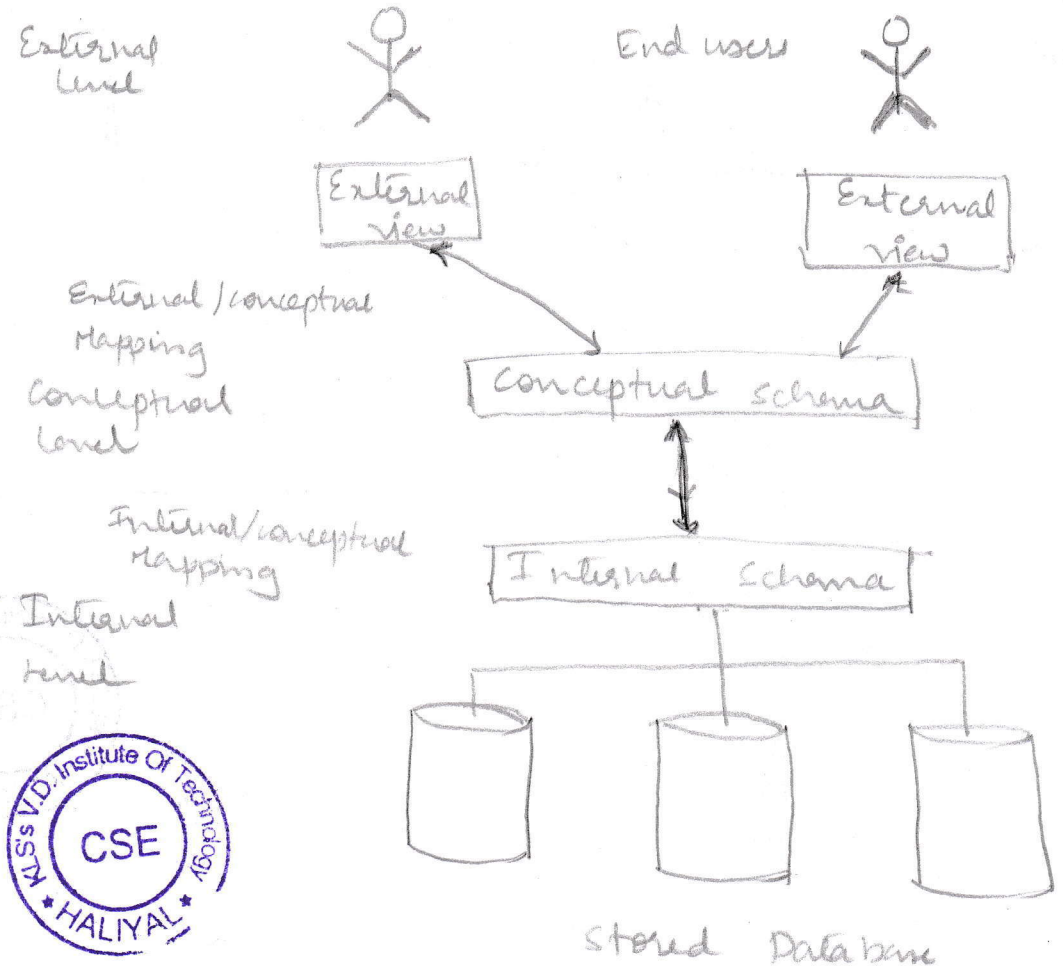
Soln: The three schema architecture can be defined at the following three levels.

(1) the internal level has an internal schema, which describes the physical storage structure of the database.

(2) The conceptual level has a conceptual schema which describes, the structures of the whole database for a community of users.

It hides the details of physical storage structures & concentrates on describing entities, data types, relationships, user operations & constraints.

③ External view or view level include a number of external schemas or user views. Each external schema describes a part of the database, that a particular user group is interested in & hides the rest of the database from that user group.



