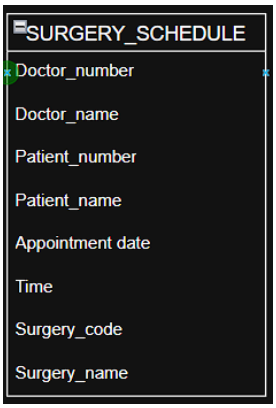
EXERCISE 4: Database Normalization – Doctor & Patient Appointment

Original Table: SURGERY_SCHEDULE

Doctor_ number	Doctor_ name	Patient_ number	Patient_ name	Appointment Date Time	Surgery_ code	Surgery_ name
D1204	Sam Evans	P101	Gil Ramirez	01-10-2025 10:00	S22	Appendicitis
D1204	Sam Evans	P108	John Smith	01-10-2025 12:00	S22	Appendicitis
D5050	Julia Roberts	P106	Paul McCall	01-10-2025 10:00	S10	Cholecystectomy
D5050	Julia Roberts	P106	Paul McCall	02-10-2025 14:00	S10	Cholecystectomy
D3553	Ann Pearson	P108	John Smith	02-10-2025 10:00	S22	Appendicitis
D3553	Ann Pearson	P110	Andres Walker	03-10-2025 18:00	S13	Hip Replacement



STEP 1: First Normal Form (1NF)

1.1.Rule 1NF

- The first normal form (1NF) requires:

    - “Each attribute must contain only a single value".

    - "Each attribute must be atomic (not be a combination of multiple columns)"

1.2.Issue

- The table has repeating data (Doctor, Surgery, Patient repeated many times).

- But no multi-valued attributes — so it’s technically in 1NF, but not well-structured (repetition leads to anomalies).

1.3.Solution

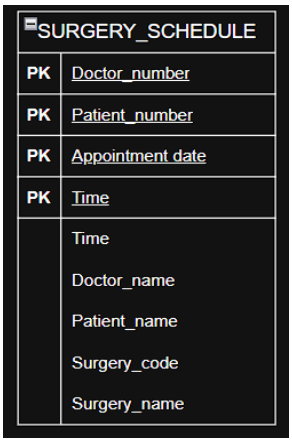
- We identify a unique composite key: {Doctor_number, Patient_number, Appointment_Date, Time} – This uniquely identifies each appointment.

- All values are atomic => so the table is now 1NF.

1.4.PK and FKS

The PK for the SURGERY_SCHEDULE table is a unique composite key: {Doctor_number, Patient_number, Appointment_Date, Time}.

1.5.ERDs



1.6.Example data

PK: Doctor_number	PK: Patient_number	PK: Appointment_Date	PK: Time	Doctor_name	Patient_name	Surgery_code	Surgery_name
D1204	P101	01-10-2025	10:00	Sam Evans	Gil Ramirez	S22	Appendicitis
D1204	P108	01-10-2025	12:00	Sam Evans	John Smith	S22	Appendicitis
D5050	P106	01-10-2025	10:00	Julia Roberts	Paul McCall	S10	Cholecystectomy
D5050	P106	02-10-2025	14:00	Julia Roberts	Paul McCall	S10	Cholecystectomy
D3553	P108	02-10-2025	10:00	Ann Pearson	John Smith	S22	Appendicitis
D3553	P110	03-10-2025	18:00	Ann Pearson	Andres Walker	S13	Hip Replacement

1.7.Reflections

- Table is in 1NF because:
    - No repeating groups.
    - All fields atomic.
- But still redundant – e.g., Sam Evans repeated multiple times for D1204 => violates 2NF.

STEP 2: Second Normal Form (2NF)

2.1 Rule (2NF)

- Must be in 1NF.

- All non-key attributes must depend on the whole primary key (**no partial dependency**).

2.2 Issue

- In 1NF table:

    - Doctor_name depends only on Doctor_number.

    - Patient_name depends only on Patient_number.

    - Surgery_name depends only on Surgery_code.

