**Step 6.1**

**Ward Table**

- Functional Dependencies:

-Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number

- BCNF Normalization:

The table is already in BCNF since the Ward\_Number is the primary key, and all non-key attributes are fully dependent on the Ward\_Number.

**Patient Table**

- Functional Dependencies:

- Patient\_Number -> Patient\_Name, Age, Address, Contact\_Number, Ward\_Number

- Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number

- BCNF Normalization:

The table is not in BCNF because of the transitive dependency Patient\_Number -> Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number. To normalize, we can split the table into two:

- Table `Patients`:

- Patient\_Number -> Patient\_Name, Age, Address, Contact\_Number

- Primary Key: Patient\_Number

- Table `Patient\_Ward`:

- Patient\_Number -> Ward\_Number

- Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number

- Primary Key: (Patient\_Number, Ward\_Number)

**Supplier Table**

- Functional Dependencies:

- Supplier\_Number -> Supplier\_Name, Address, Email, Telephone\_Number, Fax\_Number

- BCNF Normalization:

The table is already in BCNF since the Supplier\_Number is the primary key, and all non-key attributes are fully dependent on the Supplier\_Number.

**Supply Table**

- Functional Dependencies:

- Supply\_Number -> Supply\_Name, Description, Dosage, Method\_Of\_Administration, Quantity\_In\_Stock, Reorder\_Level, Cost\_Per\_Unit, Type\_Of\_Supply, Supplier\_Number

- Supplier\_Number -> Supplier\_Name, Address, Email, Telephone\_Number, Fax\_Number

- BCNF Normalization:

The table is not in BCNF because of the transitive dependency Supply\_Number -> Supplier\_Number -> Supplier\_Name, Address, Email, Telephone\_Number, Fax\_Number. To normalize, we can split the table into two:

- Table `Supply\_Info`:

- Supply\_Number -> Supply\_Name, Description, Dosage, Method\_Of\_Administration, Quantity\_In\_Stock, Reorder\_Level, Cost\_Per\_Unit, Type\_Of\_Supply

- Primary Key: Supply\_Number

- Table `Supply\_Supplier`:

- Supply\_Number -> Supplier\_Number

- Supplier\_Number -> Supplier\_Name, Address, Email, Telephone\_Number, Fax\_Number

- Primary Key: (Supply\_Number, Supplier\_Number)

**Medication Table**

- Functional Dependencies:

- Medication\_Number -> Medication\_Name, Dosage, Method\_Of\_Administration, Start\_Date, End\_Date, Patient\_Number, Drug\_Number

- Patient\_Number -> Patient\_Name, Age, Address, Contact\_Number, Ward\_Number

- Drug\_Number -> Supply\_Name, Description, Dosage, Method\_Of\_Administration, Quantity\_In\_Stock, Reorder\_Level, Cost\_Per\_Unit, Type\_Of\_Supply, Supplier\_Number

- BCNF Normalization:

The table is not in BCNF due to the transitive dependencies Medication\_Number -> Patient\_Number -> Patient\_Name, Age, Address, Contact\_Number, Ward\_Number, and Medication\_Number -> Drug\_Number -> Supply\_Name, Description, Dosage, Method\_Of\_Administration, Quantity\_In\_Stock, Reorder\_Level, Cost\_Per\_Unit, Type\_Of\_Supply, Supplier\_Number. To normalize, we can split the table into three:

- Table `Medication\_Info`:

- Medication\_Number -> Medication\_Name, Dosage, Method\_Of\_Administration, Start\_Date, End\_Date

- Primary Key: Medication\_Number

- Table `Medication\_Patient`:

- Medication\_Number -> Patient\_Number

- Patient\_Number -> Patient\_Name, Age, Address, Contact\_Number, Ward\_Number

- Primary Key: (Medication\_Number, Patient\_Number)

- Table `Medication\_Supply`:

- Medication\_Number -> Drug\_Number

- Drug\_Number -> Supply\_Name, Description, Dosage, Method\_Of\_Administration, Quantity\_In\_Stock, Reorder\_Level, Cost\_Per\_Unit, Type\_Of\_Supply, Supplier\_Number

- Primary Key: (Medication\_Number, Drug\_Number)

**Staff Table**

- Functional Dependencies:

- Staff\_Number -> Staff\_Name, Qualification, Past\_Experience, Ward\_Number

- Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number

- BCNF Normalization:

The table is not in BCNF because of the transitive dependency Staff\_Number -> Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number. To normalize, we can split the table into two:

- Table `Staff\_Info`:

- Staff\_Number -> Staff\_Name, Qualification, Past\_Experience

- Primary Key: Staff\_Number

- Table `Staff\_Ward`:

- Staff\_Number -> Ward\_Number

- Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number

- Primary Key: (Staff\_Number, Ward\_Number)

**Requisition Table**

- Functional Dependencies:

- Requisition\_Number -> Quantity\_Required, Date\_Of\_Order, Delivery\_Date, Charge\_Nurse\_Signature, Staff\_Number, Ward\_Number, Supply\_Number

- Staff\_Number -> Staff\_Name, Qualification, Past\_Experience, Ward\_Number

- Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number

- Supply\_Number -> Supply\_Name, Description, Dosage, Method\_Of\_Administration, Quantity\_In\_Stock, Reorder\_Level, Cost\_Per\_Unit, Type\_Of\_Supply, Supplier\_Number

