**MIDTERM PROJECT**

**HEALTHCARE INDUSTRY DATABASE**

## TEAM:

Bharath Anand (02044023) – 2

Pradyoth Singenahalli Prabhu (02071847) - 42

Varun W R (02070206) - 49

# Part-1: Planned Database-driven Application requirements

Developing a comprehensive database for the US healthcare sector requires careful consideration of several factors to ensure the accuracy, privacy, and security of the data.

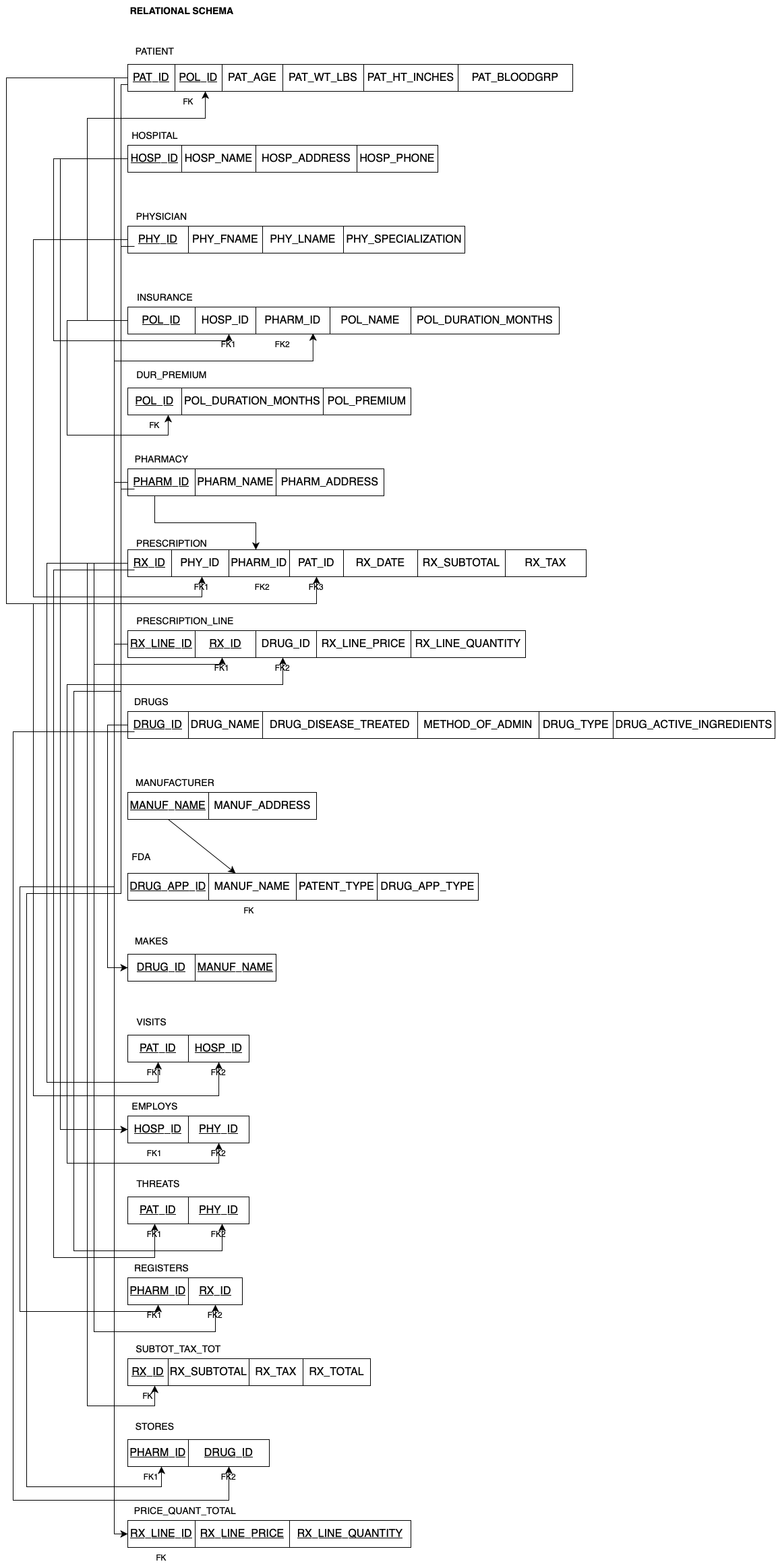
The database will need to store and maintain various critical information related to patient health, such as treatment records, prescribed medications, physician visits, and other important medical data. It will also need to store confidential patient demographics, including personal identifying information such as name, date of birth, address, and phone number.

To ensure data consistency and integrity, the database will need to incorporate various mechanisms such as validation checks, constraints, and triggers to prevent incorrect data entry or manipulation. Additionally, to ensure secure storage and maintenance of the data, the database will need to incorporate appropriate security measures, such as access control, encryption, and backup and recovery procedures.

