**Project 1: Medicine inventory**

**Supervised by: Dr. Muhammad Elsayeh**

|  |  |
| --- | --- |
| Mohammad Omar | 231002343 |
| Omnia Abdelnaser Ezzat | 211000476 |
| Mariam ashraf bassaly | 231002049 |
| Mohammad Ahmed Atea | 231002184 |

**Team Members**

**SQL Schema**

**A screenshot of a computer

AI-generated content may be incorrect.**

**SQL Code**

CREATE DATABASE medicine\_inventryy;

USE medicine\_inventryy;

CREATE TABLE Doctor (

doctor\_id INT PRIMARY KEY,

name VARCHAR(100),

qualification VARCHAR(100),

specialization VARCHAR(100)

);

CREATE TABLE Disease (

disease\_id INT PRIMARY KEY,

name VARCHAR(100),

type ENUM('Infectious','Deficiency','Genetic Hereditary','Non-genetic Hereditary')

);

CREATE TABLE Medicine (

medicine\_id INT PRIMARY KEY,

name VARCHAR(100),

manufacture\_date DATE,

expiry\_date DATE,

price DECIMAL(10,2) CHECK (price >= 0),

dosage VARCHAR(50)

);

Many-to-many: Medicine <-> Disease

CREATE TABLE Medicine\_Disease (

medicine\_id INT,

disease\_id INT,

PRIMARY KEY (medicine\_id, disease\_id),

FOREIGN KEY (medicine\_id) REFERENCES Medicine(medicine\_id) ON DELETE CASCADE,

FOREIGN KEY (disease\_id) REFERENCES Disease(disease\_id) ON DELETE CASCADE

);

CREATE TABLE Prescription (

prescription\_id INT PRIMARY KEY,

doctor\_id INT,

issue\_date DATE,

patient\_name VARCHAR(100),

FOREIGN KEY (doctor\_id) REFERENCES Doctor(doctor\_id) ON DELETE SET NULL

);

Many-to-many: Prescription <-> Medicine with quantity

CREATE TABLE Prescription\_Medicine (

prescription\_id INT,

medicine\_id INT,

quantity INT CHECK (quantity > 0),

PRIMARY KEY (prescription\_id, medicine\_id),

FOREIGN KEY (prescription\_id) REFERENCES Prescription(prescription\_id) ON DELETE CASCADE,

FOREIGN KEY (medicine\_id) REFERENCES Medicine(medicine\_id) ON DELETE RESTRICT

);

Bill: one-per-prescription

CREATE TABLE Bill (

bill\_id INT PRIMARY KEY AUTO\_INCREMENT,

prescription\_id INT UNIQUE,

tax DECIMAL(10,2) DEFAULT 0.00,

discount DECIMAL(10,2) DEFAULT 0.00,

total DECIMAL(12,2) DEFAULT 0.00,

FOREIGN KEY (prescription\_id) REFERENCES Prescription(prescription\_id) ON DELETE CASCADE

);

INSERT INTO Doctor VALUES

(1, 'Dr. Rowaida', 'MBBS, MD', 'Heart Disease'),

(2, 'Dr. Mohab', 'MBBS', 'Dermatology'),

(3, 'Dr. Abdullah', 'MBBS, MS', 'Neurology'),

(4, 'Dr. Mariam', 'MBBS, MD', 'Heart Disease'),

(5, 'Dr. Sameh', 'MBBS', 'General Physician'),

(6, 'Dr. Sara', 'MBBS', 'Pediatrics'),

(7, 'Dr. Ahmed', 'MBBS, MS', 'Orthopedics'),

(8, 'Dr. Mohammad', 'MBBS', 'ENT'),

(9, 'Dr. Omar', 'MBBS, MD', 'Gastroenterology'),

(10, 'Dr. Nada', 'MBBS', 'Heart Disease');

INSERT INTO Disease VALUES

(1, 'Anemia', 'Deficiency'),

(2, 'Scurvy', 'Deficiency'),

(3, 'Diabetes', 'Genetic Hereditary'),

(4, 'Hypertension', 'Non-genetic Hereditary'),

(5, 'Asthma', 'Non-genetic Hereditary'),

(6, 'COVID-19', 'Infectious'),

(7, 'Influenza', 'Infectious'),

(8, 'Rickets', 'Deficiency'),

(9, 'Cancer', 'Genetic Hereditary'),

(10, 'Tuberculosis', 'Infectious');

INSERT INTO Medicine VALUES

(1, 'MedT', '2023-01-01', '2025-01-01', 50.00, '10mg'),

(2, 'MedC', '2023-03-15', '2024-09-15', 30.00, '5mg'),

(3, 'MedR', '2023-05-01', '2024-05-01', 60.00, '20mg'),

(4, 'MedI', '2022-12-01', '2024-12-01', 40.00, '15mg'),

(5, 'MedCC', '2023-06-10', '2025-06-10', 25.00, '5mg'),

(6, 'MedA', '2023-08-05', '2026-08-05', 75.00, '50mg'),

(7, 'MedH', '2022-11-20', '2024-11-20', 20.00, '2mg'),

(8, 'MedD', '2023-09-09', '2025-09-09', 35.00, '7mg'),

(9, 'MedS', '2023-04-04', '2025-04-04', 90.00, '25mg'),

(10,'MedAA', '2023-07-07', '2026-07-07', 55.00, '30mg');

