**INTRODUCTION**

This software project is to develop the automate the functionalities of a bakery store. The purpose of software projects to develop the Bakery Management System (B.M.S) to automate the record of the different items, sales, graphical analysis, profit with a view to enhance the decision making of the Functionaries.

A BMS is mainly consist of a computerized database, a collection of interrelated tables for a particular subject or purpose, capable to produce different reports relevant to the user. An application program is tired with the database for easy access and interface to database. Using application program or front end, we can store, retrieve and manage all information in proper way.

This software being simple in design and working, does not require much of training to users and can be used as a powerful tool for managing a big bakery.

During coding and design of the software project, Python IDLE, a powerful front end tool is used for getting integrated platform and coding simplicity. As a back end a powerful My SQL is used as per requirement for CBSE curriculum of informatic practices course.

**PROBLEMS OF CURRENT MANUEL SYSYTEM**

Bakeries which have several branches and has expanded business model faces lot of problems regarding to it sales report, profit, stock left, etc. It lacks an organised management that can run their business with transparency.it becomes difficult for bakery which has branches all over the state to supervise every branch of it.

Staff working in this bakery also faces difficulty to serve their customers properly as it becomes almost impossible to remember the product availability in such a huge bakery, as a result even if the product is available then also the customer leaves the shop unsatisfactory.

Many such problem are faced by bakery daily, so to vanish this problem a customized bakery management system is needed in order to run the bakery in a proper organised manner.

**BENEFITS OF THE COMPUTERIZED SYSTEM**

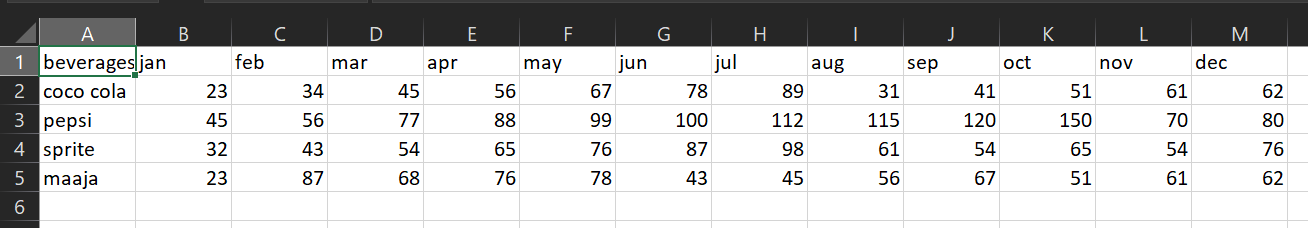
Problem definition and analysis of the activity that encompasses. Earning about the problem to be solved, understanding the needs of customer and users, trying to find out who the user really is and understanding all the constraints on the solution. It includes all activities related to the following.

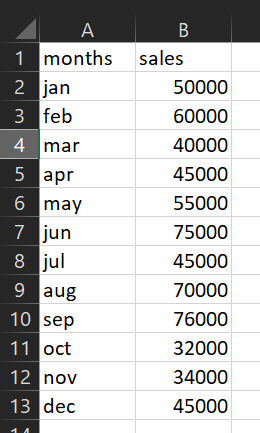
* Identification and documents of customer’s or user’s need.
* Creation of a document that describes the external behaviour and association constants that will satisfy those needs.
* Analysis and validation of the requirements documents to ensure consistency, completeness and feasibility.
* Evolution of needs.

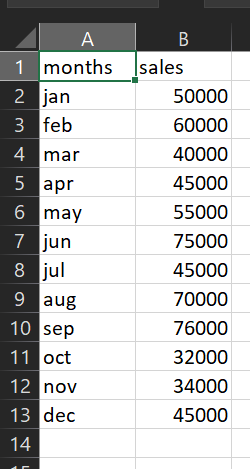
After the analysis of functioning of a bakery store, the. Proposed System is expected to the following.

* To provide a user friendly, best integrated and centralised environment for computerized bakery store.
* The proposed system should maintain all the records and transaction and should generate the required report and information when required.
* To provide efficient and secured information storage flow and retrieval system, ensuring the integrity and validity of records.
* To provide user friendly interface to interact with a centralised database based on client server architecture.
* To identify the critical operation, procedure and possibilities of simplification using modern IT tools and practices.

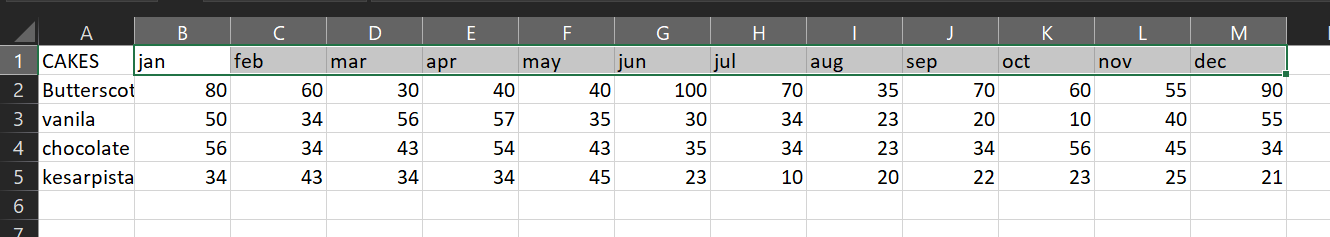
**SAMPLE DATA OF EXCEL FILE**



B1.CSV

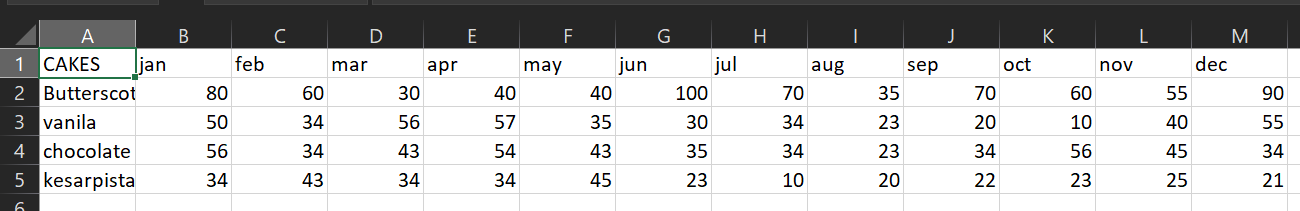


Book1.xlsx Book11.csv



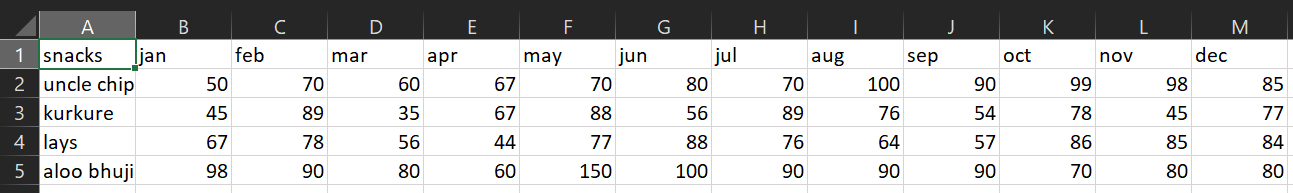
Book2.xlsx

**SAMPLE DATA OF EXCEL FILE**



ca

cakes.csv



snacks.csv

**SOURCE CODE**

import time,sys,os

msg="WELCOME TO THE BAKERY MANAGEMENT SYSTEM"

