COMPUTER SCIENCE PROJECT WORK

SUPERMARKET INVENTORY AND SALE

SHREYAS KHANDEKAR

ROLL 9202624

2019
MODERN SCHOOL
BARAKHAMBA ROAD

INDEX

- Acknowledgement
- Certificate
- About the Project and its uses
- Technical Details of the project
- Source Code
- Sample Outputs
- Scope of Improvement
- Bibliography

ACKNOWLEDGEMENT

TO WHOM SO EVER IT MAY CONCERN

I convey my sincere gratitude to Mr Gautam Sarkar for his continuous guidance and help. Without his kind support, the completion of the project would not have been possible. I also convey my sincere thanks Ms Divya Sahdev for her support and guidance.

SHREYAS KHANDEKAR
CLASS S7D
BOARD ROLL NO.

CERTIFICATE

TO WHOM SO EVER IT MAY CONCERN

This is to certify that the project on Supermarket Inventory and Sale has been designed and developed by Shreyas Khandekar under my guidance and supervision.

GAUTAM SARKAR
HEAD, COMPUTER DEPARTMENT
MODERN SCHOOL, BARAKHAMBA ROAD

ABOUT THE PROJECT AND ITS USES

WHY THIS PROJECT

This project was taken up by me to try to challenge myself by attempting to recreate a piece of software that we see being used almost every day.

It encompasses everything that I have learned in C++ and thus I have implemented my knowledge of stacks and queues, coupled with an ability to visualize the graphics screen and understanding of loops, file handling and classes.

SALIENT FEATURES

The project employs many key features that make it a really implementation oriented program which can actually be used in any supermarket. A few of them are:

- Updating Inventory
    - Adding and Deleting Items
    - Modifying Items
        - Changing Name
        - Changing Price
    - Searching for Items
        - By Name
        - By Barcode
        - By Quantity in Stock
- Transactions
    - Deleting from and Adding to Cart

    - Displaying available Items
    - Check Out and Billing
- Declaring Sales and discounts
    - Increase price of commodities in demand
    - Decrease price of Surplus commodities

And many more such features which make it almost ready to be a base for higher software.

USES

This project can be implemented in any supermarket today.

Since the requirement of such software which are easy to use and efficiently manage and protect the data of big retail chains is ever increasing, there is an actual real life application to this project.

TECHNICAL DETAILS OF THE PROJECT

BLUEPRINT FOR THE PROJECT

While developing the project, I decided to break down the source code into three distinct parts in order to achieve a clutter free layout-

1. *Inventory Update*
2. *Transaction/Sale of Goods*
3. *Declaring Discounts*

Further each part was compartmentalized and given a separate menu for its actions with dedicated functions for each of the actions mentioned in the SALIENT FEATURES . The Main menu enables us to choose a service from the three parts and takes us to the suitable sub menu of the action.

MAKING THE PROJECT

When I was writing the code to implement what I had in mind I also used subtle beautification techniques and also added an intro sequence to make the project look more amiable and welcoming.

For example: I used Empty loops to create delays and functions like gotoxy(,) and setw() in order to create a better presented look on the graphics screen

In addition to simple beautification I also used my knowledge of handling binary files to create a bill for all items that are in the cart when the customer checks out using the *CheckOut()* function.

- Displays all items and quantity of Items bought
- Calculates total amount and adds 18% **Goods and Services Tax (GST)** to produce the final amount that is to be paid.

FINAL RESULT

What emerged as the end product of this project was a program that is easy to use because of its intuitive architecture and ease of navigation coupled with a robust system to drive the code.

