

**SHAHEED RAJPAL DAV PUBLIC SCHOOL
DAYANAND VIHAR, DELHI**

**COMPUTER SCIENCE (083)
PROJECT
2023-2024**

MEDICINE STORAGE SYSTEM

SUBMITTED BY : PRANEEL MAITRA

CLASS AND SECTION : XII-C

ROLL NUMBER : 23

BOARD ROLL NO. :

SUBMITTED TO : MRS. NEERU MITTAL

INDEX

- ❖ **CERTIFICATE.**
- ❖ **ACKNOWLEDGEMENT**
- ❖ **INTRODUCTION TO THE PROJECT**
- ❖ **BACKEND DETAILS.**
- ❖ **FRONTEND DETAILS.**
- ❖ **MOTIVE.**
- ❖ **SCREEN SHOTS OF EXECUTION.**
- ❖ **BIBLIOGRAPHY.**
- ❖ **LIMITATIONS.**

CERTIFICATE

This is to certify that PRANEEL MAITRA of class XII-C, Shaheed Rajpal DAV Public School has successfully completed his project in Computer Science Practical for the AISSCE as prescribed by CBSE in the year 2023-2024.

Board Roll No :

Sign. of Internal

Sign. of External

ACKNOWLEDGEMENT

I would like to thank my Computer Science teacher Mrs. Neeru Mittal for guidance and support. I am also thankful to our principal Mrs. Vineeta Kapoor. I would also thank to my parents for encouraging during the course of this project. Finally, I would like to thank CBSE for giving me this opportunity to undertake this project.

INTRODUCTION TO THE PROJECT

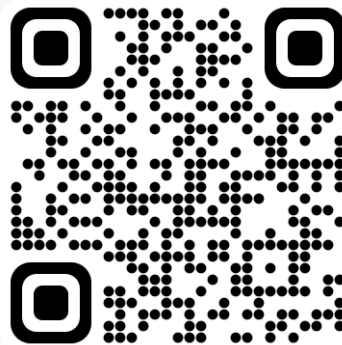
The Medicine Storage System is an ERP software that can be used by medicine shops or medicine dealers for wholesale/retail business. This software stores details of medicines and helps us to search medicines by their name and manufacturer. It is possible to edit medicine cost and sell the medicine. The balance i.e. due amount of the stock can also be checked. If the medicine is expired, the system has the provision to dispose it to the system specified warehouse. The program is also useful to check the details of the expired medicines.

PROJECT DETAILS

- **Softwares Used**

- Python
- MY SQL

- **CODE LINK:** <https://github.com/praneel1/cs-project-12>



- **Skills Used**

- Python
- Database Management
- My SQL
- My SQL Connector

BACKEND DETAILS

Database Name: MEDICINE

Create Database Medicine;

Use Medicine;

Table Name: STOCK

```
CREATE TABLE STOCK (  
Batch_no    int(11) Primary Key,  
name        varchar(50),    manuf    varchar(50),  
date_man    date,           date_exp  date,  
quantity    int,            sell        int,  
balance     int,            cost_unit int );
```

Table Name: DISPOSE

```
CREATE TABLE DISPOSE (  
Batch_no    int(11),    Name        varchar(50),  
date_exp    date,       amount      int );
```