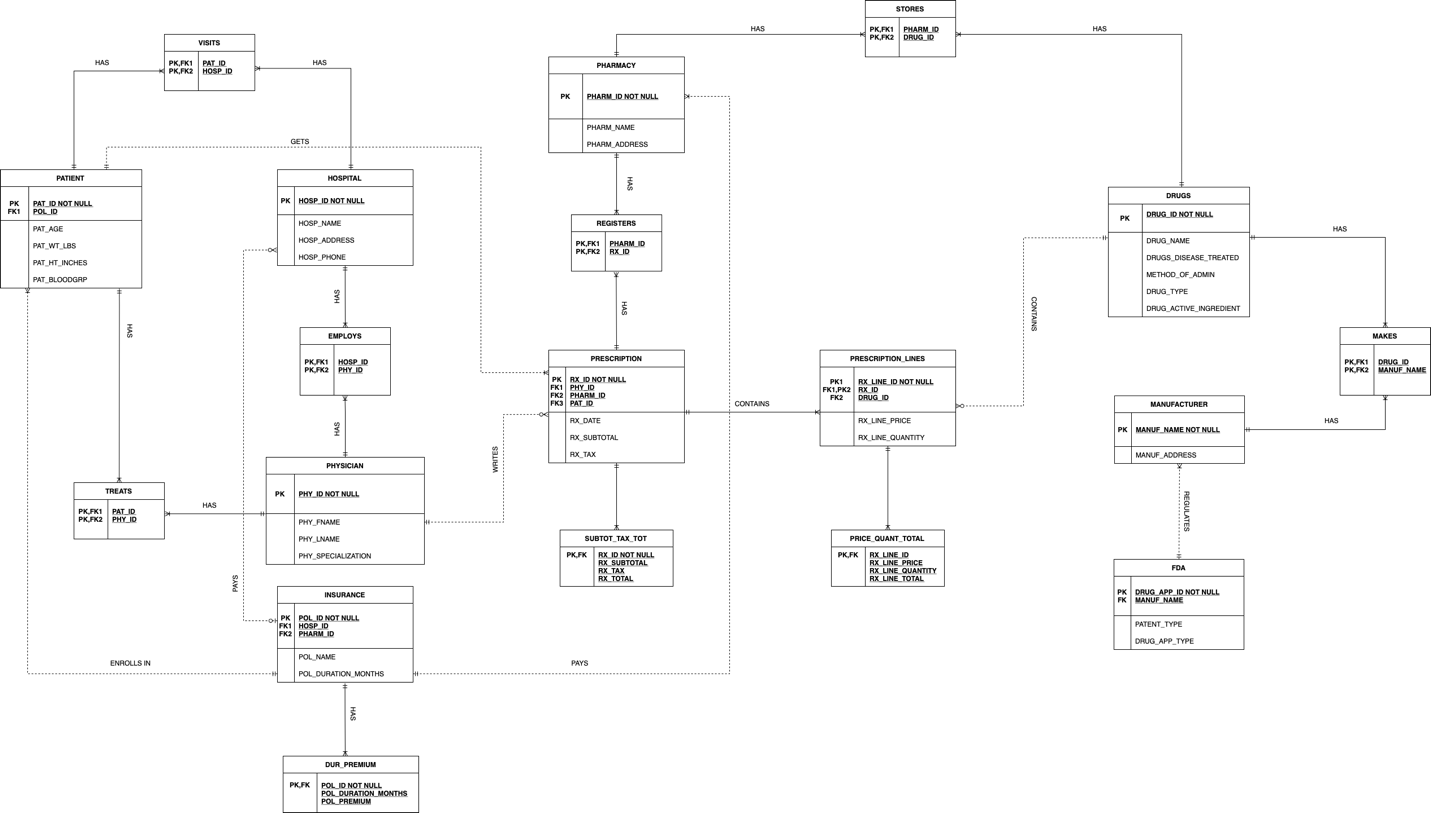
Overall, developing a comprehensive database for the US healthcare sector is a complex task that requires a thorough understanding of the healthcare industry, its data requirements, and the regulatory environment surrounding it.

# Part-2: Entity – Relationship Diagram

The Entity-Relationship diagram for a healthcare database system visually represents the different entities and relationships involved in the system. It shows the core entity, which is the PATIENT, and the other entities connected to it. Each entity is connected to other entities through relationships that show the nature of the connection between them. The diagram also includes attributes for each entity that provide additional details about the data stored within them. Overall, the Entity-Relationship diagram provides a clear and comprehensive view of the healthcare database system, making it easier to understand the relationships between the different entities and the data stored within them. It serves as a useful reference for database administrators, developers, and users to design, maintain, and use the database effectively.



# Part-3: Corresponding Relational Schema



# Part-4: Corresponding DB Implementation

In this project we have created total of 19 tables. Below we have given brief about each of the table. We have written each create statements on our own and not used forward engineering.

**Hospital table:** The hospital table will contain information about each hospital or medical center. The hospital\_id column will be the primary key, and each hospital will be identified by a unique ID.

**Drugs table:** The drugs table will contain information about each drug or medication. The drug\_id column will be the primary key, and each drug will be identified by a unique ID.

**Pharmacy table:** The pharmacy table will contain information about each pharmacy or drugstore. The pharmacy\_id column will be the primary key, and each pharmacy will be identified by a unique ID.

**Insurance table:** This table will store information about the different insurance providers in the healthcare industry. Insurance ID tells us about a unique identifier for each insurance provider.Insurance Name gives information about the name of the insurance provider.

**Patient table:** This table will store information about patients in the healthcare industry. Patient ID is a unique identifier for each patient and insurance ID is a foreign key linking to the insurance provider for the patient.

**Physician table:** This table will store information about physicians in the healthcare industry. Physician ID is a unique identifier for each physician.

**Manufacturers table:** This table will store information about the different drug manufacturers in the healthcare industry. Manufacturer ID is a unique identifier for each manufacturer.

**FDA table:** This table will store information about the different drugs approved by the Food and Drug Administration (FDA). Drug ID is a unique identifier for each drug and Drug Name is the name of the drug.

**Prescription table:** This table will store information about prescriptions given to patients by physicians. Prescription ID is a unique identifier for each prescription and Patient ID is a foreign key linking to the patient the prescription was given to.

**Prescription Lines Table:** This table will store information about the different drug in each lines. RX\_Line is the unique and RX\_ID and drug\_ID.

**Stores Table:** This table will store information about pharmacies or other healthcare providers that dispense medications.

**Visits Table:** This table will store information about patient visits to healthcare and hospital ID.

**Treats Table:** This table will store information about the treatments prescribed during patient visits.

**Registers Table:** This table will store information about the registration of patients with healthcare providers.

**Employees Table:** This table will store information about the employees working in the healthcare industry, including physicians.

