PROJECT FILE

INFORMATICS PRACTICES

SESSION: 2020-2021

PREPARED BY-

NAME: Sanya Virmani

CLASS & SECTION: XII – C

MEDICAL STORE MANAGEMENT SYSTEM

INDEX

INTRODUCTION

This software project is developed to automate the functionalities of a Medical Store. This project is designed and coded in Python and information management is handled by CSV files. This software mainly focuses on basic operations in a medical store such as displaying stock, sales or employee records, adding new items/employee, deleting records, printing data visualized graphs and also maintaining the data.

Medical Store Management System is a python application written on 64-bit Windows 10 operating system, designed to maintain and organize the store records. The software is easy to use for both beginners as well as advanced users.

It contains addition, modification, deletion and generating graphs as per requirement.

SOFTWARE REQUIREMENT

- Windows OS
- Python

HARDWARE REQUIREMENT

- ***OPERATING SYSTEM:*** WINDOWS 7 AND ABOVE
- ***PROCESSOR:*** PENTIUM (ANY) OR AMD ATHALON(3800+- 4200+ DUAL CORE)
- ***MOTHERBOARD:*** 1.845 OR 915,995 FOR PENTIUM 0R MSI K9MM-V VIA K8M800+8237R PLUS CHIPSET FOR AMD ATHALON
- ***RAM:*** 4GB+
- ***HARD DISK:*** 500GB
- ***MONITOR:*** 14.1 or 15 -17 inch
- ***KEYBOARD AND MOUSE:*** YES
- ***PRINTER:*** if print is required – Hard copy