- BCNF Normalization:

The table is not in BCNF due to multiple transitive dependencies. To normalize, we can split the table into four:

- Table `Requisition\_Info`:

- Requisition\_Number -> Quantity\_Required, Date\_Of\_Order, Delivery\_Date, Charge\_Nurse\_Signature

- Primary Key: Requisition\_Number

- Table `Requisition\_Staff`:

- Requisition\_Number -> Staff\_Number

- Staff\_Number -> Staff\_Name, Qualification, Past\_Experience

- Primary Key: (Requisition\_Number, Staff\_Number)

- Table `Requisition\_Ward`:

- Requisition\_Number -> Ward\_Number

- Ward\_Number -> Ward\_Name, Ward\_Location, Bed\_Count, Extension\_Number

- Primary Key: (Requisition\_Number, Ward\_Number)

- Table `Requisition\_Supply`:

- Requisition\_Number -> Supply\_Number

- Supply\_Number -> Supply\_Name, Description, Dosage, Method\_Of\_Administration, Quantity\_In\_Stock, Reorder\_Level, Cost\_Per\_Unit, Type\_Of\_Supply, Supplier\_Number

- Primary Key: (Requisition\_Number, Supply\_Number)

**Step 6.2**

**Ward Table**

- Ward\_Number (INT): The unique identifier for a ward.

- Ward\_Name (VARCHAR(45)): The name of the ward.

- Ward\_Location (VARCHAR(45)): The location of the ward.

- Bed\_Count (INT): The number of beds available in the ward.

- Extension\_Number (VARCHAR(45)): The extension number for the ward.

Assumptions:

- Each `Ward\_Number` is unique and cannot be NULL.

- The combination of `Ward\_Name` and `Ward\_Location` should be unique.

- The `Bed\_Count` and `Extension\_Number` can be NULL.

**Patient Table**

- Patient\_Number (INT): The unique identifier for a patient.

- Patient\_Name (VARCHAR(45)): The name of the patient.

- Age (INT): The age of the patient.

- Address (VARCHAR(45)): The address of the patient.

- Contact\_Number (VARCHAR(45)): The contact number of the patient.

- Ward\_Number (INT): The ward number where the patient is admitted.

Assumptions:

- Each `Patient\_Number` is unique and cannot be NULL.

- The combination of `Patient\_Number` and `Ward\_Number` should be unique.

- The `Age`, `Address`, and `Contact\_Number` can be NULL.

**Supplier Table**

- Supplier\_Number (INT): The unique identifier for a supplier.

- Supplier\_Name (VARCHAR(45)): The name of the supplier.

- Address (VARCHAR(45)): The address of the supplier.

- Email (VARCHAR(45)): The email address of the supplier.

- Telephone\_Number (VARCHAR(45)): The telephone number of the supplier.

- Fax\_Number (VARCHAR(45)): The fax number of the supplier.

Assumptions:

- Each `Supplier\_Number` is unique and cannot be NULL.

- The `Supplier\_Name` can be NULL.

- The combination of `Supplier\_Name` and `Address` should be unique.

- The `Email`, `Telephone\_Number`, and `Fax\_Number` can be NULL.

**Supply Table**

- Supply\_Number (INT): The unique identifier for a supply.

- Supply\_Name (VARCHAR(45)): The name of the supply.

- Description (VARCHAR(255)): A description of the supply.

- Dosage (VARCHAR(45)): The dosage information for the supply.

- Method\_Of\_Administration (VARCHAR(45)): The method of administration for the supply.

- Quantity\_In\_Stock (INT): The quantity of the supply in stock.

- Reorder\_Level (INT): The level at which the supply needs to be reordered.

- Cost\_Per\_Unit (FLOAT): The cost per unit of the supply.

- Type\_Of\_Supply (VARCHAR(45)): The type or category of the supply.

- Supplier\_Number (INT): The supplier number associated with the supply.

Assumptions:

- Each `Supply\_Number` is unique and cannot be NULL.

- The `Supply\_Name` can be NULL.

- The combination of `Supply\_Name` and `Type\_Of\_Supply` should be unique.

- The `Description`, `Dosage`, `Method\_Of\_Administration`, `Quantity\_In\_Stock`, `Reorder\_Level`, `Cost\_Per\_Unit`, and `Supplier\_Number` can be NULL.

**Medication Table**

- Medication\_Number (INT): The unique identifier for a medication record.

- Medication\_Name (VARCHAR(45)): The name of the medication.

- Dosage (VARCHAR(45)): The dosage information for the medication.

- Method\_Of\_Administration (VARCHAR(45)): The method of administration for the medication.

- Start\_Date (DATE): The date when the medication was started.

- End\_Date (DATE): The date when the medication was ended.

- Patient\_Number (INT): The patient number associated with the medication.

- Drug\_Number (INT): The drug number associated with the medication.

Assumptions:

- Each `Medication\_Number` is unique and cannot be NULL.

- The `Medication\_Name` can be NULL.

- The combination of `Medication\_Name`, `Patient\_Number`, and `Drug\_Number` should be unique.

- The `Dosage`, `Method\_Of\_Administration`, `Start\_Date`, and `End\_Date` can be NULL.

**Staff Table**

- Staff\_Number (INT): The unique identifier for a staff member.

- Staff\_Name (VARCHAR(45)): The name of the staff member.