**Drug\_Premium Table:** This table will store information about premium drugs, including policy ID, policy premium.

**Price\_Quant\_Total Table:** This table will store information about the total price and quantity of a drug.

**Makes Table:** This table will store information about the drugs manufactured by different manufacturers.

Below screenshots shows creation of tables in MySQL Workbench.

**CREATE SCHEMA** HEALTHCARE;  
**USE** HEALTHCARE;

**CREATE TABLE** HOSPITAL(  
 HOSP\_ID **INT PRIMARY KEY auto\_increment**,  
 HOSP\_NAME **VARCHAR**(20),  
 HOSP\_ADDRESS **VARCHAR**(50),  
 HOSP\_PHONE **BIGINT**(13)  
);  
  
**CREATE TABLE** DRUGS(  
 DRUG\_ID **INT PRIMARY KEY auto\_increment**,  
 DRUG\_NAME **VARCHAR**(100),  
 DRUGS\_DISEASE\_TREATED **VARCHAR**(100),  
 METHOD\_OF\_ADMIN **VARCHAR**(100),  
 DRUG\_TYPE **VARCHAR**(100),  
 DRUG\_ACTIVE\_INGREDIENT **VARCHAR**(100)  
);  
  
**CREATE TABLE** PHARMACY(  
 PHARM\_ID **INT PRIMARY KEY auto\_increment**,  
 PHARM\_NAME **VARCHAR**(100),  
 PHARM\_ADDRESS **VARCHAR**(100)  
);  
  
**CREATE TABLE** INSURANCE(  
 POL\_ID **INT PRIMARY KEY auto\_increment**,  
 HOSP\_ID **INT**,  
 PHARM\_ID **INT**,  
 POL\_NAME **VARCHAR**(100),  
 POL\_DURATION\_MONTHS **INT**,  
 **FOREIGN KEY** (HOSP\_ID) **REFERENCES** HOSPITAL(HOSP\_ID),  
 **FOREIGN KEY** (PHARM\_ID) **REFERENCES** PHARMACY(PHARM\_ID)  
);  
  
**CREATE TABLE** PATIENT(  
 PAT\_ID **INT PRIMARY KEY auto\_increment**,  
 POL\_ID **INT**,  
 PAT\_AGE **INT**,  
 PAT\_WT\_LBS **FLOAT**,  
 PAT\_HT\_INCHES **FLOAT**,  
 PAT\_BLOODGRP **VARCHAR**(4),  
 **FOREIGN KEY** (POL\_ID) **REFERENCES** INSURANCE(POL\_ID)  
);  
  
**CREATE TABLE** PHYSICIAN(  
 PHY\_ID **INT PRIMARY KEY auto\_increment**,  
 PHY\_FNAME **VARCHAR**(50),  
 PHY\_LNAME **VARCHAR**(50),  
 PHY\_SPECIALIZATION **VARCHAR**(100)  
);  
  
**CREATE TABLE** MANUFACTURER(  
 MANUF\_NAME **VARCHAR**(100) **PRIMARY KEY**,  
 MANUF\_ADDRESS **VARCHAR**(100)  
);  
  
**CREATE TABLE** FDA(  
 DRUG\_APP\_ID **INT PRIMARY KEY auto\_increment**,  
 MANUF\_NAME **VARCHAR**(100),  
 PATENT\_TYPE **VARCHAR**(100),  
 DRUG\_APP\_TYPE **VARCHAR**(100),  
 **FOREIGN KEY** (MANUF\_NAME) **REFERENCES** MANUFACTURER(MANUF\_NAME)  
);  
  
**CREATE TABLE** PRESCRIPTION(  
 RX\_ID **INT PRIMARY KEY auto\_increment**,  
 PHY\_ID **INT**,  
 PHARM\_ID **INT**,  
 PAT\_ID **INT**,  
 RX\_DATE **DATE**,  
 RX\_SUBTOTAL **FLOAT**,  
 RX\_TAX **FLOAT**,  
 **FOREIGN KEY** (PHY\_ID) **REFERENCES** PHYSICIAN(PHY\_ID),  
 **FOREIGN KEY** (PHARM\_ID) **REFERENCES** PHARMACY(PHARM\_ID),  
 **FOREIGN KEY** (PAT\_ID) **REFERENCES** PATIENT(PAT\_ID)  
);  
  
**CREATE TABLE** PRESCRIPTION\_LINES(  
 RX\_LINE\_ID **INT PRIMARY KEY auto\_increment**,  
 RX\_ID **INT**,  
 DRUG\_ID **INT**,  
 RX\_LINE\_PRICE **FLOAT**,  
 RX\_LINE\_QUANTITY **INT**,  
 **FOREIGN KEY** (DRUG\_ID) **REFERENCES** DRUGS(DRUG\_ID),  
 **FOREIGN KEY** (RX\_ID) **REFERENCES** PRESCRIPTION(RX\_ID)  
);  
  
**CREATE TABLE** STORES(  
 PHARM\_ID **INT**,  
 DRUG\_ID **INT**,  
 **PRIMARY KEY**(PHARM\_ID, DRUG\_ID),  
 **FOREIGN KEY** (DRUG\_ID) **REFERENCES** DRUGS(DRUG\_ID),  
 **FOREIGN KEY** (PHARM\_ID) **REFERENCES** PHARMACY(PHARM\_ID)  
);  
  
**CREATE TABLE** VISITS(  
 PAT\_ID **INT**,  
 HOSP\_ID **INT**,  
 **PRIMARY KEY**(PAT\_ID, HOSP\_ID),  
 **FOREIGN KEY** (PAT\_ID) **REFERENCES** PATIENT(PAT\_ID),  
 **FOREIGN KEY** (HOSP\_ID) **REFERENCES** HOSPITAL(HOSP\_ID)  
);  
  