CSV TABLES

Table 1: *username and password of the medical store management system*

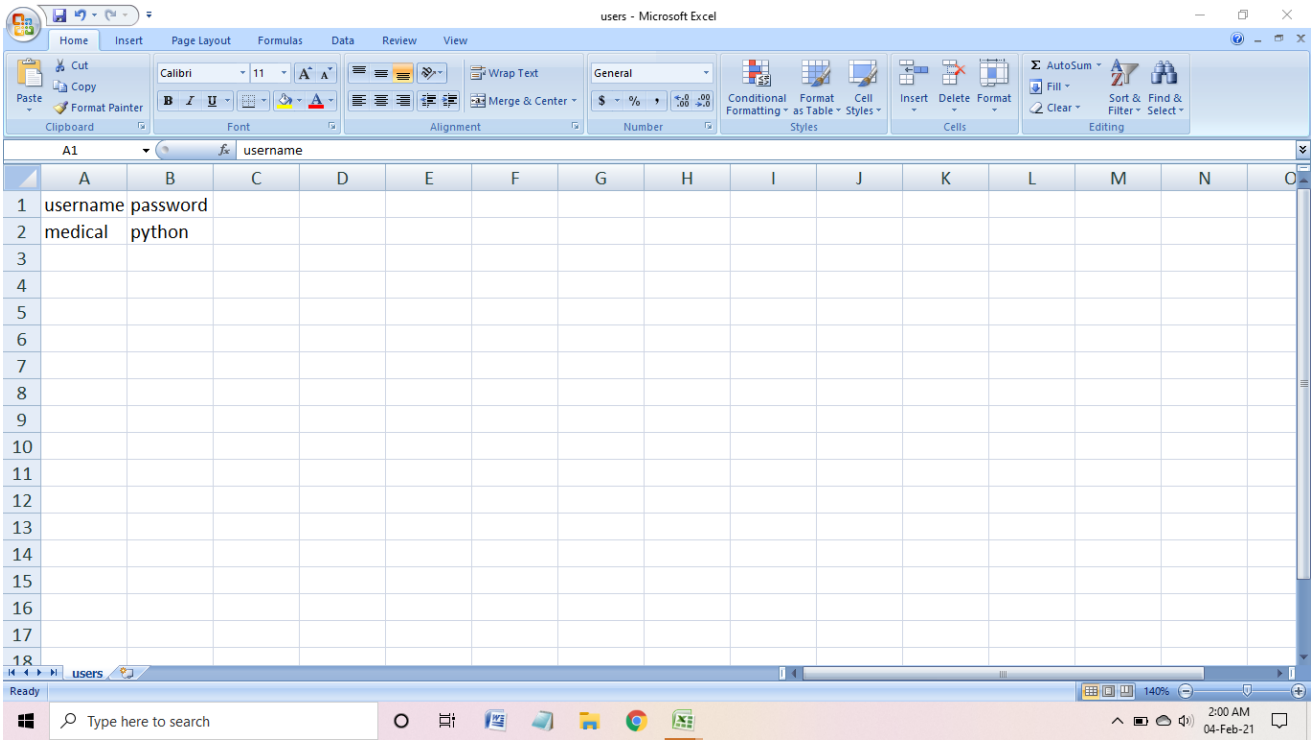


Table 2: *stock available in the medical store*

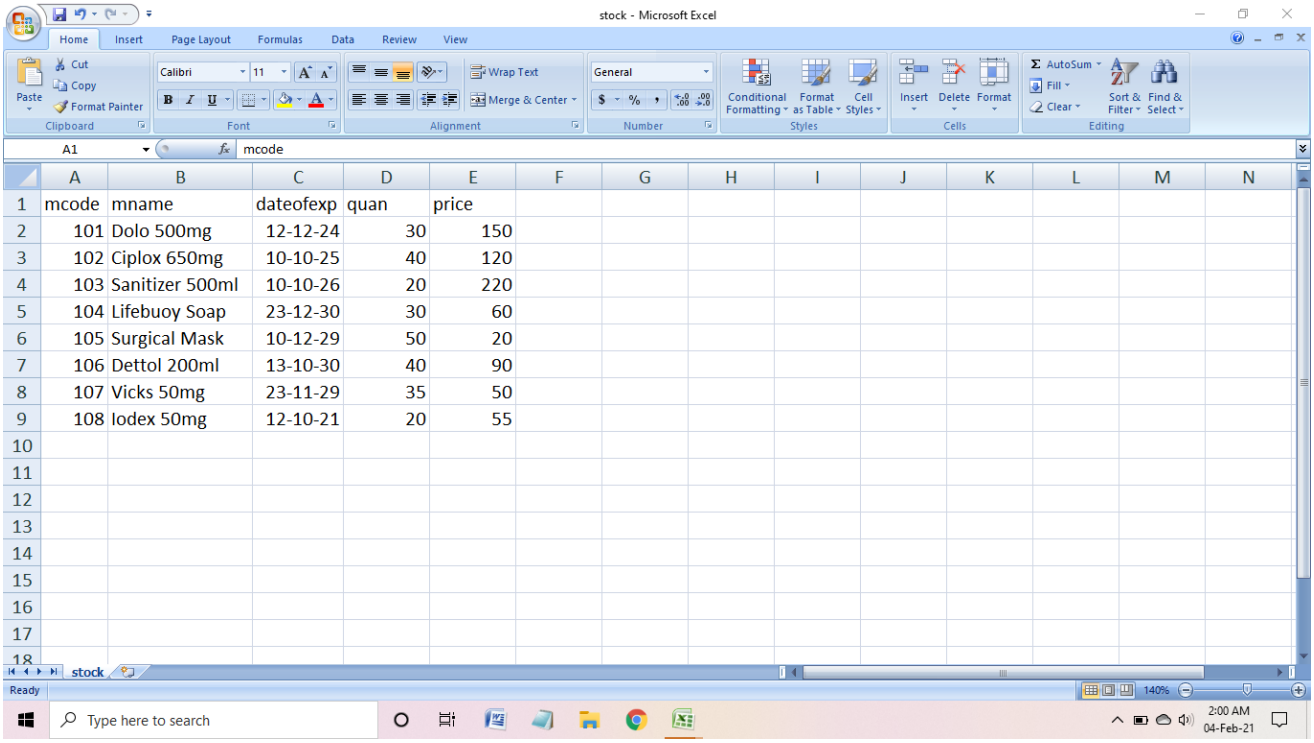
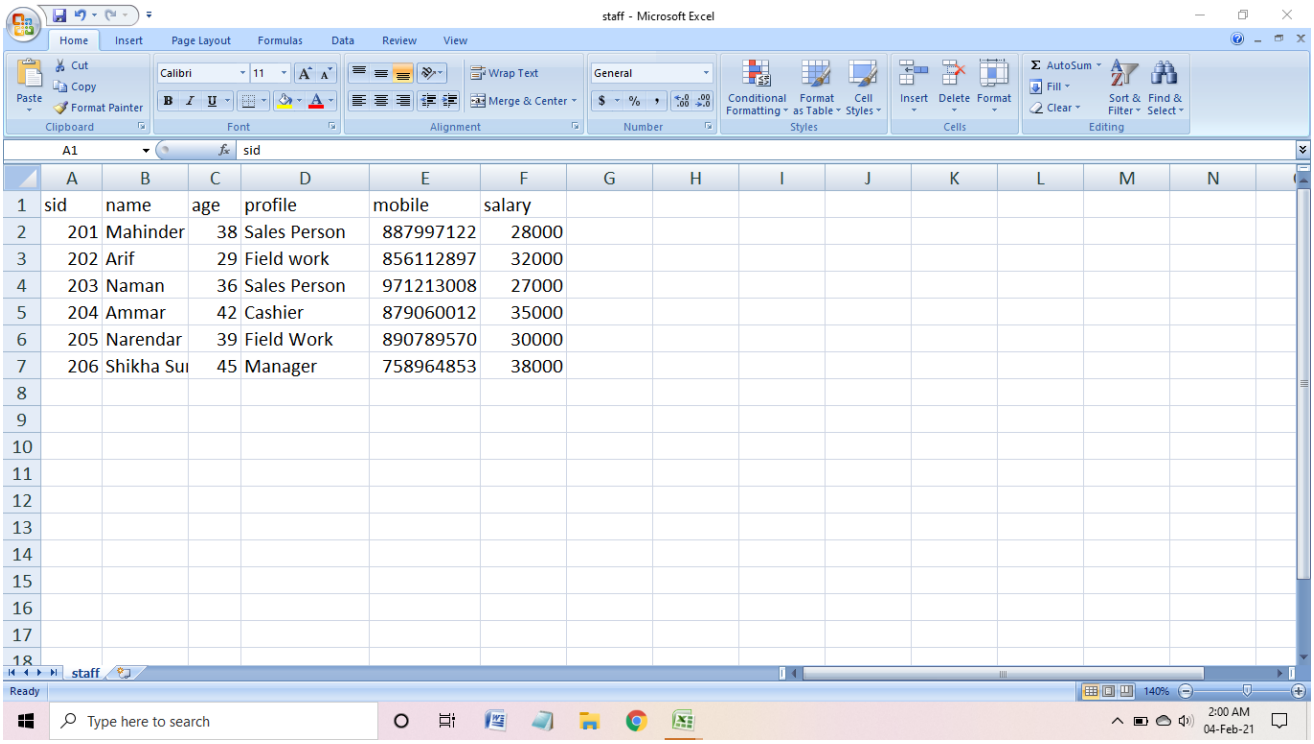
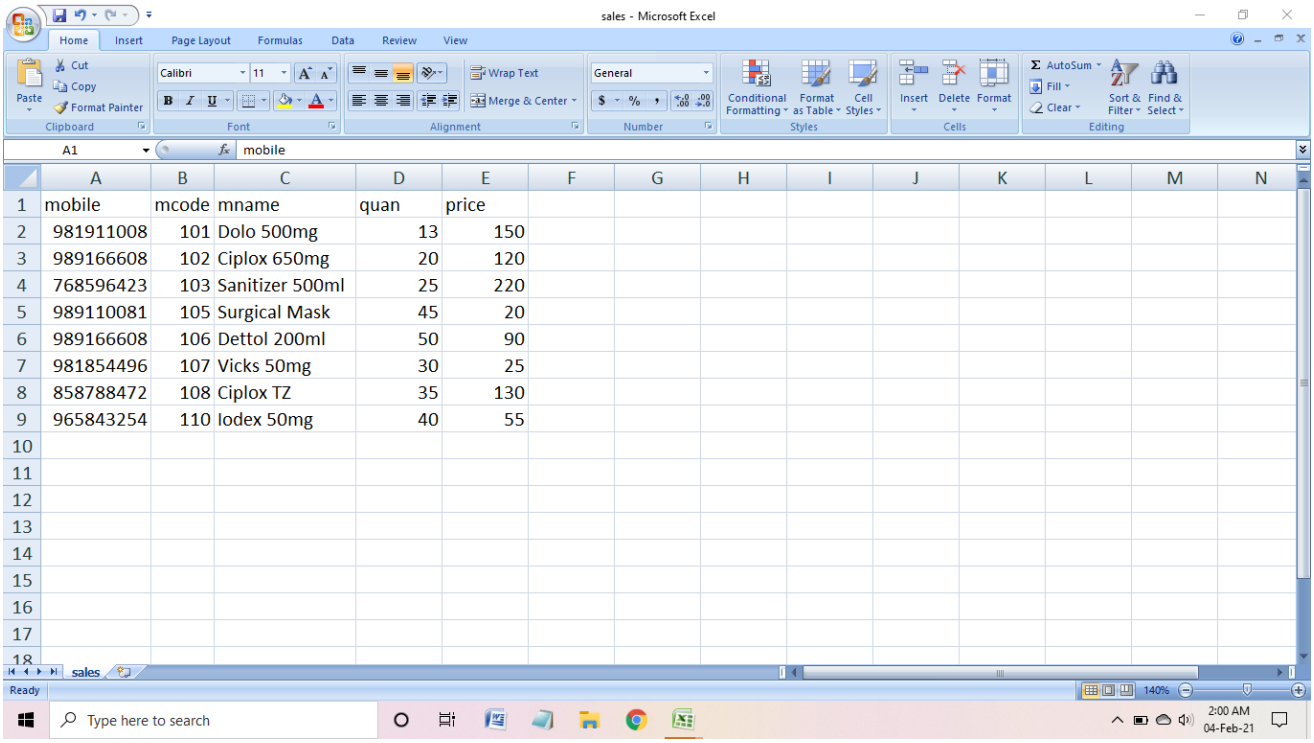