SOURCE CODE

```
#include<fstream.h>
#include<string.h>
#include<stdio.h>
#include<conio.h>
#include<iomanip.h>
#include<stdlib.h>

struct Item
{
	char name[40];
	long barcode;
	int price;
	int stock;



	void getdata();
	void showdata();
	int checkN(char nm[]);
	int checkB(long bc);
	int checkS(int st);
	void Edit();
};

//******************Input a new item***************
void Item::getdata()
{
   cout<<setw(25)<<"\n\n Enter Name : ";
   gets(name);
```

```
      cout<<setw(25)<<"\n Enter barcode number : ";
      cin>>barcode;
      cout<<setw(25)<<"\n Enter price : ";
      cin>>price;
      cout<<setw(25)<<"\n Enter quantity in stock : ";
      cin>>stock;
   }


   //  ****************** Display an item****************
   void Item::showdata()
   {
      cout<<setw(25)<<"\n\n Name : "<<name;
      cout<<setw(25)<<"\n Barcode Number : "
<<barcode;
      cout<<setw(25)<<"\n Price : "<<price;
      cout<<setw(25)<<" \n Quantity in stock : "<<stock;
   }

   //******************Check by Name***************
   int Item::checkN(char *nm)
   {
      if(strcmp(nm,name)==0)
         return 1;
      else
         return 0;
   }

   //******************Check by barcode*****************
   int Item::checkB(long bc)
   {
      if(bc==barcode)
```

```
        return 1;
    else
        return 0;
}
//******************Check by stock*****************
int Item::checkS(int st)
{
    if(stock<st)
        return 0;
    else if(st==stock)
        return 1;
    else
        return 2;
}
// ******************Edit A record******************
void Item::Edit()
{
    char ch;
    do{
        cout<<"Details of the record : \n";
        showdata();
        cout<<"\n Press A if you wish to update the name
\n";
        cout<<"\n Press B if you wish to update the
barcode \n";
        cout<<"\n Press C if you wish to update the price
\n";
        cout<<"\n Press D if you wish to update the stock
\n";
        cout<<"\n Press E to make no futher changes\n";
        ch=getche();
        switch(ch)
```

```
          {
             case 'A':
             case 'a': cout<<setw(30)<<"\n Enter new name :
";
                          gets(name);
                          break;
             case 'B':
             case 'b':cout<<setw(30)<<"\n Enter new Barcode
Number : ";
                          cin>>barcode;
                          break;
             case 'C':
             case 'c': cout<<setw(30)<<"\n Enter new price : ";
                          cin>>price;
                          break;
             case 'D':
             case 'd':cout<<setw(30)<<"\n Enter new stock
left : ";
                          cin>>stock;
                          break;
             case 'E':
             case 'e': break;

             default : cout<<"\n Invalid Input !! \n Please
enter a relevant choice. \n";

          }
       }while(ch!='E' && ch!='e');
    }
    //****************Append(ADD) a new item***********
    void Append()
    {
```

```
    Item I;
    cout<<"\n Enter the Details of the item : \n";
    I.getdata();

    fstream f1;
    f1.open("INVENTORY.DAT",ios::binary|ios::app);

    f1.write((char*) &I,sizeof(I));

    f1.close();
}


//*****************Display Inventory**************
void DisplayInventory()
{
    Item I;

    fstream f1;
    f1.open("INVENTORY.DAT",ios::binary|ios::in);

    while(f1.read((char*) &I,sizeof(I)) )
    {
        I.showdata();
    }
    f1.close();
}


//******************Search by Name****************
void SearchN()
{
    char sn[40];
    Item I;
    int found =0;

    cout<<"\n\n Enter the name to be searched : ";
    gets(sn);

    fstream f1("INVENTORY.DAT",ios::binary|ios::in);

    while(f1.read((char*)&I,sizeof(I)))
    {
        if(I.checkN(sn)==1)
        {
            found++;
            I.showdata();
        }
    }

    f1.close();

    if (found==0)
        cout<<"\n NO MATCH FOUND ! \N";
    else
        cout<<"\n TOTAL "<<found<<" ITEMS FOUND. \N";
}


// ******************Search by Barcode****************
void SearchB()
{
    long sb;
    Item I;
    int found =0;

    cout<<"\n\n Enter the barcode to be searched : ";
    cin>>sb;

    fstream f1;
    f1.open("INVENTORY.DAT",ios::binary|ios::in);

    while(f1.read((char*)&I,sizeof(I)))
    {
      if(I.checkB(sb)==1)
      {
        found++;
        I.showdata();
      }
    }

    f1.close();

    if (found==0)
      cout<<"\n NO MATCH FOUND ! \N";
    else
      cout<<"\n TOTAL "<<found<<" ITEMS FOUND. \N";
  }
  // ******************search by stock******************
  void SearchS()
  {
    int ss;
    Item I;
    int found=0;
    fstream f1;
    char sp;
    cout<<"\n\n Select choice for search by stock
quantity";
    cout<<"\n A.To search stock quantiy less than x";
    cout<<"\n B.To search stock quantity greater than x";
    cout<<"\n C.To search stock quantity  equal to x";
    sp=getche();
    switch(sp)
    {
      case 'a':
      case 'A':   cout<<"\n\n Enter the value of x : ";
              cin>>ss;



  f1.open("INVENTORY.DAT",ios::binary|ios::in);

              while(f1.read((char*)&I,sizeof(I)))
              {
                if(I.checkS(ss)==0)
                {
                  found++;
                  I.showdata();
                }
              }
```

```
                f1.close();

                if (found==0)
                  cout<<"\n NO MATCH FOUND !
\N";
                else
                  cout<<"\n   TOTAL   "<<found<<"
ITEMS FOUND. \N";


                break;



        case 'B':
        case 'b':cout<<"\n\n Enter the value of x : ";
                cin>>ss;



  f1.open("INVENTORY.DAT",ios::binary|ios::in);

                while(f1.read((char*)&I,sizeof(I)))
                {
                  if(I.checkS(ss)==2)
                  {
                    found++;
                    I.showdata();
                  }
                }
```

```
                    f1.close();

                    if (found==0)
                       cout<<"\n  NO  MATCH  FOUND  !
\N";
                    else
                       cout<<"\n    TOTAL    "<<found<<"
ITEMS FOUND. \N";

                    break;


          case 'c':
          case 'C':   cout<<"\n\n Enter the value of x : ";
                    cin>>ss;



   f1.open("INVENTORY.DAT",ios::binary|ios::in);

                    while(f1.read((char*)&I,sizeof(I)))
                    {
                       if(I.checkS(ss)==1)
                       {
                          found++;
                          I.showdata();
                       }
                    }

                    f1.close();

                    if (found==0)
```

```
                        cout<<"\n NO MATCH FOUND ! \N";
                     else
                        cout<<"\n TOTAL "<<found<<" ITEMS FOUND. \N";

                     break;
         default:    cout<<"\n Invalid Input !! \n Please enter a relevant choice. \n";
      }

   }

   //****************Modify a record*****************
   void modify()
   {
      char sn[40];
      Item I;
      int Modified=0;

      cout<<"\n\n Enter the Item to be modified : ";
      gets(sn);

      fstream f1,f2;
      f1.open("INVENTORY.DAT",ios::binary|ios::in);
      f2.open("TEMP.DAT",   ios::binary|ios::out);

      while( f1.read((char*) &I, sizeof(I)) )
      {
         if(I.checkN(sn)==1)
         {
            Modified++ ;
            I.Edit();
        }
        f2.write((char *) &I, sizeof(I));
    }

    f1.close();
    f2.close();


    if (Modified == 0)
        cout<<"\n\n NO MATCH FOUND !! \n\n";
          else
        {
            remove("INVENTORY.DAT");
            rename("TEMP.DAT" , "INVENTORY.DAT");
        }
}




//******************Delete a record*******************
void delete1()
{
    char sn[40];
    Item I;
    int Deleted=0;

    cout<<"\n Enter Item to be deleted : ";
    gets(sn);

    fstream f1,f2;
    f1.open("INVENTORY.DAT",ios::binary|ios::in);
    f2.open("TEMP.DAT",   ios::binary|ios::out);

    while( f1.read((char*) &I, sizeof(I)) )
    {
      if(I.checkN(sn)==1)
        Deleted ++ ;
         else
        f2.write((char *) &I, sizeof(I));
    }

    f1.close();
    f2.close();



    if (Deleted == 0)
      cout<<"\n\n NO MATCH FOUND !! \n\n";
       else
      {
        remove("INVENTORY.DAT");
        rename("TEMP.DAT"                              ,
"INVENTORY.DAT");
      }

  }

  void UpdateInventory()
  {
    char choice;
    do{
      clrscr();
```

```
        cout<<"Menu\n";
        cout<<"Press 1 for APPEND AN ITEM \n";
        cout<<"Press 2 for DISPLAY COMPLETE
INVENTORY \n";
        cout<<"Press 3 for SEARCH BY NAME \n";
        cout<<"Press 4 for SEARCH BY BARCODE
NUMBER \n";
        cout<<"Press 5 for SEARCH BY QUANTITY IN
STOCK \n";
        cout<<"Press 6 for MODIFY A ITEM \n";
        cout<<"Press 7 for DELETE A DELETE \n";
        cout<<"Press 8 for MAIN MENU \n";
        cout<<"Enter your choice : ";
        choice=getche();
        switch(choice)
        {
           case '1':Append();
                    break;
           case '2':DisplayInventory();
                    break;
           case '3':SearchN();
                    break;
           case '4':SearchB();
                    break;
           case '5':SearchS();
                    break;
           case '6':modify();
                    break;
           case '7':delete1();
                    break;
           case '8':
                    break;
        default :   cout<<"\n Invalid Input !! \n Please
enter a relevant choice. \n";
      }
      cout<<"\n\n Press any key to continue!";
      getch();
    }while(choice!='8');
  }

  //***************** ADD TO CART**************
  void AddtoCart()
  {
    Item I;
    long bc;
    cout<<"\n\nEnter Barcode of Item:";
    cin>>bc;
    int flag=0,quantity;
    ifstream f1("INVENTORY.DAT",ios::binary);

    while(f1.read((char*)&I,sizeof(I))&&flag==0)
    {
      if(I.checkB(bc)==1)
      {
        cout<<"\n\nItem Selected:\n";
        I.showdata();
        flag=1;
      }
    }

    f1.close();

    if (flag==0)
      cout<<"\n NO MATCH FOUND ! \n";
```

```
        else
        {
            cout<<"\n\nEnter Quantiy to be bught:\n";
            cin>>quantity;
            if(quantity>I.stock)
                cout<<"\nNot enough in stock!!\n";
            else
            {
                cout<<quantity<<" "<<I.name<<" added to
cart.\n\n";
                ofstream f2("CART.DAT",ios::binary|ios::app);
                I.stock=quantity;

                f2.write((char*)&I,sizeof(I));
                f2.close();

                fstream f3,f4;
                f3.open("INVENTORY.DAT",ios::binary|ios::in);
                f4.open("TEMP.DAT",ios::binary|ios::out);
                Item N;
                while(f3.read((char*) &N,sizeof(N)) )
                {
                    if(N.checkB(bc)==1)
                    {

                        N.stock-=I.stock;

                        f4.write((char*)&N,sizeof(N));
                    }

                    else
                    f4.write((char*)&N,sizeof(N));
                }
                f3.close();
                f4.close();

                remove("INVENTORY.DAT");
                rename("TEMP.DAT","INVENTORY.DAT");
            }

        }
    }

    //***********DELETE FROM CART**************
    void DelCart()
    {
        Item I;
        long bc;
        cout<<"\n\nEnter Barcode of Item:";
        cin>>bc;
        int flag=0;

        fstream f1,f2;
        f1.open("CART.DAT",ios::binary|ios::in);
        f2.open("TEMP.DAT",ios::binary|ios::out);

        while(f1.read((char*) &I,sizeof(I))&&flag==0)
        {
            if(I.checkB(bc)==1)
            {

                cout<<"\n\nItem DELETED:\n";
                I.showdata();
                flag=1;
```

```
      fstream f3,f4;
      f3.open("INVENTORY.DAT",ios::binary|ios::in);
      f4.open("TEMP.DAT",ios::binary|ios::out);
      Item N;
      while(f3.read((char*) &N,sizeof(N)) )
      {
         if(N.checkB(bc)==1)
         {

            N.stock+=I.stock;

            f4.write((char*)&N,sizeof(N));
         }

         else
         f4.write((char*)&N,sizeof(N));
      }
      f3.close();
      f4.close();

      remove("INVENTORY.DAT");
      rename("TEMP.DAT","INVENTORY.DAT");
   }
   else
      f2.write((char*)&I,sizeof(I));
}
if(flag==0)
   cout<<"No match found!!";
f1.close();
f2.close();
remove("CART.DAT");
```

```
        rename("TEMP.DAT","CART.DAT");

    }
```

```
    //******************UPDATE CART***************
    void UpdateCart()
    {
        char choice;
        do{
            clrscr();
            cout<<"Menu\n";
            cout<<"Press 1 to DISPLAY INVENTORY \n";
            cout<<"Press 2 to SEARCH FOR ITEM BY
NAME \n";
            cout<<"Press 3 to SEARCH FOR ITEM BY
BARCODE \n";
            cout<<"Press 4 to ADD ITEM TO CART \n";
            cout<<"Press 5 to DELETE ITEM FROM CART
\n";
            cout<<"Press 6 to FINISH SHOPPING \n";
            cout<<"Enter your choice : ";
            choice=getche();
            switch(choice)
            {
                case '1':DisplayInventory();
                                break;
                case '2':SearchN();
                                break;
                case '3':SearchB();
                                break;
        case '4':   AddtoCart();
                    break;
        case '5':DelCart();
                    break;
        case '6':
                    break;
        default :   cout<<"\n  Invalid  Input  !!  \n  Please
enter a relevant choice. \n";
      }
      cout<<"\n\n Press any key to continue!";
      getch();
    }while(choice!='6');

  }

  void ShowCart()
  {
    Item I;

    fstream f1;
    f1.open("CART.DAT",ios::binary|ios::in);

    while(f1.read((char*) &I,sizeof(I)) )
    {
      I.showdata();
    }
    f1.close();
  }

  //*****END  SHOPPING/CUSTOMER  CHECK  OUT
OF SUPERMARKET*****
  void CheckOut()
```

```
    {
        cout<<"\n\n Your Bill:\n\n";
        Item I;
        int amount=0;
        fstream f1;
        f1.open("CART.DAT",ios::binary|ios::in);

        while(f1.read((char*) &I,sizeof(I)) )
        {
            I.showdata();
            amount+=(I.price*I.stock);

        }
        f1.close();
        amount*=1.18; //GST
        cout<<"\n Total amount : "<<amount;
        cout<<"\n\n Thank You for shopping with us
today!\n";
    }



    //************SHOPPING PROCEDURE*********

    void Sale()
    {
        char choice;
        do{
            clrscr();
            cout<<"Menu\n";
            cout<<"Press 1 to ADD OR DELETE ITEMS
FROM CART \n";
```

```
      cout<<"Press 2 to DISPLAY COMPLETE CART
\n";
      cout<<"Press 3 to CHECK OUT\n";
      cout<<"Enter your choice : ";
      choice=getche();
      switch(choice)
      {
        case '1': UpdateCart();
                break;
        case '2': ShowCart();
                break;
        case '3':
                break;
        default :  cout<<"\n Invalid Input !! \n Please
enter a relevant choice. \n";
      }
      cout<<"\n\n Press any key to continue!";
      getch();
    }while(choice!='3');
    CheckOut();
  }

  //**********DECLARE CLEARENCE SALE********
  void ReducePrice()
  {
    cout<<"\n\nEnter Maximum stock threshold for
standard pricing";
    cout<<" above which a discount will be applied:";
    int threshold=0,discount=0;
    cin>>threshold;
    dis:
```