for char in msg:

sys.stdout.write(char)

time.sleep(0.05)

time.sleep(0.3)

import pandas as pd

import matplotlib.pyplot as plt

print()

n1=0

n7=input('Enter Your Name:\n')

n8='0'

while n8!='qwerty':

n8=input("Enter the Password:\n")

if n8=='qwerty':

while n1!=6:

print("------------------------------------------------------------------------")

print("'1'--> To Get the Sales Report")

time.sleep(0.2)

print("'2'--> To Get the List of Items in Shop")

time.sleep(0.2)

print("'3'--> To Update the List of Items in Shop")

time.sleep(0.2)

print("'4'--> Profit Calculator")

time.sleep(0.2)

print("'5'--> To Get the Maximum,Minimum,Total of the Sales")

time.sleep(0.2)

print("'6'--> To Exit The System")

time.sleep(0.2)

n1=int(input("Enter the Number Assigned To Above Options:\n"))

if n1==1:

print("'1'--> To Get Cakes Sales")

time.sleep(0.2)

print("'2'--> To Get Snacks Sales")

time.sleep(0.2)

print("'3'--> To Get Beverages Sales")

time.sleep(0.2)

print("'4'--> To Get Net Sales")

time.sleep(0.2)

n9=int(input("Enter the Number Assigned to Above Options:\n"))

if n9==1:

df=pd.read\_csv("cakes.csv")

df.plot(kind='bar',color=['red','blue','yellow','green','cyan','pink','magenta','black','white'],edgecolor='Black',linewidth=1,width=0.8)

ticks = df.index.tolist()

plt.xticks(ticks,df.CAKES)

plt.title('MONTHLY CAKES SALES REPORT')

plt.ylabel('Sales(in Rs)')

plt.xlabel('Cakes')

plt.grid(True)

plt.show()

elif n9==2:

df=pd.read\_csv("snacks.csv")

df.plot(kind='bar',color=['red','blue','yellow','green','cyan','pink','magenta','black','white'],edgecolor='Black',linewidth=1,width=0.8)

ticks = df.index.tolist()

plt.xticks(ticks,df.snacks)

plt.title('MONTHLY SNACKS SALES REPORT')

plt.ylabel('Sales(in Rs)')

plt.xlabel('Snacks')

plt.grid(True)

plt.show()

elif n9==3:

df=pd.read\_csv("B1.csv")

df.plot(kind='bar',color=['red','blue','yellow','green','cyan','pink','magenta','black','white'],edgecolor='Black',linewidth=1,width=0.8)

ticks = df.index.tolist()

plt.xticks(ticks,df.beverages)

plt.title('MONTHLY CAKES SALES REPORT')

plt.ylabel('Sales(in Rs)')

plt.xlabel('Beverages')

plt.grid(True)

plt.show()

else:

df=pd.read\_csv("Book11.csv")

df.plot(kind='bar',color='Yellow',edgecolor='Black',linestyle='dashed',linewidth=1)

ticks = df.index.tolist()

plt.xticks(ticks,df.months)

plt.title('MONTHLY SALES REPORT')

plt.ylabel('Sales(in Rs)')

plt.xlabel('Months')

plt.grid(True)

plt.show()

elif n1==2:

print("'1'--> CAKES")

time.sleep(0.2)

print("'2'--> SNACKS")

time.sleep(0.2)

print("'3'--> BEVERAGES")

time.sleep(0.2)

n2=int(input("Enter the Number Assigned To Above Options:\n"))

if n2==1:

Cakes=pd.read\_csv("cakes.csv")

print(Cakes)

elif n2==2:

Snacks=pd.read\_csv("snacks.csv")

print(Snacks)

elif n2==3:

Beverages=pd.read\_csv("B1.csv")

print(Beverages)

else:

print('VALUE ERROR! PLEASE CHECK')

elif n1==3:

import os

print("'1'--> CAKES")

time.sleep(0.2)

print("'2'--> SNACKS")

time.sleep(0.2)

print("'3'--> BEVERAGES")

time.sleep(0.2)

n3=int(input("Enter The Number Assigned To Above Options:\n"))

if n3==1:

os.system("start cakes.csv")

elif n3==2:

os.system("start snacks.csv")

elif n3==3:

os.system("start B1.csv")

else:

print('VALUE ERROR! PLEASE CHECK')

elif n1==4:

print("'1'--> To Get the Net Profit")

time.sleep(0.2)

print("'2'--> To Get Profit Per Item")

time.sleep(0.2)

n3=int(input("Enter The Number Assigned To Above Options:\n"))

if n3==1:

c1=int(input("Enter the Number of Cakes Sold:\n"))

c2=int(input("Enter the Number of Snacks Sold:\n"))

c3=int(input("Enter the Number of Beverages Sold:\n"))

print("Net Profit= Rs",(c1\*50)+(c2\*20)+(c3\*60))

elif n3==2:

print("'1'--> CAKES")

time.sleep(0.2)

print("'2'--> SNACKS")

time.sleep(0.2)

print("'3'--> BEVERAGES")

time.sleep(0.2)

n4=int(input("Enter The Number for Which you Want to Get the Profit:\n"))

if n4==1:

n5=int(input("Enter the Number of Cakes Sold:\n"))

print("Net Profit is=",n5\*50)

elif n4==2:

n6=int(input("Enter the Number of Snacks Sold:\n"))

print("Net Profit is=",n6\*20)

elif n4==3:

n7=int(input("Enter the Number of Beverages Sold:\n"))

print("Net Profit is=",n7\*60)

else:

print('VALUE ERROR! PLEASE CHECK')

elif n1==5:

df = pd.read\_csv("Book11.csv")

print("Net sales:Rs",df["sales"].sum())

print("Maximum Sales with Month:\n",df.max())

print("Minimum Sales With Month:\n",df.min())

elif n1==6:

print("THANK YOU",n7,"FOR USING BMS.......")