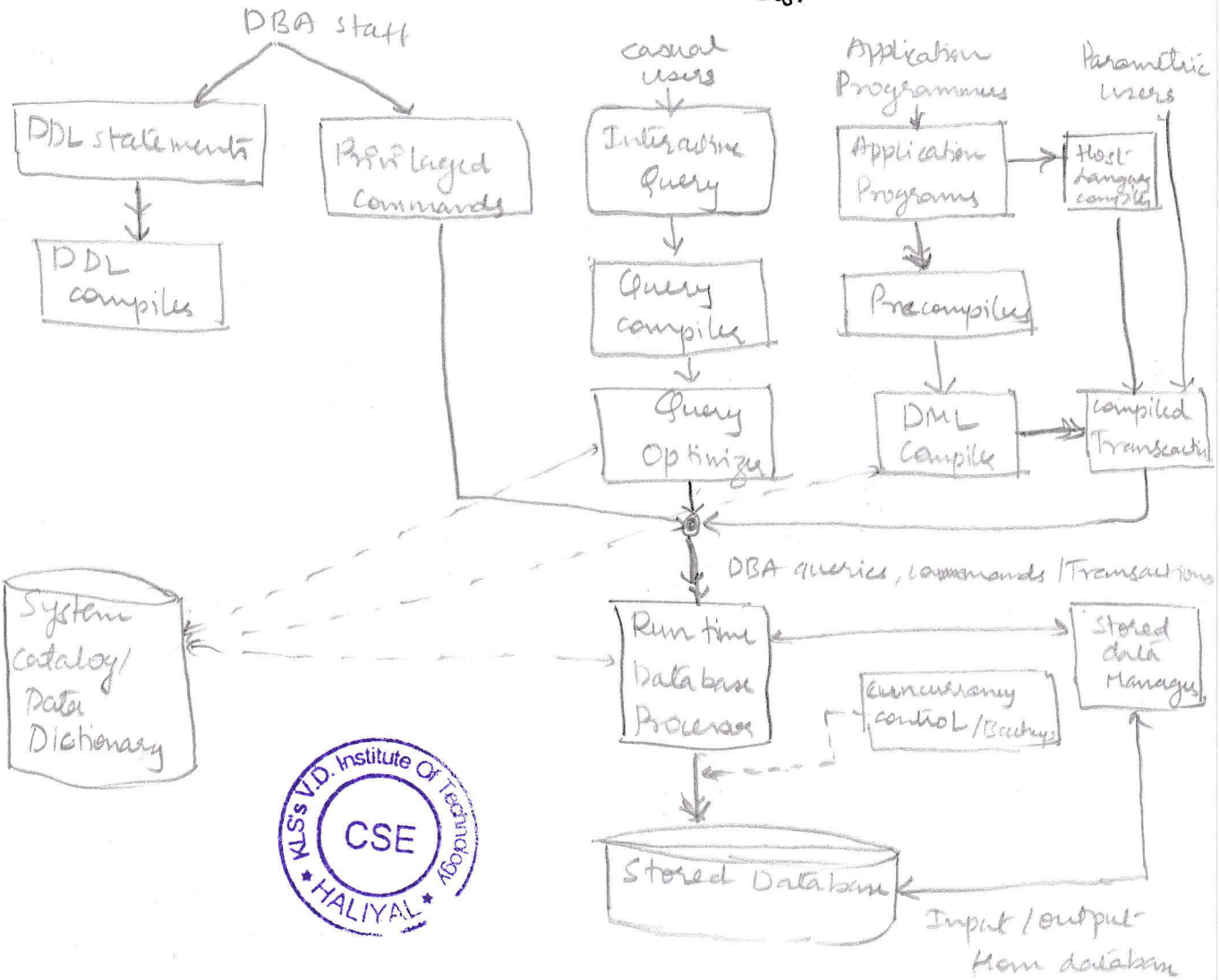
KLS's V.D. Institute Of Technology
CSE
HALIYAL

b) Explain the component models of DBMS by assuming area and their interaction with the help of diagram.

Sol:-

* A DBMS a collection of software modules that manage database creation, access and maintainance.
* DBMS components are divided into two major parts.
  - → User level components
  - → Internal DBMS components.