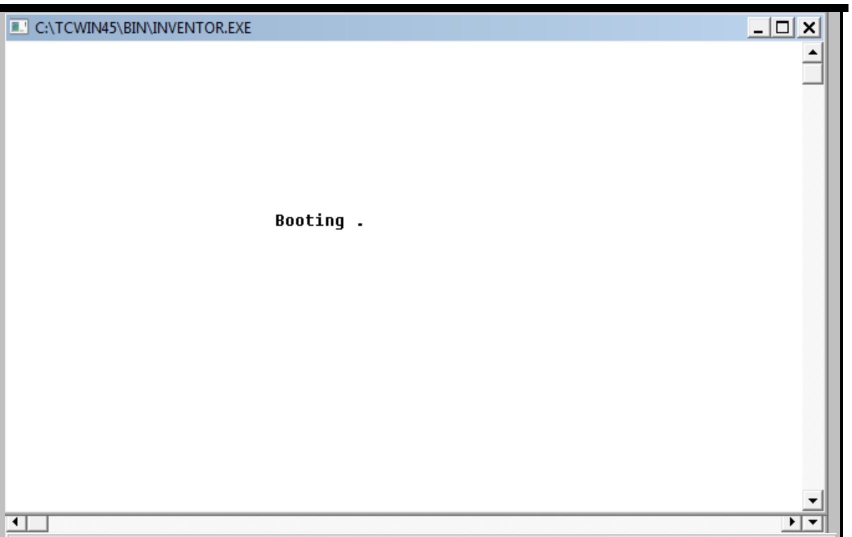
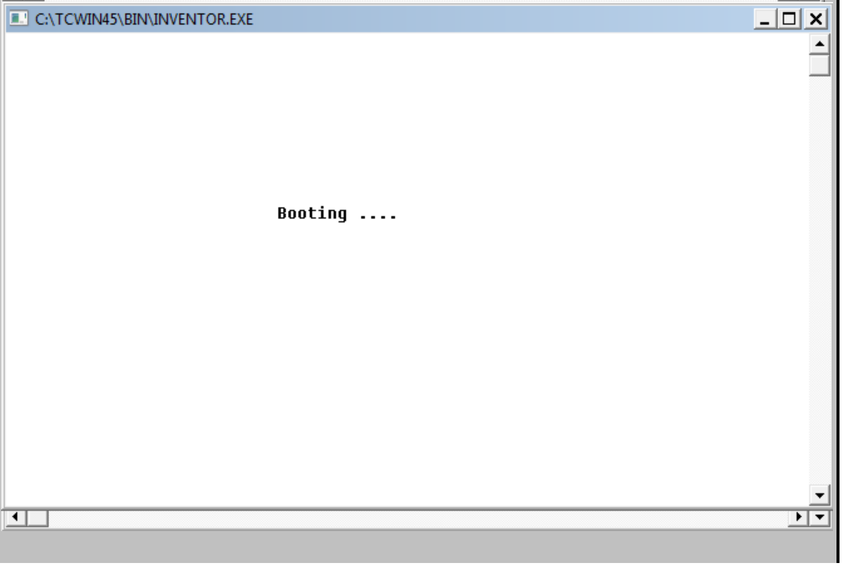
```
    cout<<"\n\nEnter %age discount for items above
maximum threshold:";
    cin>>discount;
    if(discount>100)
    {
      cout<<"\nDiscount cannot be more than
100%!!\n Try Again.\n";
      goto dis;
    }
    Item I;

    fstream f1,f2;
    f1.open("INVENTORY.DAT",ios::binary|ios::in);
    f2.open("TEMP.DAT",ios::binary|ios::out);

    while(f1.read((char*) &I,sizeof(I)) )
    {
      if(I.stock>threshold)
      {
        cout<<"Original Pricing :\n ";
        I.showdata();
        I.stock*=((100-discount)/100);
        cout<<"Updated Pricing :\n";
        f2.write((char*)&I,sizeof(I));
      }

      else
      f2.write((char*)&I,sizeof(I));
    }
    f1.close();
    f2.close();
```

