DATABASE MANAGEMENT SYSTEMS DESIGN

ASSIGNMENT 5

DAVID GOMEZ CAMARGO

I. UPDATE ANOMALIES

- **INSERTION ANOMALY:** In the given relation *BOOK_PUBLISHER*, if a new publisher needs to be entered in the relation without any *Book_Id* or *Title,* this publisher cannot be inserted into the table, therefore, it will result in database inconsistencies.

- **UPDATE ANOMALY:** If a *Publisher_Name* is wrongly entered and needs to be updated, this name needs to be updated in all the records from said publisher, otherwise that results in data inconsistencies.

- **DELETE ANOMALY:** If *Publisher_Name* for a book needs to be deleted from the table, then all records for the given publisher will be deleted and that can cause inconsistencies.

II. FUNCTIONAL DEPENDENCY

A *functional dependency (FD)* is a constraint between two sets of attributes in a relation. Given a relation *R*, a functional dependency *X* → *Y* holds if, for each instance of *R*, the values of attributes in *X* uniquely determine the values of attributes in *Y*.

Possible sources of information that define functional dependencies in a relation schema include:

- **DOMAIN CONSTRAINTS:** Rules derived from the nature of the attributes and their relationships, based on the real-world context.

- **BUSINESS RULES:** Policies and regulations that govern how the data should be organized and managed.

- **DATA SEMANTICS:** The meaning of the data and the relationships between different attributes.

III. NORMALIZATION - PATIENT/DOCTOR

R(Doctor#, Patient#, Date, Diagnosis, Treat_code, Charge)

I'd infer the following functional dependencies:

- *{Doctor#, Patient#, Date} → {Diagnosis, Treat_code, Charge}*
- *{Treat_code} → {Charge}*

Since there are no partial dependencies, the given relation *R* is in 2NF already, meaning the relation is in First Normal Form and every non-primary-key attribute is fully functionally dependent on the primary key.

However, relation *R* is not in 3NF because the *Charge* is a non-prime attribute that is determined by another non-prime attribute, *Treat_code.* Therefore, *R* needs to decompose:

- *R1(Doctor#, Patient#, Date, Diagnosis, Treat_code)*
- *R2(Treat_code, Charge)*

I could further infer that *R1* and *R2* relations are in First and Second Normal Form and no non-primary-key attribute is transitively dependent on the primary key.

IV. KEY - SUPERKEY - NON-PRIME ATTRIBUTES

R(A, B, C, D, E, F, G, H, I, J)

Functional dependencies:

- $F \rightarrow A, B$
- $C, D \rightarrow E$
- $C \rightarrow F, G$
- $H \rightarrow I, J$
- $D \rightarrow H$

a. Yes, *CDE* is a superkey of *R* because if we take the closure of *CDE* it will be: $\{C, D, E\}+ = \{C, D, E, F, G, H, I, J, A, B\}$ which includes all the attributes in the relation *R.*

b. No, *CDE* is not a key of *R* because if we take the closure of *CD:* $\{C, D\}+ = \{C, D, E, F, G, H, I, J, A, B\}$ we get all the attributes. Hence *E* is not required, *CD* alone is enough to determine all the attributes as *CD* itself determines *E*.

c. *CD* is the only key because when we try to add any other attribute in *CD* it becomes a superkey. Therefore, the prime attributes of *R* are *C* and *D* and *A, B, E, F, G, H, I, J* are non-prime attributes.