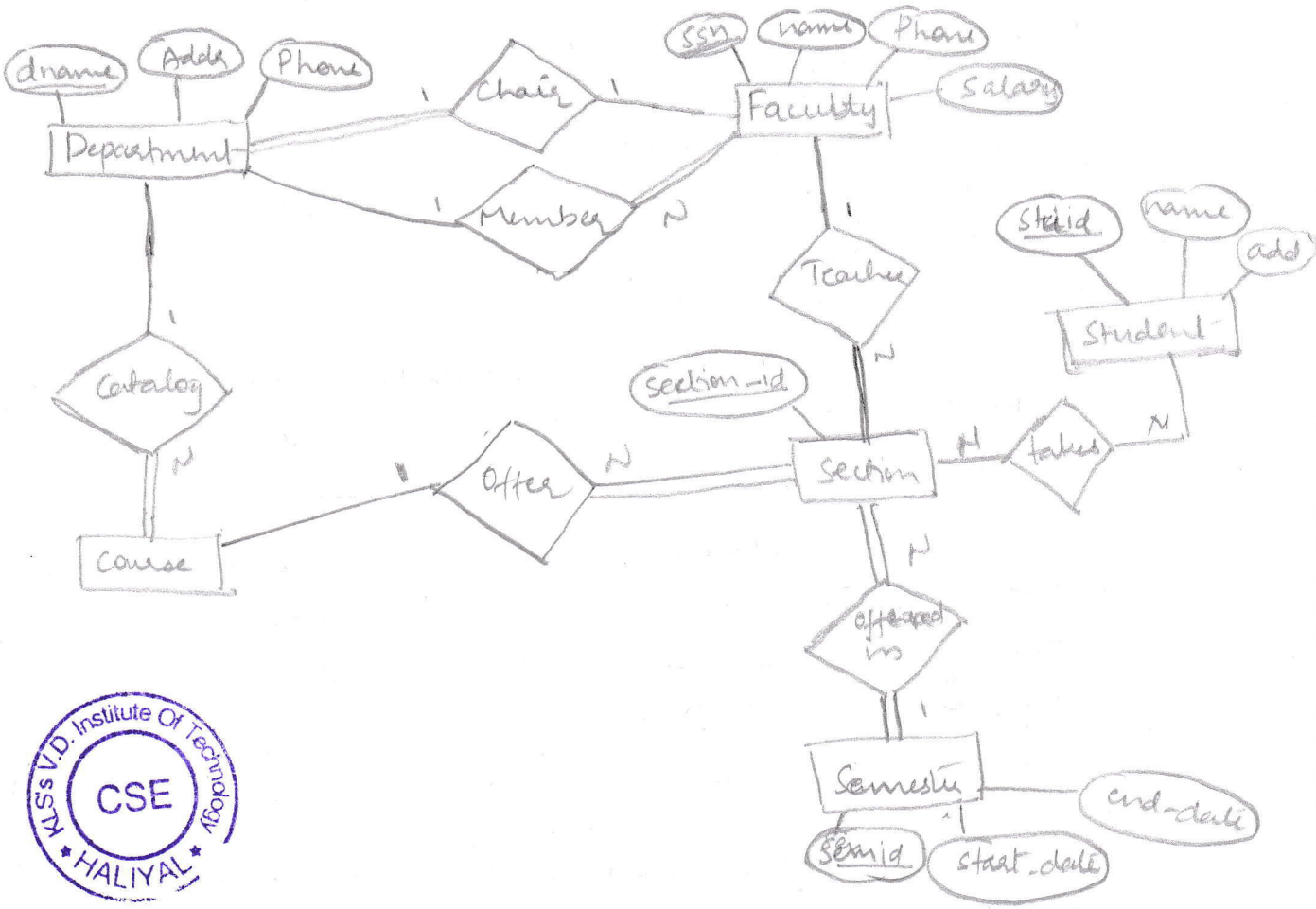
* DBA controls & manages the entire database system, he defines database schemas, storage structures, constraints, access rights
* Casual users access the database using interactive query interfaces.
* Application programmers write database programs using host languages such as Java, C, Python.
* Parametric users execute predefined or canned transactions.
* DDL compiler processes databases schema definition
* The query compiler checks SQL query syntax & validates attribute & table names.



component modules of a DBMS & their interactions

KLS's V.D. Institute of Technology CSE HALIYAL

c) Design E R diagram for a university database by assuming your own entities. Mention primary key, constraints & Relationships.



KLS's V.D. Institute Of Technology
CSE
HALIYAL

Q3.

a) Explain relational model constraints.

Soln: constraints on databases can generally be divided into three main categories.