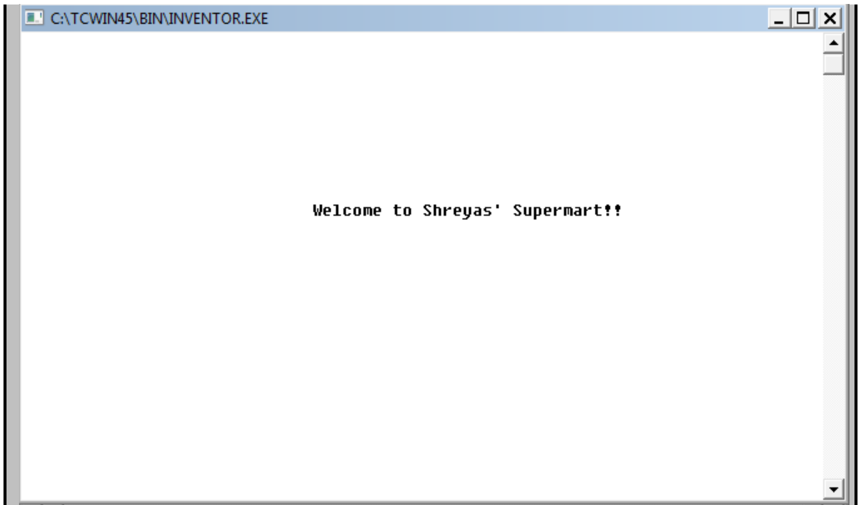
```
        remove("INVENTORY.DAT");
        rename("TEMP.DAT","INVENTORY.DAT");

    }

    //********DECLARE SURPLUS PRICES***********
    void IncreasePrice()
    {
        cout<<"\n\nEnter Minimum stock threshold for
standard pricing";
        cout<<" below which a surcharge will be applied:";
        int threshold=0,surcharge=0;
        cin>>threshold;
        cout<<"\n\nEnter %age surcharge for items below
minimum threshold:";
        cin>>surcharge;

        Item I;

        fstream f1,f2;
        f1.open("INVENTORY.DAT",ios::binary|ios::in);
        f2.open("TEMP.DAT",ios::binary|ios::out);

        while(f1.read((char*) &I,sizeof(I)) )
        {
            if(I.stock>threshold)
            {
                cout<<"Original Pricing :\n ";
                I.showdata();
                I.stock*=((100+surcharge)/100);
                cout<<"Updated Pricing :\n";
                f2.write((char*)&I,sizeof(I));
            }

            else
            f2.write((char*)&I,sizeof(I));
        }
        f1.close();
        f2.close();

        remove("INVENTORY.DAT");
        rename("TEMP.DAT","INVENTORY.DAT");

    }

    //**********ALTER COMMODITIES PRICES******
    void Discount()
    {
        char choice;
        do{
            clrscr();
            cout<<"Menu\n";
            cout<<"Press 1 to REDUCE PRICES OF ITEMS WITH SURPLUS STOCK \n";
            cout<<"Press 2 to INCREASE PRICES OF ITEMS WITH DEPLETING STOCK \n";
            cout<<"Press 3 for MAIN MENU \n";
            cout<<"Enter your choice : ";
            choice=getche();
            switch(choice)
            {
                case '1':ReducePrice();
                                break;
                case '2':IncreasePrice();
```

```
                        break;
                case '3':
                        break;
                default :  cout<<"\n Invalid Input !! \n Please
enter a relevant choice. \n";
            }
            cout<<"\n\n Press any key to continue!";
            getch();
        }while(choice!='3');
    }

    //******************MAIN MENU************
    void main()
    {

        int introsequence=0;
        if(introsequence==0)
        {
        gotoxy(30,10);
        cout<<"Welcome to Shreyas' Supermart!!"<<endl;
//Game name

        for(int i=0;i<1000;i++)
        for(int j=0;j<1000;j++)
        for(int l=0;l<1000;l++);  //Delay using empty loop

        clrscr();

        gotoxy(28,10);
        cout<<"Booting"<<endl;
        for( i=0;i<4;i++)
        {
```

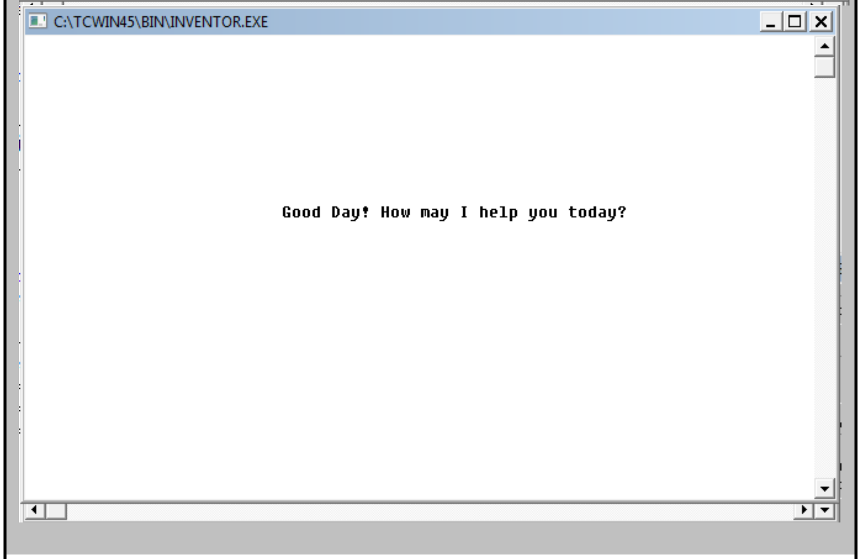
```
     gotoxy(36+i,10);
      cout<<"."; //Display pseudo booting
animation using dots with ~250µs delay
      for(int j=0;j<1000;j++)
     for(int k=0;k<250;k++)
      for(int l=0;l<1000;l++);
    }

    clrscr();

    gotoxy(27,10);
    cout<<"Good Day! How may I help you
today?"<<endl;
    for( i=0;i<1000;i++)
    for(int j=0;j<1000;j++)
    for(int l=0;l<1000;l++);
    clrscr();

    introsequence=1;
    }  mainmenu:
    char choice;
    do{
      clrscr();
      cout<<"Main Menu\n";
      cout<<"Press 1 to UPDATE INVENTORY\n";
      cout<<"Press 2 to MAKE A SALE \n";
      cout<<"Press 3 to CHANGE PRICES \n";
      cout<<"Press 4 to QUIT\n";
      cout<<"Enter your choice : ";
      choice=getche();
      switch(choice)
      {
        case '1':UpdateInventory();
                break;
        case '2':Sale();
                break;
        case '3':Discount();
                break;
        case '4':
                break;
        default :   cout<<"\n Invalid Input !! \n Please enter a relevant choice. \n";
      }
      cout<<"\n\n Press any key to continue!";
      getch();
    }while(choice!='4');
    remove("CART.DAT");
    cout<<”\nThank You for Shooping with Shreyas’ Supermart”;
    getch();
  }
```

SAMPLE OUTPUTS

Function Name	Output
Intro Sequence	 Booting .  Booting ....