Table 3: *staff details of the medical store*



sid	name	age	profile	mobile	salary
201	Mahinder	38	Sales Person	887997122	28000
202	Arif	29	Field work	856112897	32000
203	Naman	36	Sales Person	971213008	27000
204	Ammar	42	Cashier	879060012	35000
205	Narendar	39	Field Work	890789570	30000
206	Shikha Sui	45	Manager	758964853	38000

Table 4: *sales record of the medical store*



mobile	mcode	mname	quan	price
981911008	101	Dolo 500mg	13	150
989166608	102	Ciplox 650mg	20	120
768596423	103	Sanitizer 500ml	25	220
989110081	105	Surgical Mask	45	20
989166608	106	Dettol 200ml	50	90
981854496	107	Vicks 50mg	30	25
858788472	108	Ciplox TZ	35	130
965843254	110	Iodex 50mg	40	55

PROGRAM CODE

```
import pandas as pd
import matplotlib.pyplot as plt
import sys

def stockrec():
    print("Reading Stock Records")
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\stock.csv")
    print(df)

def newmed():
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\stock.csv",index_col=0)
    print(df)
    a=int(input("Enter medicine code: "))
    b=(input("Enter medicine name: "))
    c=(input("Enter Date of Expiry: "))
    d=int(input("Enter Quantity: "))
    e=int(input("Enter Price: "))
    df.loc[a]=[b,c,d,e]
    df.to_csv("C:\\Users\\welch\\Desktop\\stock.csv")
    print("Data successfully added")
    print(df)

def removemed():
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\stock.csv",index_col='mcode')
    mcode=int(input("Enter mcode: "))
    df.drop(mcode,axis=0,inplace=True)
    print("Record of Medicine Temporarily deleted")
    print(df)
```

```
def updatemed():
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\stock.csv",index_col='mcode')
    print(df)
    a=int(input("Enter Medicine Code needs to be updated: "))
    b=(input("Enter Medicine Name: "))
    c=(input("Enter Date of Expiry: "))
    d=int(input("Enter Quantity: "))
    e=int(input("Enter Price: "))
    df.loc[a]=[b,c,d,e]
    df.to_csv("C:\\Users\\welch\\Desktop\\stock.csv")
    print("Data successfully updated:")
    print(df)

def stockplot():
    print("Printing Stock Report")
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\stock.csv")
    x=df['mname']
    y=df['quan']
    plt.title("Medicine Names and Quantity")
    plt.xlabel("Medicine")
    plt.ylabel("Quantity")
    plt.xticks(rotation=15)
    plt.plot(x,y,marker='X',ls="dashed",linewidth=4,color="red")
    plt.show()
```

```
def staffrec():
    print("Reading Employee Records")
    print()
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\staff.csv")
    print(df)

def newstaff():
    print()
    print("Old Employees Record in File Staff:")
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\staff.csv",index_col=0)
    print(df)
    a=int(input("Enter Employee ID: "))
    b=(input("Enter Employee Name: "))
    c=(input("Enter Age: "))
    d=(input("Enter Employee profile: "))
    e=int(input("Enter Mobile No. : "))
    f=int(input("Enter Salary: "))
    df.loc[a]=[b,c,d,e,f]
    df.to_csv("C:\\Users\\welch\\Desktop\\staff.csv")
    print("New Employee Added")
    print(df)

def salesrec():
    print("Reading Sales Records")
    print()
    print()
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\sales.csv")
    print(df)
```

```
def newsales():
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\sales.csv",index_col=0)
    print(df)
    a=int(input("Enter Customer's mobile no. : "))
    b=(input("Enter medicine code: "))
    c=(input("Enter medicine name: "))
    d=int(input("Enter Quantity: "))
    e=int(input("Enter Price: "))
    df.loc[a]=[b,c,d,e]
    df.to_csv("C:\\Users\\welch\\Desktop\\sales.csv")
    print("Data succesfully added")
    print(df)

def salesplot():
    print("Printing Sales Report")
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\sales.csv")
    x=df['mname']
    y=df['quan']
    plt.title('Sold Medicines and their Quantity')
    plt.xlabel('Medicine')
    plt.ylabel('Quantity')
    plt.xticks(rotation=15)
    plt.bar(x,y,color='blue')
    plt.show()
```

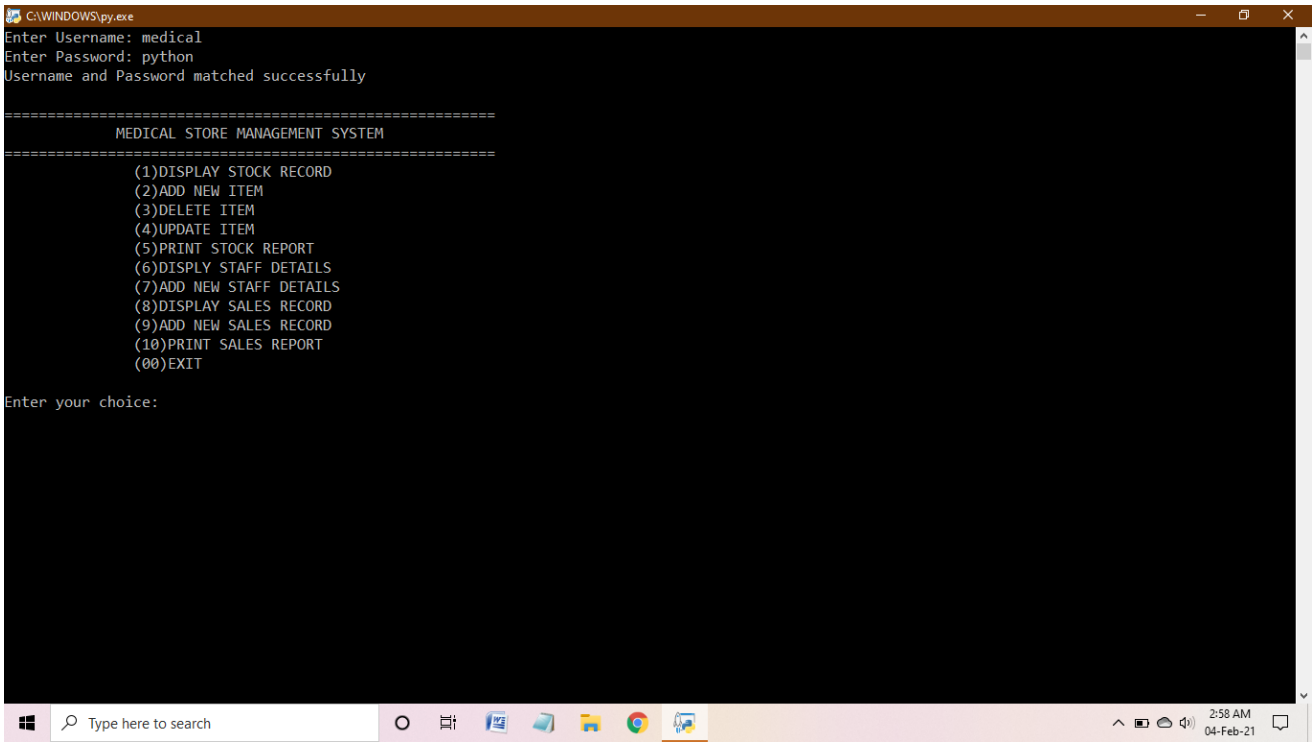
```
def login():
    uname=input("Enter Username: ")
    pwd=input("Enter Password: ")
    df=pd.read_csv("C:\\Users\\welch\\Desktop\\users.csv")
    df=df.loc[df["username"]==uname]
    if df.empty:
        print("Invalid Username given")
        return False
    else:
        df=df.loc[df["password"]==pwd]
        if df.empty:
            print("Invalid Password")
            return False
        else:
            print("Username and Password matched successfully")
            return True
```

```
def menu():
    print()
    print("============================================================")
    print("                    MEDICAL STORE MANAGEMENT SYSTEM                 ")
    print("============================================================")
    print("                          (1)DISPLAY STOCK RECORD                   ")
    print("                          (2)ADD NEW ITEM                           ")
    print("                          (3)DELETE ITEM                            ")
    print("                          (4)UPDATE ITEM                            ")
    print("                          (5)PRINT STOCK REPORT                     ")
    print("                          (6)DISPLAY STAFF DETAILS                  ")
    print("                          (7)ADD NEW STAFF DETAILS                  ")
    print("                          (8)DISPLAY SALES RECORD                   ")
    print("                          (9)ADD NEW SALES RECORD                   ")
    print("                          (10)PRINT SALES REPORT                    ")
    print("                          (00)EXIT                                  ")
    print()
    choice=int(input("Enter your choice: "))
    return choice
```