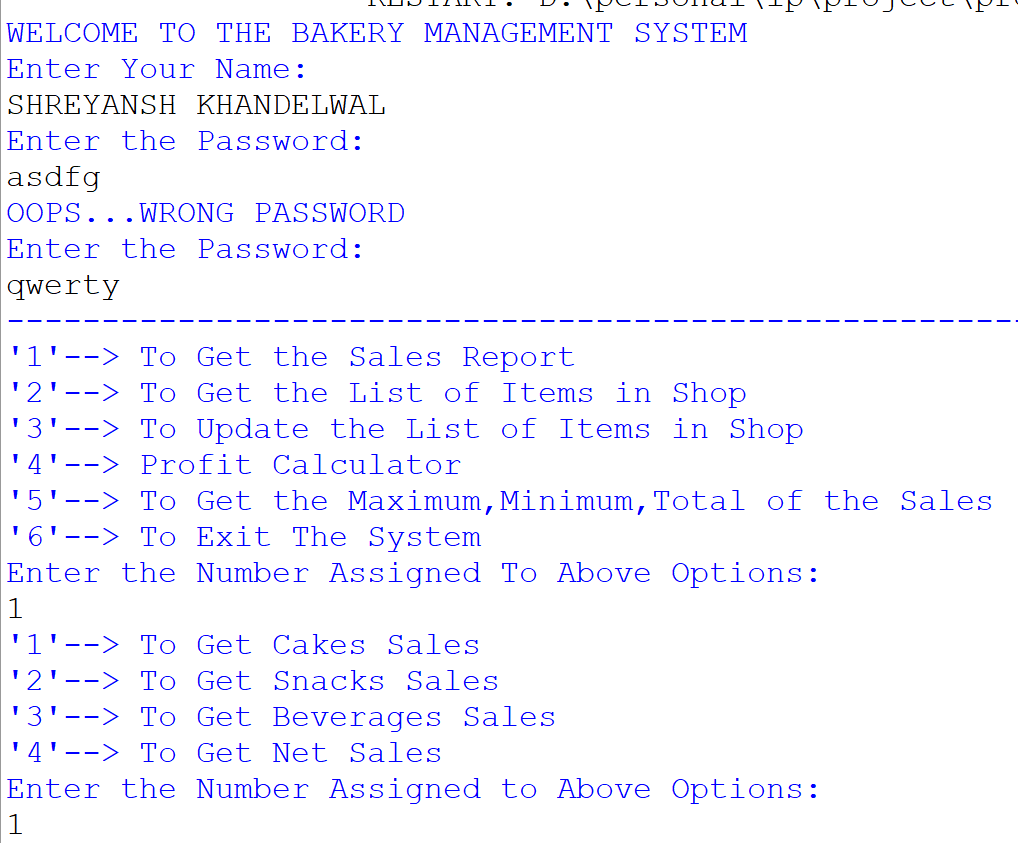
else:

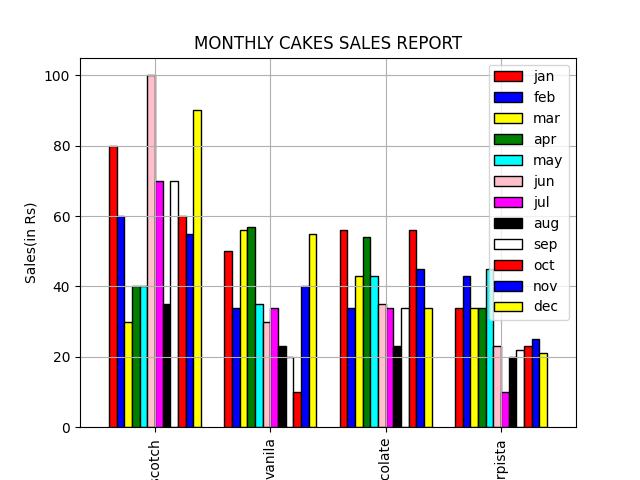
print('VALUE ERROR! PLEASE CHECK')

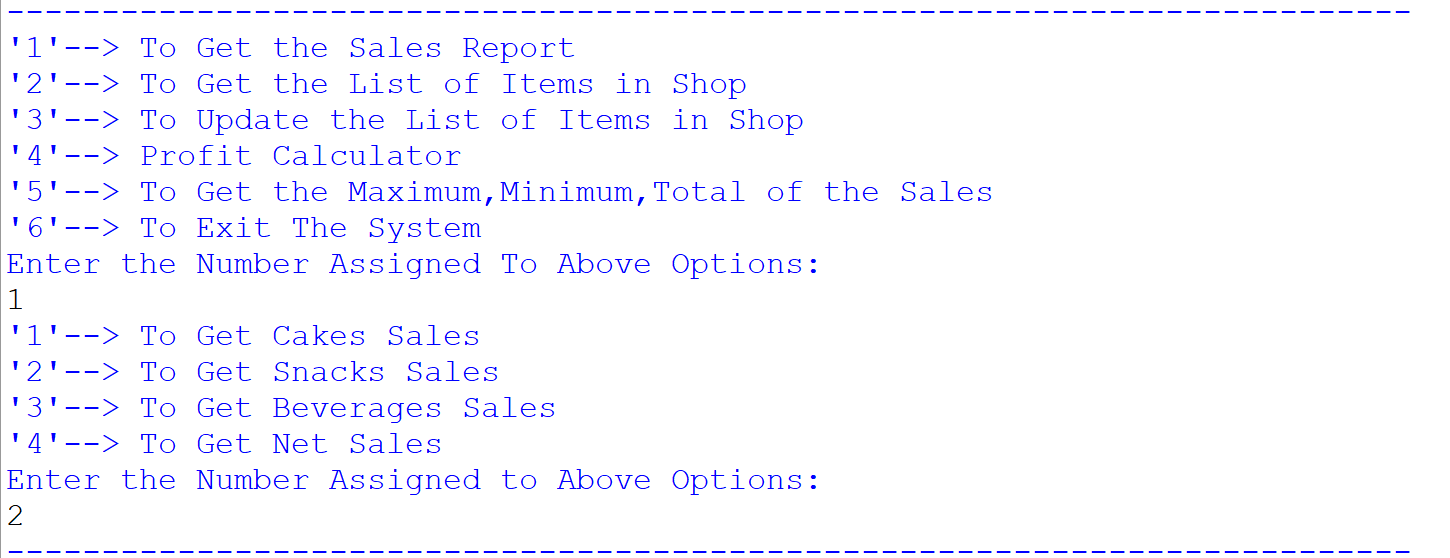
else:

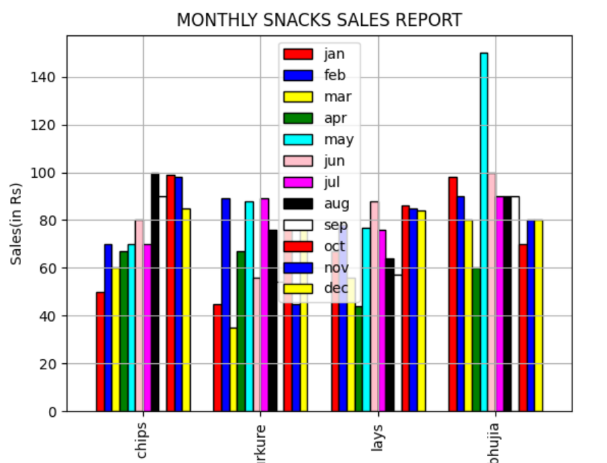
print('OOPS...WRONG PASSWORD')

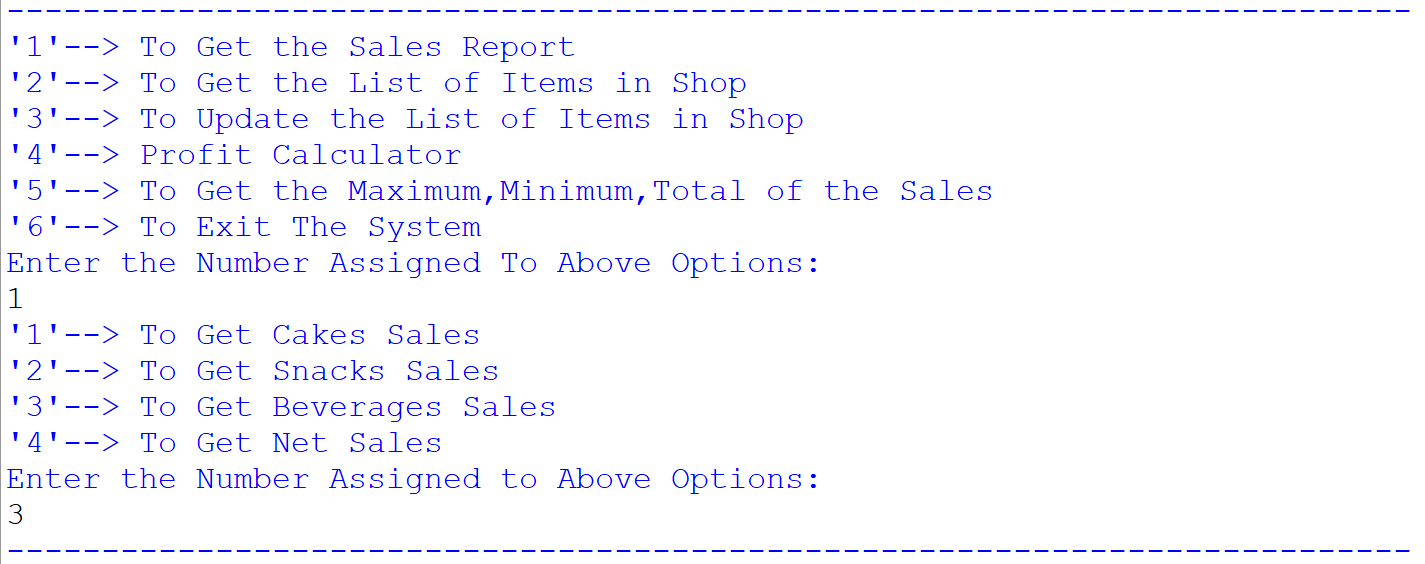
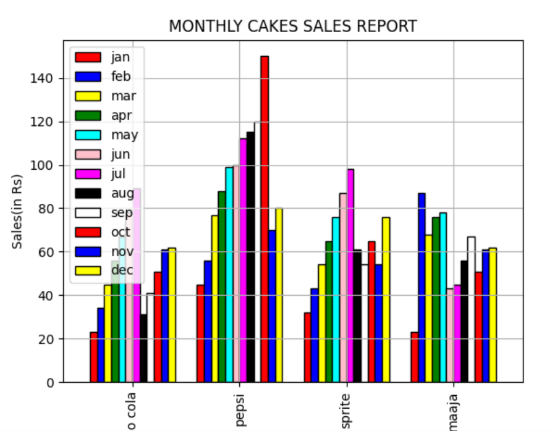
**OUTPUT SCREEN LAYOUTS**

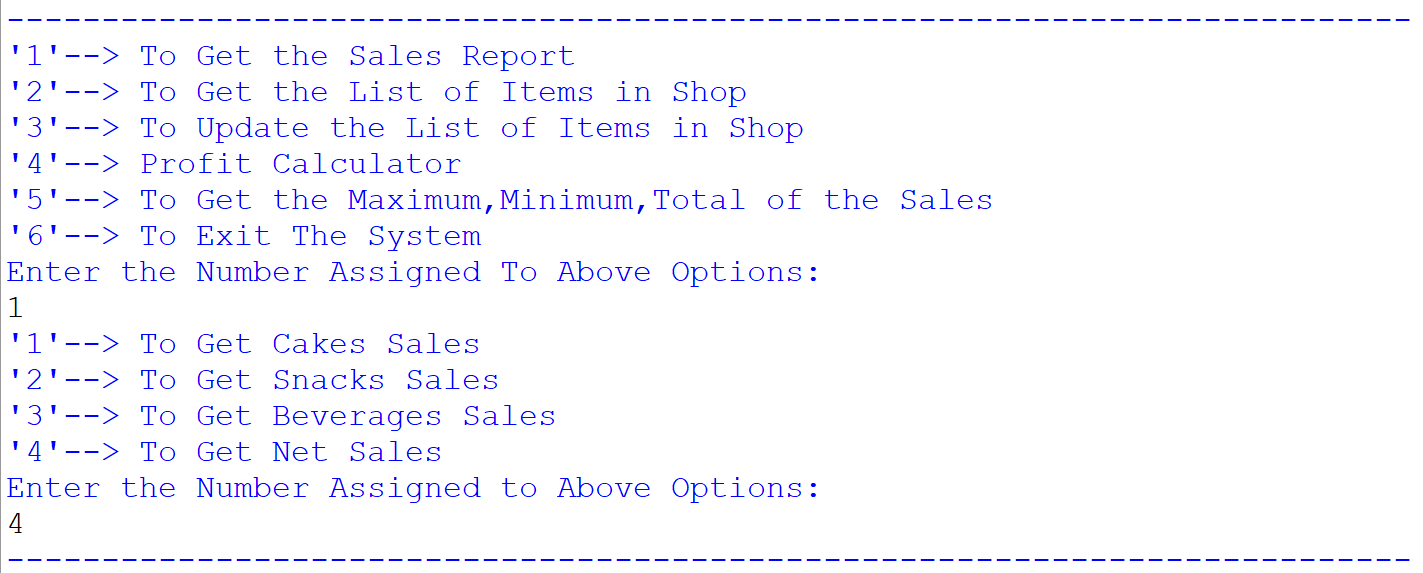
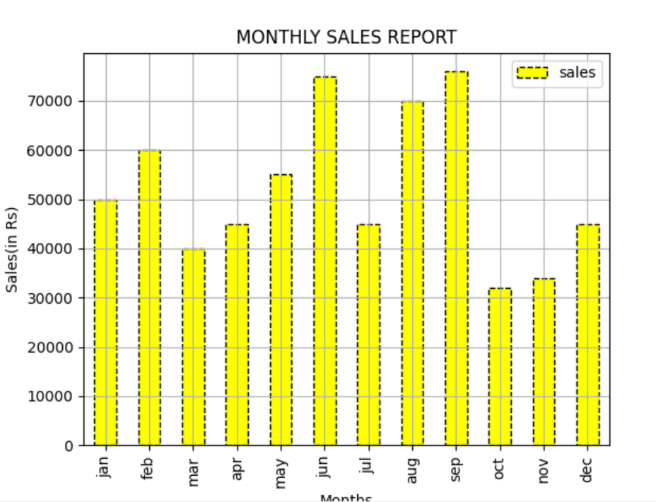