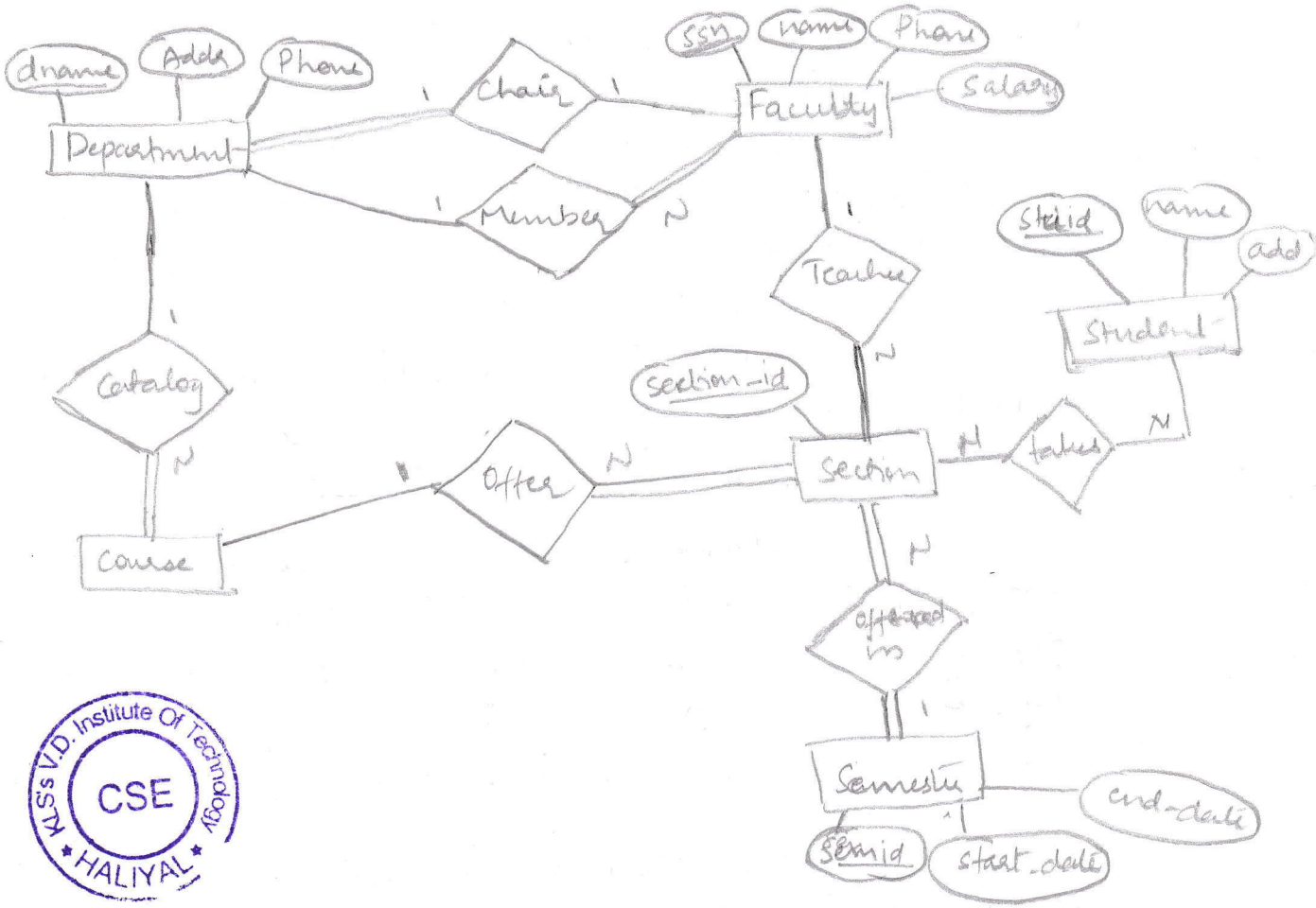
(1) constraints that are inherent in the data model. called implicit constraints

(2) Constraints that can be directly expressed in the schemas of the data model, called Data Definition language. called explicit constraints.

(3) Constraints that cannot be directly expressed in the schemas of the data model, called Application based or constraint rules.

* Domain constraints specify that each attribute value must be an atomic value from its predefined domain. It defines data types, value range, enumerated values, special types.

c) Design E R diagram for a university database by assuming your own entities. Mention primary key, constraints & Relationships.



KLS's V.D. Institute Of Technology
CSE
HALIYAL

Q3.

a) Explain relational model constraints.

Soln: constraints on databases can generally be divided into three main categories.

(1) constraints that are inherent in the data model. called implicit constraints

(2) Constraints that can be directly expressed in the schemas of the data model, called Data Definition language. called explicit constraints.

(3) Constraints that cannot be directly expressed in the schemas of the data model, called Application based or constraint rules.

* Domain constraints specify that each attribute value must be an atomic value from its predefined domain. It defines data types, value range, enumerated values, special types.

c) Considering the following schema
Sailors (sid, sname, rating, age)
Boats (bid, bname, color)
Reserves (sid, bid, day)

Soln: (1) Find the names of sailors, who have reserved red & green boat

$R_1 = \pi_{sid}(\sigma_{color='red'}(Boats) \bowtie Reserves)$ $\pi_{sname}(Sailors \bowtie R)$
$R_2 = \pi_{sid}(\sigma_{color='green'}(Boats) \bowtie Reserves)$

(2) Find the names of sailors, who have reserved a red boat.

$R_2 \leftarrow \pi_{sid}(\sigma_{}$

$\pi_{sname}(Sailors \bowtie Reserves \bowtie \sigma_{color='red'}(Boats))$

(3) Find the names of sailors who have reserved a red or green boat

$\pi_{sname}(Sailors \bowtie Reserves \bowtie \sigma_{color='red' \vee color='green'}(Boats))$

(4) Find the names of the sailors who have reserved all boats

$\pi_{sname}(sailors \bowtie s)$

KLS's V.D. Institute Of Technology CSE HALIYAL

Q.4
a) Explain the steps to convert the basic ER model to relational database schema.