**CREATE TABLE** TREATS(  
 PAT\_ID **INT**,  
 PHY\_ID **INT**,  
 **PRIMARY KEY**(PAT\_ID, PHY\_ID),  
 **FOREIGN KEY** (PAT\_ID) **REFERENCES** PATIENT(PAT\_ID),  
 **FOREIGN KEY** (PHY\_ID) **REFERENCES** PHYSICIAN(PHY\_ID)  
);  
  
**CREATE TABLE** EMPLOYS(  
 HOSP\_ID **INT**,  
 PHY\_ID **INT**,  
 **PRIMARY KEY**(HOSP\_ID, PHY\_ID),  
 **FOREIGN KEY** (HOSP\_ID) **REFERENCES** HOSPITAL(HOSP\_ID),  
 **FOREIGN KEY** (PHY\_ID) **REFERENCES** PHYSICIAN(PHY\_ID)  
);  
  
**CREATE TABLE** DUR\_PREMIUM(  
 POL\_ID **INT PRIMARY KEY**,  
 POL\_DURATION\_MONTHS **INT**,  
 POL\_PREMIUM **FLOAT**,  
 **FOREIGN KEY** (POL\_ID) **REFERENCES** INSURANCE(POL\_ID)  
);  
  
**CREATE TABLE** REGISTERS(  
 PHARM\_ID **INT**,  
 RX\_ID **INT**,  
 **PRIMARY KEY**(RX\_ID, PHARM\_ID),  
 **FOREIGN KEY** (RX\_ID) **REFERENCES** PRESCRIPTION(RX\_ID),  
 **FOREIGN KEY** (PHARM\_ID) **REFERENCES** PHARMACY(PHARM\_ID)  
);  
  
**CREATE TABLE** SUBTOT\_TAX\_TOT(  
 RX\_ID **INT PRIMARY KEY**,  
 RX\_SUBTOTAL **FLOAT**,  
 RX\_TAX **FLOAT**,  
 RX\_TOTAL **FLOAT**,  
 **FOREIGN KEY** (RX\_ID) **REFERENCES** PRESCRIPTION(RX\_ID)  
);  
  
**CREATE TABLE** PRICE\_QUANT\_TOTAL(  
 RX\_LINE\_ID **INT PRIMARY KEY**,  
 RX\_LINE\_PRICE **FLOAT**,  
 RX\_LINE\_QUANTITY **INT**,  
 RX\_LINE\_TOTAL **FLOAT**,  
 **FOREIGN KEY** (RX\_LINE\_ID) **REFERENCES** PRESCRIPTION\_LINES(RX\_LINE\_ID)  
);  
  
**CREATE TABLE** MAKES(  
 DRUG\_ID **INT**,  
 MANUF\_NAME **VARCHAR**(100),  
 **PRIMARY KEY**(DRUG\_ID, MANUF\_NAME),  
 **FOREIGN KEY** (DRUG\_ID) **REFERENCES** DRUGS(DRUG\_ID),  
 **FOREIGN KEY** (MANUF\_NAME) **REFERENCES** MANUFACTURER(MANUF\_NAME)  
);

# Part-5: Initial Population of DB

In order to populate a database with data, insert statements are used to add new information to existing tables.

The process of inserting data is essential to building a functional database, as it allows for the storage and retrieval of information in a structured format.

**INSERT INTO** HOSPITAL(HOSP\_NAME, HOSP\_ADDRESS, HOSP\_PHONE) **VALUES**("Hospital A","New Bedford", *FLOOR*( *RAND*() \* 10000000000));  
**INSERT INTO** HOSPITAL(HOSP\_NAME, HOSP\_ADDRESS, HOSP\_PHONE) **VALUES**("Hospital B","Boston", *FLOOR*( *RAND*() \* 10000000000));  
**INSERT INTO** HOSPITAL(HOSP\_NAME, HOSP\_ADDRESS, HOSP\_PHONE) **VALUES**("Hospital C","Fair Haven", *FLOOR*( *RAND*() \* 10000000000));  
**INSERT INTO** HOSPITAL(HOSP\_NAME, HOSP\_ADDRESS, HOSP\_PHONE) **VALUES**("Hospital D","Fall River", *FLOOR*( *RAND*() \* 10000000000));  
**INSERT INTO** HOSPITAL(HOSP\_NAME, HOSP\_ADDRESS, HOSP\_PHONE) **VALUES**("Hospital E","Providence", *FLOOR*( *RAND*() \* 10000000000));  
  
**SELECT** \* **FROM** HOSPITAL;  
  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Ventolin", "Asthma", "Orally inhaled", "Branded", "Salbutamol");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Lisinopril", "High Blood Pressure", "Orally ingested", "Generic", "Prinivil");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Proair", "Asthma", "Orally inhaled", "Branded", "Albuterol Sulfate");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Azithromycin", "Bacterial Infections", "Orally ingested", "Generic", "Azithromycin Dihydrate");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Omeprazole", "Acid Reflux", "Orally ingested", "Generic", "Omeprazole Magnesium");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Losartan Potassium", "High Blood Pressure", "Orally ingested", "Generic", "Hydrochlorothiazide");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Sertraline", "Depression", "Orally ingested", "Generic", "Sertraline Hydrochloride");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Amoxicillin", "Bacterial Infections", "Injection", "Generic", "Amoxicillin Trihydrate");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Gabapentin", "Seizures", "Orally ingested", "Generic", "Gabapentin");  
**INSERT INTO** DRUGS(DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT) **VALUES**("Basaglar", "Diabetes", "Injection", "Branded", "Insulin Glargine");  
  