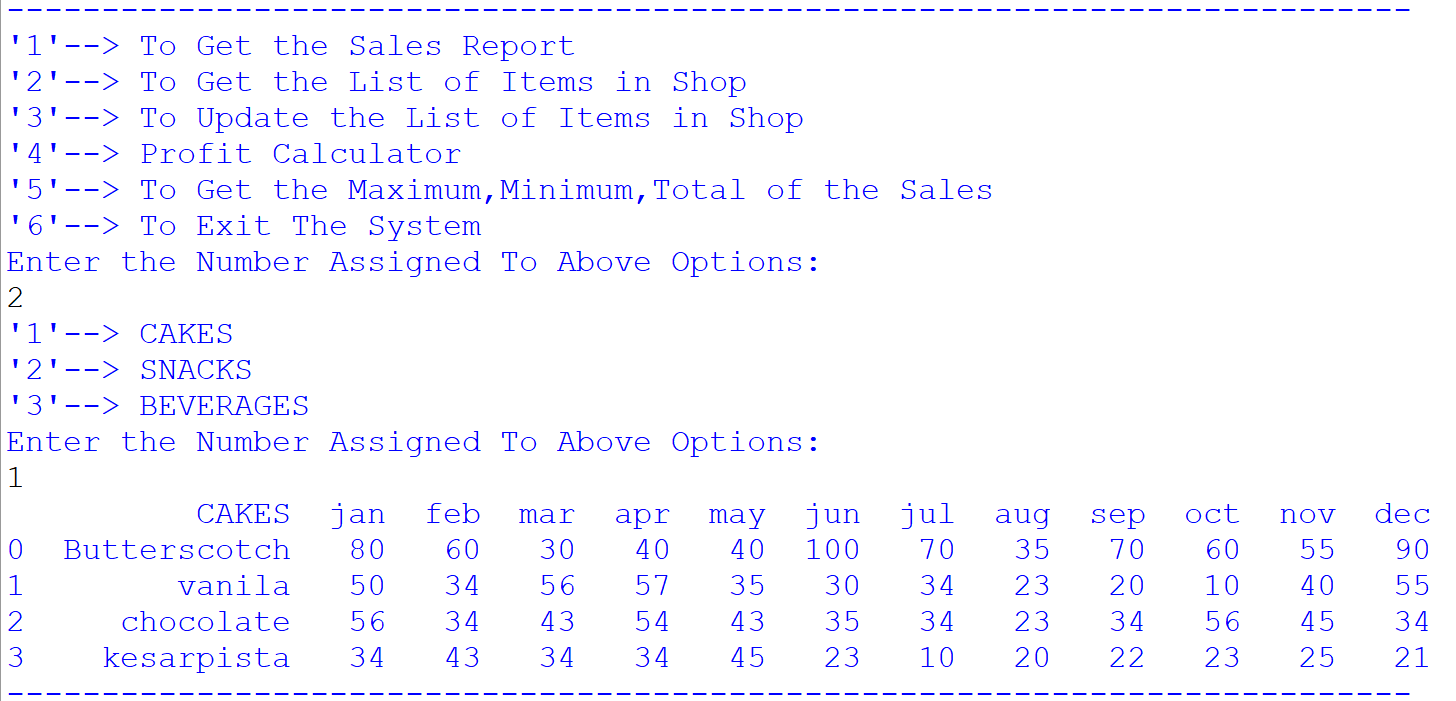


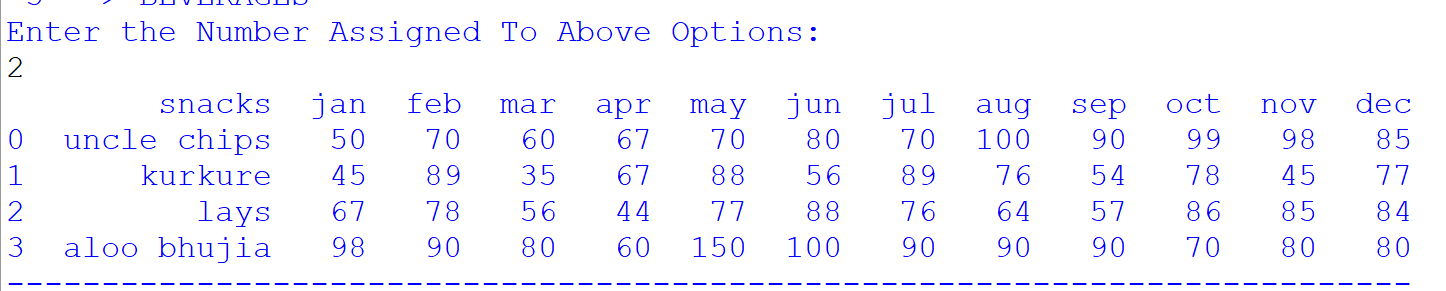


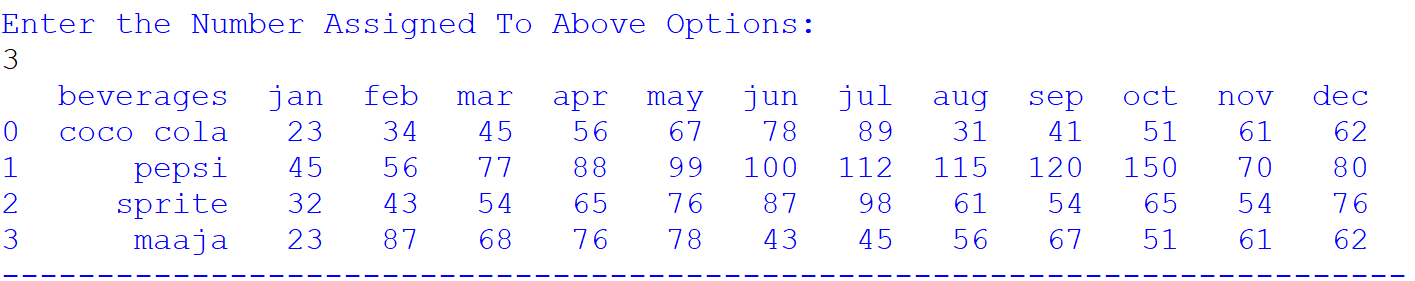


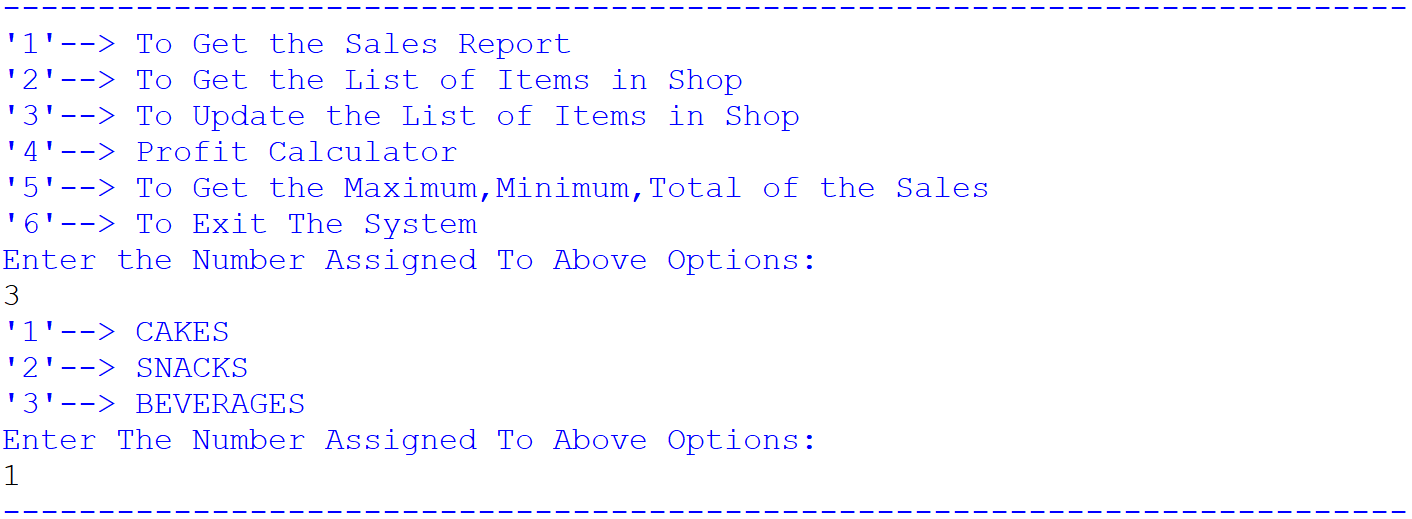