- Qualification (VARCHAR(45)): The qualification of the staff member.

- Past\_Experience (VARCHAR(45)): The past experience of the staff member.

- Ward\_Number (INT): The ward number where the staff member works.

Assumptions:

- Each `Staff\_Number` is unique and cannot be NULL.

- The `Staff\_Name` can be NULL.

- The combination of `Staff\_Name` and `Ward\_Number` should be unique.

- The `Qualification` and `Past\_Experience` can be NULL.

**Requisition Table**

- Requisition\_Number (INT): The unique identifier for a requisition record.

- Quantity\_Required (INT): The quantity of the supply required in the requisition.

- Date\_Of\_Order (DATE): The date when the requisition was placed.

- Delivery\_Date (DATE): The date when the requisition was delivered.

- Charge\_Nurse\_Signature (VARCHAR(45)): The signature of the charge nurse for the requisition.

- Staff\_Number (INT): The staff number associated with the requisition.

- Ward\_Number (INT): The ward number associated with the requisition.

- Supply\_Number (INT): The supply number associated with the requisition.

Assumptions:

- Each `Requisition\_Number` is unique and cannot be NULL.

- The combination of `Requisition\_Number`, `Staff\_Number`, `Ward\_Number`, and `Supply\_Number` should be unique.

- The `Quantity\_Required`, `Date\_Of\_Order`, `Delivery\_Date`, and `Charge\_Nurse\_Signature` can be NULL.

**Step 6.3**

**Ward Table**

- Ward\_Number (INT): Primary key, NOT NULL.

- Ward\_Name (VARCHAR(45)): NOT NULL.

- Ward\_Location (VARCHAR(45)): NULL.

- Bed\_Count (INT): NULL.

- Extension\_Number (VARCHAR(45)): NULL.

**Patient Table**

- Patient\_Number (INT): Primary key, NOT NULL.

- Patient\_Name (VARCHAR(45)): NULL.

- Age (INT): NULL.

- Address (VARCHAR(45)): NULL.

- Contact\_Number (VARCHAR(45)): NULL.

- Ward\_Number (INT): Foreign key referencing `Ward(Ward\_Number)`, NOT NULL.

**Supplier Table**

- Supplier\_Number (INT): Primary key, NOT NULL.

- Supplier\_Name (VARCHAR(45)): NULL.

- Address (VARCHAR(45)): NULL.

- Email (VARCHAR(45)): NULL.

- Telephone\_Number (VARCHAR(45)): NULL.

- Fax\_Number (VARCHAR(45)): NULL.

**Supply Table**

- Supply\_Number (INT): Primary key, NOT NULL.

- Supply\_Name (VARCHAR(45)): NULL.

- Description (VARCHAR(255)): NULL.

- Dosage (VARCHAR(45)): NULL.

- Method\_Of\_Administration (VARCHAR(45)): NULL.

- Quantity\_In\_Stock (INT): NULL.

- Reorder\_Level (INT): NULL.

- Cost\_Per\_Unit (FLOAT): NULL.

- Type\_Of\_Supply (VARCHAR(45)): NULL.

- Supplier\_Number (INT): Foreign key referencing `Supplier(Supplier\_Number)`, NOT NULL.

**Medication Table**

- Medication\_Number (INT): Primary key, NOT NULL.

- Medication\_Name (VARCHAR(45)): NULL.

- Dosage (VARCHAR(45)): NULL.

- Method\_Of\_Administration (VARCHAR(45)): NULL.

- Start\_Date (DATE): NULL.

- End\_Date (DATE): NULL.

- Patient\_Number (INT): Foreign key referencing `Patient(Patient\_Number)`, NOT NULL.

- Drug\_Number (INT): Foreign key referencing `Supply(Supply\_Number)`, NOT NULL.

**Staff Table**

- Staff\_Number (INT): Primary key, NOT NULL.

- Staff\_Name (VARCHAR(45)): NULL.

- Qualification (VARCHAR(45)): NULL.

- Past\_Experience (VARCHAR(45)): NULL.

- Ward\_Number (INT): Foreign key referencing `Ward(Ward\_Number)`, NOT NULL.

**Requisition Table**

- Requisition\_Number (INT): Primary key, NOT NULL.

- Quantity\_Required (INT): NULL.

- Date\_Of\_Order (DATE): NULL.

- Delivery\_Date (DATE): NULL.

- Charge\_Nurse\_Signature (VARCHAR(45)): NULL.

- Staff\_Number (INT): Foreign key referencing `Staff(Staff\_Number)`, NOT NULL.

- Ward\_Number (INT): Foreign key referencing `Ward(Ward\_Number)`, NOT NULL.

- Supply\_Number (INT): Foreign key referencing `Supply(Supply\_Number)`, NOT NULL.

**Step 6.4**

-- Create the 'green\_valley' database if it doesn't exist (if you haven't created it yet).

CREATE DATABASE IF NOT EXISTS green\_valley;

-- Use the 'green\_valley' database.

