Normalization - Assignment 4

# E4.10 Consider the following relation with sample data.

A white sheet with black text

AI-generated content may be incorrect.

## E4.10a Depict full key functional dependencies, partial functional dependencies (if any), and transitive functional dependencies (if any) in the Surgery Schedule Table.

Partial functional dependencies; Partial functional dependencies occur when a proper subset of the primary key determines another attribute

PatientID PatientName

NurseID NurseName

**Transitive functional dependencies**; Transitive functional dependencies occur when an attribute is determined by another non-key attribute, which is itself determined by the primary key.

(PatientID, NurseID, SurgeryDate) SurgeonID SurgeonName

**Full key functional dependencies**; (These are dependencies where the entire composite primary key (PatientID, NurseID, SurgeryDate) determines another attribute, and no proper subset of the key can determine it alone.)

PatientID, NurseID, SurgeryDate SurgeryType

(PatientID, NurseID, SurgeryDate) NurseRole

(PatientID, NurseID, SurgeryDate) SurgeonID

## E4.10b Show the result of normalizing the Surgery Schedule Table to 2NF.

To normalize the Surgery Schedule Table to 2NF, eliminate partial functional dependencies.

1. Patients Table

This table stores patient information, removing the partial dependency of PatientName on PatientID.

|  |  |
| --- | --- |
| **PatientID** | PatientName |
| 111 | Joe |
| 222 | Pat |
| 333 | Bob |
| 444 | Pat |

Primary Key: PatientID

2. Nurses Table

This table stores nurse information, removing the partial dependency of NurseName on NurseID.

|  |  |
| --- | --- |
| **NurseID** | NurseName |
| N1 | Mike |
| N2 | Sue |
| N3 | Tina |
| N4 | Lee |
| N5 | Sue |
| N6 | Pam |

Primary Key: NurseID

3. SurgerySchedule Table

This table holds the core surgery scheduling information, ensuring all non-key attributes (SurgeonID, SurgeonName, SurgeryType, NurseRole) depend on the entire composite primary key (PatientID, NurseID, SurgeryDate). Removing PatientName and NurseName since they are now in separate tables.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **PatientID** | **NurseID** | **SurgeryDate** | SurgeonID | SurgeonName | SurgeryType | NurseRole |
| 111 | N1 | 1-Jan | AAA | Dr. Adams | Knee Surgery | 1st Assistant |
| 111 | N2 | 1-Jan | AAA | Dr. Adams | Knee Surgery | 1st Assistant |
| 111 | N2 | 2-Jan | AAA | Dr. Adams | Ankle Surgery | 2nd Assistant |
| 111 | N2 | 3-Jan | BBB | Dr. Brown | Elbow Surgery | 1st Assistant |
| 222 | N3 | 1-Jan | CCC | Dr. Crown | Knee Surgery | 1st Assistant |
| 333 | N4 | 1-Jan | AAA | Dr. Adams | Hip Surgery | 1st Assistant |
| 333 | N3 | 1-Jan | AAA | Dr. Adams | Hip Surgery | 2nd Assistant |
| 333 | N5 | 2-Jan | DDD | Dr. Adams | Shoulder Surgery | 1st Assistant |
| 444 | N6 | 1-Jan | BBB | Dr. Brown | Hip Surgery | 1st Assistant |

Primary Key: (PatientID, NurseID, SurgeryDate)

Foreign Keys:

* PatientID references Patients (PatientID)
* NurseID references Nurses (NurseID)
* SurgeonID references Surgeons (SurgeonID)

## E4.10c Show the result of normalizing the Surgery Schedule Table to 3NF.

To normalize the Surgery Schedule Table to 3NF, we need to build on the 2NF normalization and eliminate transitive functional dependencies. A table is in 3NF if it is in 2NF and no non-key attribute is transitively dependent on the primary key (i.e., no non-key attribute depends on another non-key attribute, which in turn depends on the primary key).

1. Patients Table

This table stores patient information, as in 2NF.

|  |  |
| --- | --- |
| **PatientID** | PatientName |
| 111 | Joe |
| 222 | Pat |
| 333 | Bob |
| 444 | Pat |

Primary Key: PatientID

2. Nurses Table

This table stores nurse information, removing the partial dependency of NurseName on NurseID.

|  |  |
| --- | --- |
| **NurseID** | NurseName |
| N1 | Mike |
| N2 | Sue |
| N3 | Tina |
| N4 | Lee |
| N5 | Sue |
| N6 | Pam |

Primary Key: NurseID

3. Surgeons Table

This table stores surgeon information, addressing the transitive dependency (SurgeonID → SurgeonName)

|  |  |
| --- | --- |
| **SurgeonID** | SurgeonName |
| AAA | Dr. Adams |
| BBB | Dr. Brown |
| CCC | Dr. Crown |
| DDD | Dr. Adams |

Primary Key: SurgeonID

4. SurgerySchedule Table

This table holds the core surgery scheduling information, ensuring all non-key attributes depend directly on the primary key (PatientID, NurseID, SurgeryDate) and removing any transitive dependencies. Removing SurgeonName, as it is transitively dependent on SurgeonID, and keep only SurgeonID as a foreign key referencing the Surgeons table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PatientID** | **NurseID** | **SurgeryDate** | SurgeonID | SurgeryType | NurseRole |
| 111 | N1 | 1-Jan | AAA | Knee Surgery | 1st Assistant |
| 111 | N2 | 1-Jan | AAA | Knee Surgery | 1st Assistant |
| 111 | N2 | 2-Jan | AAA | Ankle Surgery | 2nd Assistant |
| 111 | N2 | 3-Jan | BBB | Elbow Surgery | 1st Assistant |
| 222 | N3 | 1-Jan | CCC | Knee Surgery | 1st Assistant |
| 333 | N4 | 1-Jan | AAA | Hip Surgery | 1st Assistant |
| 333 | N3 | 1-Jan | AAA | Hip Surgery | 2nd Assistant |
| 333 | N5 | 2-Jan | DDD | Shoulder Surgery | 1st Assistant |
| 444 | N6 | 1-Jan | BBB | Hip Surgery | 1st Assistant |

Primary Key: (PatientID, NurseID, SurgeryDate)

Foreign Keys:

* PatientID references Patients (PatientID)
* NurseID references Nurses (NurseID)
* SurgeonID references Surgeons (SurgeonID)

The SurgerySchedule table is now in 3NF because all non-key attributes (SurgeonID, SurgeryType, NurseRole) depend directly on the primary key (PatientID, NurseID, SurgeryDate), and there are no transitive dependencies. SurgeonName is stored in the Surgeons table and linked via SurgeonID, ensuring it doesn’t transitively depend on the primary key through SurgeonID.