Activity (Moderate Learner) – With Solution

Context

You have been given the following unnormalized table representing a student enrollment system:

Student_Course(Student_ID, Student_Name, Course_ID, Course_Name, Instructor, Instructor_Office)

Task 1

Normalize the table step-by-step through 1NF, 2NF, 3NF, and BCNF, clearly identifying functional and transitive dependencies at each step.

Solution

Step 1: Identify Anomalies (Unnormalized Relation)
- Insertion Anomaly: Cannot insert instructor details without student enrollment.
- Deletion Anomaly: Deleting student records might lose instructor details.
- Update Anomaly: Instructor details repeated, causing inconsistencies if updates occur.

Step 2: First Normal Form (1NF)
- The table is already in 1NF since there are no repeating groups.

Step 3: Second Normal Form (2NF)
- Identify partial dependency:
  Student_ID → Student_Name
  Course_ID → Course_Name, Instructor, Instructor_Office
- Create separate tables:

Students(Student_ID, Student_Name)
Courses(Course_ID, Course_Name, Instructor, Instructor_Office)
Enrollment(Student_ID, Course_ID)

Step 4: Third Normal Form (3NF)
- Identify transitive dependency:
  Course_ID → Instructor → Instructor_Office
- Further split the Courses table:

Courses(Course_ID, Course_Name, Instructor)
Instructors(Instructor, Instructor_Office)

Step 5: Boyce-Codd Normal Form (BCNF)
- Check BCNF:
  Each determinant is a candidate key in all tables.
- All resulting tables satisfy BCNF.

Final Normalized Tables:
- Students(Student_ID, Student_Name)
- Courses(Course_ID, Course_Name, Instructor)
- Enrollment(Student_ID, Course_ID)
- Instructors(Instructor, Instructor_Office)

Task 2: Verify Normal Form of Another Relation

Task 6: Consider another relation below with the given dependencies. Determine the highest normal form (1NF, 2NF, 3NF, BCNF) satisfied by the relation. Justify.

Relation:
Employee_Project(Emp_ID, Emp_Name, Project_ID, Project_Name, Project_Manager)

Functional Dependencies:
- Emp_ID → Emp_Name
- Project_ID → Project_Name, Project_Manager
- Project_Manager → Project_ID

Solution for Task 2

Step-by-step Analysis:

Step 1: Check 1NF
- Attributes are atomic; thus, relation is in 1NF.

Step 2: Check 2NF
- Relation has composite primary key (Emp_ID, Project_ID).
- Partial dependencies exist:
  Emp_ID → Emp_Name (depends only on Emp_ID)
  Project_ID → Project_Name, Project_Manager (depends only on Project_ID)
- Thus, the relation is NOT in 2NF.

Therefore, the highest normal form currently satisfied by this relation is **1NF**.

Recommendation for 2NF:
Split the relation into separate tables:
- Employee(Emp_ID, Emp_Name)
- Project(Project_ID, Project_Name, Project_Manager)
- Employee_Project(Emp_ID, Project_ID)

This decomposition removes partial dependencies, achieving 2NF and subsequently checking higher forms:
- Project_Manager → Project_ID indicates a further transitive dependency. This must be resolved for 3NF.

For Moderate Learners (Intermediate)

Task 3: Identify Functional Dependencies

Question:
From the following data, identify any functional dependencies:

EmpID	EmpName	DeptID	DeptName
101	Alice	D1	Sales
102	Bob	D2	HR
103	Charlie	D1	Sales

Answer:

- EmpID → EmpName
- DeptID → DeptName

Task 4: Find Candidate Keys

Question:
Given relation R(A,B,C) with FD:

- A → B
- B → C
  What are the candidate keys?

Answer:

- Closure of A: $A^+ = \{A,B,C\}$ (since A → B and B → C)
- So A is a candidate key.

- $B^+ = \{B,C\}$ (not all attributes)
- $C^+ = \{C\}$
- Candidate key: A only.

Task 5: Determine Normal Form

Question:
Relation R(A,B,C) with FD:

- A → B
- B → C
  What is the highest normal form of R?
- **Answer:**
- R is in 2NF becaUse no partial dependencies.
- It is not in 3NF Due to transitive dependency A → B → C.
- The highest normal form is 2NF.

Task 6: Decompose into 3NF

Question:
Normalize R(A,B,C) with FD:

- A → B
- B → C
  into 3NF.

Answer:

- Decompose into:
  - R1(A,B) with A → B
  - R2(B,C) with B → C
- Both satisfy 3NF.

Task 7: Explain Transitive Dependency

Question:
What is transitive dependency? Give an example.

Answer:

- A transitive dependency exists if A → B and B → C, so A → C indirectly.
- Example: In R(A,B,C), if A → B and B → C, then C is transitively dependent on A.