Advanced Databases - Exercise Sheet No. 2 Solutions

Exercise I

1- Show that R is in Second Normal Form (2NF)

Let's find the candidate keys first.

C does not appear on the right-hand side of the functional dependencies,

So, the candidate keys must contain C.

Thus, the possible combinations are: {A,C}, {B,C}, {E,C}, {D,C}.

Let's compute closures:

$\{E,C\}^+$ and $\{D,C\}^+$ are not equal to U.

Let's compute AC^+ and BC^+:

Compute closures:

- AC^+: from $AC \to BE$ get B, E. Now we have A, B, C, so $ABC \to D$ gives D. Hence $AC^+ = \{A, B, C, D, E\} = U. \Rightarrow AC$ is a key.
- BC^+: from $BC \to A$ get A. With A, B, C we get D by $ABC \to D$, and from $AC \to BE$ we get E. Hence $BC^+ = U. \Rightarrow BC$ is a key.

Check minimality: Both AC and BC are minimal (A alone or B alone or C alone are not super keys).

Therefore, the candidate keys are: AC and BC.

Now, let's check whether R is in 2NF:

Examine FDs:

- AC → BE: AC is a whole candidate key (not a proper part) and determines E (non-key). This is a full-key dependency, not a partial dependency.
- ABC → D: ABC is a super key (contains AC), so not a partial dependency.

So, non-key attributes D and E depend on the whole candidate keys AC and BC.

Thus, R is in 2NF.

2- Determine whether R is in Third Normal Form (3NF).
   We need to see if a non-key (D or E) depends on a non-key attribute.
   There is no case where a non-key attribute (D or E) depends on another non-key (like D depending on E). So, R is in 3NF.

Exercise II

1) Is CourseRegistration in 1NF?

Yes. All attributes shown are atomic (atomic values for StudentName, InstructorEmail, Semester, Grade, etc.) and rows are uniquely identifiable. So, the relation is in **1NF**.

2) Candidate key(s)

From the FDs:

(StudentID, CourseID, Semester) determines Grade,

and using StudentID → StudentName

and CourseID → (CourseName, Instructor, InstructorEmail), we can derive all attributes.

So:
Candidate key = **{StudentID, CourseID, Semester}.**

It is minimal (dropping any attribute from it would not determine Grade or the other attributes), so there is a single candidate key.

3) The relation is not in 2NF

We have: StudentID → StudentName.

StudentName is a **non-key** attribute and is determined by **StudentID**, which is a proper subset of the candidate key.

=> **partial dependency** (violates 2NF).

Hence, the relation is **not in 2NF**.

Decomposition to achieve 2NF:

1. **Student**(StudentID PK, StudentName)
2. **Course**(CourseID PK, CourseName, Instructor, InstructorEmail)
3. **Registration**(StudentID, CourseID, Semester, Grade)

4) Are the 2NF tables in 3NF?

Check each table for transitive dependencies:

- **Student(StudentID → StudentName)**
  - Left side StudentID is the key for the Student table. So FD left side is a key ⇒ satisfies 3NF. No transitive dependency present. => **Student is in 3NF** (also BCNF).
- **Course(CourseID → CourseName, Instructor, InstructorEmail)**
  - CourseID is the key for Course. Left side is a key ⇒ satisfies 3NF. No transitive dependency given. => **Course is in 3NF**.
  - (Note: if there is an FD Instructor → InstructorEmail in reality, you might further split Instructor into its own table; but that FD was not given, so we assume none.)
- **Registration((StudentID,CourseID,Semester) → Grade)**
  - Left side is the key for Registration; Grade depends on the whole key. No other non-key attributes present to create a transitive chain. => **Registration is in 3NF**
