By Uraz C Turker
2024
Data Engineering

Nomenclature for Functional Dependencies, Normal Forms, Attributes, and Keys

Attributes and Keys

1. **Candidate Key**:
   - **Definition**: A minimal set of attributes that can uniquely identify a record in a table. A candidate key must be unique (no two rows can have the same value) and minimal (removing any attribute from the key should prevent it from uniquely identifying records).
   - **Example**: In a "Student" table, both "StudentID" and the combination of "Email" can serve as candidate keys if they uniquely identify students. If "StudentID" is chosen as the primary key, "Email" remains a candidate key.
2. **Composite Key**:
   - **Definition**: A key that consists of two or more attributes used together to uniquely identify a record in a table.
   - **Example**: In a "CourseEnrollment" table, the combination of "StudentID" and "CourseID" could be a composite key to uniquely identify each enrollment.
3. **Superkey**:
   - **Definition**: A set of one or more attributes that, taken collectively, can uniquely identify a record in a table. A superkey may contain extra attributes that are not necessary for uniqueness.
   - **Example**: In a "Student" table, (StudentID) is a superkey, and (StudentID,StudentName) is also a superkey (even though "StudentName" is not needed for uniqueness).
4. **Prime Attribute**:
   - **Definition**: An attribute that is part of a candidate key.
   - **Example**: In a "Student" table with a candidate key of (StudentID,CourseID), both "StudentID" and "CourseID" are prime attributes.
5. **Non-Prime Attribute**:
   - **Definition**: An attribute that is not part of any candidate key.
   - **Example**: In the same "Student" table, if "StudentName" and "CourseName" are not part of the candidate key, they are non-prime attributes.

Functional Dependencies

1. **Functional Dependency (FD)**:
   - **Definition**: A relationship between two attributes, where the value of one attribute (or a group of attributes) determines the value of another attribute.
   - **Notation**: If attribute A functionally determines attribute B, it is denoted as A→B.
   - **Example**: In a "Student" entity, if the "StudentID" uniquely determines the "StudentName," it can be expressed as StudentID→StudentName.
2. **Partial Dependency**:

    - **Definition**: A functional dependency where a non-key attribute is functionally dependent on part of a composite primary key.
    - **Example**: In a table with a composite key (CourseID,StudentID), if "CourseName" is dependent only on "CourseID," it represents a partial dependency: CourseID→CourseName.
3. **Transitive Dependency**:
    - **Definition**: A functional dependency that exists when one attribute indirectly depends on another through a third attribute.
    - **Example**: If A→B and B→C, then there is a transitive dependency: A→C.
4. **Trivial Dependency**:
    - **Definition**: A functional dependency where the dependent attribute is a subset of the determinant.
    - **Example**: A→A is a trivial dependency.

Normal Forms

1. **First Normal Form (1NF)**:
    - **Definition**: A relation is in 1NF if all attributes contain only atomic (indivisible) values, and each record is unique.
    - **Example**: A "Student" table containing multiple phone numbers in a single field violates 1NF; it should be split into separate rows or a separate table.
2. **Second Normal Form (2NF)**:
    - **Definition**: A relation is in 2NF if it is in 1NF and all non-key attributes are fully functionally dependent on the primary key (no partial dependencies).
    - **Example**: If a "Course" table with a composite key (CourseID,StudentID) has "Instructor" as a non-key attribute dependent only on "CourseID," it is not in 2NF.
3. **Third Normal Form (3NF)**:
    - **Definition**: A relation is in 3NF if it is in 2NF and there are no transitive dependencies among non-key attributes.
    - **Example**: If a "Student" table has "Advisor" as an attribute dependent on "Department" (which is dependent on "StudentID"), it violates 3NF.

Term	Definition	Example
Functional Dependency (FD)	A relationship where one attribute determines another.	StudentID → StudentName
Partial Dependency	A dependency where a non-key attribute is dependent on part of a composite key.	CourseID → CourseName
Transitive Dependency	A dependency where one attribute depends on another through a third attribute.	StudentID → Advisor → Department
Trivial Dependency	A dependency where the dependent attribute is a subset of the determinant.	A → A
First Normal Form (1NF)	All attributes have atomic values; each record is unique.	No repeating groups or arrays within a single field.
Second Normal Form (2NF)	In 1NF; all non-key attributes are fully functionally dependent on the primary key.	No partial dependencies for composite keys.
Third Normal Form (3NF)	In 2NF; no transitive dependencies among non-key attributes.	Non-key attributes are not dependent on other non-key attributes.
Prime Attribute	An attribute that is part of a candidate key.	In (StudentID, CourseID), both are prime attributes.
Non-Prime Attribute	An attribute that is not part of any candidate key.	"StudentName" and "CourseName" in the "Student" table.
Candidate Key	A minimal set of attributes that can uniquely identify a record in a table.	"StudentID" and "Email" can both be candidate keys.
Composite Key	A key consisting of two or more attributes that together uniquely identify a record.	(StudentID, CourseID) in a "CourseEnrollment" table.
Superkey	A set of attributes that can uniquely identify a record; may contain unnecessary attributes.	(StudentID) and (StudentID, StudentName) in a "Student" table.