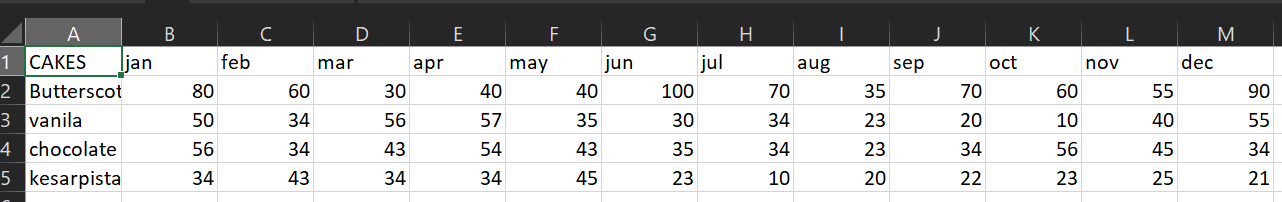


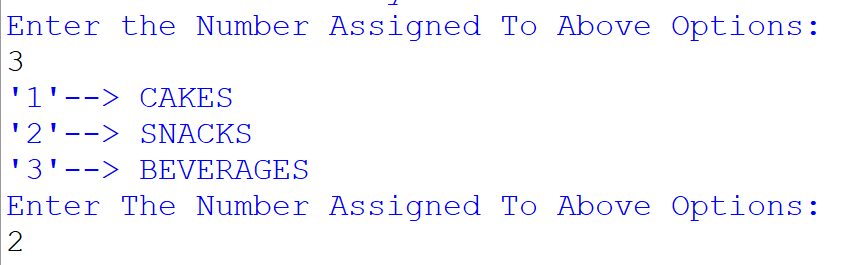


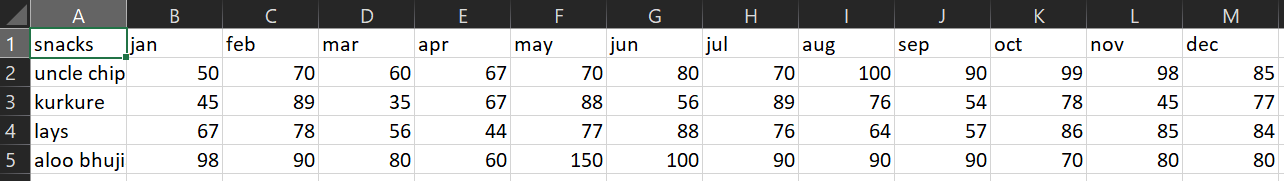


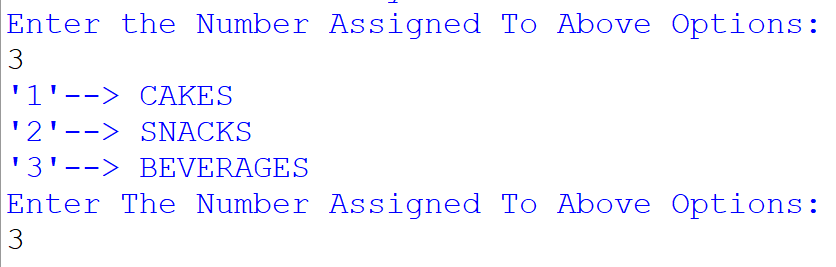


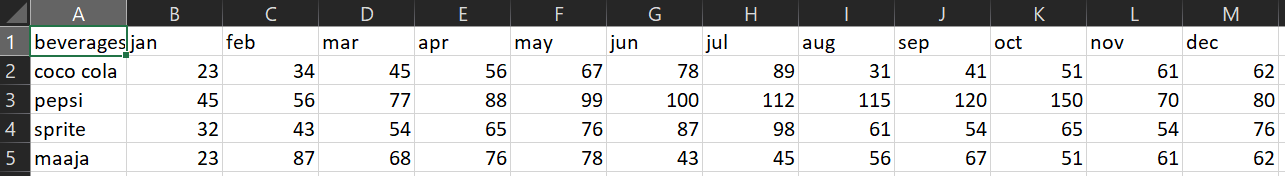