- These are partial dependencies, so the table is not in 2NF.

2.3 Solution

- Split the table into separate entities

1. DOCTOR (Doctor info)

2. PATIENT (Patient info)

3. SURGERY (Surgery info)

4. APPOINTMENT (Linking Doctor, Patient, Surgery and Date/Time)

2.4 PK and FKs

- The primary key for DOCTOR table is the Doctor_number.

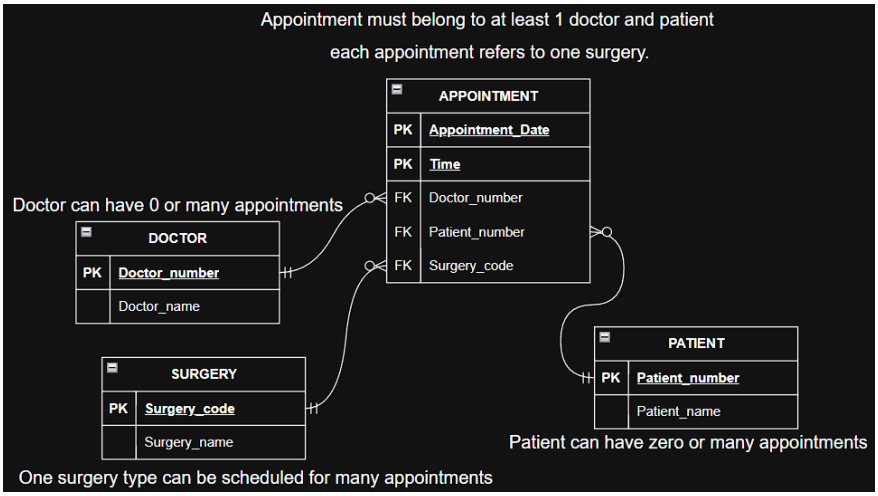
- The primary key for PATIENT table is the Patient_number.

- The primary key for SURGERY table is the Surgery_code.

- The primary key for APPOINTMENT table is the Appointment_Date and Time.

- Doctor_number, Patient_number, Surgery_code are FKs in APPOINTMENT table.

2.5 ERDs



2.6 Sample data

DOCTOR	
PK: Doctor_number	**Doctor_name**
D1204	Sam Evans
D5050	Julia Roberts
D3553	Ann Pearson

PATIENT	
PK: Patient_number	**Patient_name**
P101	Gil Ramirez
P108	John Smith
P106	Paul McCall
P110	Andres Walker

SURGERY	
PK: Surgery_code	**Surgery_name**
S22	Appendicitis
S10	Cholecystectomy
S13	Hip Replacement

APPOINTMENT				
PK: Doctor_number (FK)	PK: Patient_number (FK)	PK: Appointment_Date	PK: Time	FK: Surgery_code
D1204	P101	01-10-2025	10:00	S22
D1204	P108	01-10-2025	12:00	S22
D5050	P106	01-10-2025	10:00	S10
D5050	P106	02-10-2025	14:00	S10
D3553	P108	02-10-2025	10:00	S22
D3553	P110	03-10-2025	18:00	S13

2.7 Reflections

- All non-key attributes now depend on the full PK.

- Redundancy reduced.

- But still possible ***transitive dependency*** (e.g., Surgery_name depends on Surgery_code, which depends on Appointment).

STEP 3: Third Normal Form (3NF)

3.1 Rule (3NF)

- Must be in 2NF

- All non-key attributes must depend only on the primary key (no transitive dependency)

3.2 Issue

- Our design already separates dependencies — there are no transitive dependencies left. Each non-key column depends **only** on its table's PK.

3.3 Solution

The structure from 2NF already satisfies 3NF — this is the final normalized form.

3.4, 3.5, 3.6 same like step 2

3.7 Reflection

- All redundancy eliminated.

- No multi-valued or derived fields.

- All tables in 3NF.

- Primary and foreign keys defined correctly.