SPRING MAKEUP MID SEMESTER EXAMINATION-2023

School of Computer Engineering
Kalinga Institute of Industrial Technology, Deemed to be University
Database Management System
[CS-2004]

Time: 1 1/2 Hours **Full Mark: 20**

All Questions are compulsory.
The figures in the margin indicate full marks. Candidates are required to give theiranswers in their own words as far as practicable and all parts of a question should beanswered at one place only.

1. Answer all the questions. [1 x 5]

a) What is the dependency preservation property for a decomposition? Why is it important?

Ans: The dependency preservation refers to the property that when a relation is decomposed into smaller relations, the functional dependencies that hold in the original relation are preserved in the smaller relations. This helps to maintain data integrity and consistency, prevent data anomalies, and improve the efficiency of queries and data retrieval operations.

b) Differentiate between weak entity set and strong entity set. How is weak entity set identified?

Ans:

Strong Entity	Weak Entity
A strong entity set can exist independently of other entity sets and has a primary key that uniquely identifies each entity within the set	A weak entity set cannot exist independently and doesn't contain any key attribute of it's own.

Weak entity sets are represented using **Double Rectangle**

c) Explain disjointness and completeness constraints on specialization.

Ans: In specialization, which is the process of creating new entity sets based on specific characteristics, there are two important constraints that determine the relationship between the new specialized entity sets and the original entity set: disjointness and completeness. The disjointness constraint specifies that an entity can belong to at most one of the specialized entity sets, while the completeness constraint specifies whether an entity in the original entity set must belong to one of the specialized entity sets or not. Total completeness means that every entity must belong to at least one of the specialized entity sets, while partial completeness means that some entities may not belong to any specialized entity set.

d) What are the rules that must be satisfied by the foreign key?

Ans: The following rules must be satisfied by the foreign key:

- Uniqueness
- Referential Integrity
- Consistency

e) Why SQL doesn't allow to compare an attribute value to NULL using = and <> operators – justify.

Ans: SQL doesn't allow comparing attribute values to NULL using the equals (=) and not equals (<>) operators because NULL is not a value, but a placeholder that represents the absence of a value. Comparing a value to NULL using these operators would result in an unknown or NULL value. Instead, SQL provides the IS NULL and IS NOT NULL operators to test for the presence or absence of a value.

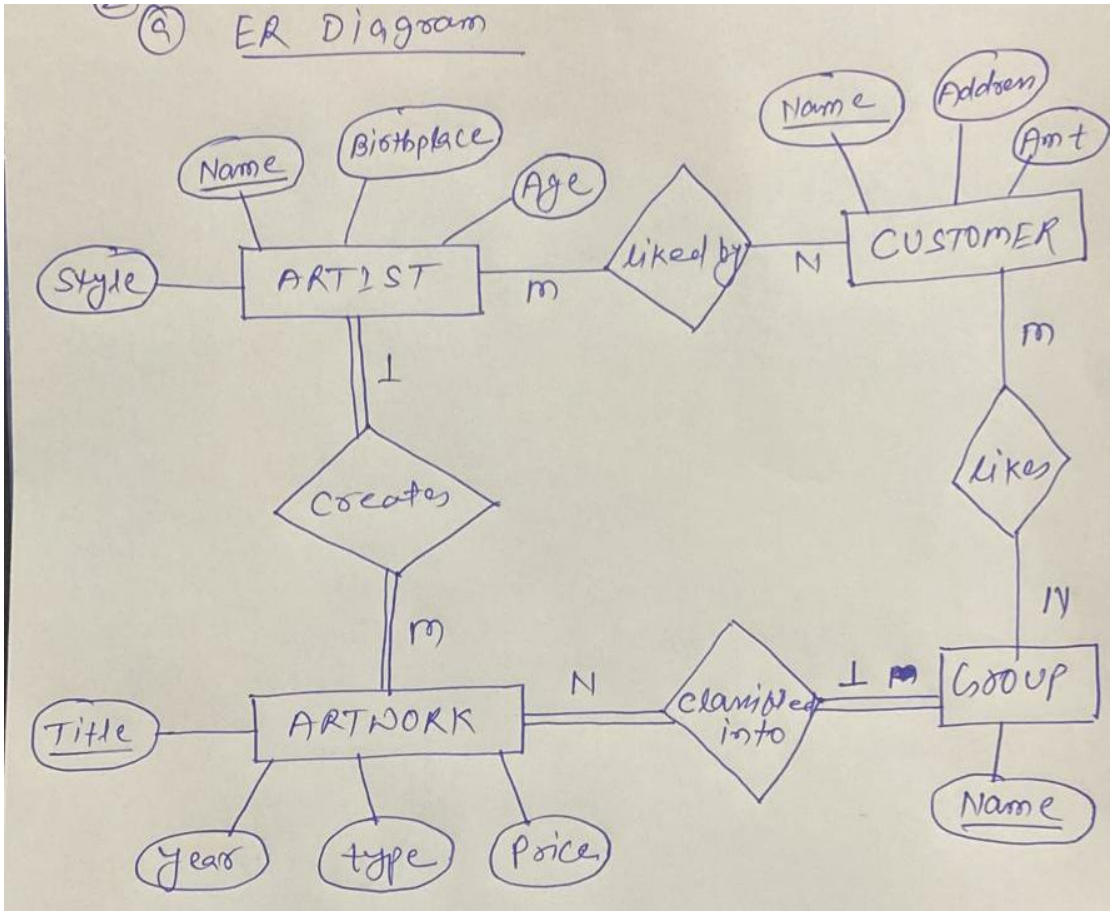
2.