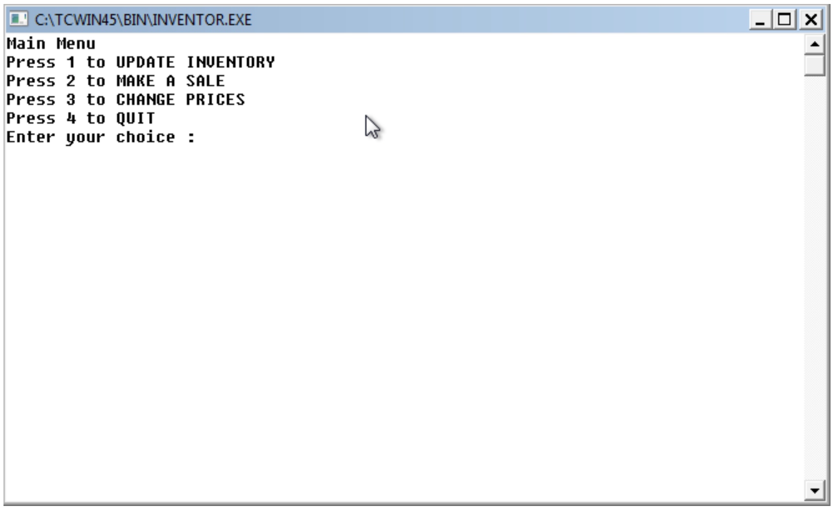


```
Welcome to Shreyas' Supermart!!
```



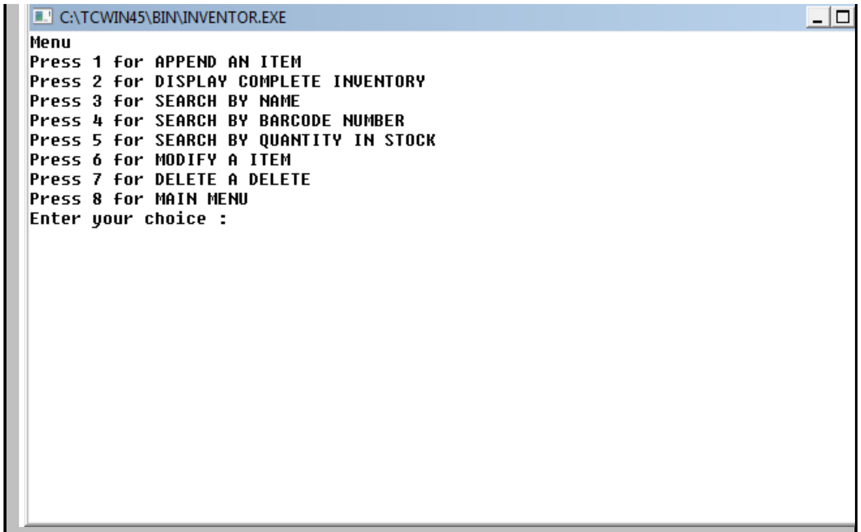
```
Good Day! How may I help you today?
```

Main Menu



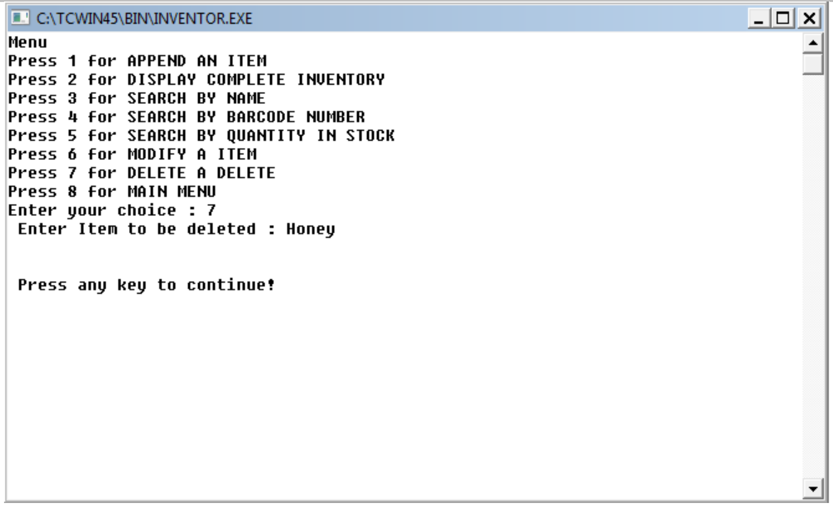
```
Main Menu
Press 1 to UPDATE INVENTORY
Press 2 to MAKE A SALE
Press 3 to CHANGE PRICES
Press 4 to QUIT
Enter your choice :
```

Update Inventory Menu

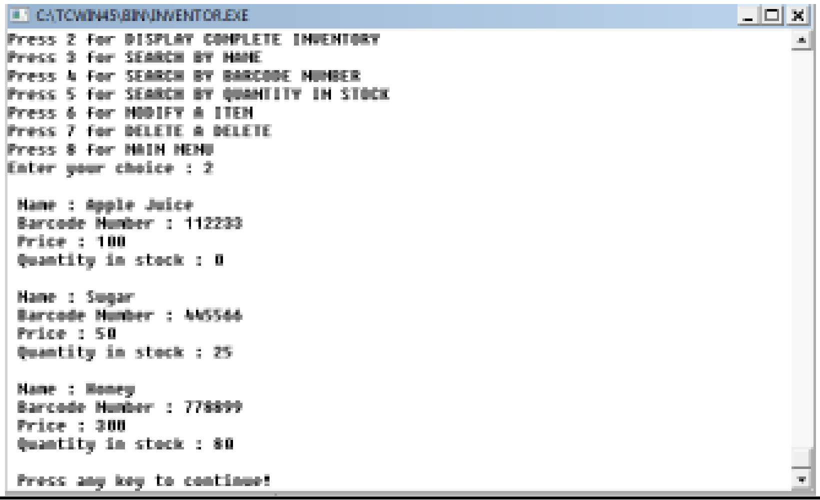


```
Menu
Press 1 for APPEND AN ITEM
Press 2 for DISPLAY COMPLETE INVENTORY
Press 3 for SEARCH BY NAME
Press 4 for SEARCH BY BARCODE NUMBER
Press 5 for SEARCH BY QUANTITY IN STOCK
Press 6 for MODIFY A ITEM
Press 7 for DELETE A DELETE
Press 8 for MAIN MENU
Enter your choice :
```

Append Item, Display Inventory, Delete Item.

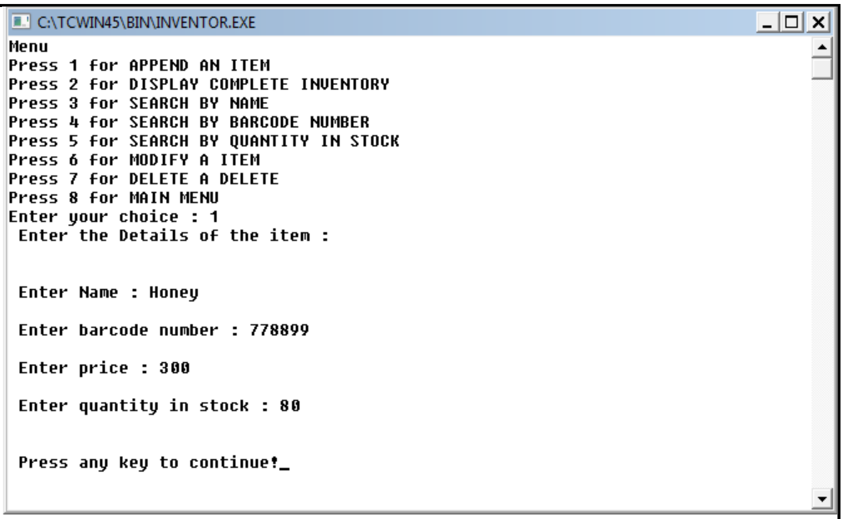


```
Menu
Press 1 for APPEND AN ITEM
Press 2 for DISPLAY COMPLETE INVENTORY
Press 3 for SEARCH BY NAME
Press 4 for SEARCH BY BARCODE NUMBER
Press 5 for SEARCH BY QUANTITY IN STOCK
Press 6 for MODIFY A ITEM
Press 7 for DELETE A DELETE
Press 8 for MAIN MENU
Enter your choice : 7
 Enter Item to be deleted : Honey


 Press any key to continue!
```

