SET B

Doctor – Hospital Database

Consider the following database

Doctor (d_no int, d_namevarchar(30), specialization varchar(35), charges int)

Hospital (h_no int, h_namevarchar(20), city varchar(10))

Doctor and Hospital are related with many to one relationship.

Create the above database in PostGreSQL and insert sufficient records.

a. Write a trigger before insert/update on Doctor table. Raise exception if charges are <0.

b. Write a trigger that restricts insertion of charges value greater than 500.

```
-- Create the Hospital Table
CREATE TABLE Hospital (
    h_no INT PRIMARY KEY,
    h_name VARCHAR(20),
    city VARCHAR(10)
);


-- Create the Doctor Table
CREATE TABLE Doctor (
    d_no INT PRIMARY KEY,
    d_name VARCHAR(30),
    specialization VARCHAR(35),
    charges INT,
    h_no INT,
    FOREIGN KEY (h_no) REFERENCES Hospital (h_no)
);
```

-- Insert data into the Hospital table

INSERT INTO Hospital VALUES (1, 'CityHospital', 'NewYork');

INSERT INTO Hospital VALUES(2, 'HealthCareCenter', 'LosAngeles');

INSERT INTO Hospital VALUES(3, 'MedClinic', 'Chicago');

-- Insert data into the Doctor table

INSERT INTO Doctor VALUES (101, 'Dr. Smith', 'Cardiology', 300, 1);

INSERT INTO Doctor VALUES(102, 'Dr. Brown', 'Orthopedics', 400, 2);

INSERT INTO Doctor VALUES(103, 'Dr. Taylor', 'Neurology', 350, 3);

INSERT INTO Doctor VALUES(104, 'Dr. White', 'Pediatrics', 450, 1);

-- Q1. Write a trigger before insert/update on Doctor table. Raise exception if charges are <0.

CREATE OR REPLACE FUNCTION validate_doctor_charges()

RETURNS TRIGGER AS '

BEGIN

   IF NEW.charges < 0 THEN

     RAISE EXCEPTION 'Charges for doctor % cannot be negative.', NEW.d_name;

   END IF;

   RETURN NEW;

END;

' LANGUAGE plpgsql;

-- Create a trigger to execute the function before an insert or update on Doctor table

CREATE TRIGGER charges_check_trigger

BEFORE INSERT OR UPDATE ON Doctor

FOR EACH ROW

EXECUTE FUNCTION validate_doctor_charges();

--Test the trigger

INSERT INTO Doctor VALUES (105, 'Dr. Green', 'Dentistry', -100, 2);

-- Q2.Write a trigger that restricts insertion of charges value greater than 500.

CREATE OR REPLACE FUNCTION restrict_high_charges() RETURNS TRIGGER AS '

BEGIN

   IF NEW.charges > 500 THEN

     RAISE EXCEPTION "Charges for doctor % cannot exceed 500.", NEW.d_name;

   END IF;

   RETURN NEW;

END;

' LANGUAGE plpgsql;

-- Create a trigger to execute the function before an insert on the Doctor table

CREATE TRIGGER restrict_charges_trigger

BEFORE INSERT ON Doctor

FOR EACH ROW

EXECUTE FUNCTION restrict_high_charges();

--Test the trigger

INSERT INTO Doctor VALUES (106, 'Dr. Blue', 'Surgery', 600, 3);