Task 4.2: Advanced Normalization

1) Primary key determination:

The primary key of the CourseSchedule table is {StudentID, TimeSlot, Room} because the combination of StudentID, TimeSlot, and Room uniquely identifies a student's enrollment in a specific course section.

2) Functional Dependencies:

- StudentID -> StudentMajor (each student has one major)
- CourseID -> CourseName (each course has a fixed name)
- InstructorID -> InstructorName (each instructor has one name)
- Room -> Building (rooms are unique across campus, so room determines building)
- {TimeSlot, Room} -> CourseID (a time slot and room determine the course section, hence the course)
- {TimeSlot, Room} -> InstructorID (a time slot and room determine the instructor teaching the section)

3) BCNF check:

The table is not in BCNF because several functional dependencies violate BCNF:

- StudentID -> StudentMajor: StudentID is not a superkey.
- CourseID -> CourseName: CourseID is not a superkey.

4) Decomposition to BCNF:

Step-by-Step Decomposition:

1. Decompose using StudentID -> StudentMajor**:**

   Students(StudentID, StudentMajor)
   FD: StudentID -> StudentMajor
   Key: StudentID

2. Decompose using CourseID -> CourseName**:**

   Courses(CourseID, CourseName)
   FD: CourseID -> CourseName
   Key: CourseID

3. Decompose using InstructorID -> InstructorName**:**

   Instructors(InstructorID, InstructorName)
   FD: InstructorID -> InstructorName
   Key: InstructorID

4. Decompose using Room -> Building**:**

Rooms(Room, Building)
FD: Room -> Building
Key: Room

5. Decompose using {TimeSlot, Room} -> CourseID and {TimeSlot, Room} -> InstructorID**:**

   Sections(TimeSlot, Room, CourseID, InstructorID)
   FDs: {TimeSlot, Room} -> CourseID, {TimeSlot, Room} -> InstructorID
   Key: {TimeSlot, Room}

6. The remaining table for enrollments:

   Enrollments(StudentID, TimeSlot, Room)
   Key: {StudentID, TimeSlot, Room}
   No non-trivial FDs beyond the key.

Final BCNF Schema:

- Students(StudentID, StudentMajor)
- Courses(CourseID, CourseName)
- Instructors(InstructorID, InstructorName)
- Rooms(Room, Building)
- Sections(TimeSlot, Room, CourseID, InstructorID)
- Enrollments(StudentID, TimeSlot, Room)

Foreign Keys:

- Sections.CourseID references Courses.CourseID
- Sections.InstructorID references Instructors.InstructorID
- Enrollments.StudentID references Students.StudentID
- Enrollments.TimeSlot, Enrollments.Room references Sections.TimeSlot, Sections.Room

5) Potential Loss of Information:

No, the data was not lost. We simply carefully laid out the general table into thematic folders: students, courses, teachers, classrooms and schedule separately.

All the information was saved, but now without duplication. For example, the audience data is now stored in one place, and is not repeated for each entry.