USE green\_valley;

-- Table `Ward`

CREATE TABLE IF NOT EXISTS Ward (

Ward\_Number INT NOT NULL,

Ward\_Name VARCHAR(45) NOT NULL,

Ward\_Location VARCHAR(45),

Bed\_Count INT,

Extension\_Number VARCHAR(45),

PRIMARY KEY (Ward\_Number),

UNIQUE INDEX Ward\_Number\_UNIQUE (Ward\_Number) VISIBLE

) ENGINE=InnoDB;

-- Table `Patient`

CREATE TABLE IF NOT EXISTS Patient (

Patient\_Number INT NOT NULL,

Patient\_Name VARCHAR(45),

Age INT,

Address VARCHAR(45),

Contact\_Number VARCHAR(45),

Ward\_Number INT NOT NULL,

PRIMARY KEY (Patient\_Number, Ward\_Number),

UNIQUE INDEX Patien\_Number\_UNIQUE (Patient\_Number) VISIBLE,

INDEX fk\_Patient\_Ward\_idx (Ward\_Number) VISIBLE,

CONSTRAINT fk\_Patient\_Ward

FOREIGN KEY (Ward\_Number)

REFERENCES Ward (Ward\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION

) ENGINE=InnoDB;

-- Table `Supplier`

CREATE TABLE IF NOT EXISTS Supplier (

Supplier\_Number INT NOT NULL,

Supplier\_Name VARCHAR(45),

Address VARCHAR(45),

Email VARCHAR(45),

Telephone\_Number VARCHAR(45),

Fax\_Number VARCHAR(45),

PRIMARY KEY (Supplier\_Number),

UNIQUE INDEX Supplier\_Number\_UNIQUE (Supplier\_Number) VISIBLE

) ENGINE=InnoDB;

-- Table `Supply`

CREATE TABLE IF NOT EXISTS Supply (

Supply\_Number INT NOT NULL,

Supply\_Name VARCHAR(45),

Description VARCHAR(255),

Dosage VARCHAR(45),

Method\_Of\_Administration VARCHAR(45),

Quantity\_In\_Stock INT,

Reorder\_Level INT,

Cost\_Per\_Unit FLOAT,

Type\_Of\_Supply VARCHAR(45),

Supplier\_Number INT NOT NULL,

PRIMARY KEY (Supply\_Number, Supplier\_Number),

UNIQUE INDEX Drug\_Number\_UNIQUE (Supply\_Number) VISIBLE,

INDEX fk\_Supply\_Supplier1\_idx (Supplier\_Number) VISIBLE,

CONSTRAINT fk\_Supply\_Supplier1

FOREIGN KEY (Supplier\_Number)

REFERENCES Supplier (Supplier\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION

) ENGINE=InnoDB;

-- Table `Medication`

CREATE TABLE IF NOT EXISTS Medication (

Medication\_Number INT NOT NULL,

Medication\_Name VARCHAR(45),

Dosage VARCHAR(45),

Method\_Of\_Administration VARCHAR(45),

Start\_Date DATE,

End\_Date DATE,

Patient\_Number INT NOT NULL,

Drug\_Number INT NOT NULL,

PRIMARY KEY (Medication\_Number, Patient\_Number, Drug\_Number),

UNIQUE INDEX Medication\_Number\_UNIQUE (Medication\_Number) VISIBLE,

INDEX fk\_Medication\_Patient1\_idx (Patient\_Number) VISIBLE,

INDEX fk\_Medication\_Pharmaceutical\_Supply1\_idx (Drug\_Number) VISIBLE,

CONSTRAINT fk\_Medication\_Patient1

FOREIGN KEY (Patient\_Number)

REFERENCES Patient (Patient\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT fk\_Medication\_Pharmaceutical\_Supply1

FOREIGN KEY (Drug\_Number)

REFERENCES Supply (Supply\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION

) ENGINE=InnoDB;

-- Table `Staff`

CREATE TABLE IF NOT EXISTS Staff (

Staff\_Number INT NOT NULL,

Staff\_Name VARCHAR(45),

Qualification VARCHAR(45),

Past\_Experience VARCHAR(45),

Ward\_Number INT NOT NULL,

PRIMARY KEY (Staff\_Number, Ward\_Number),

UNIQUE INDEX Staff\_Number\_UNIQUE (Staff\_Number) VISIBLE,

INDEX fk\_Staff\_Ward1\_idx (Ward\_Number) VISIBLE,

CONSTRAINT fk\_Staff\_Ward1

FOREIGN KEY (Ward\_Number)

REFERENCES Ward (Ward\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION

) ENGINE=InnoDB;

-- Table `Requisition`

CREATE TABLE IF NOT EXISTS Requisition (

Requisition\_Number INT NOT NULL,

Quantity\_Required INT,

Date\_Of\_Order DATE,

Delivery\_Date DATE,

Charge\_Nurse\_Signature VARCHAR(45),

Staff\_Number INT NOT NULL,

Ward\_Number INT NOT NULL,

Supply\_Number INT NOT NULL,

PRIMARY KEY (Requisition\_Number, Staff\_Number, Ward\_Number, Supply\_Number),

UNIQUE INDEX Requisition\_Number\_UNIQUE (Requisition\_Number) VISIBLE,

INDEX fk\_Requisition\_Staff1\_idx (Staff\_Number) VISIBLE,

INDEX fk\_Requisition\_Ward1\_idx (Ward\_Number) VISIBLE,

INDEX fk\_Requisition\_Supply1\_idx (Supply\_Number) VISIBLE,

CONSTRAINT fk\_Requisition\_Staff1

FOREIGN KEY (Staff\_Number)