INSERT INTO Medicine\_Disease VALUES

(1, 10),(2, 3),(3, 4),(4, 9),(5, 7),(6, 5),(7, 2),(8, 6),(9, 8),(10, 1);

INSERT INTO Prescription (prescription\_id, doctor\_id, issue\_date, patient\_name) VALUES

(1,1,'2023-01-01','Patient A'),

(2,2,'2023-02-01','Patient B'),

(3,3,'2023-03-01','Patient C'),

(4,4,'2023-04-01','Patient D'),

(5,4,'2023-05-01','Patient E'),

(6,2,'2023-06-01','Patient F'),

(7,1,'2023-07-01','Patient G'),

(8,3,'2023-08-01','Patient H'),

(9,2,'2023-09-01','Patient I'),

(10,3,'2023-10-01','Patient J');

INSERT INTO Prescription\_Medicine VALUES

(1,6,2),(1,1,1),(2,5,1),(2,2,1),(3,3,2),(4,4,1),

(5,6,1),(5,2,3),(6,2,2),(6,8,1),(7,9,1),(8,5,2),

(8,4,1),(9,4,1),(9,8,2),(10,6,1),(10,3,2);

-- create initial Bill rows (one per prescription)

INSERT INTO Bill (prescription\_id)

SELECT prescription\_id FROM Prescription;

Procedure to recalc bill for a prescription id

DELIMITER $$

CREATE PROCEDURE recalc\_bill\_proc (IN pid INT)

BEGIN

DECLARE subtotal DECIMAL(12,2) DEFAULT 0.00;

SELECT IFNULL(SUM(pm.quantity \* m.price),0.00) INTO subtotal

FROM Prescription\_Medicine pm

JOIN Medicine m ON pm.medicine\_id = m.medicine\_id

WHERE pm.prescription\_id = pid;

UPDATE Bill

SET tax = ROUND(subtotal \* 0.05,2), -- example: 5% tax

discount = 0.00,

total = ROUND(subtotal + ROUND(subtotal \* 0.05,2) - 0.00,2)

WHERE prescription\_id = pid;

END$$

Trigger: after inserting a prescription -> ensure bill row and calc

CREATE TRIGGER trg\_after\_insert\_prescription

AFTER INSERT ON Prescription

FOR EACH ROW

BEGIN

INSERT INTO Bill (prescription\_id)

SELECT NEW.prescription\_id

WHERE NOT EXISTS (SELECT 1 FROM Bill b WHERE b.prescription\_id = NEW.prescription\_id);

CALL recalc\_bill\_proc(NEW.prescription\_id);

END$$

Trigger: after inserting a prescription\_medicine -> recalc bill

CREATE TRIGGER trg\_after\_insert\_prescription\_medicine

AFTER INSERT ON Prescription\_Medicine

FOR EACH ROW

BEGIN

CALL recalc\_bill\_proc(NEW.prescription\_id);

END$$

Trigger: after updating quantity or medicine -> recalc bill

CREATE TRIGGER trg\_after\_update\_prescription\_medicine

AFTER UPDATE ON Prescription\_Medicine

FOR EACH ROW

BEGIN

CALL recalc\_bill\_proc(NEW.prescription\_id);

END$$

Trigger: after deleting a medicine from a prescription -> recalc bill

CREATE TRIGGER trg\_after\_delete\_prescription\_medicine

AFTER DELETE ON Prescription\_Medicine

FOR EACH ROW

BEGIN

CALL recalc\_bill\_proc(OLD.prescription\_id);

END$$

DELIMITER ;

initial recalculation for all existing prescriptions (optional)

CALL recalc\_bill\_proc(1); CALL recalc\_bill\_proc(2); CALL recalc\_bill\_proc(3);

CALL recalc\_bill\_proc(4); CALL recalc\_bill\_proc(5); CALL recalc\_bill\_proc(6);

CALL recalc\_bill\_proc(7); CALL recalc\_bill\_proc(8); CALL recalc\_bill\_proc(9);

CALL recalc\_bill\_proc(10);

-- 1) List the name of doctors whose specialty is heart disease

SELECT name FROM Doctor WHERE specialization = 'Heart Disease';

2) List the deficiency diseases

SELECT name FROM Disease WHERE type = 'Deficiency';

3) List the most sold medicine in 2023

SELECT m.name, SUM(pm.quantity) AS total\_sold

FROM Prescription\_Medicine pm

JOIN Prescription p ON pm.prescription\_id = p.prescription\_id

JOIN Medicine m ON pm.medicine\_id = m.medicine\_id

WHERE YEAR(p.issue\_date) = 2023

GROUP BY m.name

ORDER BY total\_sold DESC

LIMIT 1;

**ER Diagram and Relational Data**