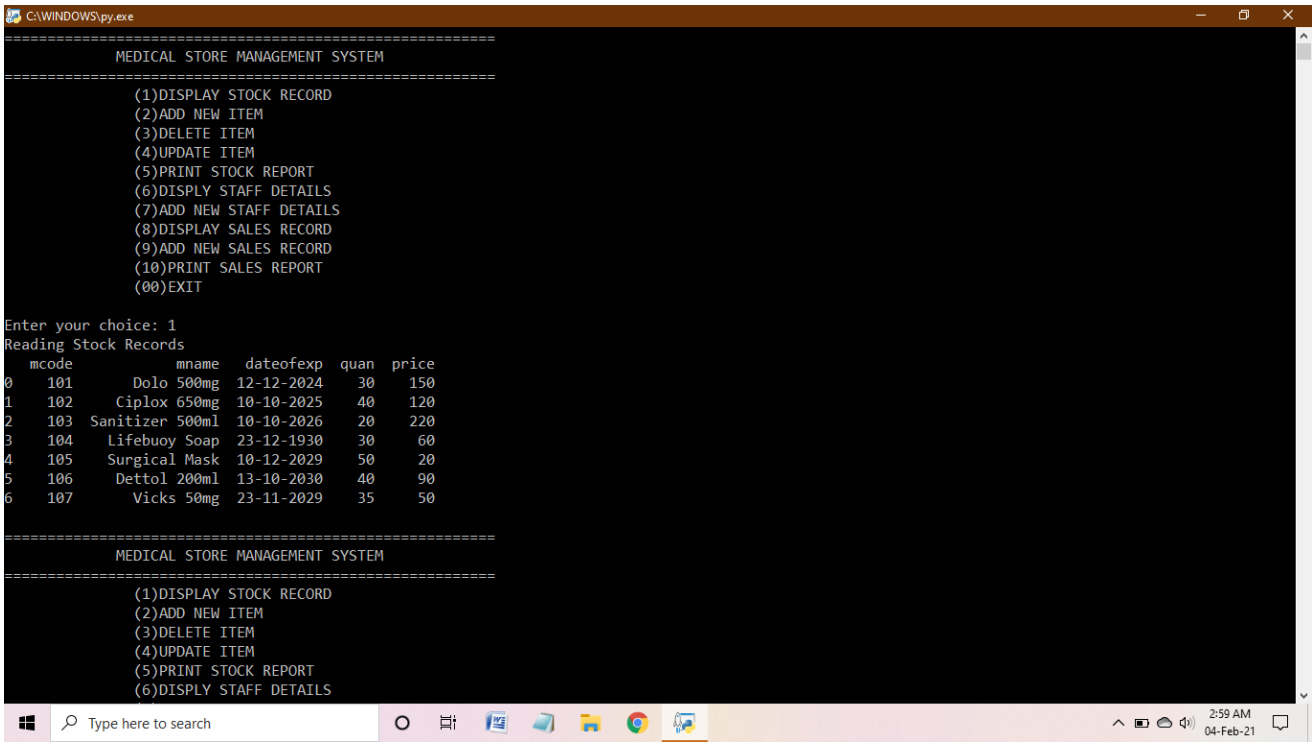
```
if login():
    while True:
        opt=menu()
        if opt==1:
            stockrec()
        elif opt==2:
            newmed()
        elif opt==3:
            removemed()
        elif opt==4:
            updatemed()
        elif opt==5:
            stockplot()
        elif opt==6:
            staffrec()
        elif opt==7:
            newstaff()
        elif opt==8:
            salesrec()
        elif opt==9:
            newsales()
        elif opt==10:
            salesplot()
        elif opt==00:
            break
        else:
            print("Invalid Option Selected")
```