REFERENCES Staff (Staff\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT fk\_Requisition\_Ward1

FOREIGN KEY (Ward\_Number)

REFERENCES Ward (Ward\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT fk\_Requisition\_Supply1

FOREIGN KEY (Supply\_Number)

REFERENCES Supply (Supply\_Number)

ON DELETE NO ACTION

ON UPDATE NO ACTION

) ENGINE=InnoDB;

**Step 6.5**

Indexes for foreign keys and other constraints added above in Step 6.4

**Step 6.6**

INSERT INTO Ward (`Ward\_Number`, `Ward\_Name`, `Ward\_Location`, `Bed\_Count`, `Extenstion\_Number`) VALUES

(1, 'Pediatrics', 'East Wing', 20, 'Ext-101'),

(2, 'Orthopedics', 'West Wing', 15, 'Ext-102'),

(3, 'Cardiology', 'North Wing', 30, 'Ext-103'),

(4, 'Neurology', 'South Wing', 25, 'Ext-104'),

(5, 'Oncology', 'West Wing', 18, 'Ext-105'),

(6, 'Gynecology', 'East Wing', 12, 'Ext-106'),

(7, 'Urology', 'North Wing', 10, 'Ext-107'),

(8, 'ENT', 'South Wing', 8, 'Ext-108'),

(9, 'Dermatology', 'West Wing', 14, 'Ext-109'),

(10, 'Psychiatry', 'East Wing', 16, 'Ext-110'),

(11, 'General Surgery', 'North Wing', 22, 'Ext-111'),

(12, 'Internal Medicine', 'South Wing', 25, 'Ext-112'),

(13, 'Nephrology', 'West Wing', 9, 'Ext-113'),

(14, 'Ophthalmology', 'East Wing', 10, 'Ext-114'),

(15, 'Pulmonology', 'North Wing', 12, 'Ext-115'),

(16, 'Rheumatology', 'South Wing', 7, 'Ext-116'),

(17, 'Endocrinology', 'West Wing', 11, 'Ext-117'),

(18, 'Hematology', 'East Wing', 8, 'Ext-118'),

(19, 'Gastroenterology', 'North Wing', 15, 'Ext-119'),

(20, 'Dentistry', 'South Wing', 6, 'Ext-120');

-------------------------

INSERT INTO Patient (`Patient\_Number`, `Patient\_Name`, `Age`, `Address`, `Contact\_Number`, `Ward\_Number`) VALUES

(1, 'John Doe', 35, '123 Main St', '555-123-4567', 3),

(2, 'Jane Smith', 45, '456 Oak Ave', '555-987-6543', 1),

(3, 'Michael Johnson', 28, '789 Elm Rd', '555-555-5555', 2),

(4, 'Emily Williams', 62, '321 Maple Ln', '555-222-3333', 4),

(5, 'William Brown', 50, '555 Pine St', '555-444-7890', 5),

(6, 'Olivia Lee', 19, '777 Birch Ave', '555-777-8888', 6),

(7, 'James Davis', 72, '888 Cedar Rd', '555-999-1111', 9),

(8, 'Sophia Garcia', 55, '999 Spruce Ln', '555-333-2222', 8),

(9, 'Liam Rodriguez', 42, '111 Willow St', '555-222-1111', 11),

(10, 'Emma Martinez', 30, '222 Palm Ave', '555-444-2222', 12),

(11, 'Noah Hernandez', 25, '333 Rose Rd', '555-666-5555', 14),

(12, 'Ava Lopez', 68, '444 Ivy Ave', '555-777-4444', 15),

(13, 'Sophia Brown', 37, '555 Vine St', '555-999-3333', 16),

(14, 'Michael Smith', 41, '666 Oak Ave', '555-111-7777', 18),

(15, 'Olivia Johnson', 29, '777 Elm Rd', '555-555-5555', 20),

(16, 'William Williams', 52, '888 Maple Ln', '555-222-8888', 19),

(17, 'Emily Davis', 48, '999 Pine St', '555-444-1111', 17),

(18, 'James Martinez', 33, '111 Birch Ave', '555-777-3333', 13),

(19, 'Sophia Rodriguez', 39, '222 Cedar Rd', '555-999-6666', 10),

(20, 'Ava Garcia', 61, '333 Spruce Ln', '555-333-4444', 7);

------------------------------

INSERT INTO Supplier (`Supplier\_Number`, `Supplier\_Name`, `Address`, `Email`, `Telephone\_Number`, `Fax\_Number`) VALUES

(1, 'ABC Medical Supplies', '123 Main St', 'abc@example.com', '555-123-4567', '555-987-6543'),

(2, 'XYZ Pharma', '456 Oak Ave', 'xyz@example.com', '555-222-3333', '555-444-7890'),

(3, 'MediCorp', '789 Elm Rd', 'medicorp@example.com', '555-555-5555', '555-777-8888'),

(4, 'Global Healthcare', '321 Maple Ln', 'globalhealth@example.com', '555-999-1111', '555-333-2222'),

(5, 'HealthPlus', '555 Pine St', 'healthplus@example.com', '555-222-1111', '555-444-2222'),

(6, 'PharmaLink', '777 Birch Ave', 'pharmalink@example.com', '555-666-5555', '555-777-4444'),

(7, 'MedEquip', '888 Cedar Rd', 'medequip@example.com', '555-999-3333', '555-111-7777'),