Soln: Step1: Mapping of Regular Entity Types: For each regular entity type E in ER schema, create a relation R that includes all the simple attributes of E. Choose one of the key attributes of E as the primary key for R.

Step2: Mapping of the Weak Entity types: For each weak entity type W in the ER schema with owner entity type E, create a relation R and include all simple attributes of W as attributes of R.

Step3: Mapping of Binary 1:1 Relation types: For each binary 1:1 relationship type R in the ER schema, identify the relations S & T that corresponds to the entity types participating in R.

Step 4: Mapping of the Binary 1:N Relationship types:

① Foreign key approach: For each regular binary 1:N relationship type R identify the relation S that represents the participating entity type at the N-side of the relationship type.

② Relationship relation approach: is to use the cross reference option as in the third option for binary 1:1 relationships

Step 5: Mapping of the Binary M:N Relationship Types:
In the additional relation model with no multivalued attributes, the only option for M:N relationships ~~types~~ is the relationship relation option.

Step 6: Mapping of the Multivalued Attributes:
For each Multivalued attribute A, create a new Relation R. This relation R will include an attribute corresponding to A plus the primary key attribute K, as a foreign key in R.

Step 7: Mapping of N-ary relationship Types:
For each n-ary relationship type R, where n>2 create a new relationship relation S to represent R. Include as foreign key attributes in S the primary keys of the relations that represents the participating entity

b) Explain Unary relational operations with example.

Soln: Unary select operation:

* The select operation is used to choose a subset of the tuples from a relation that satisfies a selection condition.
* We can select consider the select operation to restrict the tuples in a relation to only those tuples that satisfy the condition.

Ex: To select the Employee tuples whose department is 4, or those whose salary is greater than $30,000

$\sigma_{Dno=4}(Employee)$

$\sigma_{Salary>30000}(Employee)$

KLS's V.D. Institute Of Technology CSE HALIYAL

The Project Operation: The project operation selects certain columns from the table and discards the other columns. If we are interested in only certain attributes of a relation, we use the Project operation to project the relation over these attributes.

Eg: To list each employee's first & last name & salary, use the Project operation as follows,

$\pi_{Lname, Fname, Salary}(Employee)$

c) Consider the relation schema Employee database

Employee (Fname, Minit, Lname, SSn, Bdate, Address, Sex, Salary, Super_ssn, Dno)

Department (Dname, Dnumber, Mgr_ssn, Mgr_Start_date)

Project (Pname, PNumber, Plocation, Dnum)

Works_on (Essn, Pno, Hours)

Dependent (Essn, Dependent_name, Sex, Bdate, Relationship)

Write relational queries for the following:

(a) Retrieve the name & address of all employees who work for the 'Research' Department.

$\pi_{Fname, Lname, Address}(\sigma_{Dname = 'Research'}(Department) \bowtie_{Dnumber = Dno} Employee)$

(b) List the names of all employees with 2 or more dependents

$\pi_{Fname, Lname}(Employee \bowtie_{Ssn = Essn} \sigma_{count > 2}(\gamma_{Essn, count(Dependent\_name) \rightarrow count(Dependent)}))$

(c) Find the names of employees who work on all the projects controlled by department no = 5

$\pi_{Fname, Lname}(Employee \bowtie_{Ssn = Essn} (\pi_{Essn, Pno}(Works\_on) \div \pi_{PNo}(\sigma_{Dnum = 5}(Project))))$

(d) List the names of Employees who have no dependents

$\pi_{Fname, Lname}(Employee) - \pi_{Fname, Lname}(Employee \bowtie_{Ssn = Essn} Dependent)$

KLS's V.D. Institute of Technology CSE HALIYAL

Q.5

a) What is the need for Normalization ? Explain 2nd & 3rd normal form with examples.

Soln:

Normalization is a systematic process of Organizing data in a relational database to reduce redendancy and improve data integrity. It provides database designers with the following.

* A formal framework for analyzing relation schemas, based on their keys & on the functional dependencies among their attributes.
* A series of normal form test that can be carried out Individual relation schemas so that the relational database can be normalized to any desired degree.

→ Second Normal Form: A relation schema R is in 2NF if every non prime attribute A in R is fully functionally dependent on the primary key of R.

2nd NF is based on the concept of full functional dependency A functional dependency X→Y is full function dependency if removal of any attribute A from X means that the dependency does not hold any more.