**SELECT** \* **FROM** DRUGS;  
  
**INSERT INTO** PHARMACY (PHARM\_NAME, PHARM\_ADDRESS) **VALUES**("CVS Health", "New Bedford");  
**INSERT INTO** PHARMACY (PHARM\_NAME, PHARM\_ADDRESS) **VALUES**("Walgreens Boots Alliance", "Dartmouth") ;  
**INSERT INTO** PHARMACY (PHARM\_NAME, PHARM\_ADDRESS) **VALUES** ("Cigna", "Boston") ;  
**INSERT INTO** PHARMACY (PHARM\_NAME, PHARM\_ADDRESS) **VALUES** ("UnitedHealth Group", "Fair Haven") ;  
**INSERT INTO** PHARMACY (PHARM\_NAME, PHARM\_ADDRESS) **VALUES** ("Walmart", "Providence") ;  
**INSERT INTO** PHARMACY (PHARM\_NAME, PHARM\_ADDRESS) **VALUES** ("Kroger", "New Bedford") ;  
**INSERT INTO** PHARMACY (PHARM\_NAME, PHARM\_ADDRESS) **VALUES**("Rite Aid", "Fall River");  
  
**SELECT** \* **FROM** PHARMACY;  
  
**INSERT INTO** INSURANCE(HOSP\_ID, PHARM\_ID, POL\_NAME, POL\_DURATION\_MONTHS) **VALUES**(1, 7, "POS", 3);  
**INSERT INTO** INSURANCE(HOSP\_ID, PHARM\_ID, POL\_NAME, POL\_DURATION\_MONTHS) **VALUES**(2, 6, "PPO", 5);  
**INSERT INTO** INSURANCE(HOSP\_ID, PHARM\_ID, POL\_NAME, POL\_DURATION\_MONTHS) **VALUES**(3, 5, "HMO", 7);  
**INSERT INTO** INSURANCE(HOSP\_ID, PHARM\_ID, POL\_NAME, POL\_DURATION\_MONTHS) **VALUES**(4, 4, "EPO", 12);  
  
**SELECT** \* **FROM** INSURANCE;  
  
**INSERT INTO** DUR\_PREMIUM **VALUES** (1, 3, 1500.00);  
**INSERT INTO** DUR\_PREMIUM **VALUES** (2, 5, 1200.00);  
**INSERT INTO** DUR\_PREMIUM **VALUES** (3, 7, 800.00);  
**INSERT INTO** DUR\_PREMIUM **VALUES** (4, 12, 500.00);  
  
**SELECT** \* **FROM** DUR\_PREMIUM;  
  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(2, 35, 189, 68, "B+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(2, 28, 168, 70, "O+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 60, 171, 72, "AB+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 73, 140, 75, "O+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(4, 18, 162, 69, "AB-");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 94, 185, 70, "B+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 46, 201, 66, "B+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 58, 193, 63, "O-");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 83, 213, 70, "B+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(3, 26, 169, 71, "B-");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(2, 39, 188, 75, "B+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 50, 200, 68, "A+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 59, 150, 67, "B+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(4, 20, 196, 71, "O+");  
**INSERT INTO** PATIENT(POL\_ID, PAT\_AGE, PAT\_WT\_LBS, PAT\_HT\_INCHES, PAT\_BLOODGRP) **VALUES**(1, 57, 250, 73, "B+");  
  
**SELECT** \* **FROM** PATIENT;  
  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Jim", "Halpert", "Pediatrician");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Michael", "Scott", "Orthopedic");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Dwight", "Schrute", "Neurologist");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Mose", "Schrute", "Dermatologist");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Creed", "Bratton", "Psychiatrist");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Pam", "Beesly", "OB/GYN");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Stanley", "Hudson", "Cardiologist");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Jon", "Snow", "Radiologist");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Thomas", "Shelby", "Oncologist");  
**INSERT INTO** PHYSICIAN(PHY\_FNAME, PHY\_LNAME, PHY\_SPECIALIZATION) **VALUES** ("Jesse", "Pinkman", "Anesthesiologist");  
  
**SELECT** \* **FROM** PHYSICIAN;  
  
**INSERT INTO** MANUFACTURER **VALUES** ("GSK", "New Bedford");  
**INSERT INTO** MANUFACTURER **VALUES** ("TEVA", "Boston");  
**INSERT INTO** MANUFACTURER **VALUES** ("Eli Lilly and Company", "Fairhaven");  
**INSERT INTO** MANUFACTURER **VALUES** ("Lupin", "Dartmouth");  
**INSERT INTO** MANUFACTURER **VALUES** ("Zydus", "Fall River");  
  
**SELECT** \* **FROM** MANUFACTURER;  
  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("GSK", "AIP", "NDA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("GSK", "FP", "BLA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("TEVA", "AIP", "NDA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("Lupin", "FP", "BLA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("TEVA", "FP", "NDA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("Eli Lilly and Company", "AIP", "BLA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("Zydus", "FP", "BLA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("Lupin", "FP", "NDA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("Zydus", "AIP", "BLA");  
**INSERT INTO** FDA(MANUF\_NAME, PATENT\_TYPE, DRUG\_APP\_TYPE) **VALUES**("Eli Lilly and Company", "FP", "NDA");  
  