(8, 'Sunrise Medical', '999 Spruce Ln', 'sunrise@example.com', '555-555-5555', '555-222-8888'),

(9, 'Best Care Pharmaceuticals', '111 Willow St', 'bestcare@example.com', '555-444-1111', '555-444-1111'),

(10, 'MediSupply', '222 Palm Ave', 'medisupply@example.com', '555-777-7777', '555-999-9999'),

(11, 'HealthTech', '333 Rose Rd', 'healthtech@example.com', '555-333-3333', '555-111-1111'),

(12, 'PharmaWorld', '444 Ivy Ave', 'pharmaworld@example.com', '555-222-2222', '555-888-8888'),

(13, 'Vital Healthcare', '555 Vine St', 'vitalhealth@example.com', '555-666-6666', '555-333-3333'),

(14, 'MediLife', '666 Oak Ave', 'medilife@example.com', '555-999-9999', '555-666-6666'),

(15, 'Prime Medical Supplies', '777 Elm Rd', 'prime@example.com', '555-444-4444', '555-222-2222'),

(16, 'Allied Pharma', '888 Maple Ln', 'alliedpharma@example.com', '555-777-7777', '555-444-4444'),

(17, 'HealthFirst', '999 Pine St', 'healthfirst@example.com', '555-111-1111', '555-777-7777'),

(18, 'MediAid', '111 Birch Ave', 'mediaid@example.com', '555-888-8888', '555-999-9999'),

(19, 'PharmaDirect', '222 Cedar Rd', 'pharmadirect@example.com', '555-333-3333', '555-555-5555'),

(20, 'United Medical', '333 Spruce Ln', 'unitedmedical@example.com', '555-666-6666', '555-333-3333');

----------------------------

INSERT INTO Supply (`Supply\_Number`, `Supply\_Name`, `Description`, `Dosage`, `Method\_Of\_Administration`, `Quantity\_In\_Stock`, `Reorder\_Level`, `Cost\_Per\_Unit`, `Type\_Of\_Supply`, `Supplier\_Number`) VALUES

(1, 'Paracetamol', 'Pain reliever and fever reducer', '500mg', 'Oral', 1000, 100, 2.5, 'Tablet', 1),

(2, 'Ibuprofen', 'Nonsteroidal anti-inflammatory drug', '400mg', 'Oral', 800, 150, 3.0, 'Tablet', 2),

(3, 'Aspirin', 'Pain reliever and antiplatelet', '325mg', 'Oral', 1200, 200, 1.8, 'Tablet', 1),

(4, 'Amoxicillin', 'Antibiotic', '250mg', 'Oral', 500, 50, 4.2, 'Capsule', 3),

(5, 'Ciprofloxacin', 'Broad-spectrum antibiotic', '500mg', 'Oral', 300, 30, 6.7, 'Tablet', 2),

(6, 'Lisinopril', 'ACE inhibitor for hypertension', '10mg', 'Oral', 400, 40, 2.9, 'Tablet', 4),

(7, 'Metformin', 'Antidiabetic medication', '500mg', 'Oral', 700, 70, 1.5, 'Tablet', 5),

(8, 'Loratadine', 'Antihistamine', '10mg', 'Oral', 600, 60, 3.8, 'Tablet', 6),

(9, 'Fluoxetine', 'Selective serotonin reuptake inhibitor', '20mg', 'Oral', 250, 25, 5.1, 'Capsule', 7),

(10, 'Simvastatin', 'HMG-CoA reductase inhibitor', '40mg', 'Oral', 300, 30, 4.9, 'Tablet', 8),

(11, 'Albuterol', 'Bronchodilator', '2mg', 'Inhalation', 200, 20, 10.5, 'Inhaler', 9),

(12, 'Insulin Lispro', 'Rapid-acting insulin', '100 units/mL', 'Subcutaneous', 150, 15, 12.2, 'Vial', 10),

(13, 'Omeprazole', 'Proton pump inhibitor', '20mg', 'Oral', 400, 40, 3.7, 'Capsule', 12),

(14, 'Atorvastatin', 'HMG-CoA reductase inhibitor', '20mg', 'Oral', 500, 50, 6.0, 'Tablet', 11),

(15, 'Prednisone', 'Corticosteroid', '5mg', 'Oral', 300, 30, 2.4, 'Tablet', 14),

(16, 'Escitalopram', 'Selective serotonin reuptake inhibitor', '10mg', 'Oral', 200, 20, 4.8, 'Tablet', 13),

(17, 'Warfarin', 'Anticoagulant', '5mg', 'Oral', 600, 60, 1.9, 'Tablet', 17),

(18, 'Metoprolol', 'Beta-blocker for hypertension', '25mg', 'Oral', 450, 45, 2.3, 'Tablet', 15),

(19, 'Losartan', 'Angiotensin II receptor blocker', '50mg', 'Oral', 350, 35, 3.5, 'Tablet', 18),

(20, 'Hydrochlorothiazide', 'Diuretic', '12.5mg', 'Oral', 500, 50, 2.7, 'Tablet', 19);

----------------------------

INSERT INTO Medication (`Medication\_Number`, `Medication\_Name`, `Dosage`, `Method\_Of\_Administration`, `Start\_Date`, `End\_Date`, `Patient\_Number`, `Drug\_Number`) VALUES

