**C++ PROJECT**

**RETAIL SHOPPING MANAGEMENT**

[](https://www.google.com.kw/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=&url=http://www.bloggang.com/index.php?category%3D21&bvm=bv.104819420,d.d2s&psig=AFQjCNGtjgyM3SUQYKFwMV-U2nOc6g-qhg&ust=1444562545477260)

PREPARED BY –

SHRADHEYA THAKRE

**INDIAN EDUCATIONAL SCHOOL, KUWAIT**

Affiliated to CBSE, New Delhi

(Affiliation No. 5930015)

**BONAFIDE CERTIFICATE**

This is to certify that ……………………………………………………………………. of class XII A with registration number ………………………. has done a project work in Computer Science for **All India Senior School Certificate Examination**  during the year 2015-2016 at the Indian Educational School, Kuwait.

Date: \_\_\_\_\_\_\_\_\_\_\_

Principal School Seal Teacher-in-Charge

Date: \_\_\_\_\_\_\_\_\_\_\_\_ External Examiner

**Acknowledgement**

*"There are times when silence speaks so much more loudly than words of praise to only as good as belittle a person, whose words do not express, but only put a veneer over true feelings, which are of gratitude at this point of time."*

I would like to express my sincere gratitude to GOD ALMIGHTY and my computer science mentor Mrs Asha Ebey for her vital support, guidance and encouragement – without which this project would not have come forth. I would also like to express my gratitude to my parents for their support during the making of this project.

|  |  |  |  |
| --- | --- | --- | --- |
| **Serial No.** | **TOPICS** | **Page No.** | **Remarks** |
| **1** | **AIM** | **5** |  |
| **2** | **DOCUMENTATION** | **6** |  |
| **3** | **SOURCE CODE** | **10** |  |
| **4** | **OUTPUT** | **49** |  |
| **5** | **CONCLUSION** | **65** |  |
| **6** | **BIBLIOGRAPHY** | **66** |  |

**INDEX**

**AIM**

The project is designed as a Digitalised Shopping centre.

The Project Aims to CREATE, STORE, MODIFY and SEARCH the data related to Retail Marketing, i.e.

* It handles various types of Data like,

1. Name, Code, Quantity, Price and Discount percentages for each Item present in Stock.
2. Various Details of each customer.

* It can be used by either,
* Administrator

To ADD, UPDATE, SEARCH Stocks and view the registered Members.

* Customer

To Register, Shop, Buy and view the Bill of the Purchase.

Thus a complete DIGITALISED SHOPPING SOFTWARE is ready for use.

DOCUMENTATION

* HEADER FILES USED

| HEADER FILE | USE | FUNCTIONS |
| --- | --- | --- |
| #include<iostream.h> | To insert data and show them on the output screen | cin  cout |
| #include<conio.h> | To clear the screen and to position the cursor to the desired position | clrscr();  getch(); |
| #inlcude<stdio.h> | To get or put data into a file | gets();  remove();  rename(); |
| #inlcude<process.h> | To exit from a file or loop | exit(); |
| #include<string.h> | To concatenate, copy or compare two strings | strcmpi();  strcpy(); |
| #include<fstream.h> | Declares the C++ stream classes that supports file input and output | open();  close();  read();  write();  seekp();  tellg(); |

* FUNCTIONS USED

|  |  |
| --- | --- |
| **FUNCTIONS** | USE |
| void AddS(); | To Add details of Items into the file |
| void AddC(); | To Add details of Customer into the file |
| void DispS(); | To Display the details of all item in Stock |
| void DispC(); | To Display the details of Registered Customers |
| void DeleteBill(); | To remove the data in the file Bill.dat |
| int Login(); | To check the password and returns 1 for correct password and allows access to Administrator Section |
| void SearchItem(); | To display the details of the searched item code |
| void Update(); | To change the quantity of an Item after Stock has been received |
| void ShopStart(); | To check for Registered customer and allow access to the Shopping Menu |
| void Buy(); | To make necessary changes when the customer buys a product and to append data in the Bill file of a Customer |
| void Purchase(); | To ask customer for Quantity of the product to be purchased |
| void ShowPrice(); | To display the Price and Discount of the product chosen by the Customer |
| void Bill(); | To Show the Bill of the particular customer after the purchase and assign points to the customer |
| void DeleteC(); | To Delete a Registered customer |
| void DeleteI(); | To Delete an item after purchase |
| void Admin(); | To show the Menu of the Administrator |
| void CustomerF(); | To display the Menu for the Customer and the Shopping Menu for registered Customer |
| void Choice(); | To show the Menu for the whole program |