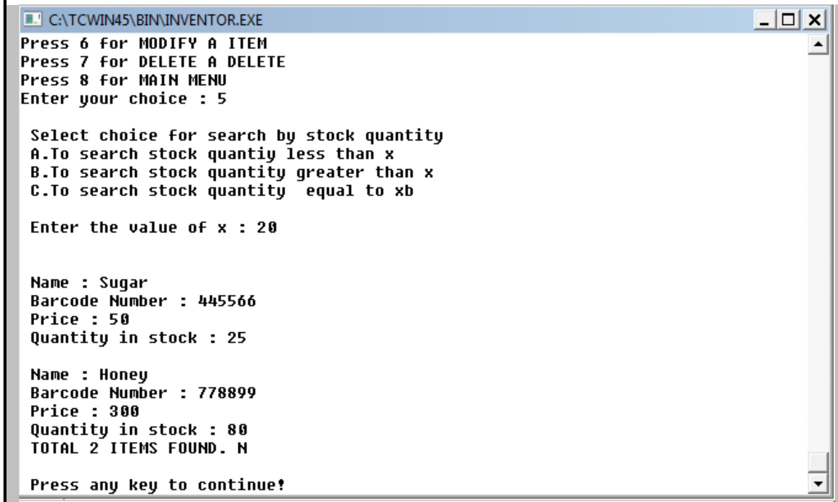


```
Press 2 for DISPLAY COMPLETE INVENTORY
Press 3 for SEARCH BY NAME
Press 4 for SEARCH BY BARCODE NUMBER
Press 5 for SEARCH BY QUANTITY IN STOCK
Press 6 for MODIFY A ITEM
Press 7 for DELETE A DELETE
Press 8 for MAIN MENU
Enter your choice : 2

 Name : Apple Juice
 Barcode Number : 112233
 Price : 100
 Quantity in stock : 0

 Name : Sugar
 Barcode Number : 445566
 Price : 50
 Quantity in stock : 25

 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 80

 Press any key to continue!
```

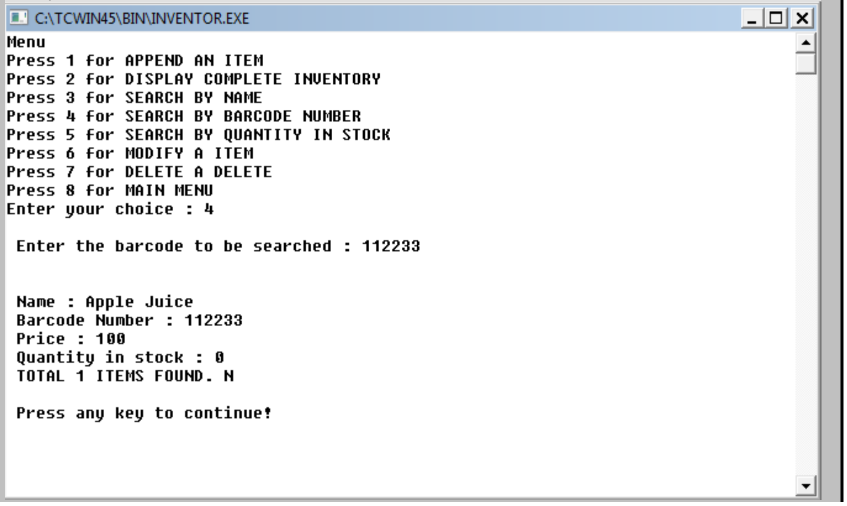
Search Functions



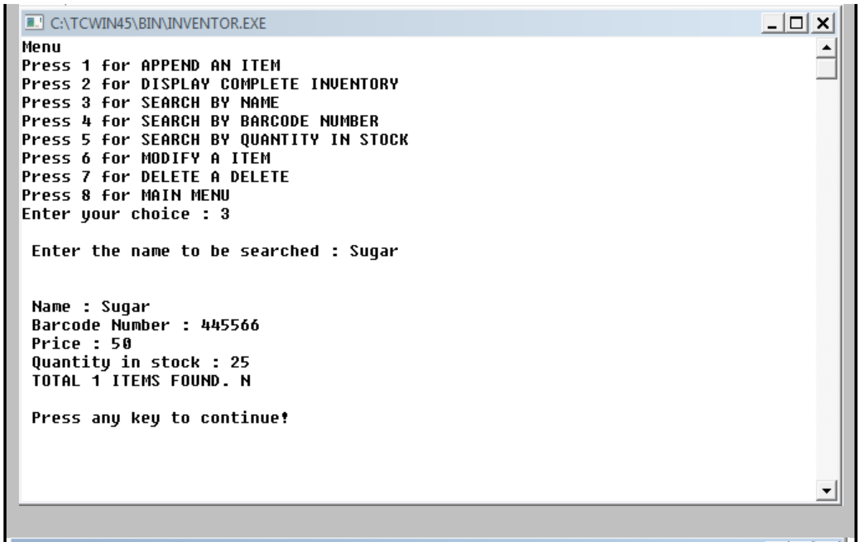
```
Menu
Press 1 for APPEND AN ITEM
Press 2 for DISPLAY COMPLETE INVENTORY
Press 3 for SEARCH BY NAME
Press 4 for SEARCH BY BARCODE NUMBER
Press 5 for SEARCH BY QUANTITY IN STOCK
Press 6 for MODIFY A ITEM
Press 7 for DELETE A DELETE
Press 8 for MAIN MENU
Enter your choice : 1
 Enter the Details of the item :


 Enter Name : Honey

 Enter barcode number : 778899

 Enter price : 300

 Enter quantity in stock : 80


 Press any key to continue!_
```



```
Press 6 for MODIFY A ITEM
Press 7 for DELETE A DELETE
Press 8 for MAIN MENU
Enter your choice : 5

 Select choice for search by stock quantity
 A.To search stock quantiy less than x
 B.To search stock quantity greater than x
 C.To search stock quantity  equal to xb

 Enter the value of x : 20


 Name : Sugar
 Barcode Number : 445566
 Price : 50
 Quantity in stock : 25

 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 80
 TOTAL 2 ITEMS FOUND. N

 Press any key to continue!
```

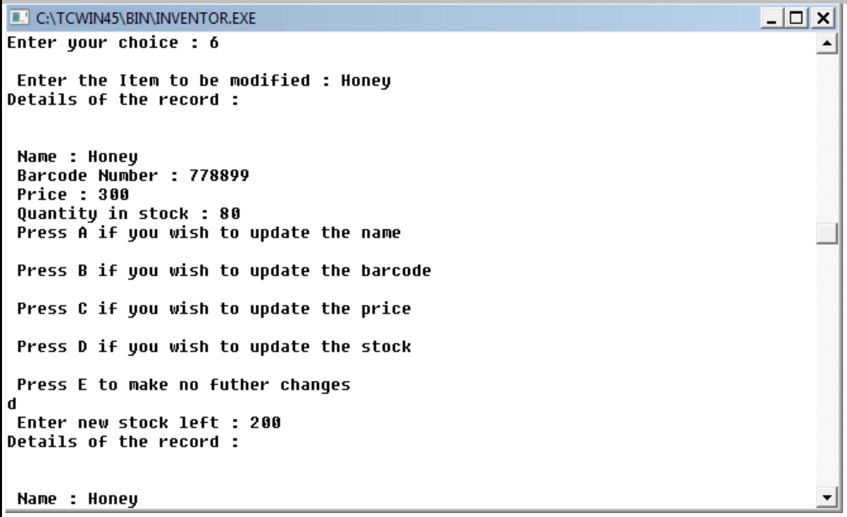


```
Menu
Press 1 for APPEND AN ITEM
Press 2 for DISPLAY COMPLETE INVENTORY
Press 3 for SEARCH BY NAME
Press 4 for SEARCH BY BARCODE NUMBER
Press 5 for SEARCH BY QUANTITY IN STOCK
Press 6 for MODIFY A ITEM
Press 7 for DELETE A DELETE
Press 8 for MAIN MENU
Enter your choice : 4

 Enter the barcode to be searched : 112233


 Name : Apple Juice
 Barcode Number : 112233
 Price : 100
 Quantity in stock : 0
 TOTAL 1 ITEMS FOUND. N

 Press any key to continue!
```