(1, 'Pain Relief', '500mg', 'Oral', '2023-07-01', '2023-07-10', 1, 1),

(2, 'Fever Reducer', '400mg', 'Oral', '2023-07-03', '2023-07-08', 2, 2),

(3, 'Antibiotic', '250mg', 'Oral', '2023-07-05', '2023-07-15', 3, 4),

(4, 'Allergy Relief', '10mg', 'Oral', '2023-07-10', '2023-07-20', 4, 8),

(5, 'Anti-inflammatory', '20mg', 'Oral', '2023-07-12', '2023-07-25', 5, 7),

(6, 'Asthma Control', '2mg', 'Inhalation', '2023-07-15', '2023-07-30', 6, 11),

(7, 'Antidiabetic', '500mg', 'Oral', '2023-07-20', '2023-08-05', 7, 12),

(8, 'Antidepressant', '20mg', 'Oral', '2023-07-22', '2023-08-15', 8, 16),

(9, 'Blood Pressure Control', '10mg', 'Oral', '2023-07-25', '2023-08-10', 9, 6),

(10, 'Cholesterol Lowering', '40mg', 'Oral', '2023-07-28', '2023-08-18', 10, 10),

(11, 'Antiplatelet', '325mg', 'Oral', '2023-08-01', '2023-08-20', 11, 3),

(12, 'Anticoagulant', '5mg', 'Oral', '2023-08-05', '2023-08-25', 12, 17),

(13, 'Bronchodilator', '2mg', 'Inhalation', '2023-08-10', '2023-08-28', 13, 11),

(14, 'Insulin Therapy', '100 units/mL', 'Subcutaneous', '2023-08-15', '2023-09-05', 14, 12),

(15, 'Heart Medication', '25mg', 'Oral', '2023-08-20', '2023-09-08', 15, 18),

(16, 'Diuretic', '12.5mg', 'Oral', '2023-08-25', '2023-09-10', 16, 20),

(17, 'Pain Relief', '500mg', 'Oral', '2023-08-28', '2023-09-15', 17, 1),

(18, 'Antibiotic', '500mg', 'Oral', '2023-09-01', '2023-09-18', 18, 5),

(19, 'Anti-inflammatory', '20mg', 'Oral', '2023-09-05', '2023-09-20', 19, 7),

(20, 'Antidepressant', '10mg', 'Oral', '2023-09-10', '2023-09-25', 20, 16);

-----------------------------------

INSERT INTO Staff (`Staff\_Number`, `Staff\_Name`, `Qualification`, `Past\_Experience`, `Ward\_Number`) VALUES

(1, 'John Smith', 'Registered Nurse', '5 years at ABC Hospital', 1),

(2, 'Emily Johnson', 'Licensed Practical Nurse', '3 years at XYZ Clinic', 2),

(3, 'Michael Williams', 'Medical Doctor', '10 years at City Hospital', 3),

(4, 'Sophia Brown', 'Certified Nursing Assistant', '2 years at Community Care Center', 4),

(5, 'James Davis', 'Medical Technologist', '7 years at Regional Medical Center', 5),

(6, 'Olivia Garcia', 'Physician Assistant', '4 years at HealthPlus Clinic', 6),

(7, 'William Martinez', 'Pharmacist', '6 years at MediCorp Pharmacy', 7),

(8, 'Emma Rodriguez', 'Physical Therapist', '9 years at Sports Rehab Center', 8),

(9, 'Liam Hernandez', 'Occupational Therapist', '5 years at Senior Care Facility', 9),

(10, 'Ava Lopez', 'Speech-Language Pathologist', '3 years at Speech Center', 10),

(11, 'Noah Rivera', 'Medical Doctor', '8 years at County Hospital', 11),

(12, 'Isabella Turner', 'Registered Nurse', '6 years at Medical Center', 12),

(13, 'Mason Mitchell', 'Licensed Practical Nurse', '4 years at Home Healthcare Agency', 13),

(14, 'Sophia Phillips', 'Certified Nursing Assistant', '2 years at Elderly Care Home', 14),

(15, 'Alexander Campbell', 'Medical Technologist', '5 years at Diagnostic Lab', 15),

(16, 'Mia Scott', 'Physician Assistant', '3 years at Family Practice Clinic', 16),

(17, 'Ethan Adams', 'Pharmacist', '7 years at HealthMart Pharmacy', 17),

(18, 'Abigail Hernandez', 'Physical Therapist', '4 years at Sports Medicine Center', 18),

(19, 'Jayden Wood', 'Occupational Therapist', '6 years at Rehabilitation Hospital', 19),

(20, 'Ella Foster', 'Speech-Language Pathologist', '5 years at Language Development Center', 20);

-------------------------

INSERT INTO `green\_valley`.`Requisition` (`Requisition\_Number`, `Quantity\_Required`, `Date\_Of\_Order`, `Delivery\_Date`, `Charge\_Nurse\_Signature`, `Staff\_Number`, `Ward\_Number`, `Supply\_Number`) VALUES