* FILES USED

|  |  |
| --- | --- |
| **FILES** | **USE** |
| Stock.dat | To store data of Stocks and Items |
| Customer.dat | To store data of registered Memebers |
| Bill.dat | To store data of Purchased item for a Cutomer |

SOURCE CODE

#include<iostream.h>

#include<conio.h>

#include<stdio.h>

#include<string.h>

#include<fstream.h>

#include<process.h>

//-------------------------DECLARATION----------------------------------

void Choice();

fstream B("Bill.dat",ios::binary|ios::trunc);

void CustomerF();

float Amount=0;

//----------------------------------CLASS STOCK-------------------------

class Stock

{

long Iid;

int Qty;

int Price;

float Discount;

char Name[50];

public:

Stock()

{

Iid=1000;

Qty=0;

Price=1;

Discount=0;

strcpy(Name , "");

}

void GetS()

{

cout<<"\n\n Enter the item id : ";

cin>>Iid;

cout<<"\n\n Enter the Item name : ";

gets(Name);

cout<<"\n\n Enter the quatity of the item : ";

cin>>Qty;

cout<<"\n\n Enter the price of each item : ";

cin>>Price;

cout<<"\n\nEnter the discount % : ";

cin>>Discount;

}

void ShowS()

{

clrscr();

cout<<"\n\n\n--------ITEM DETAILS----------- \n\n";

cout<<"\n\n Item ID number : "<<Iid;

cout<<"\n\n Item Name : "<<Name;

cout<<"\n\n Item Quantity : "<<Qty;

cout<<"\n\n Item Price : "<<Price;

cout<<"\n\n Discount % : "<<Discount;

getch();

}

void Decrement(int loc)

{

Qty=Qty-loc;

}

char\* Rtnch()

{

return Name;

}

int Rtnid()

{

return Iid;

}

void DispB()

{

float k=(Price\*Qty)-((Discount/100.0)\*Price\*Qty);

Amount=Amount+k;

cout<<Iid<<" "<<Name<<" "<<Qty<<" "<<Price\*Qty<<" "<<(Price\*Qty)-((Discount/100.0)\*Price\*Qty);

}

void Change(int loc)

{

Qty=loc;

}

void Gain(int loc)

{

Qty=Qty+loc;

}

int Rtnprice()

{

return Price;

}

int Rprice()

{

return Price;

}

int Rdisc()

{

return Discount;

}

};

//----------------------------- CLASS CUSTOMER-----------------------

class Customer

{

long telno;

char Name[20];

int points;

char Address[50];

public:

Customer()

{

telno=0;

points=0;

strcpy(Address,"");

strcpy(Name, "");

}

void GetC()

{

cout<<"--------------------------------------------------";

cout<<"\n\n\nEnter the telephone number : ";

cin>>telno;

cout<<"\n\n\nEnter the customer name : ";

gets(Name);

cout<<"\n\n\nEnter your Address : ";

gets(Address);

points=0;

}

void ShowC()

{

clrscr();

cout<<"------------------------DETAILS-----------------------\n\n";

cout<<"\n\n Customer telephone number : "<<telno;

cout<<"\n\n Customer Name : "<<Name;

cout<<"\n\n Delivery Address : "<<Address;

cout<<"\n\n Points earned : "<<points;

cout<<"\n\n\n\n";

getch();