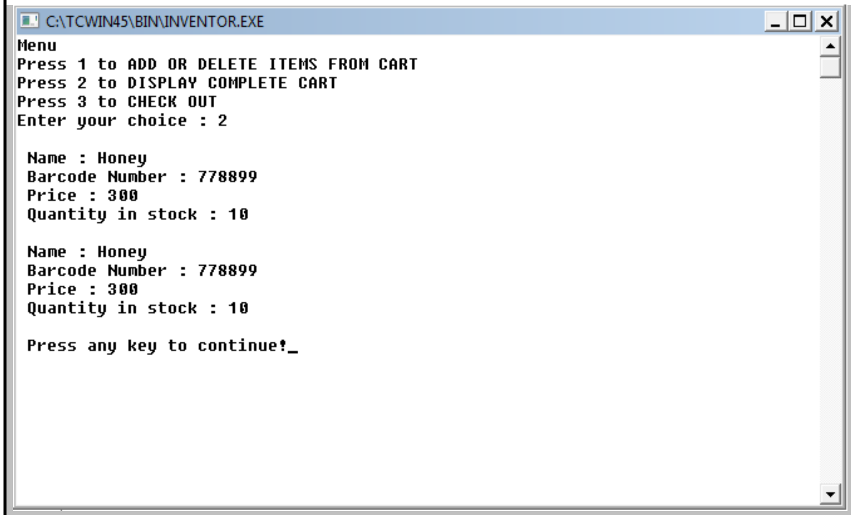
```
Menu
Press 1 for APPEND AN ITEM
Press 2 for DISPLAY COMPLETE INVENTORY
Press 3 for SEARCH BY NAME
Press 4 for SEARCH BY BARCODE NUMBER
Press 5 for SEARCH BY QUANTITY IN STOCK
Press 6 for MODIFY A ITEM
Press 7 for DELETE A DELETE
Press 8 for MAIN MENU
Enter your choice : 3

 Enter the name to be searched : Sugar


 Name : Sugar
 Barcode Number : 445566
 Price : 50
 Quantity in stock : 25
 TOTAL 1 ITEMS FOUND. N

 Press any key to continue!
```

Modify Function



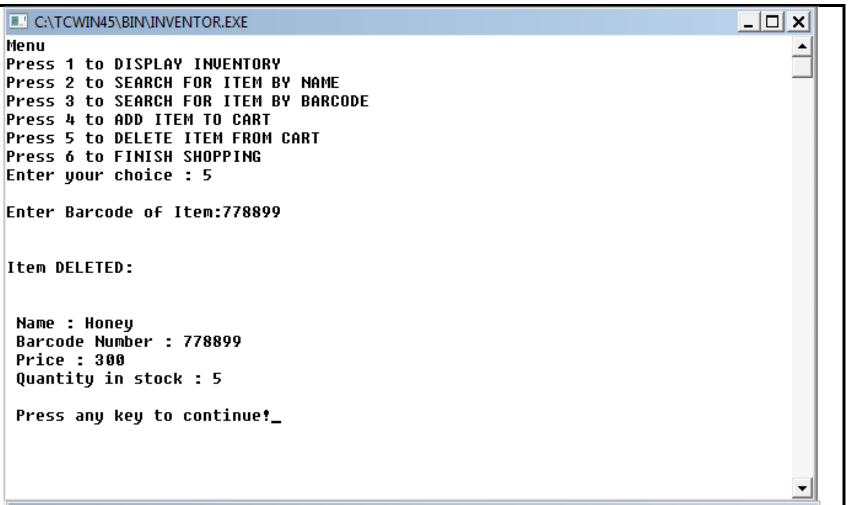
```
Enter your choice : 6

 Enter the Item to be modified : Honey
Details of the record :


 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 80
 Press A if you wish to update the name

 Press B if you wish to update the barcode

 Press C if you wish to update the price

 Press D if you wish to update the stock

 Press E to make no futher changes
d
 Enter new stock left : 200
Details of the record :


 Name : Honey
```

Sale Menu(Displaying Cart)

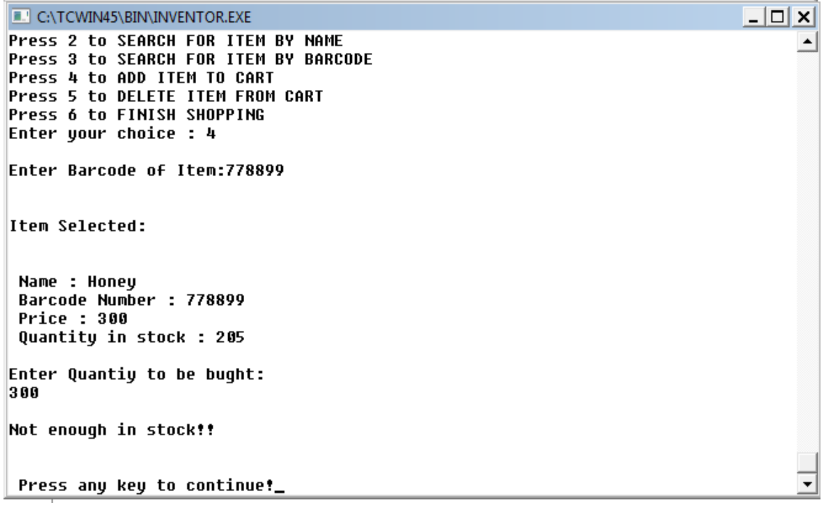


```
Menu
Press 1 to ADD OR DELETE ITEMS FROM CART
Press 2 to DISPLAY COMPLETE CART
Press 3 to CHECK OUT
Enter your choice : 2

 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 10

 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 10

 Press any key to continue!_
```

Add to and Delete from cart In ADD TO CART MENU

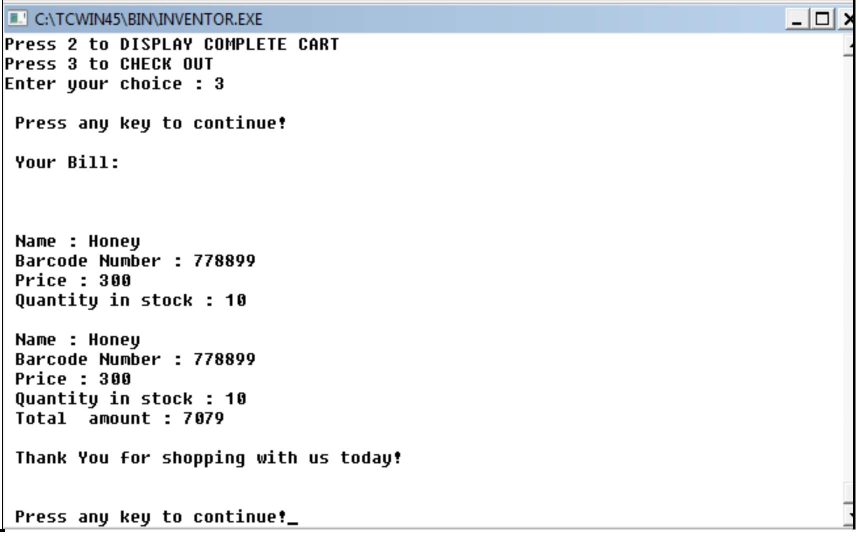


```
Menu
Press 1 to DISPLAY INVENTORY
Press 2 to SEARCH FOR ITEM BY NAME
Press 3 to SEARCH FOR ITEM BY BARCODE
Press 4 to ADD ITEM TO CART
Press 5 to DELETE ITEM FROM CART
Press 6 to FINISH SHOPPING
Enter your choice : 5

Enter Barcode of Item:778899


Item DELETED:


 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 5

 Press any key to continue!_
```



