**Decomposition to 2NF**

Students

|  |  |
| --- | --- |
| StudentID | StudentName |

Courses

|  |  |  |
| --- | --- | --- |
| CourseID | CourseName | Instructor |

Enrollment

|  |  |  |  |
| --- | --- | --- | --- |
| StudentID | CourseID | Grade | Semester |

In the context of Second Normal Form (2NF), the focus is on eliminating redundancy arising from partial dependencies, where a non-prime attribute is functionally dependent on only a part of a composite primary key. In the original "Enrollments" table, for instance, StudentName is solely dependent on StudentID, not on the composite key (StudentID, CourseID). Similarly, CourseName and Instructor are each dependent only on CourseID. To align with 2NF principles, these dependencies necessitate the division of the original table into distinct entities, ensuring that each non-prime attribute is fully functionally dependent on the entire primary key.

**Decomposition to 3NF**

Students

|  |  |
| --- | --- |
| StudentID | StudentName |

Courses

|  |  |
| --- | --- |
| CourseID | CourseName |

Instructors

|  |  |
| --- | --- |
| CourseName | Instructor |

Enrollment

|  |  |  |  |
| --- | --- | --- | --- |
| StudentID | CourseID | Grade | Semester |

In the 2NF-structured tables, a notable transitive dependency is observed with the Instructor attribute, which is dependent on CourseName, a non-prime attribute. To rectify this, 3NF dictates the segregation of such attributes into separate tables. This separation effectively ensures that all non-prime attributes in a table are directly dependent on the primary key, thereby enhancing the logical consistency and integrity of the database schema.