a. Draw the ER diagram for the ArtBase gallery:

Galleries keep information about artists, their names (unique), birthplaces, age, and style of art. For each piece of artwork, the artist, the year it was made, its unique title, its type of art, and its price must be stored. Pieces of artwork are also classified into groups of various kinds, a given piece may belong to more than one group. Each group is identified by a name that describes the group. Finally, galleries keep information about customers. For each customer, galleries keep that person's unique name, address, total amount spent in the gallery, and the artists and groups of art that the customer tends to like.

Make necessary assumptions. [3 Marks]

Ans:



b. Also, convert the above ER diagram into relational schemas and specify primary and foreign keys. [2 Marks]

Artist(Name, Birthplace, Age, Style)
Customer(Name, Address, Amt)
Artwork(Title, year, type, price, Artistname) (FK) groupname (FK)
Group (Name)
LikedBy (Artist_Name, C_Name)
PK/FK PK/FK
Likes (C_Name, Group_name)
PK/FK PK/FK.

3.

a. Consider the relation R (A, B, C, D, E) and the set of functional dependencies A→BC, CD→E, B→D, E→A. Decompose R into 2NF and 3NF relations. [3 Marks]

$A^+ = \{A, B, C, D, E\}$
$B^+ = \{B, D\}$
$C^+ = \{C\}$
$D^+ = \{D\}$
$E^+ = \{A, B, C, D, E\}$

Key of the relation A

As there is no partial dependency, the relation R is already in 2NF.

3NF

As per the rule of transitive dependency, in B→D, B being a non prime attribute determines another non prime attribute D. Hence not in 3NF. Decomposing R in R_1 & R_2

R_1 (B, D)
R_2 (A, B, C, D, E)

Now R_1 and R_2 are in 3NF.

b. Consider two sets of functional dependencies F1={A→C, AC→D, E→AD, E→H} and F2={A→CD, E→AH}. Are they equivalent? [2 Marks]

Ans:

$F1^{+}$={A→C, AC→D, E→AD, E→H, A→CD, E→AH }

$F2^{+}$={ A→CD, E→AH, A→C, AC→D, E→AD, E→H}

F1 covers F2. F2 covers F1. So, F1 and F2 are equivalent.

4. Consider the COMPANY database schema:

Employee (ssn, fname, lname, bdate, address, gender, salary, super_ssn, dno)

Department (dno, dname, mgr_ssn, mgr_start_date)

Dept_Location (dno, dloc)

Project (pno, pname, plocation, dno)

Works_On (essn, pno, hours)

Dependent (essn, dependent_name, gender, bdate, relationship)

Solve the following querie using relational algebra expressions. [1 Mark X 5]

a. List the names of all employees who have a dependent with the same first name as themselves.

Ans: $\pi_{lname, fname}(Employee \bowtie_{fname=dependent\ name\ AND\ ssn=essn} (Dependent))$

b. Find the names of employees who are supervised by ‘ Rakesh’.

Ans: $\pi_{lname, fname}(Employee \bowtie_{ssn=super\_ssn}(\pi_{ssn}(\sigma_{fname='Rakesh'}(Employee))))$

c. Retrieve the names of all employees who work on every project.

Ans:
$$proj\_emp(pno, ssn) \leftarrow \sigma_{fpno, ssn}(Works\_on)$$
$$all\_proj \leftarrow \pi_{pno}(Project)$$
$$emp\_all\_proj \leftarrow proj\_emp \div all\_proj$$
$$result \leftarrow \pi_{lname, fname}(Employee \bowtie emp\_all\_proj)$$

d. Retrieve the average salary of all female employees.

Ans: $g_{average(salary)}(\sigma_{gender='Female'}(Employee))$

e. List the last names of department managers who have no dependents.

Ans:

$$dept\_mngr(ssn) <- \pi_{mgr\_ssn}(Department)$$
$$emp\_depnt(ssn) <- \pi_{essn}(Dependent)$$
$$result\_emp <- dept\_mngr - emp\_depnt$$
$$result <- \pi_{lname}(Employee \bowtie result\_emp)$$

*** Best of Luck ***