Ex: Enrollment (Studentid, courseid, Sname, Coursename)

Functional Dependencies

Studentid → Student_name

courseid → course_name

conversion into 2NF

Student (Studentid, Student_name)

course (courseid, course_name)

Enrollment (Student.id, courseid)

Third Normal Form: a relation schema R is in 3NF if it satisfies 2NF & no non prime attribute of R is transitively dependent on the primary key

Ex: Employee (Empid, Ename, deptid, dept_name)

conversion to 3NF

Employee (Empid, Emp_name, dept_id)

Dept (Dept_id, Dept_name)

KLS's V.D. Institute Of Technology CSE HALIYAL

b) Outline the constraints in SQL.

Soln: ① Specifying Attribute constraints & Attribute Defaults:

→ A constraint NOT NULL may be specified if NULL is not permitted for a particular attribute.

→ Default value for an attribute by appending the clause DEFAULT <value> to an attribute defination.

② Specifying key & Referential Integrity constraints:

→ The Primary key clause specifies one or more attributes that make up the primary key of a relation.

→ Unique key clause specifies alternate, known as candidate keys if it is a single attribute.

→ Referential Integrity is specified via the FOREIGN KEY clause,

③ Giving names to constraints:

A constraint name is used to identify a particular constraint in case the constraint must be dropped later or replaced with another constraint

c) Identify the given relation R(ABCDE) and its instance, check whether FDs given hold or not. Give reason.

① A → B holds if for any two tuples with same value of A they must have the same value of B

For A = a1, B values are b1 and b2

For A = a2, B values are b2 and b3

Therefor A → B does not hold as A value determines multiple B values, violating the FD defination.

② B → C

B = b2 appear in two rows with C = c1 in both cases

B → C holds, since whenever B is the same, C is also same

③ D → E, For D = d1, E = e1

For D = d2, E values are e3 & e2

Therefor D → E does not hold, since D value determine different E values.

KLSs V.D. Institute Of Technology CSE HALIYAL

(4) CD → E

C,D = (C1,d1) → E = e1 [consistant]

(C1,d2) → C3

(C3,d2) → e2

No two tuples with same (C,D) have different E values

CD → E holds, as the combination of C & D uniquely determines E.

6)

a) What is Multivalued dependency? Explain 4NF and 5NF with suitable example.

Sol: A multivalued dependency X → Y specified on relation schema R, where X & Y are both subsets of R, specifies the following constraint on any relation state r of R: If two tuples t1 & t2 exists in r such that t1[X] = t2[X] then tuples t3 & t4 should also exist in r with the following properties,

$t_3[X] = t_4[X] = t_1[X] = t_2[X]$

$t_3[Y] = t_1[Y]$ and $t_4[Y] = t_2[Y]$

$t_3[Z] = t_2[Z]$ and $t_4[Z] = t_1[Z]$

Where Z to denote (R - (X ∪ Y))

When X →→ Y holds, we say that X multidetermines Y.

Eg: Emp (Empid, Skill, Project)

Empid	Skill	Project
E1	Java	P1
E1	Java	P2
E1	Python	P1
E1	Python	P2

Then, Empid →→ Skill

Empid →→ Project

b) Outline the informal design guidelines for relational schemas

Soln: Informal design guidelines:

1) Design a relation schema so that it is easy to explain its meaning. Do not combine attributes of multiple entity types and relationship types into a single relation.

2) Design the base relation schemas so that no insertion deletion or modification anamolies are present in the relations.

3) As fas as possible, avoid attributes in a base relation whose value may frequently be NULL.

4) Design relation schemas so that they can be joined with equality conditions on attributes that are appropriately related (primary keys, foreign keys) pairs in a way that guarantees that no spurious tuples are generated.

c) Consider relation R with following funtion dependency

EMPPROJ (SSN, PNo, Hours, Ename, Pname, Plocation)

SSN, Pnumber → Hours

SSN → Ename

Pnumber → Pname, Plocation. Is it in 2NF? verify.

KLSS V.D. Institute Of Technology CSE

Soln: (a) The value of an employee's SSN uniquely determines the employee name (Ename)

(b) The value of a project's number (Pnumber) uniquely determines the project name (Pname) & Plocation.

(c) A combination of SSN & PNumber uniquely determines the number of hours the employee currently works on the project per week.

Therefor, the relation is NOT in 2nd Normal Form.

The relation violates 2NF if any non prime attribute is partially dependent on a candidate key.