OUTPUT

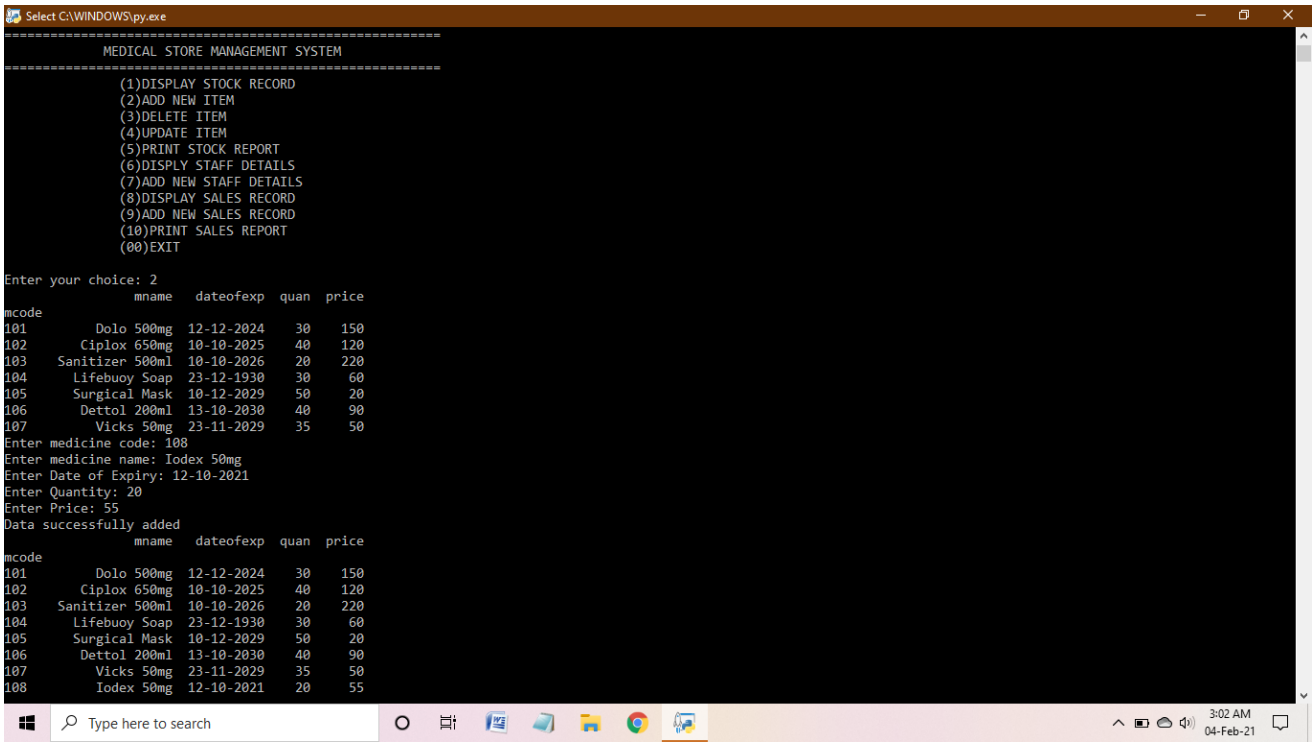
Press F5 for output: *username* and *password* entered correctly



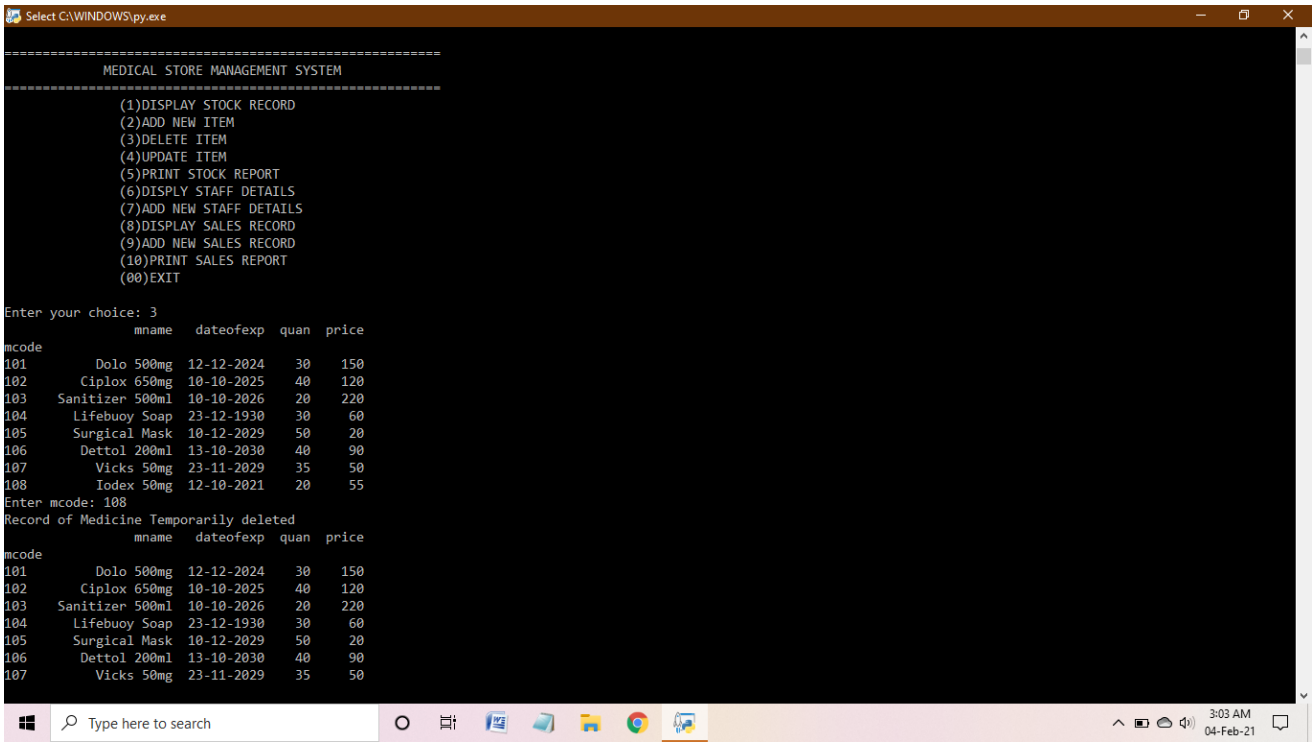
Press 1: Reading the stock records from *mstock* CSV file



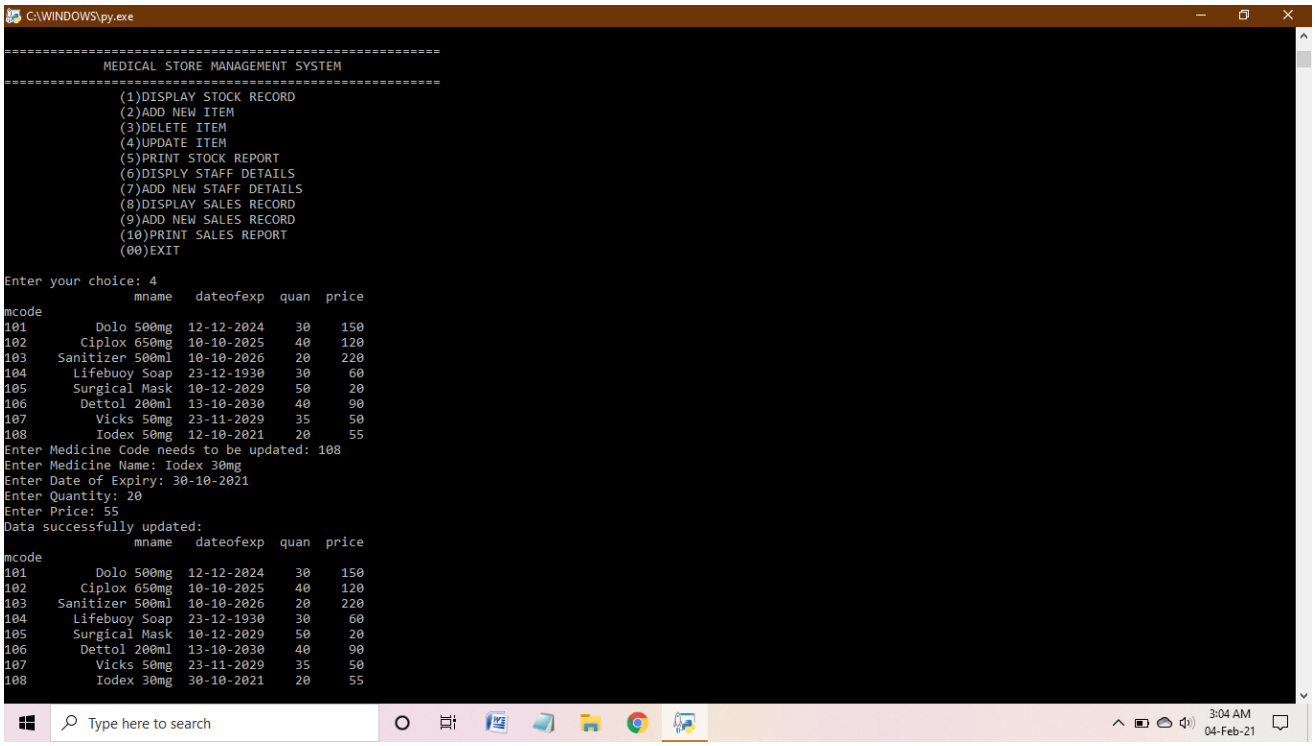
Press 2: Adding new item to the *mstock* CSV file