}

long Rtntel()

{

return telno;

}

void Updatepoint(int temp)

{

points=points+temp;

}

};

//-----------------------FUCTIONS----------------------------------

void AddS(Stock S1)

{

Stock T;

int c=0;

fstream S("Stock.dat",ios::binary|ios::in|ios::out);

while(S.read((char\*)&T,sizeof(T)))

{

if(stricmp(T.Rtnch(),S1.Rtnch())==0)

c++;

}

S.close();

S.open("Stock.dat",ios::binary|ios::in|ios::out|ios::app);

if(c==0)

{

S.write((char\*)&S1,sizeof(S1));

}

else

{

cout<<"\n\n\n Item already exists";

getch();

}

S.close();

}

void AddC(Customer C1)

{

fstream C("Customer.dat",ios::binary|ios::app|ios::in|ios::out);

C.write((char\*)&C1,sizeof(C1));

C.close();

}

void DispS()

{

int k=0;

Stock S1;

fstream S("Stock.dat",ios::binary|ios::in|ios::out);

while(S.read((char\*)&S1,sizeof(S1)))

{

S1.ShowS();

k++;

}

if(k==0)

{

clrscr();

cout<<"\n\n\nNO STOCKS PRESENT CURRENTLY";

getch();

}

S.close();

}

void DispC()

{

int k=0;

Customer C1;

fstream C("Customer.dat",ios::binary|ios::in|ios::out);

while(C.read((char\*)&C1,sizeof(C1)))

{

C1.ShowC();

k++;

}

if(k==0)

{

clrscr();

cout<<"\n\nNO REGISTERED CUSTOMERS ";

getch();

}

C.close();

}

void DeleteBill()

{

remove("Bill.dat");

}

void Buy(int no, char Identifier[50])

{

fstream B("Bill.dat",ios::binary|ios::app|ios::in|ios::out);

fstream S("Stock.dat",ios::binary|ios::in|ios::out);

Stock S1;

while(S.read((char\*)&S1,sizeof(S1)))

{

if(stricmp(S1.Rtnch(),Identifier)==0)

{

S1.Decrement(no);

S.seekp(S.tellp()-sizeof(S1),ios::beg);

S.write((char\*)&S1,sizeof(S1));

S1.Change(no);

B.write((char\*)&S1,sizeof(S1));

}

}

S.close();

B.close();

}

void Update(int temp)

{

clrscr();

Stock S1;

int no,k=0;

fstream S("Stock.dat",ios::binary|ios::in|ios::out);

while(S.read((char\*)&S1,sizeof(S1)))

{

if(S1.Rtnid()==temp)

{

cout<<"Enter the number of stock recieved ";

cin>>no;

S1.Gain(no);

S.seekp(S.tellp()-sizeof(S1),ios::beg);

S.write((char\*)&S1,sizeof(S1));

k++;

}

}

if(k==0)

{

cout<<"\n\n Item Id doesnt exist ";

getch();

}

S.close();

}

void ShopStart(long temptel)

{

int k=0;

Customer C1;

fstream C("Customer.dat",ios::binary|ios::in|ios::out);

while(C.read((char\*)&C1,sizeof(C1)))

{

if(C1.Rtntel()==temptel)

{

cout<<"\n\n\n\n";

C1.ShowC();

cout<<"\n\nLets Shop now ";

getch();

k++;

}

}

if(k==0)

{

cout<<"\n\n\n Please register first";

getch();

CustomerF();

}

C.close();

}

void Purchase(char\* product)

{

int qty1;

clrscr();

cout<<"\n\nEnter the quantity ";

cin>>qty1;

Buy(qty1,product);

cout<<"\n\n\n\nItem Saved in cart ";

getch();

clrscr();

}

void ShowPrice(char \* Name)

{

Stock S1;

int k=0;

fstream S("Stock.dat",ios::binary|ios::in|ios::out);

while(S.read((char\*)&S1,sizeof(S1)))