Since Ename, Pname, Plocation are partially dependent on SSN & Pnumber, the relation is not in 2NF

Q7.

a) i. List the names of managers who have atleast one dependent.

```
select E.Fname, E.Lname
from Employee e
where E.ssn in ( select D.mgr-ssn
                 from Department D
                 where Exists ( select *
                                from Dependent DP
                                where DP.Essn = D.mgr-ssn
                 ));
```

ii. Retrieve the name of each employee who has a dependent with the same first name & same sex

```
Select E.name, E.Lname
from Employee E
where Exists( select *
              from Dependent D
              where D.Essn = E.SSN
              AND D.Dependent_name = E.Fname
              AND D.sex = E.sex );
```

iii. For each project, retrieve project no., project name, and no. of employees working on it.

```
Select P.Pnumber, P.Pname, Count (W.Essn) As No_of_Employee
from Project P, Works_On W
where P.Pno = W.Pno.
Group by P.Pno, P.Pname;
```

KLS's V.D. Institute Of Technology CSE HALIYAL

iv. Retrieve the ssn of all employees who work on project no 1 2 or 3.

```
Select Distinct Essn
from Works_On
where Pno in (1, 2, 3);
```

v. Find sum, max, min, avg salary of employees in Research dept.

```
Select sum(E.Salary), MAX(E.salary), Min(E.Salary), Avg(E.Salary)
from Employee E, department D
where E.Pno = D.Dnumber AND D.Dname = 'Research';
```

b) Why concurrency is needed? Demonstrate with example.

Soln: Several problems can occur when concurrent transactions execute in an uncontrolled manner.

(1) The Lost Update Problem: occurs when two transactions that access the same database items have their operations interleaved in a way that makes the value of some database item incorrect.

(2) The temporary update (or Dirty Read) Problem: occurs when one transactions updates a database item and then the transaction fails for some reason. Meanwhile the updated item is accessed by another transaction before it is changed back to its original value.

(3) The Incorrect Summary Problem: If one transaction is calculating an aggregate summary function on a number of data base items while other transactions are updating some of these items, the aggregate function may calculate some values before they are updated and others after they are updated.

(4) The Unrepeatable Read Pattern: May occur, where a transaction T reads the same item twice and the item is changed by another transaction T' between the two reads. Hence, T receives different values for its two reads of the same item.

8)

a) Given schedules:

S1
T_1 : R(X)
T_2 : R(X)
T_1 : W(Y)
T_2 : W(Y)
T_1 : R(Y)
T_1 : R(Y)

S2
T_3 : W(X)
T_1 : R(X)
T_1 : W(Y)
T_2 : R(Z)
T_2 : W(Z)
T_3 : R(Z)

S3:

KLS's V.D. Institute Of Technology CSE

i) What is the precedence graph for the schedule?

$T_1 : W(X) \rightarrow T_2 : W(4)$
$T_1 : W(4) \rightarrow T_2 : R(4)$
$T_2 : W(4) \rightarrow T_1 : R(4)$

Graph: $T_1 \longleftrightarrow T_2$

ii) Is the schedule conflict serializable? If so what are all the conflicts equivalent serial schedules?

Conflict operations:

① $T_3 : W(X) \rightarrow T_1 : R(X)$
$\Rightarrow$ edge $T_3 \rightarrow T_1$

② $T_2 : W(2) \rightarrow T_3 : R(2)$
$\Rightarrow$ edge $T_2 \rightarrow T_3$

KLSs V.D. Institute Of Technology CSE HALIYAL

The precedence graph is acyclic & view serializable.
The view serial schedule : $T_2 \rightarrow T_3 \rightarrow T_1$

8)

b) Explain triggers with example. write a trigger in SQL to call a procedure "Inform_Supervisor" whenever an employees salary is greater than the salary of his or her direct supervisor in the company database.

Soln: A triggered is a special kind of stored procedure that is automatically executed by the database management system whenever a specified event occurs in the database.

A trigger consisst of three main components: an event, a condition, an action.

Event specifies the database operation that activates the trigger, such as INSERT, DELETE or UPDATE on a particular relation.

A condition is a Boolean Expression that is evaluated once the event occurs.

The action part may consist of one of SQL statement including invocation of stored procedure.

Consider the company database, where each employee has a direct supervisor and a salary. Suppose an employee's salary becomes greater than the salary of

his supervisor, the database system must automatically call a stored procedure named Inform_Supervisor. The trigger can be written as follows,

```
Create Trigger check_salary
After update of salary on employee
for each row
when (new.salary >
        (select salary
         from employee
         where ssn = new.super_ssn))
begin
        call inform_supervisor;
end;
```

Q9.

a) Describe the two phase locking protocol for concurrency control provide example to illustrate how it is ensures serializability in transaction schedule.

Soln: * Two phase locking protocol is a concurrency control protocol that ensure conflict serializability of transaction schedules by controlling how locks are acquired and released.

