5. Relational Database Design

Q.1 Define Normalization.

⇒ Normalization is a step by step decomposition of complex records into simple records.

- Normalization is a process of organizing data in database in more efficient forms.
- This process is also called as canonical synthesis.

Q.2 Write short note on : Functional Dependency

⇒ i) Functional dependency provides a constraints between various attributes of a relation.

ii) The Functional dependency is a relationship that exists between two attributes.

iii) It typically exists between primary key and non-key attribute within a table.

iv) In a relation R with attributes X and Y represented as R(X, Y), where Y is functionally dependent on X or we can say X functionally determines Y.

v) This dependency is denoted by (→)
$X \rightarrow Y$.

vi) All the attributes before arrow is called determinant and attribute after arrow is called as determine.

For e.g.;
Consider an employee table -

Emp_id	Ename	Salary	Proj_id	Hours	Allowance
10	Mahesh	50k	E1	44	40k
12	Suresh	20k	E4	31	30k
15	Ganesh	25k	E6	23	12k
18	Mahesh	50k	E2	12	15k

CASE 1: $X \rightarrow Y$

Emp_id → E_name

⇒ As per above constraint, it is possible to have multiple employees with same Ename but different emp_id.
But it is not allowed to have two employees with same emp_id and different ename.

CASE 2: $X \rightarrow YZ$

Emp_id → Ename, Salary

⇒ As per above constraint, it is possible to have multiple employees with same ename and salary.

CASE 3: $XY \rightarrow ZW$

Emp_id, Proj_id → Hours, Allowance

⇒ As per above constraint, it is possible to have multiple emp_id & proj_id pair with same values of hours & allowance

Q.2 Explain various normal Forms with an example.

⇒ • Various Normal Forms as follows -

1) First Normal Form (1NF)

⇒ 1NF states that all attributes in relation must have atomic values and all attributes in a tuple must have a single value from the domain of that attribute.

- A relation is in 1NF, if every row contains exactly one value for each attribute.
- For e.g.,

Consider an employee table,

a) The relation schema not in 1NF is represented as -

Empid	Ename	Esalary	Ecity

b)

Empid	Ename	Esalary	Ecity
10	X	10k	Mumbai, Pune
12	Y	20k	Mumbai
15	Z	7k	Pune
17	W	8k	Mumbai, Delhi

c) To convert into 1NF, the Ecity attribute is divided in atomic domains it may introduce some data redundancy

Empid	Ename	Esalary	Ecity
10	X	10k	Mumbai
10	X	10k	Pune
12	Y	20k	Mumbai
13	Z	7k	Pune
17	W	8k	Mumbai
17	W	8k	Delhi

- So, 1NF will solve all problems related to domain redundancy.

2) Second Normal Form (2NF)

⇒ A relation is in 2NF, if it is in 1NF and all non-key attribute in relation are fully functionally dependent on the primary key of the relation.

- In short 2NF means, it should be in 1NF.
- There should not be any partial dependency on primary key attributes.
- For e.g.;

Consider relation R(A, B, C, D, E, F) with FD's as below,

$A \rightarrow BC$, $B \rightarrow DC$, $D \rightarrow EF$

i) The candidate key is {AD}

→ {A, D, B, C, E, F} selected as primary key.

All attributes are partially dependent on primary key.

Hence, Relation R is not in 2NF.

ii) The 2NF Relation Schema is,
- R_1 (A, B, C, D) with FDs, $A \to BC$, $B \to DC$
- R_2 (D, E, F) with FDs $D \to EF$

3) Third Normal Form (3NF)

⇒ A relation is in 3NF, if it is 2NF and all non-prime attributes of relation are non-transitively dependent on the every key.

- A relation R is in 3NF, if non-trivial FD $X \to A$ holds true where X is superkey and A is prime attributes.
- For e.g.:

Consider relation R (A, B, C, D, E, F) and the FD's below -

$A \to BC$, $B \to D$, $D \to EF$

i) Here, {A} is the candidate key and selected as primary key.
All attributes are full functionally dependent on primary key.
Hence, R is in 2NF
But non-prime attributes B, D, E, F are transitively depend on key.
So, R is not in 3NF.

ii) 3NF relation schema is,

R_1 (A, B, C) with $A \to BC$

R_2 (B, D) with $B \to D$

R_3 (D, E, F) with $D \to EF$

4) Boyce- Codd- Normal Form (BCNF)

⇒ A relation R is said to be in BCNF, if and only if every determinant is a candidate key.

For, above example (3)

⇒

Relation	FDs	Determinant	key	BCNF?
$R_1(\underline{A}, B, C)$	$A \to BC$	A	A	YES
$R_2(\underline{B}, D)$	$B \to D$	B	B	YES
$R_3(\underline{D}, E, F)$	$D \to EF$	D	D	YES