{

if(stricmp(Name,S1.Rtnch())==0)

{

clrscr();

cout<<"\n\n\nThe Price of item "<<Name<<" is = "<<S1.Rprice();

cout<<"\n\n\n\n\nDiscount is = "<<S1.Rdisc();

cout<<"\n\n\n\n\n Press Enter to buy the product ";

getch();

k++;

Purchase(Name);

}

}

if(k==0)

{

clrscr();

cout<<"\n\n Item out of Stock ";

getch();

clrscr();

}

}

void Bill(long tel)

{

Customer C1;

Stock S;

fstream B("Bill.dat",ios::binary|ios::in);

clrscr();

cout<<"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*INVOICE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n";

getch();

cout<<"ITEM ID NAME Quantity Total Price Discounted Price ";

while(B.read((char\*)&S,sizeof(S)))

{

cout<<"\n";

S.DispB();

getch();

cout<<"\n";

}

int tpoints=0;

fstream C("Customer.dat",ios::binary|ios::out|ios::in);

while(C.read((char\*)&C1,sizeof(C1)))

{

if(tel==C1.Rtntel())

{

tpoints=Amount/10.0;

C1.Updatepoint(tpoints);

C.seekp(C.tellg()-sizeof(C1),ios::beg);

C.write((char\*)&C1,sizeof(C1));

}

}

C.close();

getch();

B.close();

}

int Login()

{

clrscr();

int k=0;

char pass[20]="project410";

start:

clrscr();

char check[20];

cout<<"\n\n ENTER PASSWORD : ";

for(int i=0;i<10;i++)

{

check[i]=getch();

cout<<"\*";

}

check[i]='\0';

if(strcmp(check,pass)==0)

{

cout<<"\n\n\nACCESS GRANTED ";

getch();

return 1;

}

else

{

k++;

cout<<"\\n\n\nWRONG PASSWORD";

getch();

cout<<"\n\n\n\n\nYou have "<<3-k<<" chances remaining ";

getch();

clrscr();

if(k==3)

{

cout<<"\n\n\n\nAccount FROZEN ";

getch();

exit(0);

}

goto start;

}

}

void SearchItem(int code)

{

fstream S("Stock.dat",ios::in|ios::binary);

int k=0;

Stock S1;

while(S.read((char\*)&S1,sizeof(S1)))

{

if(code==S1.Rtnid())

{

k++;

clrscr();

S1.ShowS();

getch();

}

}

if(k==0)

{

clrscr();

cout<<"\n\nItem out of Stock";

getch();

}

}

void DeleteC(long tel)

{

int k=0;

fstream C("Customer.dat",ios::binary|ios::in);

fstream Temp("Temp.dat",ios::binary|ios::in|ios::out);

Customer C1;

while(C.read((char\*)&C1,sizeof(C1)))

{

if(tel!=C1.Rtntel())

{

Temp.write((char\*)&C1,sizeof(C1));

}

else

{

cout<<"\n\n DATA with TELEPHONE number "<<tel<<" DELETED ";

getch();

k++;

}

}

if(k==0)

{

cout<<"\n\n\n NO such data found";

getch();

}

remove("Customer.dat");

rename("TEMP.dat","Customer.dat");

C.close();

Temp.close();

}

void DeleteI()

{

char ch[20];

cout<<"\n\n\n\nEnter the item name to be deleted from cart ";

gets(ch);

int k=0;

fstream B("Bill.dat",ios::binary|ios::in);

fstream TempI("ChangeI.dat",ios::binary|ios::in|ios::out);

Stock S1;

while(B.read((char\*)&S1,sizeof(S1)))

{

if(strcmpi(ch,S1.Rtnch())!=0)

{

TempI.write((char\*)&S1,sizeof(S1));

}

else

{

cout<<"\n\n Item DELETED from CART ";

getch();

k++;

}

}

if(k==0)

{

cout<<"\n\n\n NO such item in cart";

getch();