FRONT-END DETAILS

PROGRAM CODE

```
import os # For clean terminal window
import datetime # For tracking expired meds
import mysql.connector
mycon=mysql.connector.connect(host='localhost',user='root',
password='tiger',database='medicine')
mycur=mycon.cursor()
def Store(): # This Function is used to add medicines
    print('\nPLEASE PROVIDE THE REQUIRED INFORMATION\n')
    batch_no=int(input('\nENTER THE BATCH NUMBER:'))
    name=input('\nENTER THE NAME OF THE MEDICINE WITH POWER:')
    manuf=input('\nENTER THE NAME OF THE MANUFACTURER:')
    date_man=input('\nENTER THE DATE OF MANUFACTURE(YYYY-MM-DD):')
    date_exp=input('\nENTER THE DATE OF EXPIRY(YYYY-MM-DD):')
    quantity=int(input('\nENTER THE QUANTITY OF THE IMPORTED MEDICINE:'))
    sell=0
    balance=quantity
    cost_unit=int(input('\nENTER THE COST OF THE IMPORTED MEDICINE PER
UNIT:'))
    sql="Insert into stock
values("+str(batch_no)+',"'+name+'",'+manuf+',",'+date_man+',",'+date_exp+',
",'+str(quantity)+',','+str(sell)+',','+str(balance)+',','+str(cost_unit)+');"

    try:
        mycur.execute(sql)
        print(name,'ADDED TO THE STOCK')
        mycon.commit()
    except:
        print('UNABLE TO ADD MEDICINE!!!!!!')

def Search_by_Name(): # Used to search meds by name
    ph=input('\nENTER THE MEDICINE NAME TO SEARCH:')
    sql="Select * from Stock where name='"+ph+"";"
    mycur.execute(sql)
    rec=mycur.fetchone()
    if(rec==None):
        print(ph,'IS NOT AVAILABLE')
    else:
        print('BATCH NUMBER:\t',rec[0])
        print('MEDICINE NAME:\t',rec[1])
        print('MANUFACTURER:\t',rec[2])
        print('DATE OF MANUFACTURE:\t',rec[3])
```



```
print('DATE OF EXPIRY:\t',rec[4])
print('QUANITTY STORED:\t',rec[5])
print('INITIAL COST:\t',rec[8])
```

```
def Search_by_Manu(): # Used to search by manufacturer's name
    ph=input('\nENTER THE MANUFACTURER NAME TO SEARCH:')
    sql="Select name from Stock where manuf='"+ph+"'";
    mycur.execute(sql)
    rec=mycur.fetchall()
    if(rec==None):
        print(ph,'IS A WRONG MANUFACTURER')
    else:
        print('-----MEDICINES MANUFACTURED BY',ph,'-----')
    for name in rec:
        print(name[0])
```

```
def Cost_Update(): # Used to update cost of medicine

    name=input('\nENTER THE MEDICINE NAME TO CHANGE COST:')
    cost=int(input('\nENTER THE NEW COST PER UNIT:'))
    sql="Update stock set cost_unit="+str(cost)+" where name='"+name+"'";
    try:
        mycur.execute(sql)
        mycon.commit()
        print('NEW COST OF',name,'IS=RS',cost)
    except:
        print('UNABLE TO CHANGE COST!!!!')
```

```
def Sell(): # Used when meds are sold
    sql="Update stock set sell=%s,balance=%s where name=%s;"
    ph=input('\nENTER THE MEDICINE NAME TO SELL:')
    addr=int(input('\nENTER THE QUANTITY TO SELL:'))
    sql2='select quantity from stock where name=%s'
    value2=(ph,)
    mycur.execute(sql2,value2)
    rec=mycur.fetchone()
    if(addr>rec[0]):
        print('INSUFFICIENT STOCK IN HAND!!!!!!') # Printed when quantity
        required is less than present
        return
    else:
        balance=rec[0]-addr
        value=(addr,balance,ph)
        try:
```

```

        mycur.execute(sql,value)
        mycon.commit()
        print(addr,'UNITS OF',ph,'SOLD')
        print(balance,'UNITS LEFT')
except:
    print('UNABLE TO SELL MEDICINE!!!!')

def Available(): # Check no. of units of med available
    name=input('\nENTER THE MEDICINE NAME TO SEARCH:')
    sql="Select balance from Stock where name='"+name+"';"
    mycur.execute(sql)
    rec=mycur.fetchone()
    if(rec==None):
        print(name,'IS NOT AVAILABLE')
    else:
        print(rec[0],'UNITS OF',name,'IS AVAILABLE')

def Dispose(): # Used to dispose expired meds
    sql="Insert into
dispose(batch_no,name,date_exp,amount)values(%s,%s,%s,%s)"
    nm=input('\nENTER THE MEDICINE NAME TO DISPOSE:')
    sql2="Select batch_no,name,date_exp,balance from stock where name=%s and
date_exp<=%s"
    t_date=datetime.date.today()
    value2=(nm,t_date)
    mycur.execute(sql2,value2)
    rec=mycur.fetchone()
    if(rec==None):
        print(nm,'IS NOT EXPIRED YET')
    else:
        print(nm,'IS EXPIRED')
        c=int(input('\nPRESS 1 TO DISPOSE IT:'))
        if(c==1):
            b=rec[0]
            n=rec[1]
            d=rec[2]
            am=rec[3]
            value=(b,n,d,am)
            sql3='Delete from stock where name=%s'
            value3=(n,)
            try:
                mycur.execute(sql,value)
                mycon.commit()
                print(n,'SUCCESSFULLY DISPOSED')
                mycur.execute(sql3,value3)
                mycon.commit()
            except:
                print('UNABLE TO DISPOSE MEDICINE')