(1, 50, '2023-07-01', '2023-07-05', 'CN-Sign-001', 1, 1, 1),

(2, 30, '2023-07-02', '2023-07-06', 'CN-Sign-002', 2, 2, 2),

(3, 40, '2023-07-03', '2023-07-07', 'CN-Sign-003', 3, 3, 3),

(4, 25, '2023-07-04', '2023-07-08', 'CN-Sign-004', 4, 4, 4),

(5, 60, '2023-07-05', '2023-07-09', 'CN-Sign-005', 5, 5, 5),

(6, 35, '2023-07-06', '2023-07-10', 'CN-Sign-006', 6, 6, 6),

(7, 20, '2023-07-07', '2023-07-11', 'CN-Sign-007', 7, 7, 7),

(8, 45, '2023-07-08', '2023-07-12', 'CN-Sign-008', 8, 8, 8),

(9, 55, '2023-07-09', '2023-07-13', 'CN-Sign-009', 9, 9, 9),

(10, 70, '2023-07-10', '2023-07-14', 'CN-Sign-010', 10, 10, 10),

(11, 28, '2023-07-11', '2023-07-15', 'CN-Sign-011', 11, 11, 11),

(12, 38, '2023-07-12', '2023-07-16', 'CN-Sign-012', 12, 12, 12),

(13, 42, '2023-07-13', '2023-07-17', 'CN-Sign-013', 13, 13, 13),

(14, 32, '2023-07-14', '2023-07-18', 'CN-Sign-014', 14, 14, 14),

(15, 48, '2023-07-15', '2023-07-19', 'CN-Sign-015', 15, 15, 15),

(16, 52, '2023-07-16', '2023-07-20', 'CN-Sign-016', 16, 16, 16),

(17, 18, '2023-07-17', '2023-07-21', 'CN-Sign-017', 17, 17, 17),

(18, 63, '2023-07-18', '2023-07-22', 'CN-Sign-018', 18, 18, 18),

(19, 24, '2023-07-19', '2023-07-23', 'CN-Sign-019', 19, 19, 19),

(20, 36, '2023-07-20', '2023-07-24', 'CN-Sign-020', 20, 20, 20);

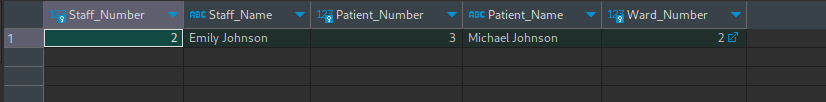
**Step 6.7**

* **Request: Find the name of the staff who is in ward 2 and taking care of patient with ID 3**

**SQL**

SELECT S.Staff\_Number, S.Staff\_Name, P.Patient\_Number, P.Patient\_Name, P.Ward\_Number FROM Staff AS S INNER JOIN Patient AS P ON S.Ward\_Number = P.Ward\_Number WHERE P.Patient\_Number = 3 AND S.Ward\_Number = 2;

**Screenshot**

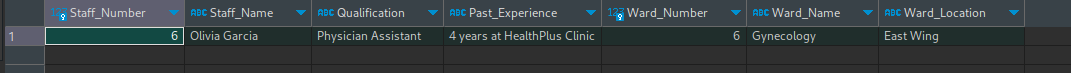


* **Request: Find the staffs who are assigned to Ward 6**

**SQL**

SELECT S.Staff\_Number, S.Staff\_Name, S.Qualification, S.Past\_Experience, W.Ward\_Number, W.Ward\_Name, W.Ward\_Location FROM Staff AS S INNER JOIN Ward AS W ON S.Ward\_Number = W.Ward\_Number WHERE S.Ward\_Number = 6;

**Screenshot**

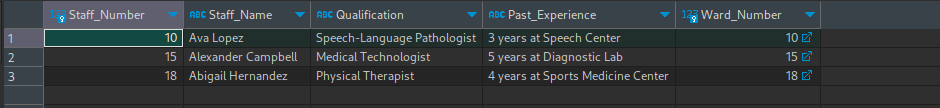


* **Request: Find all staff whose name starts with “A”**

**SQL**

SELECT \* FROM Staff AS S WHERE S.Staff\_Name LIKE 'A%';

**Screenshot**

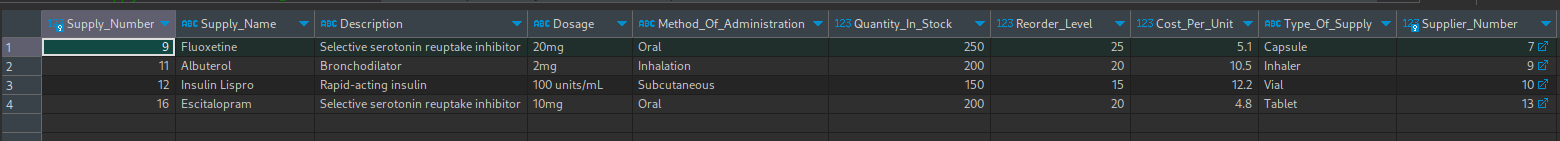


* **Request: Find all supplies that need to be reordered if the reorder level is less than 30**

**SQL**

SELECT \* FROM Supply AS S WHERE S.Reorder\_Level < 30;

**Screenshot**



* **Request: Find all patients who have been given the “Pain Relief”**

**SQL**

SELECT P.Patient\_Number, P.Patient\_Name, P.Address, M.Dosage, M.Method\_Of\_Administration, M.Start\_Date, M.End\_Date FROM Patient AS P INNER JOIN Medication AS M ON P.Patient\_Number = M.Patient\_Number WHERE M.Medication\_Name = "Pain Relief";

**Screenshot**