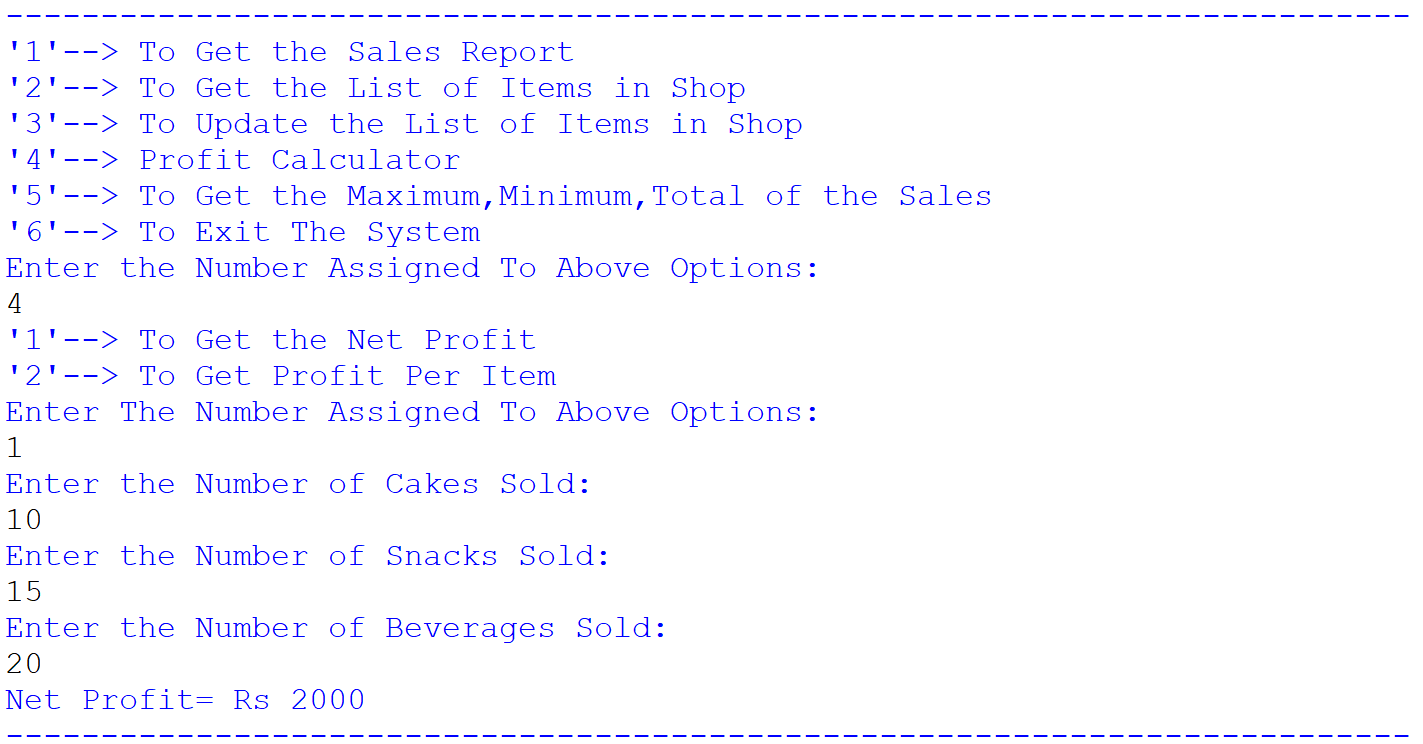


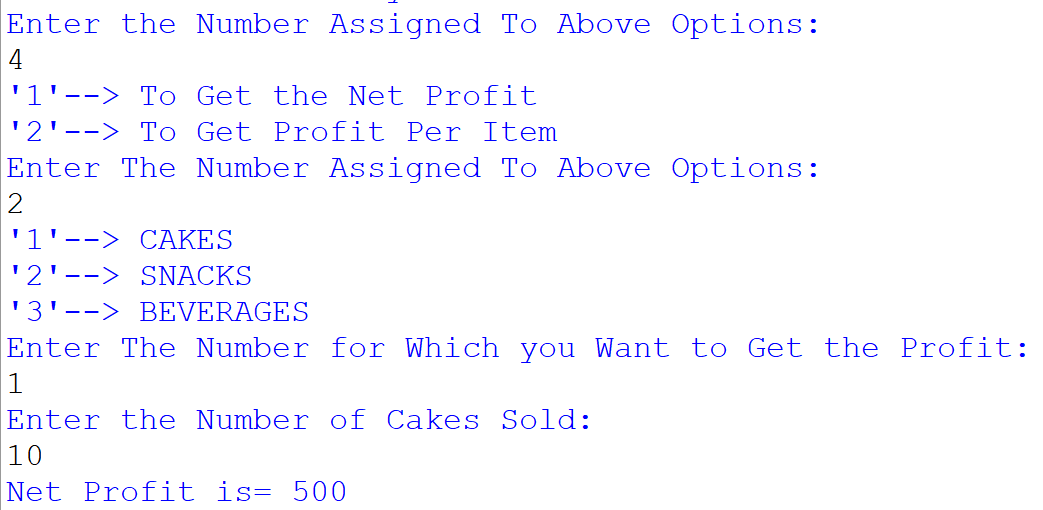


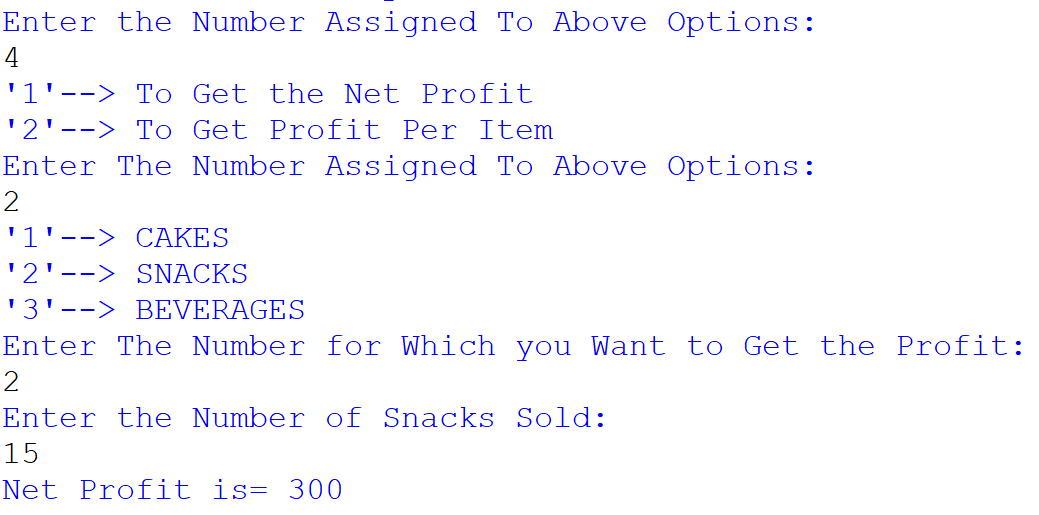


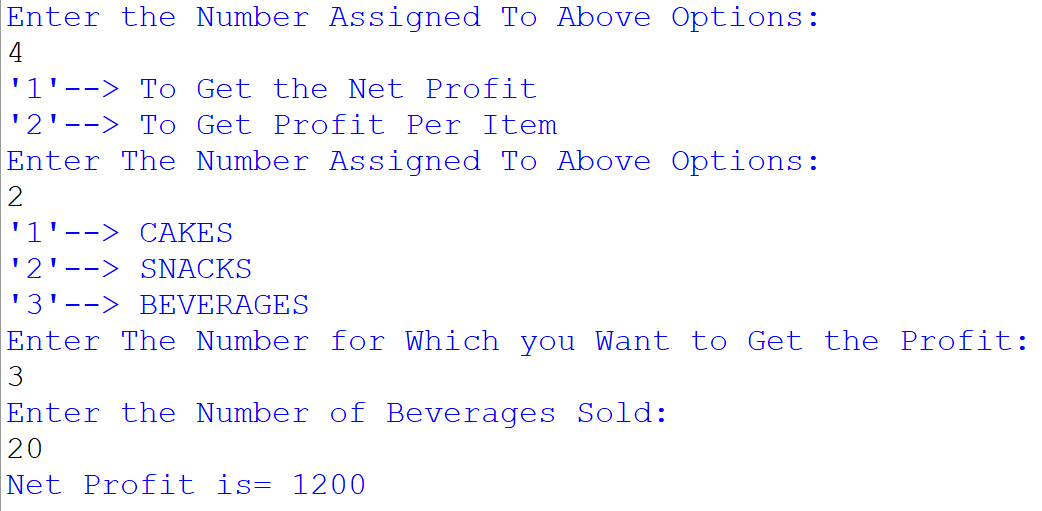