```

```

        else:
            print('WARNING!!!!!!',nm,'MUST BE DISPOSED LATER')
            return

def Search_Dispose(): # Searches for disposed meds
    name=input('\nENTER THE DISPOSED MEDICINE NAME TO SEARCH:')
    sql="Select * from Dispose where name='"+name+"';"
    try:
        mycur.execute(sql)
        rec=mycur.fetchone()
        if(rec==None):
            print(name,'IS NOT AVAILABLE')
        else:
            print('BATCH NUMBER:\t',rec[0])
            print('MEDICINE NAME:\t',rec[1])
            print('DATE OF EXPIRY:\t',rec[2])
            print('BALANCE AMOUNT:\t',rec[3])
    except:
        print('NOT ACCESSIBLE')


def Close():
    os.system('cls')
    print('\nTHANK YOU FOR USING THE APPLICATION')
    quit()


while(True):
    print('-----WELCOME TO MEDICINE STOCK CHECKING SYSTEM-----')
    -\n\n')
    print('\nPRESS 1 TO ADD A NEW MEDICINE')
    print('PRESS 2 TO SEARCH A MEDICINE BY NAME')
    print('PRESS 3 TO SEARCH A MEDICINE BY MANUFACTURER')
    print('PRESS 4 TO UPDATE MEDICINE COST')
    print('PRESS 5 TO SELL MEDICINE')
    print('PRESS 6 TO CHECK AVAILABILITY')
    print('PRESS 7 TO DISPOSE EXPIRED MEDICINE')
    print('PRESS 8 TO SEARCH EXPIRED MEDICINE BY NAME')
    print('PRESS 9 TO CLOSE THE APPLICATION')
    try:
        choice=int(input('ENTER YOUR CHOICE : '))
    except:
        choice=10

    if(choice==1):          # Add
        os.system('cls')

```

```
        Store()
    elif(choice==2):          #Name Search
        os.system('cls')
        Search_by_Name()
    elif(choice==3):          #Manufacturer Search
        os.system('cls')
        Search_by_Manu()
    elif(choice==4):          #Cost Update
        os.system('cls')
        Cost_Update()
    elif(choice==5):          #Sell
        os.system('cls')
        Sell()
    elif(choice==6):          #Check Availability
        os.system('cls')
        Available()
    elif(choice==7):          #Dispose
        os.system('cls')
        Dispose()
    elif(choice==8):          #Search Dispose
        os.system('cls')
        Search_Dispose()
    elif(choice==9):          #Close
        Close()
    else:
        print('Invalid Entry. Press 9 to close app.')
```




MOTIVE

- ❖ To maintain the medicine stock details, sell medicine, update stock details, providing medicine amount enquiry by simple search technique.
- ❖ To dispose medicines which are expired and provide the facility to search the disposed medicines.
- ❖ To display the amount, sold amount, balance amount of a particular medicine by graphical analysis technique.
- ❖ Globalized usage.