**SELECT** \* **FROM** FDA;  
  
**INSERT INTO** PRESCRIPTION(PHY\_ID, PHARM\_ID, PAT\_ID, RX\_DATE, RX\_TAX) **VALUES**(*FLOOR*(*RAND*()\*9+1), *FLOOR*(*RAND*()\*6+1), *FLOOR*(*RAND*()\*14+1), *curdate*(), 6.25);  
**INSERT INTO** PRESCRIPTION(PHY\_ID, PHARM\_ID, PAT\_ID, RX\_DATE, RX\_TAX) **VALUES**(*FLOOR*(*RAND*()\*9+1), *FLOOR*(*RAND*()\*6+1), *FLOOR*(*RAND*()\*14+1), *curdate*(), 6.25);  
**INSERT INTO** PRESCRIPTION(PHY\_ID, PHARM\_ID, PAT\_ID, RX\_DATE, RX\_TAX) **VALUES**(*FLOOR*(*RAND*()\*9+1), *FLOOR*(*RAND*()\*6+1), *FLOOR*(*RAND*()\*14+1), *curdate*(), 6.25);  
**INSERT INTO** PRESCRIPTION(PHY\_ID, PHARM\_ID, PAT\_ID, RX\_DATE, RX\_TAX) **VALUES**(*FLOOR*(*RAND*()\*9+1), *FLOOR*(*RAND*()\*6+1), *FLOOR*(*RAND*()\*14+1), *curdate*(), 6.25);  
**INSERT INTO** PRESCRIPTION(PHY\_ID, PHARM\_ID, PAT\_ID, RX\_DATE, RX\_TAX) **VALUES**(*FLOOR*(*RAND*()\*9+1), *FLOOR*(*RAND*()\*6+1), *FLOOR*(*RAND*()\*14+1), *curdate*(), 6.25);  
  
**SELECT** \* **FROM** PRESCRIPTION;  
  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(1, 10, 70.0, 30);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(1, 9, 63.0, 60);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(2, 8, 56.0, 30);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(2, 7, 48.99, 90);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(3, 6, 42.0, 30);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(3, 5, 34.99, 60);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(4, 4, 27.5, 90);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(4, 3, 20.99, 30);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(5, 2, 14.0, 30);  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(5, 1, 6.99, 60);  
  
**SELECT** \* **FROM** PRESCRIPTION\_LINES;  
  
**INSERT INTO** MAKES **VALUES** (1, "Eli Lilly and Company");  
**INSERT INTO** MAKES **VALUES** (2, "Lupin");  
**INSERT INTO** MAKES **VALUES** (3, "GSK");  
**INSERT INTO** MAKES **VALUES** (4,"TEVA");  
**INSERT INTO** MAKES **VALUES** (5, "GSK");  
**INSERT INTO** MAKES **VALUES** (6, "Zydus");  
**INSERT INTO** MAKES **VALUES** (7, "Eli Lilly and Company");  
**INSERT INTO** MAKES **VALUES** (8, "Zydus");  
**INSERT INTO** MAKES **VALUES** (9, "Lupin");  
**INSERT INTO** MAKES **VALUES** (10, "TEVA");  
  
**SELECT** \* **FROM** MAKES;  
  
**INSERT INTO** STORES **VALUES** (7, 1);  
**INSERT INTO** STORES **VALUES** (6, 2);  
**INSERT INTO** STORES **VALUES** (5, 3);  
**INSERT INTO** STORES **VALUES** (4, 4);  
**INSERT INTO** STORES **VALUES** (3, 5);  
**INSERT INTO** STORES **VALUES** (2, 6);  
**INSERT INTO** STORES **VALUES** (1, 7);  
**INSERT INTO** STORES **VALUES** (5, 8);  
**INSERT INTO** STORES **VALUES** (7, 9);  
**INSERT INTO** STORES **VALUES** (3, 10);  
  
**SELECT** \* **FROM** STORES;  
  
**INSERT INTO** VISITS **VALUES** (1, 1);  
**INSERT INTO** VISITS **VALUES** (2, 2);  
**INSERT INTO** VISITS **VALUES** (3, 3);  
**INSERT INTO** VISITS **VALUES** (4, 4);  
**INSERT INTO** VISITS **VALUES** (5, 5);  
**INSERT INTO** VISITS **VALUES** (6, 1);  
**INSERT INTO** VISITS **VALUES** (7, 2);  
**INSERT INTO** VISITS **VALUES** (8, 3);  
**INSERT INTO** VISITS **VALUES** (9, 4);  
**INSERT INTO** VISITS **VALUES** (10, 5);  
**INSERT INTO** VISITS **VALUES** (11, 1);  
**INSERT INTO** VISITS **VALUES** (12, 2);  
**INSERT INTO** VISITS **VALUES** (13, 3);  
**INSERT INTO** VISITS **VALUES** (14, 4);  
**INSERT INTO** VISITS **VALUES** (15, 5);  
  
**SELECT** \* **FROM** VISITS;  
  
**INSERT INTO** TREATS **VALUES** (1, 1);  
**INSERT INTO** TREATS **VALUES** (2, 2);  
**INSERT INTO** TREATS **VALUES** (3, 3);  
**INSERT INTO** TREATS **VALUES** (4, 4);  
**INSERT INTO** TREATS **VALUES** (5, 5);  
**INSERT INTO** TREATS **VALUES** (6, 10);  
**INSERT INTO** TREATS **VALUES** (7, 9);  
**INSERT INTO** TREATS **VALUES** (8, 8);  
**INSERT INTO** TREATS **VALUES** (9, 7);  
**INSERT INTO** TREATS **VALUES** (10, 6);  
**INSERT INTO** TREATS **VALUES** (11, 1);  
**INSERT INTO** TREATS **VALUES** (12, 2);  
**INSERT INTO** TREATS **VALUES** (13, 3);  
**INSERT INTO** TREATS **VALUES** (14, 4);  
**INSERT INTO** TREATS **VALUES** (15, 5);  
  
**SELECT** \* **FROM** TREATS;  
  
**INSERT INTO** REGISTERS **VALUES** (5, 1);  
**INSERT INTO** REGISTERS **VALUES** (7, 2);  
**INSERT INTO** REGISTERS **VALUES** (2, 3);  
**INSERT INTO** REGISTERS **VALUES** (3, 4);  
**INSERT INTO** REGISTERS **VALUES** (1, 5);  
  