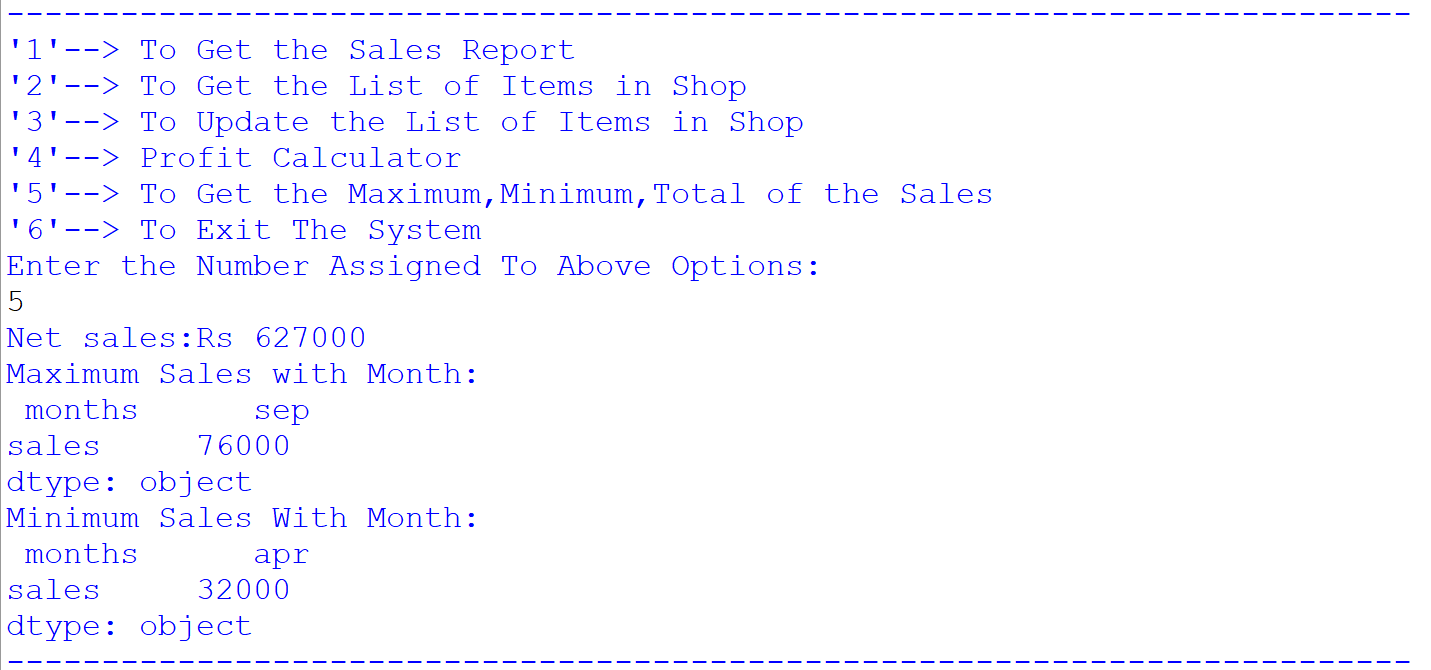


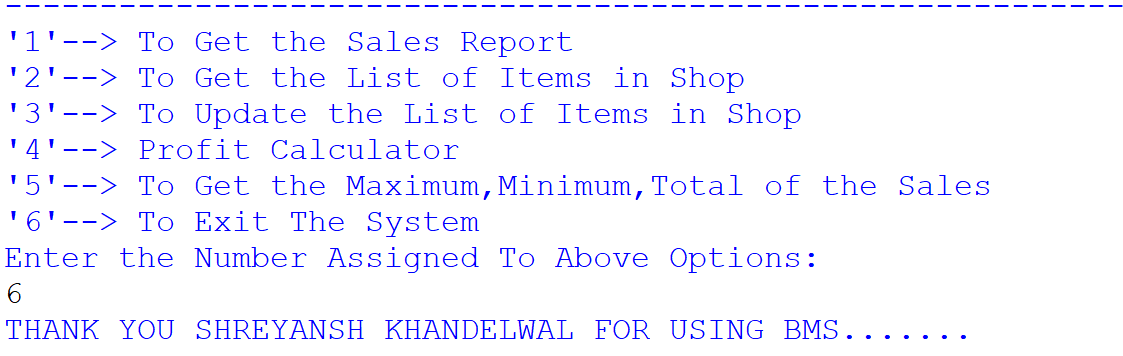












**BIBLOGRAPHY**

* Stacksoverflow.com
* GEEKSFORGEEKS.com

* Youtube.com
* IP NCERT
* Learncode.com