* A transaction is said to follow the two phase locking protocol if all locking operations (read_lock, write_lock) precede with first unlock operation in the transaction.
* Such a transaction can be divided into two phases:
  1. an expanding, during which new locks on items can be equipped with, but none can be released.
  2. shrinking, during which existing locks can be released but no new locks can be acquired.

T_1	T_2
read_lock(Y);	read_lock(X);
read_item(Y);	read_item(X);
unlock(Y);	unlock(X);

initial values X=20, Y=30
Result serial schedule T_1 followed T_2: X=50, Y=80
Result of serial schedule T_2 followed by T_1 X=70, Y=50

Transactions T_1 & T_2 do not follow the two phase locking protocol because write_lock(x) operation follows the unlock(y) operation in T_1 & similarly the write_lock(y) operation follows the unlock(x) in T_2.

T_1'	T_2'
read_lock(y);	read_lock(x);
read_item(y);	read_item(x);
write_lock(x);	write_lock(y);
unlock(y);	unlock(x);
read_item(x);	read_item(y);
x := x+y;	y := x+y;
write_item(x);	write_item(y);
unlock(x);	unlock(y);

KLS's V.D. Institute Of Technology CSE HAL

If every transaction in a schedule follows the two phase locking protocol, the schedule is guaranteed to be serializable.

b) Explain the characteristics of NOSQL system

Soln: ① Scalability: In NOSQL systems horizontal scalability is generally used, where the distributed system is expanded by adding more nodes for data storage & processing as the volume of data grows.

② Availability, Replication & Eventual consistency:
Many applications that use NOSQL system require continuous system availability. To accomplish this, data is replicated over two or more nodes in a transparent manner, so that if one node fails, the data is still available on other nodes.
Replication improves data availability and can also improve read performance because read requests can often be serviced from any of the replicated data nodes.

③ Replication Models: Two major replication models are used in NOSQL systems: master-slave and master-master replication.
Master-slave replication requires one copy of the master copy; all the write operations must be applied to the

and then propagated to the slave copies. usually using eventual consistency.

(4) Sharding of Files: Sharding seeks to distributing the load of accessing the file records to multiple nodes.

(5) High Performance Data access: It is necessary to find individual records or objects from among the millions of data records or objects in a file. Two techniques used;

(1) Hashing: Hashing h(k) is applied to the key k, and location of the object with key k is determined by value of h(k)

(2) Range partitioning: the location is determined via a range of key values.

10 a) Explain binary locks and shared lock with algorithms

Soln: Binary lock can have two states or values. locked or unlocked. A distinct lock is associated with each database item x. If lock on x is 1, item cannot be accessed by database operation, if '0', item can be accessed when requested.

```
lock_item(x):
B: if Lock(x) = 0
      then LOCK(x) ← 1
   else
      begin
      wait (until LOCK(x) = 0
      go to B
      end;
unlock_item(X);
   LOCK(x) ← 0;
   if any transactions are waiting
      then wakeups one of the waiting transactions.
```

KLS's V.D. Institute of CSE HALIYAL

Shared / Exclusive locks:- If a transaction is to write an item X, it must have exclusive access to X, called shared / exclusive or read / write locks. A lock is associated with an item X, LOCK(X), has three possible states. read-locked, write locked or unlocked.

read locked item, because other transactions are allowed to read the item.

write locked item, because a single transactions exclusively holds the locks on the item.

read_lock(Y):

B: If LOCK(X) = "unlocked"
then begin LOCK(X) ← "read locked"
no-of-reads(X) ← 1
end

else if LOCK(X) = "read-locked"
then no_of_reads(X) ← no of-reads(X) + 1

else begin
wait (until LOCK(X) = "unlocked"
and the lock manager wakes up the transaction
go to B
end;

KLS's V.D. Institute Of Technology CSE HALIYAL

b) Explain MongoDB data model, CRUD operation and distributed system characteristics.

Sol no:

* The MongoDB is a document oriented NOSQL database that stores data in the form of documents rather than tables & rows.
* documents are grouped into collections, which are analogous to relations in database
* Mongodb supports embedded documents and arrays allowing complex data structures to be stored within a single document.

CRUD Operations:

(1) Create operation: data is inserted into a collection using insert operation, which add new document to a collection

(2) Read operation: Data retrieval is performed using query operations that select documents based on specified condition

(3) Update operation: existing documents can be modified using update operation

(4) Delete operation: Documents can be deleted using delete operations based on given criteria

(5) MongoDB supports rich query expressions like indexing, aggregation for efficient data access.

(Bhagyat G. Gaundar)

HOD
Computer Science & Engineering
KLS Vishwanathrao Deshpande
Institute of Technology, Haliyal