}

remove("Bill.dat");

rename("ChangeI.dat","Bill.dat");

B.close();

TempI.close();

}

//--------------------------------------------ADMIN--------------

void Admin()

{

clrscr();

Stock X;

Customer Y;

int c1;

while(1)

{

clrscr();

cout<<"\n\n1.Add new Stocks ";

cout<<"\n\n2.Display all stocks ";

cout<<"\n\n3.Update Quantity";

cout<<"\n\n4.Search Item";

cout<<"\n\n5.Customer Details ";

cout<<"\n\n6.Delete Customer";

cout<<"\n\n0.Back to Main Menu";

cout<<"\n\n\nEnter choice : ";

cin>>c1;

switch(c1)

{

case 1:clrscr();

cout<<"\n\nEnter data ";

cout<<"------------------------------------------------------";

X.GetS();

AddS(X);

break;

case 2:clrscr();

cout<<"---------DATA ----------- \n\n";

DispS();

break;

case 3:clrscr();

int t;

cout<<"\n\nEnter the item number whose quantity has to be updated ";

cin>>t;

Update(t);

break;

case 4:clrscr();

int ctemp;

cout<<"\n Enter the item code you want to search for ";

cin>>ctemp;

SearchItem(ctemp);

break;

case 5:clrscr();

cout<<"-------------------DETAILS---------------------------";

getch();

DispC();

getch();

break;

case 6:clrscr();

long tel=0;

cout<<"Enter the telephone number of the customer whose data is to be deleted ";

cin>>tel;

DeleteC(tel);

break;

case 0:Choice();

break;

default:cout<<"\n\n Try again ";

getch();

}

}

}

//---------------------------------CUSTOMER-------------------------

void CustomerF()

{

Stock X;

Customer Y;

int c2;

while(1)

{

mainc:

clrscr();

cout<<"\n\n1.New customer ";

cout<<"\n\n2.Registered customer ";

cout<<"\n\n0.Return to main menu ";

cout<<"\n\n\n\nEnter your choice : ";

cin>>c2;

switch(c2)

{

case 1:clrscr();

cout<<"\n\nEnter your new data to become our member \n\n";

Y.GetC();

AddC(Y);

cout<<"\n\nThanks for registering ";

getch();

goto mainc;

break;

case 2:clrscr();

long temptel;

cout<<"Enter your Telephone number ";

cin>>temptel;

ShopStart(temptel);

getch();

int c21;

while(1)

{

back:

clrscr();

cout<<"\n\n1.Electronics ";

cout<<"\n\n2.Sports Equipment ";

cout<<"\n\n3.Footwear ";

cout<<"\n\n4.Books and Media";

cout<<"\n\n0.Proceed to billing ";

cout<<"\n\n Enter the category which you want to purchase ";

cin>>c21;

switch(c21)

{

case 1:clrscr();

int e1;

while(1)

{

clrscr();

cout<<"\n\n1.Mobile phone";

cout<<"\n\n2.Laptop ";

cout<<"\n\n3.Television ";

cout<<"\n\n4.Tablets ";

cout<<"\n\n0.Shop other category/Proceed to Billing";

cout<<"\n\nEnter your choice ";

cin>>e1;

switch(e1)

{

case 1:ShowPrice("Mobile Phone");

break;

case 2:ShowPrice("Laptop");

break;

case 3:ShowPrice("Television");

break;

case 4:ShowPrice("Tablet");

break;

case 0:goto back;

break;

default: cout<<"\n\nInvalid";

getch();

}

}

case 2:clrscr();

int s1;

while(1)

{

cout<<"\n\n1.Football";

cout<<"\n\n2.Cricket Bats ";

cout<<"\n\n3.Badminton Racquets";

cout<<"\n\n4.Sports Jerseys ";

cout<<"\n\n0.Shop other category";

cout<<"\n\nEnter your choice/Procced to Billing ";

cin>>s1;

switch(s1)