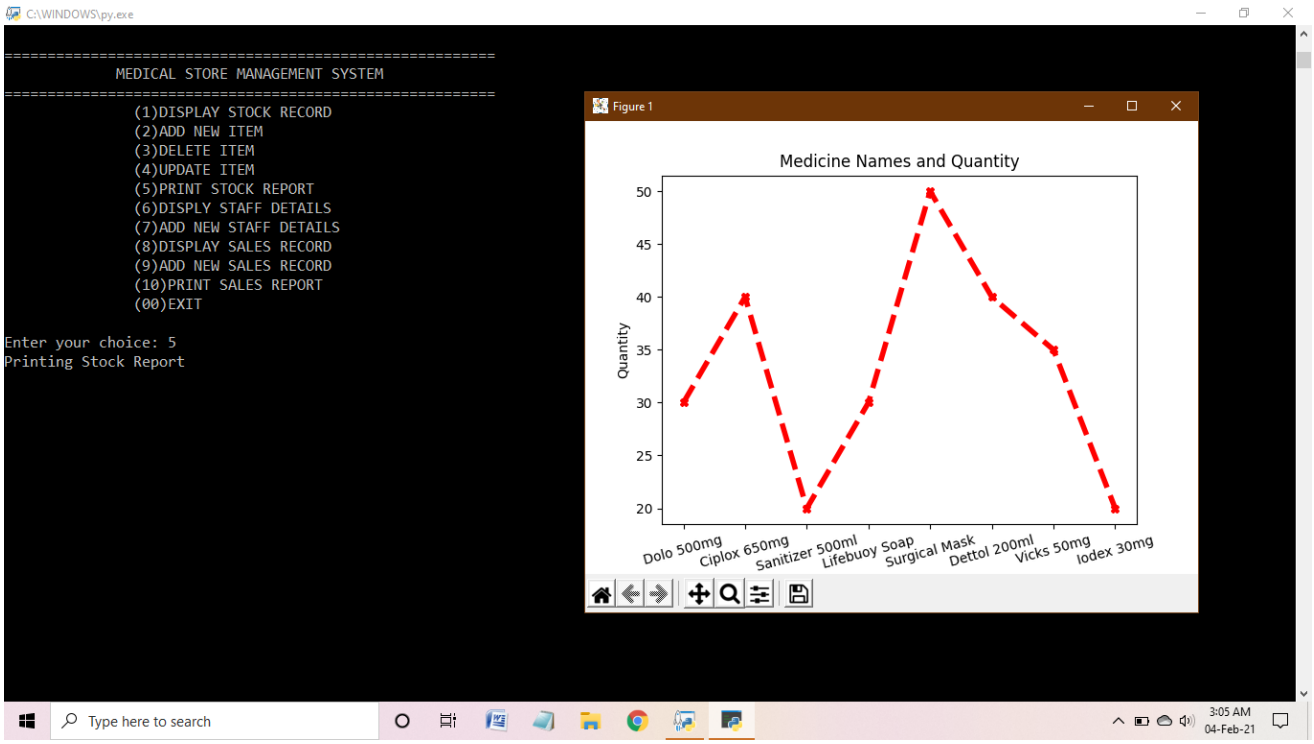
Press 3: Updating the previously entered records in *mstock* CSV file



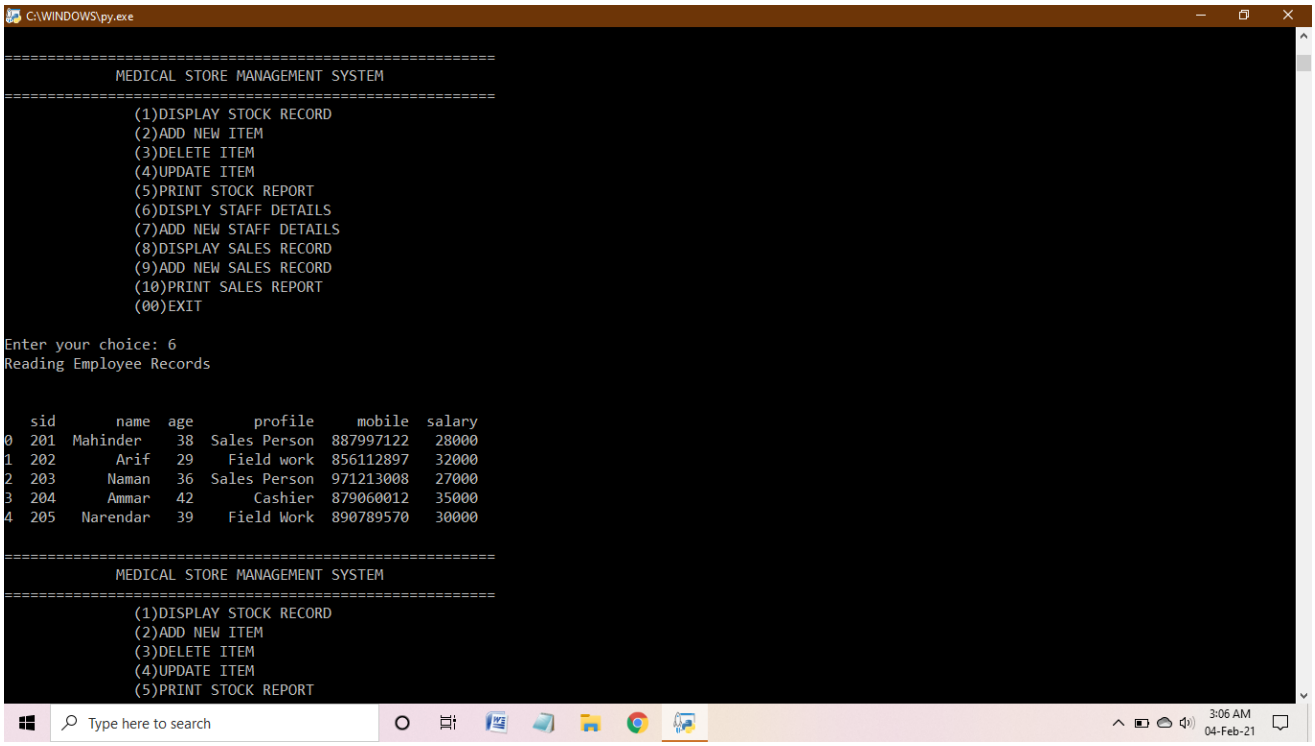
Press 4: Updating the previously entered records in *mstock* CSV file