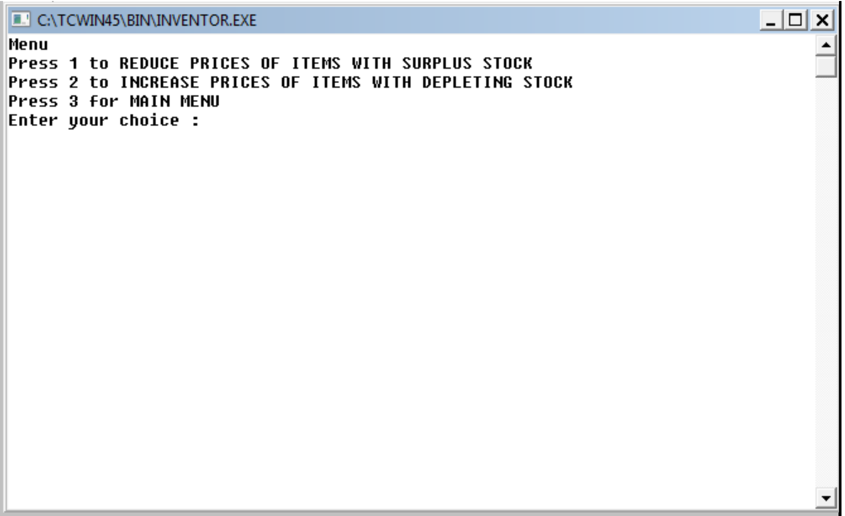
```
Press 2 to SEARCH FOR ITEM BY NAME
Press 3 to SEARCH FOR ITEM BY BARCODE
Press 4 to ADD ITEM TO CART
Press 5 to DELETE ITEM FROM CART
Press 6 to FINISH SHOPPING
Enter your choice : 4

Enter Barcode of Item:778899


Item Selected:


 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 205

Enter Quantiy to be bught:
300

Not enough in stock!!


 Press any key to continue!_
```

Check Out



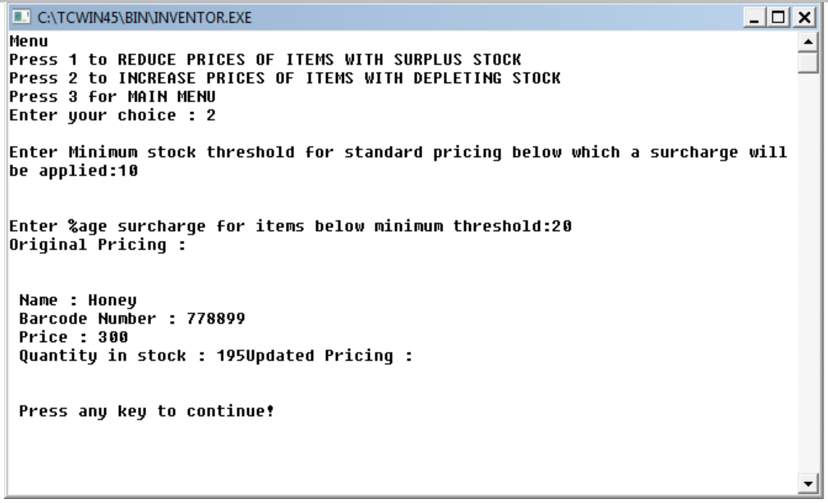
```
Press 2 to DISPLAY COMPLETE CART
Press 3 to CHECK OUT
Enter your choice : 3

 Press any key to continue!

 Your Bill:



 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 10

 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 10
 Total  amount : 7079

 Thank You for shopping with us today!


 Press any key to continue!_
```

Pricing Menu



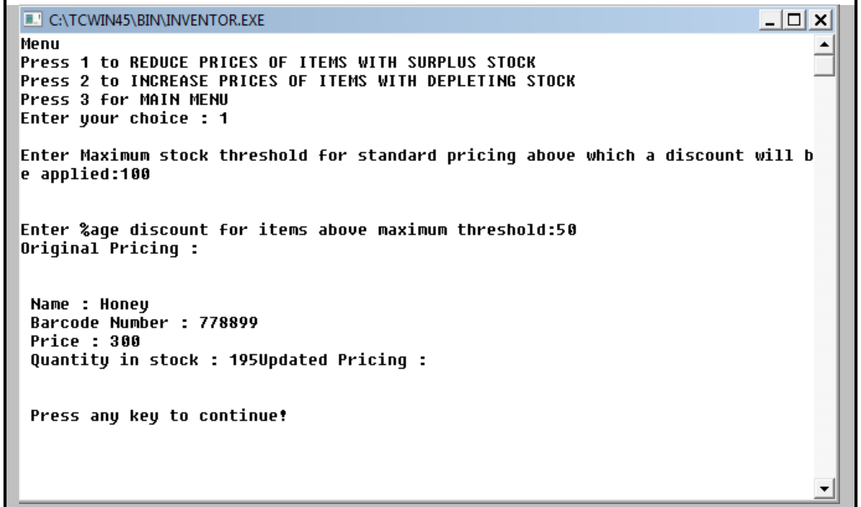
```
Menu
Press 1 to REDUCE PRICES OF ITEMS WITH SURPLUS STOCK
Press 2 to INCREASE PRICES OF ITEMS WITH DEPLETING STOCK
Press 3 for MAIN MENU
Enter your choice :
```

Surcharge Pricing



```
Menu
Press 1 to REDUCE PRICES OF ITEMS WITH SURPLUS STOCK
Press 2 to INCREASE PRICES OF ITEMS WITH DEPLETING STOCK
Press 3 for MAIN MENU
Enter your choice : 2

Enter Minimum stock threshold for standard pricing below which a surcharge will
be applied:10


Enter %age surcharge for items below minimum threshold:20
Original Pricing :


 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 195Updated Pricing :


 Press any key to continue!
```

Discount Pricing



```
Menu
Press 1 to REDUCE PRICES OF ITEMS WITH SURPLUS STOCK
Press 2 to INCREASE PRICES OF ITEMS WITH DEPLETING STOCK
Press 3 for MAIN MENU
Enter your choice : 1

Enter Maximum stock threshold for standard pricing above which a discount will b
e applied:100


Enter %age discount for items above maximum threshold:50
Original Pricing :


 Name : Honey
 Barcode Number : 778899
 Price : 300
 Quantity in stock : 195Updated Pricing :


 Press any key to continue!
```

SCOPE FOR IMPROVEMENT

ADDITIONAL MEMBER VARIABLES

THERE IS A POSSIBILITY TO CREATE ADDITIONAL MEMBER VARIABLES FOR EACH ITEM:

1. Date of Purchase
2. Date of Sale
3. Date of Expiry

These can be implemented by creating another class- date. Which would contain dd, mm, and yyyy and we could use ***containership*** *to use this as a* ***user defined data type.***

ADDITIONAL FUNCTIONS

Many more possibilities for additional member functions arise with the new member variables.

CLEARENCE SALE AUTOMIZATION

- To implement an automatic sale we could use the data from expiry dates to automatically decrease prices of commodities nearing their date of expiry.

This method can also use a complex algorithm which would also take into account rate of sale, i.e. how fast a certain product goes out of stock to either increase or decrease prices accordingly. Thus this will create a real supply and demand model.

DELETE ITEMS PAST EXPIRY

We can also implement expired item to be automatically removed from the inventories.

RETURN TO VENDOR

Damaged goods may be returned to the vendor for the supermarket using edit inventories options.

BIBLIOGRAPHY

- Class Notes - 2017-18
- Class Notes - 2018-19