# Unit-3

## 4th Normal Form (4NF) and 5th Normal Form (5NF)

# # Multivalued Dependencies

(a) A multivalued dependency exists when there are at least 3 attributes (like X,Y and Z) in a relation and for value of X there is a well defined set of values of Y and a well defined set of values of Z. However, the set of values of Y is independent of set Z and vice versa.Primary Key:(XYZ)

(b) Because if there are only 2 attributes (like A,B) in a relation and for value of A there is a well defined set of values of B, then we can easily repeat value of A for each value of B. Then the Candidate Key will be (A,B).

(c) The EMP relation with 2 MVDs: ENAME —>> PNAME and ENAME —>> DNAME.

**EMP**

| ENAME | PNAME | DNAME |
|---|---|---|
| Smith | X | John |
| | Y | Anna |

**EMP** (Redundancy bcoz of multivalued Dependency)

| ENAME | PNAME | DNAME |
|---|---|---|
| Smith | X | John |
| Smith | Y | Anna |
| Smith | X | Anna |
| Smith | Y | John |

# # 4th Normal Form (4NF)

For a table to satisfy the Fourth Normal Form, it should satisfy the following two conditions:

1.It should be in the **Boyce-Codd Normal Form**.

2. The table should not have any non-trivial **Multi-valued Dependency**. And for this the relation should have at-least 3 attributes.

# # Fourth Normal Form (4NF)

**(a) The EMP relation with 2 MVDs: ENAME —>> PNAME and ENAME —>> DNAME.**

**(b) Decomposing the EMP relation into two 4NF relations EMP_PROJECTS and EMP_DEPENDENTS.**

(a) **EMP**

| ENAME | PNAME | DNAME |
|---|---|---|
| Smith | X | John |
| Smith | Y | Anna |
| Smith | X | Anna |
| Smith | Y | John |

(b) **EMP_PROJECTS**

| ENAME | PNAME |
|---|---|
| Smith | X |
| Smith | Y |

**EMP_DEPENDENTS**

| ENAME | DNAME |
|---|---|
| Smith | John |
| Smith | Anna |

# Multivalued Dependencies and 4th Normal Form

**Decomposing a relation state of EMP that is not in 4NF:**
**(a) EMP relation with additional tuples.**
**(b) 2 corresponding 4NF relations EMP_PROJECTS and EMP_DEPENDENTS.**

(a) **EMP**

| ENAME | PNAME | DNAME |
|---|---|---|
| Smith | X | John |
| Smith | Y | Anna |
| Smith | X | Anna |
| Smith | Y | John |
| Brown | W | Jim |
| Brown | X | Jim |
| Brown | Y | Jim |
| Brown | Z | Jim |
| Brown | W | Joan |
| Brown | X | Joan |
| Brown | Y | Joan |
| Brown | Z | Joan |
| Brown | W | Bob |
| Brown | X | Bob |
| Brown | Y | Bob |
| Brown | Z | Bob |

(b) **EMP_PROJECTS**

| ENAME | PNAME |
|---|---|
| Smith | X |
| Smith | Y |
| Brown | W |
| Brown | X |
| Brown | Y |
| Brown | Z |

**EMP_DEPENDENTS**

| ENAME | DNAME |
|---|---|
| Smith | Anna |
| Smith | John |
| Brown | Jim |
| Brown | Joan |
| Brown | Bob |

# Multivalued Dependencies and 4th Normal Form

**Decomposing a relation state of EMP that is not in 4NF:**

(a) The COURSE relation with 2 MVDs: SUBJECT —>> LECTURER and SUBJECT —>> BOOKS.

(b) Decomposing the COURSE relation into two 4NF relations.

Course

| SUBJECT | LECTURER | BOOKS |
|---|---|---|
| Mathematics | Alex | Maths Book1 |
| Mathematics | Bosco | Maths Book2 |
| Physics | Rose | Physics Book |
| Chemistry | Adam | Chemistry Book |

4NF

| SUBJECT | LECTURER |
|---|---|
| Mathematics | Alex |
| Mathematics | Bosco |
| Physics | Rose |
| Chemistry | Adam |

| SUBJECT | BOOKS |
|---|---|
| Mathematics | Maths Book1 |
| Mathematics | Maths Book2 |
| Physics | Physics Book |
| Chemistry | Chemistry Book |

# # Join Dependencies and Fifth Normal Form (5NF)

## Join Dependency Definition:

- A relation R is subject to a join dependency or we can say that a relation R is having join dependency if R can always be recreated by joining multiple tables each having a subset of the attributes of R. If one of the relation in the join has all the attributes of the relation R, the join dependency is called trivial.
- If a relation can be recreated by joining multiple tables (R1, R2, R3….Rn) and each of this table have a subset of the attributes of the table, then the table is having Join Dependency.

# # Join Dependencies and Fifth Normal Form (5NF)

## Join Dependency Definition:

- Let ‘R’ be a relation schema and R1,R2,….Rn be the decomposition of R. Then R is said to satisfy the join dependency JD(R1,R2,….Rn) if and only if:

$$\prod R_1(R) \bowtie \prod R_2(R) \bowtie \ldots\ldots\ldots \bowtie \prod R_n(R) = R$$

## Join Dependency Rule:

- JD holds good only if a relation can be retransformed back without any loss of information from the join of certain specified projection (sub-relations) on it.
- JD holds good only for a relation if the join of certain specified projection (sub-relations) on it does not have any extra, missing or false(spurious) tuples.

# # Fifth Normal Form (5NF)

It is also known as **Project-Join Normal Form (PJNF).** A relation R is in 5NF if:

- R is already in 4NF.
- It can-not be further non-loss decomposed or if it is not having any join dependency.

- **Join Dependency can be related to 5NF, wherein a relation R is in 5NF if and only if it is already in 4NF and it cannot be decomposed further or in other words we can say that if R is not having any Join dependency.**

- **If a relation is in 4NF and having Join dependency means if it can be decomposed further than it is not in 5NF. After decomposing the resultant sub relations will be in 5NF.**

# # Fifth Normal Form (5NF)

**(c) The relation SUPPLY with no MVDs is in 4NF but not in 5NF as it has the join dependency : JD(R1, R2, R3).**

**(d) Decomposing the relation SUPPLY into the 5NF relations R1, R2, and R3.**

(c) **SUPPLY**

| SNAME | PARTNAME | PROJNAME |
|---|---|---|
| Smith | Bolt | ProjX |
| Smith | Nut | ProjY |
| Adamsky | Bolt | ProjY |
| Walton | Nut | ProjZ |
| Adamsky | Nail | ProjX |
| Adamsky | Bolt | ProjX |
| Smith | Bolt | ProjY |

**The relation SUPPLY has the join dependency: JD(R1, R2, R3) as R1 ⋈ R2 ⋈ R3 = R**
**So table SUPPLY is not in 5NF**

**5NF Relations: R1, R2 and R3**

(d) **R1**

| SNAME | PARTNAME |
|---|---|
| Smith | Bolt |
| Smith | Nut |
| Adamsky | Bolt |
| Walton | Nut |
| Adamsky | Nail |

**R2**

| SNAME | PROJNAME |
|---|---|
| Smith | ProjX |
| Smith | ProjY |
| Adamsky | ProjY |
| Walton | ProjZ |
| Adamsky | ProjX |

**R3**

| PARTNAME | PROJNAME |
|---|---|
| Bolt | ProjX |
| Nut | ProjY |
| Bolt | ProjY |
| Nut | ProjZ |
| Nail | ProjX |