SCREEN SHOTS OF EXECUTION


MAIN MENU

 C:\Windows\py.exe

-----WELCOME TO MEDICINE STOCK CHECKING SYSTEM-----

PRESS 1 TO ADD A NEW MEDICINE
PRESS 2 TO SEARCH A MEDICINE BY NAME
PRESS 3 TO SEARCH A MEDICINE BY MANUFACTURER
PRESS 4 TO UPDATE MEDICINE COST
PRESS 5 TO SELL MEDICINE
PRESS 6 TO CHECK AVAILABILITY
PRESS 7 TO DISPOSE EXPIRED MEDICINE
PRESS 8 TO SEARCH EXPIRED MEDICINE BY NAME
PRESS 9 TO CLOSE THE APPLICATION
ENTER YOUR CHOICE :

ADDING A NEW MEDICINE

 C:\Windows\py.exe

PLEASE PROVIDE THE REQUIRED INFORMATION

ENTER THE BATCH NUMBER:1006

ENTER THE NAME OF THE MEDICINE WITH POWER:ASPIRIN 125

ENTER THE NAME OF THE MANUFACTURER:CIPLA


ENTER THE DATE OF MANUFACTURE(YYYY-MM-DD):2023-02-01

ENTER THE DATE OF EXPIRY(YYYY-MM-DD):2024-01-05

ENTER THE QUANTITY OF THE IMPORTED MEDICINE:50


ENTER THE COST OF THE IMPORTED MEDICINE PER UNIT:12
ASPIRIN 125 ADDED TO THE STOCK

SEARCHING MEDICINE BY NAME

 C:\Windows\py.exe


```
ENTER THE MEDICINE NAME TO SEARCH:aspirin 125
BATCH NUMBER:      1006
MEDICINE NAME:     ASPIRIN 125
MANUFACTURER:      CIPLA
DATE OF MANUFACTURE:      2023-02-01
DATE OF EXPIRY:      2024-01-05
QUANITTY STORED:      50
INITIAL COST:      12
```

SEARCHING MEDICINE BY MANUFACTURER

 C:\Windows\py.exe

```
ENTER THE MANUFACTURER NAME TO SEARCH:cipla
-----MEDICINES MANUFACTURED BY cipla -----
zoryl m2
PREGABID 400
ASPIRIN 125
```

UPDATING MEDICINE COST


 Select C:\Windows\py.exe

ENTER THE MEDICINE NAME TO CHANGE COST:pregabid 400

ENTER THE NEW COST PER UNIT:20

NEW COST OF pregabid 400 IS=RS 20

SELLING MEDICINE

 C:\Windows\py.exe


ENTER THE MEDICINE NAME TO SELL:aspirin 125

ENTER THE QUANTITY TO SELL:15

15 UNITS OF aspirin 125 SOLD

35 UNITS LEFT


CHECKING AVAILABILITY

 C:\Windows\py.exe

ENTER THE MEDICINE NAME TO SEARCH:zoryl m2

75 UNITS OF zoryl m2 IS AVAILABLE


DISPOSING MEDICINES

 C:\Windows\py.exe

ENTER THE MEDICINE NAME TO DISPOSE:pregabid 400
pregabid 400 IS EXPIRED

PRESS 1 TO DISPOSE IT:1
PREGABID 400 SUCCESSFULLY DISPOSED

SEARCHING EXPIRED MEDICINE BY NAME

 C:\Windows\py.exe

ENTER THE DISPOSED MEDICINE NAME TO SEARCH:pregabid 400
BATCH NUMBER: 1004
MEDICINE NAME: PREGABID 400
DATE OF EXPIRY: 2023-04-04
BALANCE AMOUNT: 12

LIMITATIONS

- ❖ The project has no provision to calculate annual turnover of the medicine unit.
- ❖ The project does not incorporate the provision of GST Calculation.
- ❖ The project does not have the facility to take care of the medicines which are to be refunded i.e. there is no mechanism to keep the account of the refunded medicines.
- ❖ This application can be made good looking by making GUI using Tkinter or any other module.

BIBLIOGRAPHY

BOOKS:

- ✓ COMPUTER SCIENCE WITH PYTHON- BY
SUMITA ARORA
- ✓ PYTHON COOKBOOK

WEBSITES:

- ✓ www.python4me.com
- ✓ <https://docs.python.org/3/>
- ✓ <https://www.w3schools.com/python/>