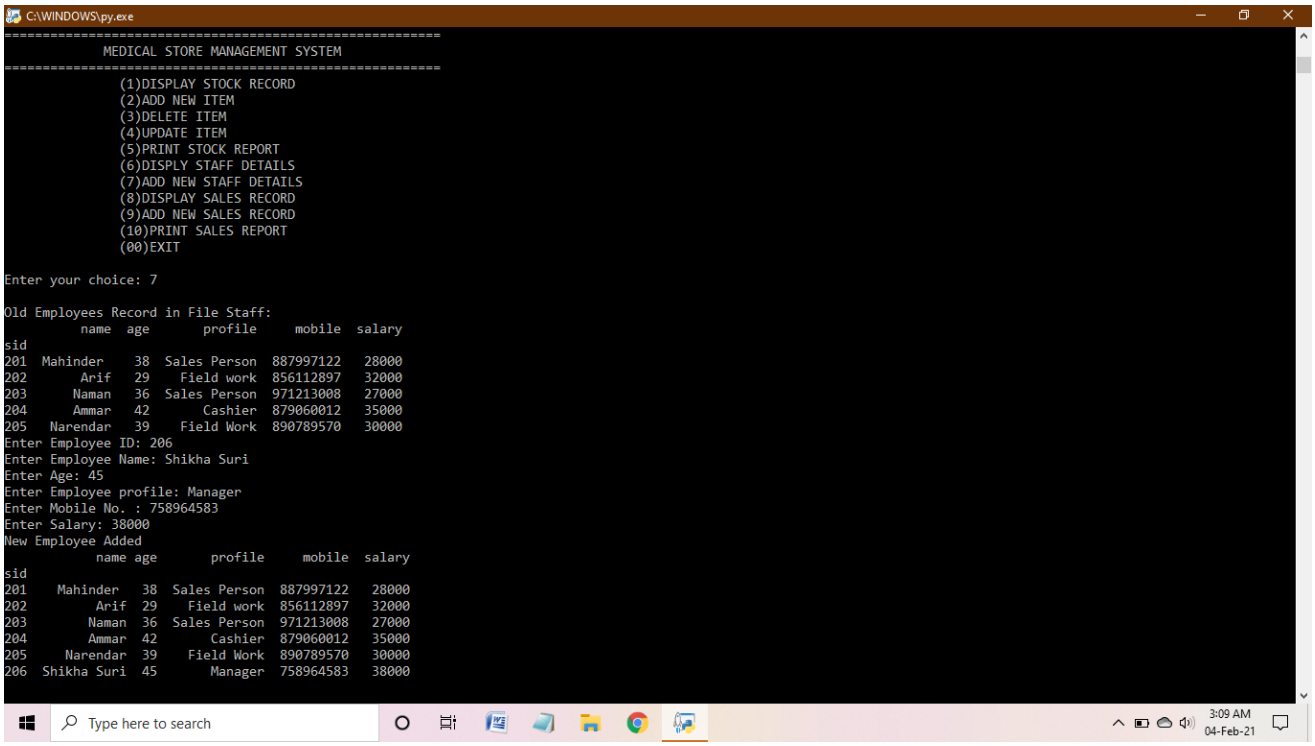
Press 5: Showing the line plot of stock records from *mstock* CSV file



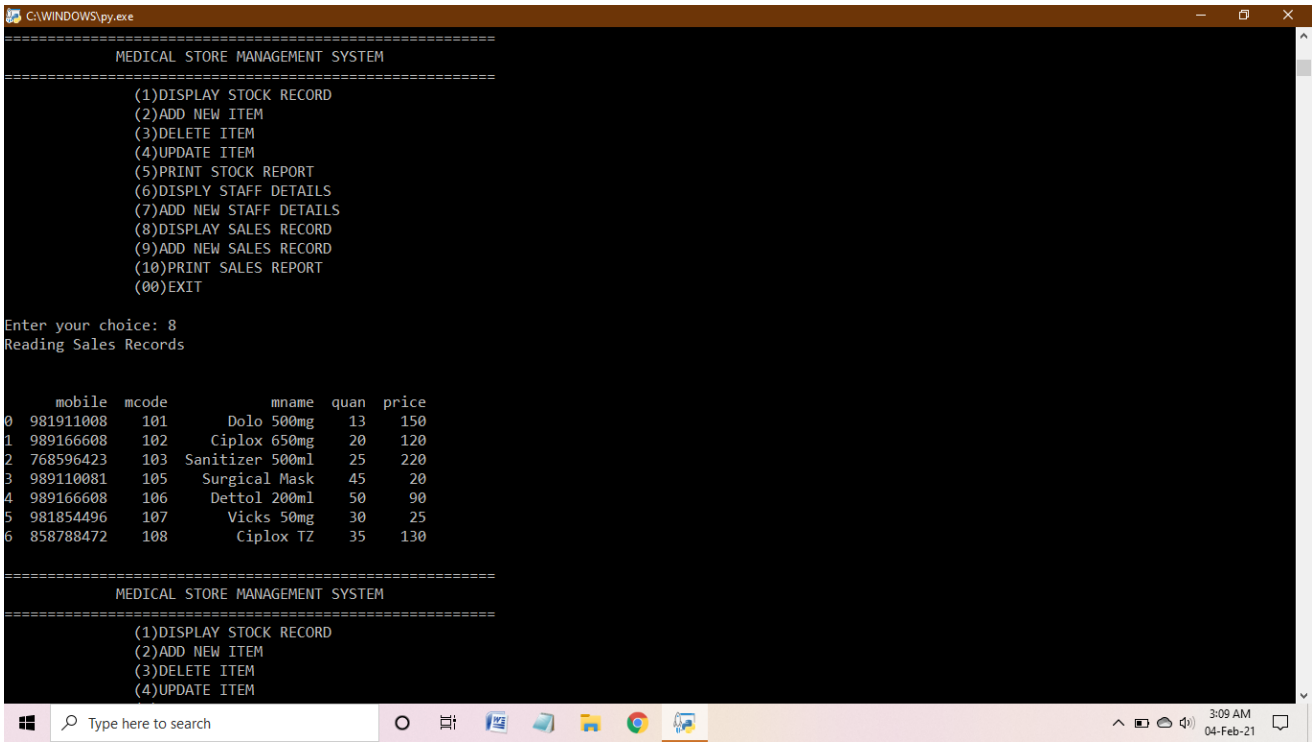
Press 6: Reading the employee records from *mstaff* CSV file