**SELECT** \* **FROM** REGISTERS;  
  
**INSERT INTO** EMPLOYS **VALUES** (1, 1);  
**INSERT INTO** EMPLOYS **VALUES** (1, 2);  
**INSERT INTO** EMPLOYS **VALUES** (2, 3);  
**INSERT INTO** EMPLOYS **VALUES** (2, 4);  
**INSERT INTO** EMPLOYS **VALUES** (3, 5);  
**INSERT INTO** EMPLOYS **VALUES** (3, 6);  
**INSERT INTO** EMPLOYS **VALUES** (4, 7);  
**INSERT INTO** EMPLOYS **VALUES** (4, 8);  
**INSERT INTO** EMPLOYS **VALUES** (5, 9);  
**INSERT INTO** EMPLOYS **VALUES** (5, 10);  
  
**SELECT** \* **FROM** EMPLOYS;

# Part-6: Initial Demonstration of DB Querying

1. This SQL query retrieves data from three tables - PATIENT, INSURANCE, and DUR\_PREMIUM - using inner join to connect them together. It selects the patient's age and their corresponding insurance policy's premium amount, and categorizes the policy premium as high, medium, or low based on certain conditions. The resulting output will display the patient's age, policy premiums, and their corresponding premium profile category. This query could be useful in analyzing patient demographics and insurance policy premiums for further research or business intelligence purposes.

**SELECT** P.PAT\_AGE **AS** 'Patient\'s Age', DP.POL\_PREMIUM **AS** 'Policy Premiums',  
**CASE  
 WHEN** DP.POL\_PREMIUM > 1200 **THEN** 'High'  
 **WHEN** DP.POL\_PREMIUM > 800 **AND** DP.POL\_PREMIUM <= 1200 **THEN** 'Medium'  
 **WHEN** DP.POL\_PREMIUM <= 800 **THEN** 'Low'  
**END AS** 'Premium Profile'  
**FROM** PATIENT **AS** P  
**INNER JOIN** INSURANCE **AS** I **ON** P.POL\_ID = I.POL\_ID  
**INNER JOIN** DUR\_PREMIUM **AS** DP **ON** I.POL\_ID = DP.POL\_ID;

2. This SQL query retrieves data from four tables - PATIENT, PRESCRIPTION, PRESCRIPTION\_LINES, and DRUGS - using inner join to connect them together. It selects patient IDs and their corresponding diseases based on the drugs that were prescribed to them, which are linked through the prescription lines. The resulting output will display the patient IDs and the diseases they were treated for based on the prescribed drugs. This query could be useful in tracking the types of diseases that patients are being treated for, and identifying any trends or patterns in prescribed medications.

**SELECT** P.PAT\_ID **AS** 'Patient IDs', DR.DRUGS\_DISEASE\_TREATED **AS** 'Diseases'  
**FROM** PATIENT **AS** P  
**INNER JOIN** PRESCRIPTION **AS** PRES **ON** P.PAT\_ID = PRES.PAT\_ID  
**INNER JOIN** PRESCRIPTION\_LINES **AS** PRES\_LINE **ON** PRES.RX\_ID = PRES\_LINE.RX\_ID  
**INNER JOIN** DRUGS **AS** DR **ON** PRES\_LINE.DRUG\_ID = DR.DRUG\_ID;

3. This SQL query retrieves data from two tables - DRUGS and PRESCRIPTION\_LINES - using a natural join to connect them together. It selects the distinct method of administration for drugs, and counts the number of times each method appears in the prescription lines. It then groups the results by method of administration, orders the results by the count of prescription lines in descending order, and limits the output to the highest count. The resulting output will display the method of administration that is most commonly used in prescription lines. This query could be useful in identifying the most frequently used method of drug administration, which could inform healthcare providers and policymakers in their decision-making processes.

**SELECT DISTINCT** METHOD\_OF\_ADMIN, *COUNT*(METHOD\_OF\_ADMIN)  
**FROM** DRUGS  
**NATURAL JOIN** PRESCRIPTION\_LINES  
**GROUP BY** METHOD\_OF\_ADMIN  
**ORDER BY** *COUNT*(RX\_LINE\_PRICE) **DESC  
LIMIT** 1;

# Part-7: Stored Procedure and Embedded Query

This SQL code creates a stored procedure named insert\_drugs, which takes six input parameters for inserting new records into the DRUGS table. The procedure executes an insert statement that adds a new record to the DRUGS table with the values provided by the input parameters. The subsequent call statement calls the insert\_drugs procedure with specific parameter values to insert a new drug record into the table. Finally, the SELECT statement is used to retrieve all records from the DRUGS table, which will include the newly inserted record. This code could be useful for inserting new drugs into the database system as needed, such as when a new drug becomes available for treating a certain disease.

**CREATE PROCEDURE** *insert\_drugs*(  
 **IN** DRUG\_ID **INT**,  
 **IN** DRUG\_NAME **VARCHAR**(100),  
 **IN** DRUGS\_DISEASE\_TREATED **VARCHAR**(100),  
 **IN** METHOD\_OF\_ADMIN **VARCHAR**(100),  
 **IN** DRUG\_TYPE **VARCHAR**(100),  
 **IN** DRUG\_ACTIVE\_INGREDIENT **VARCHAR**(100)  
)  
  
**BEGIN  
 INSERT INTO** DRUGS **VALUES** (DRUG\_ID, DRUG\_NAME, DRUGS\_DISEASE\_TREATED, METHOD\_OF\_ADMIN, DRUG\_TYPE, DRUG\_ACTIVE\_INGREDIENT);  
**END**;  
  
**call** *insert\_drugs*(  
 12,  
 "Proair\_4",  
 "Asthma",  
 "Orally inhaled",  
 "Branded",  
 "Insulin Glargine"  
);  
  
