Activity (Fast Learner) – With Solution

Context

Given an unnormalized table:

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

Task 1

Quickly normalize the table from its current form to BCNF, clearly specifying minimal intermediate steps and all functional/transitive dependencies succinctly.

Solution

Initial Dependencies:
- Student_ID → Student_Name
- Course_ID → Course_Name, Instructor, Instructor_Office
- Instructor → Instructor_Office (transitive dependency)

Normalization Steps (Concise):

• 1NF: No repeating groups (already satisfied).

• 2NF: Separate partial dependencies:
  - Students(Student_ID, Student_Name)
  - Courses(Course_ID, Course_Name, Instructor, Instructor_Office)
  - Enrollment(Student_ID, Course_ID)

• 3NF: Remove transitive dependencies:
  - Courses(Course_ID, Course_Name, Instructor)
  - Instructors(Instructor, Instructor_Office)

• BCNF Verification:
  - All tables have determinants as candidate keys.

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

Quick Check for Anomalies:
- No insertion, deletion, or update anomalies remain.

Task 2: Identifying Normal Form of Relations

Task 6: Given the relation below and the set of dependencies, identify the highest normal form (1NF, 2NF, 3NF, BCNF) that this relation currently satisfies. Justify your answer.

Relation:
R(A, B, C, D, E)

Functional Dependencies:
- A → B, C
- B → D
- D → E

Solution for Task 2

Step-by-step Analysis:

Step 1: Check 1NF
- 1NF requires atomic attributes. All attributes (A, B, C, D, E) are atomic.
- Thus, relation R is in 1NF.

Step 2: Check 2NF
- 2NF requires no partial dependency. All non-key attributes must depend on the entire candidate key.
- Assuming A is a candidate key, dependencies are:
  A → B, C (Fully dependent on key A)
  B → D (B is partially dependent on A if AB were a composite key, but here A alone is sufficient)
  D → E (D dependent on B, which is dependent on A)
- No partial dependencies, thus the relation is in 2NF.

Step 3: Check 3NF
- 3NF requires no transitive dependencies (non-key attributes shouldn't depend transitively on a candidate key).
- A → B, B → D creates a transitive dependency (A → B → D).
- B → D → E creates further transitivity.
- Due to transitive dependencies, relation R is NOT in 3NF.

Thus, the highest normal form currently satisfied by relation R is 2NF.

Task 3: Minimal Cover

Question:
Given the following set of functional dependencies on relation R(A, B, C, D):

- A → BC
- B → C
- A → B
- AB → C
  Find the minimal cover for this set.

Answer:

1. Decompose RHS to single attributes:
   - A → B
   - A → C
   - B → C
   - AB → C
2. Remove redundant dependencies:
   - Since A → B and A → C are present, AB → C is redundant (because A alone determines C).
3. Minimal cover:
   - A → B
   - A → C
   - B → C

Task 4: Attribute Closure and Candidate Keys

Question:
Given relation R(A,B,C,D,E) and functional dependencies:

- A → BC
- CD → E
- B → D
  Find the closure of {A} and identify candidate keys.

Answer:

- Compute $\{A\}^+$:
  Start with {A}
  - A → BC, add B and C → {A,B,C}
  - B → D, add D → {A,B,C,D}
  - CD → E, since C and D are in closure, add E → {A, B, C, D, E}
- $\{A\}^+$ = {A,B,C,D,E} = all attributes → A is a candidate key.

Task 5: Dependency Preservation Check

Question:
Relation R(A,B,C) is decomposed into:

- R1(A,B) with FD: A → B
- R2(B,C) with FD: B → C
  Is this decomposition dependency preserving?

Answer:

- The original FDs are: A → B, B → C
- Check if all FDs are preserved in the decomposition:
  - A → B is in R1
  - B → C is in R2
- Both FDs preserved → Decomposition is dependency preserving.