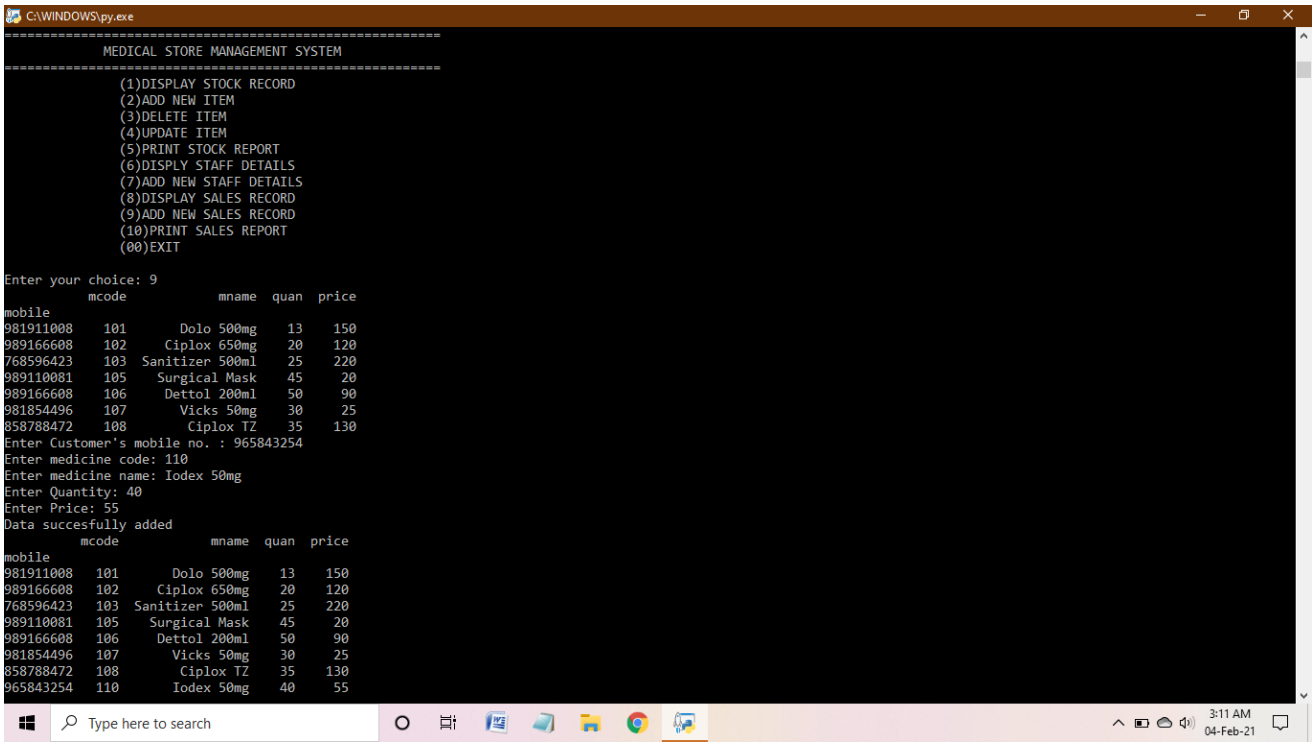
Press 7: Adding a new employee record to the *mstaff* CSV file



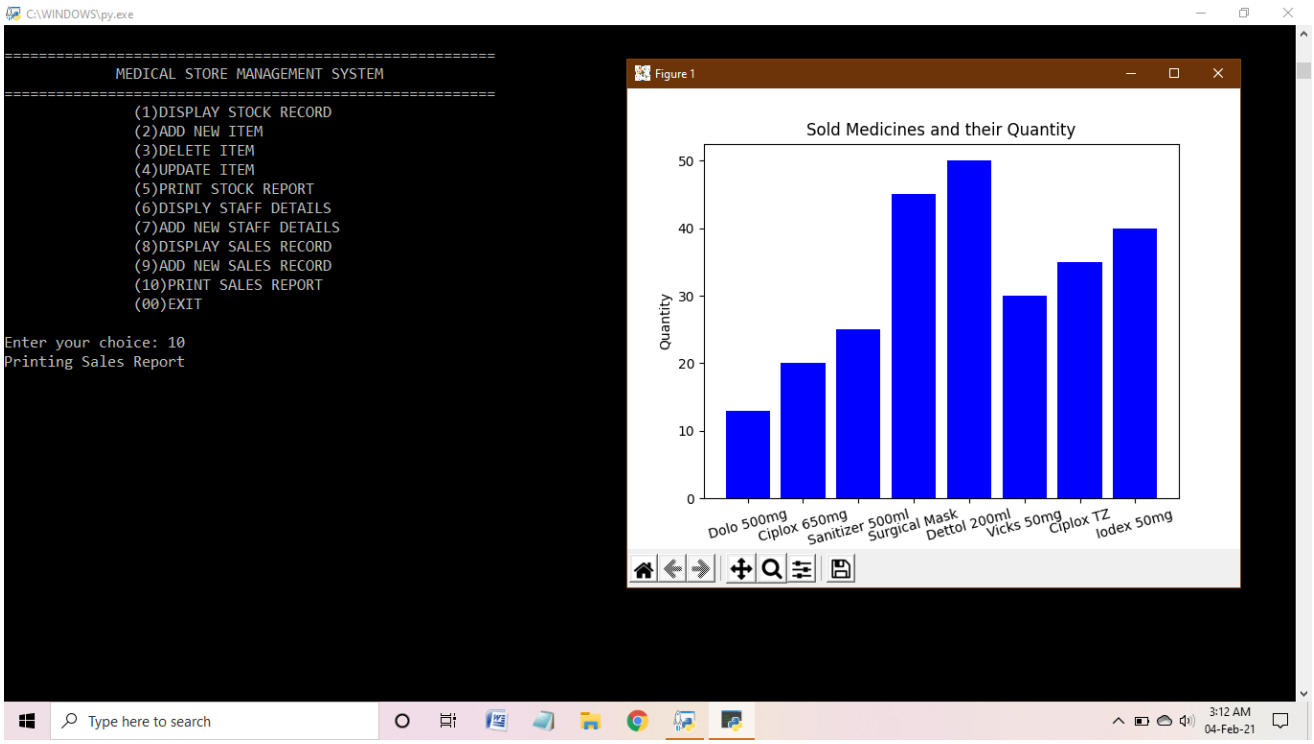
Press 8: Reading the sales records from *msales* CSV file



Press 9: Adding a new sales record to the *msales* CSV file



Press 10: Showing the bar graph of sales records from *msale* CSV file



CONCLUSION

This project has a wider scope in future and can be helpful to many. Since this project is made in python, it reduces manual work and with the help of CSV files, the system was able to store and update the database with more ease.

This project also clears the functions of various python commands. It also shows the application of python's different libraries.

Thus, the project "*Medical Store Management System*" can also be altered in accordance with the future requirements of the organization.

BIBLIOGRAPHY

- Informatics Practices Textbook (Sumita Arora)
- www.youtube.com