**SELECT** \* **FROM** DRUGS;

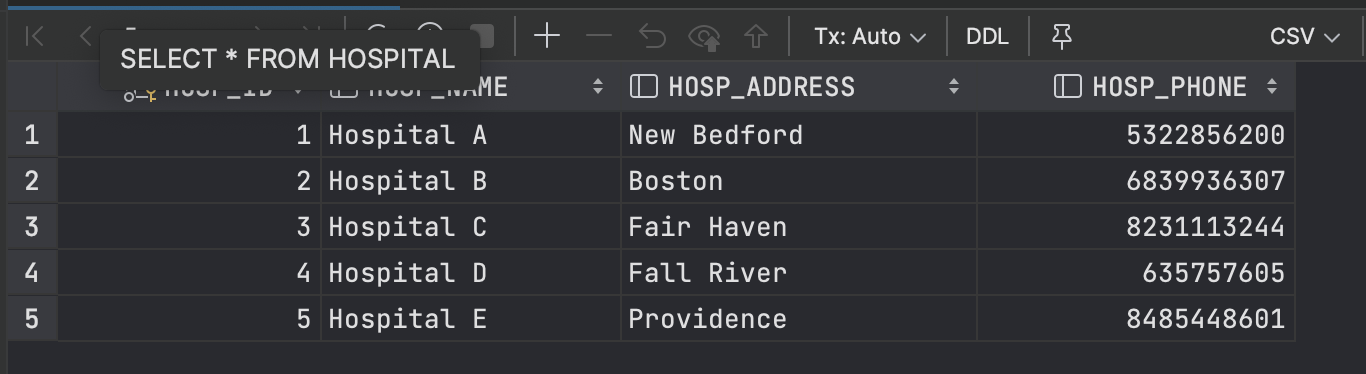
This SQL code creates a trigger named Prescription\_trigger that is executed automatically after every new record is inserted into the PRESCRIPTION\_LINES table. The trigger inserts a new record into the PRICE\_QUANT\_TOTAL table, which calculates the total price of the prescription line by multiplying the RX\_LINE\_PRICE by the RX\_LINE\_QUANTITY.

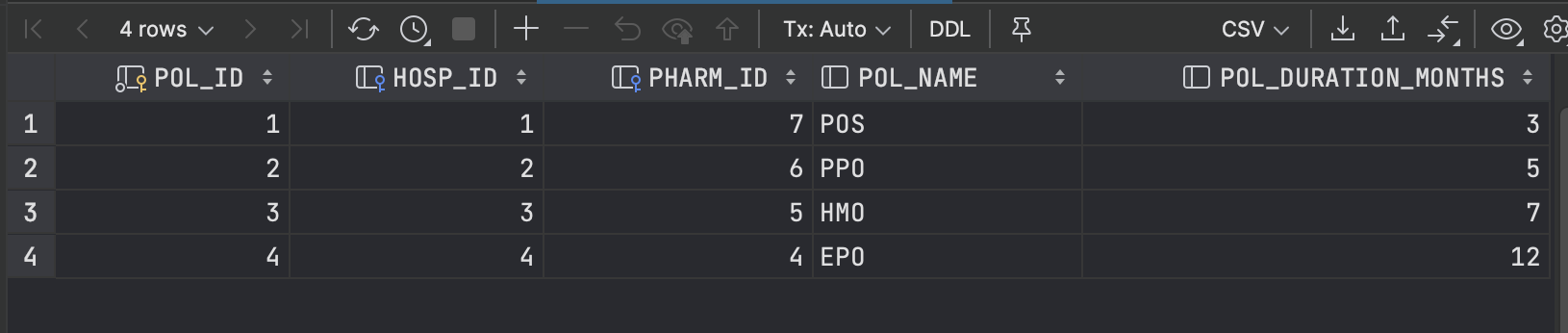
The subsequent INSERT statement adds a new record to the PRESCRIPTION\_LINES table with specific values for RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, and RX\_LINE\_QUANTITY. This will trigger the Prescription\_trigger trigger, which in turn inserts a new record into the PRICE\_QUANT\_TOTAL table.

Finally, the SELECT statement is used to retrieve all records from the PRICE\_QUANT\_TOTAL table, which will include the newly inserted record. This code could be useful in tracking the total prices of prescription lines and analyzing the costs associated with different medications or treatments.

**DELIMITER** ||  
**CREATE TRIGGER** Prescription\_trigger **AFTER INSERT ON** PRESCRIPTION\_LINES  
 **FOR EACH ROW  
 BEGIN  
 INSERT INTO** PRICE\_QUANT\_TOTAL **VALUES** (NEW.RX\_LINE\_ID, NEW.RX\_LINE\_PRICE, NEW.RX\_LINE\_QUANTITY, NEW.RX\_LINE\_PRICE\*NEW.RX\_LINE\_QUANTITY);  
 **END**||  
  
**INSERT INTO** PRESCRIPTION\_LINES(RX\_ID, DRUG\_ID, RX\_LINE\_PRICE, RX\_LINE\_QUANTITY) **VALUES**(1,7,200,30);  
  
**SELECT** \* **FROM** PRICE\_QUANT\_TOTAL;

# Results:





Query results:

1.

Graphical user interface, text

Description automatically generated

2.

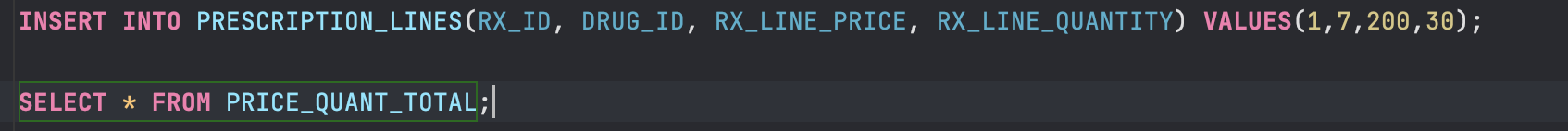
Graphical user interface, text, application

Description automatically generated

3. Graphical user interface, application

Description automatically generated

Trigger output:



A screenshot of a computer

Description automatically generated with medium confidence

Procedure output:

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated