Project Group: 0502\_01

**Medlife Pharmacy**

**README FILE**

# OVERVIEW

Medlife Pharmacy is a system that stores data related to medication use processes within pharmacies in an orderly and systematic manner. Thus, serving many purposes which ensures safety, accuracy, and efficiency within the pharmaceutical store. The main aim of this project is to develop the aforementioned system such that the pharmaceutical store can effectively manage its behemoth data and stay ahead of the curve.

This program deals with the day-to-day requirements about the pharmacy. It handles inventory details, drug details, customer details, employee details and the sales details. Thereupon, it will be facilitating the process for recording each of these data and fetch the necessary detail whenever required.

**PRIMARY OBJECTIVE**

* To develop a system that deals with the day to day requirement of any pharmacy.
* To ensure safety, accuracy, and efficiency within the pharmaceutical store.
* To provide competitive advantage to the pharmacy.
* To provide detailed information about the drugs in order to help locate it in the store easily.
* To streamline the use of inventory in the pharmacy

# DATA SOURCES

* We have referenced the type of data by considering Pharmacy Management Systems like CVS Pharmacies which deal with collecting drugs from Suppliers and selling it to the customers.
* There are three main sources of data that are being managed in this system at Medlife Pharmacy:

1. The data coming from different Pharmaceutical companies that manufacture drugs.
2. The Employees who work at Medlife
3. The customers who purchase drugs from our stores

* We have populated the tables by inserting data on our own, by keeping in mind the type of data that pharmacies like CVS use.

# DATABASE DESIGN

## TABLE DEFINITION

* **Company**: The company table contains the details of the companies that Medlife sources its drugs from. In our database, we store the company’s name, address and phone number.
* **Drug**: The drug table contains the details of the drugs that Medlife purchases from the drug-producing companies. Details such as drug ID, drug Price, name of the company that produced the drug, etc is contained in the Drug table.
* **Inventory**: The inventory details contain the information about the type of inventory (e.g. Warehouse) and its address that is used by Medlife to store its drug shipments.
* **Employee**: The employee table contains details about the employees that work at Medlife and their details.
* **HourlyEmployee**: This is the subtype of the Employee table that contains details about the employees that are employed on an hourly basis. Information such as Hourly Rate of the employee is contained within this table.
* **FullTimeEmployee**: This is the subtype of the Employee table that contains details about the employees that are employed on a full-time basis. Information such as the employee salary is contained within this table.
* **ContractEmployee**: This is the subtype of the Employee table that contains details about the employees that are employed on a contract basis. Information such as the contract rate and the contract duration of the employee is contained within this table.
* **Customer**: The customer table consists of the data of the customers that visit Medlife and make a purchase.
* **Bill**: The Bill table consists of the data of the purchases made by a particular customer.
* **Purchase**: The purchase table consists of the data of the purchases made of a drug and which inventory it is stored in and also the data about the quantity and the date of purchase.
* **Sale**: The Sale table consists of the data of the drugs and their corresponding Bill number which can help us determine which customer is purchasing which drugs.

## 

## ENTITY RELATIONSHIP DIAGRAM

An entity relationship diagram (ERD) is a graphical representation that provides an overview of the scope of the project and how each data set is related to one another. Thus, helping to define the business processes and serving as the foundation for a relational database.

## 

## 

## 

## 

## 

## LOGICAL DATABASE DESIGN

# Company (cmpId, cmpName, cmpPhone, cmpAddress)

# Drug (drgBarcode, drgName, drgType, drgDose, drgCode, drgCostPrice,

# drgSellPrice, drgProductionDate, drgExpirationDate, drgQuantity, drgProfit, *cmpName*)

# Inventory(invId, invType, invAddress)

# Bill (bilId, bilType, bilPrice, bilDate,*cstId*)

# Customer (cstId, cstName, cstPhone, cstDOB,*empId*)

# Employee (empId, empName, empPhone, empAddress, empPassword, empType)

# HourlyEmployee (*empId*, empHourlyRate)

# FullTimeEmployee (*empId*, empSalary, empStockAmount, empPaidLeave)

# ContractEmployee (*empId*, empContractDuration, empContractAmount)

# Purchase (*invId*, *drgBarcode*, prcQuantity, prcDate)

# Sale (*drgBarcode*, *bilId*, saleQty)

# TESTING THE PROJECT

## CREATE TABLE

For our system, we require the following tables to store the pharmacy details:

* Company
* Drug
* Inventory
* Employee
* Customer
* Bill
* Purchase
* Sale
* Hourly Employee
* Full time employee
* Contract Employee

But, before we proceed with creating these tables, we need to make sure that they do not exist already and if they do, we need to drop them. For this, we will run the drop table query in the backwards order of the creation of each table

USE BUDT703\_Project\_0502\_01

-- Drop the table if exists

DROP TABLE IF EXISTS ContractEmployee;

DROP TABLE IF EXISTS FullTimeEmployee;

DROP TABLE IF EXISTS HourlyEmployee;

DROP TABLE IF EXISTS Sale;

DROP TABLE IF EXISTS Purchase;

DROP TABLE IF EXISTS Bill;

DROP TABLE IF EXISTS Customer;

DROP TABLE IF EXISTS Employee;

DROP TABLE IF EXISTS Inventory;

DROP TABLE IF EXISTS Drug;

DROP TABLE IF EXISTS Company;

Now, we can move forward to create the tables

**CREATE TABLE: COMPANY**

CREATE TABLE Company (

cmpName VARCHAR (30) NOT NULL,

cmpPhone CHAR(10),

cmpAddress VARCHAR(50),

CONSTRAINT pk\_Company\_cmpName PRIMARY KEY (cmpName))

**CREATE TABLE: DRUG**

CREATE TABLE Drug (

drgBarcode CHAR (10) NOT NULL,

drgName VARCHAR(30),

drgType VARCHAR(20),

drgDose INTEGER,

drgCode VARCHAR(10),

drgCostPrice DECIMAL(6,2),

drgSellPrice DECIMAL(6,2),

drgProductionDate DATE,

drgExpirationDate DATE,

drgQuantity INTEGER,

drgProfit DECIMAL(6,2),

cmpName VARCHAR (30) NOT NULL,

CONSTRAINT pk\_Drug\_drgBarcode PRIMARY KEY (drgBarcode),

CONSTRAINT fk\_Drug\_cmpName FOREIGN KEY (cmpName )

REFERENCES Company (cmpName )

ON DELETE NO ACTION ON UPDATE CASCADE)

**CREATE TABLE: INVENTORY**

CREATE TABLE Inventory (

invId CHAR(10) NOT NULL,

invType VARCHAR(20),

invAddress VARCHAR (50),

CONSTRAINT pk\_Inventory\_invId PRIMARY KEY (invId))

**CREATE TABLE: EMPLOYEE**

CREATE TABLE Employee (

empId VARCHAR (10) NOT NULL,

empName VARCHAR (20),

empPhone CHAR (12),

empAddress VARCHAR (50),

empPassword VARCHAR (20),

emp\_type VARCHAR(20),

CONSTRAINT pk\_Employee\_empId PRIMARY KEY (empId) )

**CREATE TABLE: CUSTOMER**

CREATE TABLE Customer (

cstId VARCHAR (10) NOT NULL,

cstName VARCHAR (20),

cstPhone CHAR (10),

cstDOB DATE,

empId VARCHAR (10) NOT NULL,

CONSTRAINT pk\_Customer\_cstId PRIMARY KEY (cstId),

CONSTRAINT fk\_Customer\_cmpName FOREIGN KEY (empId)

REFERENCES Employee (empId)

ON DELETE NO ACTION ON UPDATE CASCADE)

**CREATE TABLE: BILL**

CREATE TABLE Bill (

bilId CHAR (10) NOT NULL,

bilType VARCHAR (20),

bilDate DATE,

bilPrice DECIMAL (8,2),

cstId VARCHAR (10) NOT NULL,

CONSTRAINT pk\_Bill\_bilId PRIMARY KEY (bilId),

CONSTRAINT fk\_Bill\_cstId FOREIGN KEY (cstId)

REFERENCES Customer (cstId)

ON DELETE NO ACTION ON UPDATE CASCADE )

**CREATE TABLE: PURCHASE**

CREATE TABLE Purchase (

invId CHAR (10) NOT NULL,

drgBarcode CHAR (10) NOT NULL,

prcQuantity INTEGER,

prcDate DATE,

CONSTRAINT pk\_Purchase\_invId\_drgBarcode PRIMARY KEY (invId, drgBarcode),

CONSTRAINT fk\_Purchase\_invId FOREIGN KEY (invId)

REFERENCES Inventory (invId)

ON DELETE NO ACTION ON UPDATE CASCADE,

CONSTRAINT fk\_Purchase\_drgBarcode FOREIGN KEY (drgBarcode)

REFERENCES Drug (drgBarcode)

ON DELETE CASCADE ON UPDATE CASCADE)

**CREATE TABLE: SALE**

CREATE TABLE Sale (

drgBarcode CHAR (10) NOT NULL,

bilId CHAR (10) NOT NULL,

saleQty INTEGER,

CONSTRAINT pk\_Sale\_drgBarcode\_bilId PRIMARY KEY (drgBarcode, bilId),

CONSTRAINT fk\_Sale\_drgBarcode FOREIGN KEY (drgBarcode)

REFERENCES Drug (drgBarcode)

ON DELETE NO ACTION ON UPDATE CASCADE,

CONSTRAINT fk\_Sale\_bilId FOREIGN KEY (bilId)

REFERENCES Bill (bilId)

ON DELETE CASCADE ON UPDATE CASCADE)

**CREATE TABLE: HOURLY EMPLOYEE**

CREATE TABLE HourlyEmployee(

empId VARCHAR (10) NOT NULL,

empHourlyRate DECIMAL (6,2),

CONSTRAINT pk\_HourlyEmployee\_empId PRIMARY KEY (empId),

CONSTRAINT fk\_HourlyEmployee\_cmpName FOREIGN KEY (empId)

REFERENCES Employee (empId)

ON DELETE CASCADE ON UPDATE CASCADE)

**CREATE TABLE: FULL TIME EMPLOYEE**

CREATE TABLE FullTimeEmployee (

empId VARCHAR (10) NOT NULL,

empSalary DECIMAL (10,2),

empStockAmount DECIMAL (9,2),

empPaidLeave INTEGER,

CONSTRAINT pk\_FullTimeEmployee\_empId PRIMARY KEY (empId),

CONSTRAINT fk\_FullTimeEmployee\_cmpName FOREIGN KEY (empId)

REFERENCES Employee (empId)

ON DELETE CASCADE ON UPDATE CASCADE)

**CREATE TABLE: CONTRACT EMPLOYEE**

CREATE TABLE ContractEmployee (

empId VARCHAR (10) NOT NULL,

empContractDurationMonths INTEGER,

empContractAmount DECIMAL (9,2),

CONSTRAINT pk\_ContractEmployee\_empId PRIMARY KEY (empId),

CONSTRAINT fk\_ContractEmployee\_cmpName FOREIGN KEY (empId)

REFERENCES Employee (empId)

ON DELETE CASCADE ON UPDATE CASCADE)

## INSERT RECORDS OUTPUT

Some sample queries to insert into the tables is given below:

**INSERT DATA IN THE TABLE: COMPANY**

INSERT INTO Company (cmpName, cmpPhone, cmpAddress)

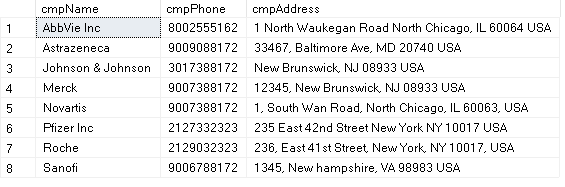
VALUES ('Pfizer Inc', '2127332323', '235 East 42nd Street New York NY 10017 USA'),

('AbbVie Inc' , '8002555162', '1 North Waukegan Road North Chicago, IL 60064 USA'),

('Johnson & Johnson' , '3017388172', 'New Brunswick, NJ 08933 USA');

SELECT \* FROM COMPANY

**OUTPUT:**

****

**INSERT DATA IN THE TABLE: DRUG**

INSERT INTO Drug (drgBarcode ,drgName, drgType, drgDose, drgCode, drgCostPrice,

drgSellPrice, drgProductionDate, drgExpirationDate, drgQuantity, drgProfit, cmpName)

VALUES ('1112908900', 'Atorvastatin', 'Prescription' , 2, '2d00', 23.4, 29.9,

'2022-08-11', '2025-08-11', 12, 6.50, 'Pfizer Inc'),

('1112908901', 'Levothyroxine', 'Prescription' , 2, '3m50', 13.4, 20.5, '2020-07-

12', '2022-07-12', 16, 7.10, 'AbbVie Inc'),

('1112908902', 'Metformin', 'Prescription' , 1, '4f30', 43.4, 55.9, '2021-11-13',

'2025-11-13', 20, 12.50, 'Johnson & Johnson')

('1112908903', 'Fosinopril', 'Prescription' , 2, '2d34', 73.40, 85.90, '2022-08-

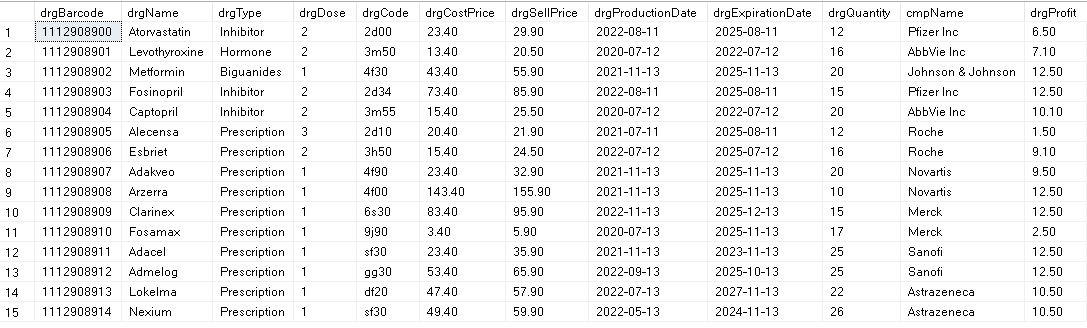
11', '2025-08-11', 15, 'AbbVie Inc', 12.50),

('1112908904', 'Captopril', 'Prescription' , 2, '3m50', 13.4, 20.5, '2020-07-

12', '2022-07-12', 16, 'AbbVie Inc', 10.10);

SELECT \* FROM DRUG

**OUTPUT:**

****

**INSERT DATA IN THE TABLE: INVENTORY**

INSERT INTO Inventory (invId, invType, invAddress)

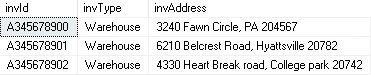
VALUES ('A345678900', 'Warehouse', '3240 Fawn Circle, PA 204567'),

('A345678901', 'Warehouse', '6210 Belcrest Road, Hyattsville 20782'),

('A345678902', 'Warehouse', '4330 Heart Break road, College park 20742');

SELECT \* FROM Inventory

**OUTPUT:**

****

**INSERT DATA IN THE TABLE: EMPLOYEE**

INSERT INTO Employee (empId, empName, empPhone, empAddress,

empPassword, empType)

VALUES ('1111192779','John Walker', '457890894', '571 Gulf Ave., Onalaska, WI

54650', 'gjkTYU&56^', 'Hourly'),

('1111192780','Raja Syan', '2405678894', '972 N. Myrtle St., Southampton, PA

18966', 'GHJKU&46^', 'Hourly'),

('1111192781','Yash Taylor', '2046378894', '829 Edgewater Street, Nashua, NH

03060', 'LKJ78&56^', 'Hourly'),

('1111192782','Zeel Patel', '1206378894', '118 Spring Dr., Poughkeepsie, NY 12601', 'UIOP34&5z^', 'FullTime'),

('1111192783','Vikram Walker', '2316378894', '78 Olive St., Vernon Hills, IL

60061', 'LKJrtU&56^', 'FullTime'),

('1111192784','Harsh Zalavadia', '4845678894', '9743 Big Rock Cove Court, Coventry,

RI 02816', '9087Y&56^', 'Contract'),

('1111192785','Nishant Vun', '2406378894', '500 Fulton Rd., Delray Beach, FL

33445', 'opLAU&56^', 'Contract'),

('1111192786','Geet Maine', '34046378894', '8082 Pilgrim Circle, Glendale, AZ

85302', 'MNghU&56^', 'Contract'),

('1111192787','Riddhi Gawda', '8906378894', '9420 Harvard Dr., Maineville, OH

45039', 'op><YU&56^', 'Contract');

SELECT \* FROM Employee

**OUTPUT:**

****

**INSERT DATA IN THE TABLE: CUSTOMER**

INSERT INTO Customer (cstId, cstName, cstPhone, cstDOB, empId)

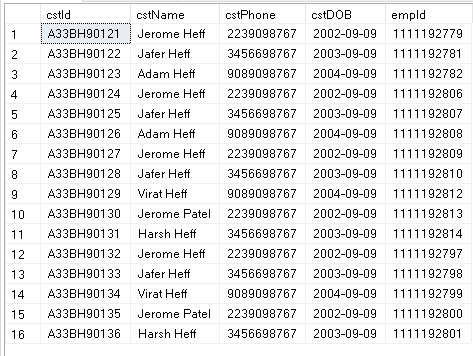
VALUES ('A33BH90121', 'Jerome Heff', '2239098767', '2002-09-09', '1111192779'),

('A33BH90122', 'Jafer Heff', '3456698767', '2003-09-09', '1111192781'),

('A33BH90123', 'Adam Heff', '9089098767', '2004-09-09', '1111192782');

SELECT \* FROM Customer;

**OUTPUT:**

****

**INSERT DATA IN THE TABLE: BILL**

INSERT INTO Bill (bilId , bilType, bilDate, bilPrice, cstId)

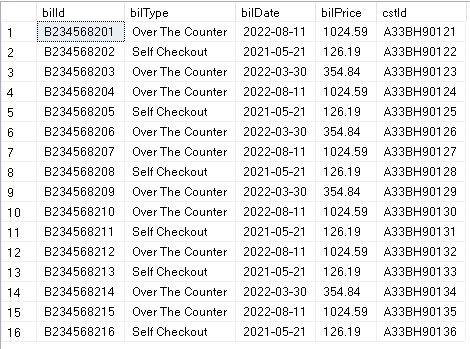
VALUES('B234568201', 'Over The Counter', '2022-08-11', 1024.59, 'A33BH90121'),

('B234568202', 'Self Checkout', '2021-05-21', 126.19, 'A33BH90122'),

('B234568203', 'Over The Counter', '2022-03-30', 354.84, 'A33BH90123');

SELECT \* FROM Bill;

**OUTPUT:**



**INSERT DATA IN THE TABLE: PURCHASE**

INSERT INTO Purchase (invId, drgBarcode , prcQuantity, prcDate)

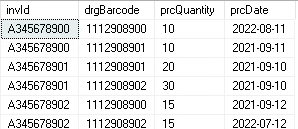
VALUES ('A345678900', '1112908900', 10, '2022-08-11'),

('A345678901', '1112908901', 20, '2021-09-10'),

('A345678902', '1112908902', 15, '2022-07-12');

SELECT \* FROM Purchase;

**OUTPUT:**



**INSERT DATA IN THE TABLE: SALE**

INSERT INTO Sale (drgBarcode, bilId)

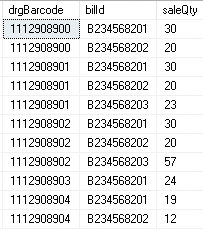
VALUES ('1112908900', 'B234568201'),

('1112908902', 'B234568202'),

('1112908901', 'B234568202');

SELECT \* FROM Sale;

**OUTPUT:**



**INSERT DATA IN THE TABLE: HOURLY EMPLOYEE**

INSERT INTO HourlyEmployee (empId, empHourlyRate)

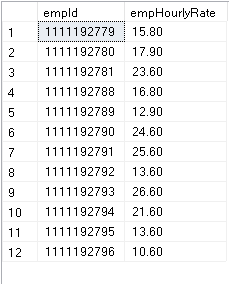
VALUES ('1111192779', 15.80),

('1111192780', 17.90),

('1111192781', 23.60);

SELECT \* FROM Employee;

**OUTPUT:**

****

**INSERT DATA IN THE TABLE: FULL TIME EMPLOYEE**

INSERT INTO FullTimeEmployee (empId, empSalary, empStockAmount, empPaidLeave)

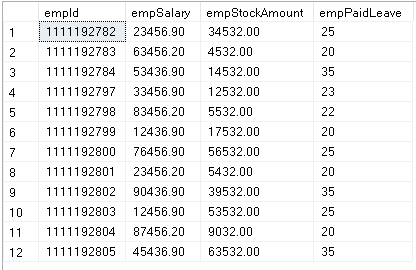
VALUES ('1111192782', 23456.90, 34532.00, 25),

('1111192783', 63456.20, 4532.00, 20),

('1111192784', 53436.90, 14532.00, 35);

SELECT \* FROMFullTimeEmployee;

**OUTPUT:**



**INSERT DATA IN THE TABLE: CONTRACT EMPLOYEE**

INSERT INTO ContractEmployee (empId, empContractDurationMonths, empContractAmount)

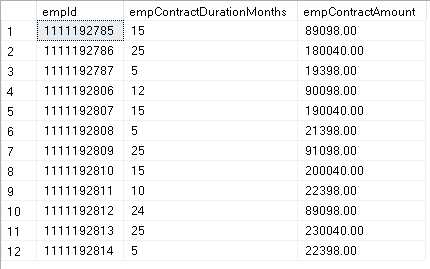
VALUES ('1111192785', 15, 89098.00),

('1111192786', 25, 180040.00),

('1111192787', 5, 19398.00);

SELECT \* FROM ContractEmployee;

**OUTPUT:**



## BUSINESS TRANSACTION 1 - QUERY & OUTPUT

As part of our customer experience, we would like to know how our employees are doing with customer assistance. Are customers who purchase from our pharmacy, helped by Full-time employees or part-time (Hourly)?

SELECT e.emp\_type AS 'Employee Type' , COUNT(e.empId) AS 'Number of Customers'

FROM Employee e

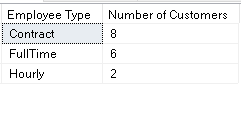
WHERE empId IN (

SELECT c.empId FROM Customer c, Bill b

WHERE c.cstId = b.cstId)

GROUP BY e.emp\_type

ORDER BY COUNT(e.empId) DESC;



## BUSINESS TRANSACTION 2 - QUERY & OUTPUT

For the purpose of efficient Inventory management, the manager wants to know the purchase quantity of drugs for any particular month and year. This will help the manager in determining which drugs to buy in bulk and also determine the kind of contracts he/she wants to make with drug selling companies.

SELECT d.drgName AS 'Drug Name', p.drgBarcode AS 'Drug Barcode',

d.cmpName AS 'Company Name', SUM(p.prcQuantity) AS 'Purchase Quantity'

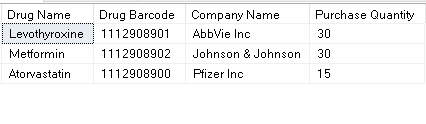
FROM Purchase p, Drug d

WHERE YEAR(p.prcDate) = '2021' AND MONTH(p.prcDate) = '09' AND

p.drgBarcode = d.drgBarcode

GROUP BY p.drgBarcode, d.drgName, d.cmpName

ORDER BY SUM(p.prcQuantity) DESC;



## BUSINESS TRANSACTION 3 - QUERY & OUTPUT

The consumption pattern of drugs changes yearly. With the rise in variety of diseases and global outbreaks, the most-selling drugs are different for every year. The Sales manager wants to see the most-selling drugs every year to analyze the trends and patterns and for forecasting purposes.

SELECT s.drgBarcode AS 'Drug Barcode', d.drgName AS 'Drug Name',

d.cmpName AS 'Company Name', YEAR(b.bilDate) AS 'Year of Sale',

SUM(s.saleQty) AS 'Sale Quantity'

FROM Drug d, Sale s, Bill b

WHERE (Year(b.bilDate) IN (

SELECT Year(b.bilDate)

FROM Bill b GROUP BY YEAR(b.bilDate) ) AND

(s.drgBarcode = d.drgBarcode AND s.bilId = b.bilId))

GROUP BY s.drgBarcode, d.drgName, d.cmpName, YEAR(b.bilDate)

ORDER BY SUM(s.saleQty) DESC, d.drgName;



## BUSINESS TRANSACTION 4 - QUERY & OUTPUT

Profitability is an important part of any business. Medlife wants to estimate which drugs are the most profitable for them. This will help the Sales manager and the Relationship manager to enter into long term contracts for companies whose drugs are sold in the highest number by Medlife.

SELECT d.drgType AS 'Drug Type', d.drgName AS 'Drug Name', d.cmpName AS 'Company Name',

SUM(s.saleQty) AS 'Sale Quantity',

((SUM(d.drgProfit)\*SUM(s.saleQty))/COUNT(s.drgBarcode)) AS 'Total Profit'

FROM Drug d, Sale s

WHERE d.drgBarcode = s.drgBarcode

GROUP BY d.drgType, d.cmpName, d.drgName

ORDER BY d.drgType ASC , SUM(s.saleQty) DESC;



# REFERENCES

[1]

https://www.g2.com/categories/pharmacy-management-systems

[2] https://en.wikipedia.org/wiki/Pharmacy\_management\_system#:~:text=The%20pharmacy%20management%20system%2C%20also,medication%20use%20process%20within%20pharmacies.

[3]

https://arirms.com/features-of-pharmacy-management-system