INFO20003 Tutorial – Week 9

(Tutorial: Normalisation)

Objectives:

This tutorial will cover:

I. Review of normalisation concepts – 15 mins
II. Normalisation exercises – 35 mins

Key Concepts:

NOTE for students: *This is a brief summary of some of the concepts taught in lectures. The lectures contain detailed content related to these and many more concepts. These notes should be considered quick revision instead of a sole resource for the course material.*

- Anomalies

  Consider the following instance of the relation Allocation:

CourseNumber	Tutor	Room	Seats
INFO20003	Farah	Alice Hoy 109	30
COMP10001	Farah	EDS 6	25
INFO30005	Patrick	Sidney Myer G09	20
COMP20005	Alan	Sidney Myer G09	20

- Functional dependency
- Determinants
- Key and non-key attributes
- Partial functional dependency
- Transitive functional dependency
- Armstrong's Axioms
- Normalisation and normal forms

Exercises:

1. Consider the relation Diagnosis with the schema Diagnosis (DoctorID, DocName, PatientID, DiagnosisClass) and the following functional dependencies:

   DoctorID → DocName
   DoctorID, PatientID → DiagnosisClass

   Consider the following instance of Diagnosis:

DoctorID	DocName	PatientID	DiagnosisClass
D001	Alicia	P888	Flu
D002	John	P999	Lactose intolerance
D003	Jennifer	P000	Flu
D002	John	P111	Fever

   Identify different anomalies that can arise from this schema using the above instance.

2. Consider a relation R (A, B, C, D) with the following FDs:

$$AB \rightarrow C,\ AC \rightarrow B,\ BC \rightarrow A,\ B \rightarrow D$$

The possible candidate keys of R are AB, AC, and BC, since each of those combinations is sufficient to uniquely identify each record. Let's consider AB for instance. From AB → C we see that AB uniquely identifies C, and since B alone uniquely identifies D, AB together have covered CD, i.e. the entire set of attributes.

List all the functional dependencies that violate 3NF. If any, decompose R accordingly. After decomposition, check if the resulting relations are in 3NF, if not decompose further.

3. Consider the following relation StaffPropertyInspection:

StaffPropertyInspection (propertyNo, pAddress, iDate, iTime, comments, staffNo, sName)

The FDs stated below hold for this relation:

propertyNo, iDate → iTime, comments, staffNo, sName
propertyNo → pAddress
staffNo → sName

From these FDs, it is safe to assume that propertyNo and iDate can serve as a primary key. Your task is to normalise this relation to 3NF. Remember in order to achieve 3NF, you first need to achieve 1NF and 2NF.

Take-Home Exercises (not covered in tutes):

4. The following Report table is used by a publishing house to keep track of the editing and design of books by a number of authors:

report_no	editor	dept_no	dept_name	dept_addr	author_id	auth_name	auth_addr
4216	woolf	15	design	argus1	53	mantel	cs-tor
4216	woolf	15	design	argus1	44	bolton	mathrev
4216	woolf	15	design	argus1	71	koenig	mathrev
5789	koenig	27	analysis	argus2	26	fry	folkstone
5789	koenig	27	analysis	argus2	38	umar	prise
5789	koenig	27	analysis	argus2	71	koenig	mathrev

By looking at the data, we see that functional dependencies in the Report table are the following:

report_no → editor, dept_no
dept_no → dept_name, dept_addr
author_id → auth_name, author_addr

The candidate key for this relation is (report_no, author_id) since we need these two attributes to uniquely identify each record. Thus we have:

Report (<u>report_no</u>, editor, dept_no, dept_name, dept_addr, <u>author_id</u>, auth_name, auth_addr)

a. Is the Report table in 2NF? If not, put the table in 2NF.
b. Are there any insert, update or delete anomalies with these 2NF relations?

5. Consider the following relation:

Class (courseNumber, roomNumber, instructorName, studentNumber,
workshopNumber, grade, tutor)

The following functional dependencies hold for this relation:

workshopNumber → tutor
studentNumber, courseNumber → grade, workshopNumber
courseNumber → roomNumber, instructorName

Normalise this relation into 3NF.

END OF TUTORIAL