Database Management System – cs422 DE

Assignment 6 – Week 7

This assignment is based on lecture 8 (chapter 16 & 17).

○ Submit your *own work* on time. No credit will be given if the assignment is submitted after the due date.
○ Note that the completed assignment should be submitted in .pdf format only.
○ In MCQs, if you think that your answer needs more explanation to get credit then please write it down.
○ You are encouraged to discuss these questions in the Sakai forum.

(1) 3NF essentially identifies

A. 1-* relationships
B. * - * relationships
C. 1-1 relationships
D. None of the above

ANS: D

(2) While checking our tables for normalization, if we find that they are not even in 2NF then we must have missed some

A. 1-* relationships
B. * - * relationships
C. 1-1 relationships
D. None of the above

ANS: B

(3) How to identify parent and child entities in a relationship?

ANS: The table being referenced is the parent; the table holding the FK is the child.

- Parent entity: It's primary key is referenced by another entity. Exists independently
- Child entity: Contain a foreign key referencing the parent's primary key. Depends on the parent for existance

(4) Solve review question 17.2/ 16.2 (a,b,c,d,g,i) from 5th /4th edition of the course text book.

ANS:

a. Strong entity type rules:
   - Create one relation for each strong entity
   - Include all simple attributes
   - The primary key of the entity becomes the PK of the relation
   - E.g: STUDENT(studentID, name, dob, address)
b. Weak entity type rules:
   - Create a relation for the weak entity.
   - Include its partial key and the PK of the owner entity.
   - The PK is a composite key (owner PK + partial key).

- Owner PK is also a foreign key (FK).
- E.g: DEPENDENT(empID, dependentName, DOB, relationship) (empID → EMPLOYEE.empID)

c. One-to-many (1:*) binary relationship type rules:

- Add the PK of the "1" side as an FK to the relation on the "many" side.
- Include any relationship attributes in the "many" side relation.
- E.g: DEPARTMENT(deptID, deptName); EMPLOYEE(empID, name, deptID)

d. One-to-one (1:1) binary relationship type rules:

- Add the PK of one entity as an FK in the other.
- Prefer the side with total participation.
- Relationship attributes go where the FK is placed.
- E.g: PERSON(personID, name); PASSPORT(passportNo, issueDate, personID)

e. One-to-one (1:1) recursive relationship type rules:

- Add a recursive FK referencing the same relation's PK.
- The FK represents the relationship role.
- E.g: EMPLOYEE(empID, name, supervisorID) (supervisorID -> EMPLOYEE.empID)

f. Superclass / subclass relationship type rules:

- Create a relation for the superclass with common attributes.
- Create a relation for each subclass.
- Subclass PK = superclass PK (also an FK)
- E.g: EMPLOYEE(empID, name, salary); MANAGER(empID, bonus); ENGINEER(empID, level)

g. Many-to-many (:) binary relationship type rules:

- Create a new relation for the relationship.
- Include PKs of both participating entities as FKs.
- Combined FKs form the composite PK.
- Include relationship attributes.
- E.g: WORKS_ON(empID, projID, hoursWorked)

h. Complex relationship type rules:

- Treat like - relationships.
- Create a new relation.
- Include PKs of all participating entities as FKs.
- Combined keys form the PK.
- E.g: SUPPLY(supplierID, partID, projectID, quantity

i. Multi-valued attributes rules:

- Create a separate relation for each multi-valued attribute.
- Include the PK of the original entity as an FK.
- PK = (entity PK + attribute value).
- E.g: EMP_PHONE(empID, phoneNumber)

(5) Discuss how the technique of normalization can be used to validate the relations derived from the conceptual data model. (17.3/16.3)

ANS: Normalization can be used to validate the relationships inferred from the conceptual data model by checking whether those relationships are well-structured and free from design flaws. By applying 1NF, 2NF, and 3NF, we ensure that attributes are atomic, fully dependent on the primary key, and free from transitive dependencies. Any violations indicate problems such as missing entities, unresolved many-to-many relationships, or misplaced attributes in the conceptual model. If the relationships can be normalized to at least 3NF without loss of information, it confirms that the conceptual data model is accurate and consistent.

(6) Solve exercise 17.8/16.8 from the 5th /4th edition of the course text book. In the ERD, only those attributes are listed which are PK for that entity. You are required to add more attributes to the relations which will be logically applicable to that entity.

ANS:

PAYMENT_METHOD(**pMethodNo**, methodName, provider, description)

CUSTOMER(**customerNo**, name, phone, email, billingAddress)

EMPLOYEE(**employeeNo**, name, title, phone, email)

PRODUCT(**productNo**, productName, unitPrice, description)

SHIPMENT_METHOD(**sMethodNo**, methodName, carrier, serviceLevel)

ORDER(**orderNo**, orderDate, status, totalAmount, → customerNo FK, → employeeNo FK)

INVOICE(**invoiceNo**, invoiceDate, amountDue, dueDate, → orderNo FK (UNIQUE), → pMethodNo FK)

SHIPMENT(**shipmentNo**, shipDate, trackingNo, shipStatus, → employeeNo FK, → sMethodNo FK)

ORDER_DETAIL(**orderNo**, **productNo**, lineNo, quantity, unitPrice, discount, → shipmentNo FK)

PK could be (**orderNo, productNo**) or (**orderNo, lineNo**) depending on whether the same product can appear multiple times.