{

case 1:ShowPrice("Football");

break;

case 2:ShowPrice("Cricket Bat");

break;

case 3:ShowPrice("Badminton Racquets");

break;

case 4:ShowPrice("Sports Jerseys");

break;

case 0:goto back;

break;

}

}

case 3:clrscr();

int f1;

while(1)

{

cout<<"\n\n1.Casual Shoes";

cout<<"\n\n2.Crocs";

cout<<"\n\n3.Formal Shoes ";

cout<<"\n\n4.Slippers";

cout<<"\n\n0.Shop other category/Proceed to Billing";

cout<<"\n\nEnter your choice ";

cin>>f1;

switch(f1)

{

case 1:ShowPrice("Casual Shoes");

break;

case 2:ShowPrice("Crocs");

break;

case 3:ShowPrice("Formal Shoes");

break;

case 4:ShowPrice("Slippers");

break;

case 0:goto back;

break;

}

}

case 4:clrscr();

int b1;

while(1)

{

cout<<"\n\n1.Fictional Books";

cout<<"\n\n2.E-book ";

cout<<"\n\n3.PS4";

cout<<"\n\n4.Board Games ";

cout<<"\n\n0.Shop other category/Proceed to Billing";

cout<<"\n\nEnter your choice ";

cin>>b1;

switch(b1)

{

case 1:ShowPrice("Fictional Books");

break;

case 2:ShowPrice("E-Book");

break;

case 3:ShowPrice("PS4");

break;

case 4:ShowPrice("Board Games");

break;

case 0:goto back;

break;

}

}

case 0:int cash;

clrscr();

char ch;

cout<<"\n\nDo you want to remove any item from cart ?(Y/N) ";

cin>>ch;

if(ch=='Y'||ch=='y')

{

DeleteI();

}

else

{

cout<<"\n\n PRESS ENTER TO VIEW BILL";

getch();

clrscr();

}

Bill(temptel);

cout<<"\n\n\n\n Total Cost : "<<Amount;

pay:

cout<<"\n\n\n\n Cash paid : ";

cin>>cash;

if(cash>=Amount)

cout<<"\n\n\n\n Balance : "<<cash-Amount;

else

{

cout<<"\n\nCash not sufficinet";

getch();

goto pay;

}

getch();

cout<<"\n\n\n\n\n\nTHANKS FOR SHOPPING";

cout<<"\n\n COME BACK AGAIN ";

getch();

exit(0);

break;

}

}

case 0:Choice();

break;

}

}

}

void Choice()

{

clrscr();

int c;

Amount=0;

while(1)

{

here:

cout<<"\n\n\n1.Administrator ";

cout<<"\n\n\n2.Customer ";

cout<<"\n\n\n0.EXIT \n";

cout<<"\n\n\nEnter your Choice: ";

cin>>c;

switch(c)

{

case 1:clrscr();

if(Login()==1)

Admin();

break;

case 2:clrscr();

CustomerF();

break;

case 0:exit(0);

break;

default:cout<<"\n\nInvalid !! Try again";

}

}

getch();

}

//-----------------------------MAIN-----------------------------------

void main()

{

clrscr();

DeleteBill();

cout<<"\n\n\n\n\n";

cout<<" \\ // |||||||||||| || // ||-- \n ";

cout<<" \\ // || || || // || -- \n ";

cout<<" \\ // || || || // || -- \n ";

cout<<" \\ // || || ||// || -- \n ";

cout<<" || || || --- ||\\ || -- \n ";

cout<<" || || || || \\ || -- \n ";

cout<<" || || || || \\ || -- \n ";

cout<<" || || || || \\ || -- \n ";

cout<<" || |||||||||||| || \\ || -- \n ";

getch();

cout<<"\n\n\n\n\n\nWelcome to the new era of Shopping";

cout<<"\n\n\n\nTo ensure safe surfing press enter only after hearing the BEEP sound";

cout<<"\a\a\a\a\a\a\a\a\a\a\a\a\a\a";

getch();

clrscr();

Choice();

getch();

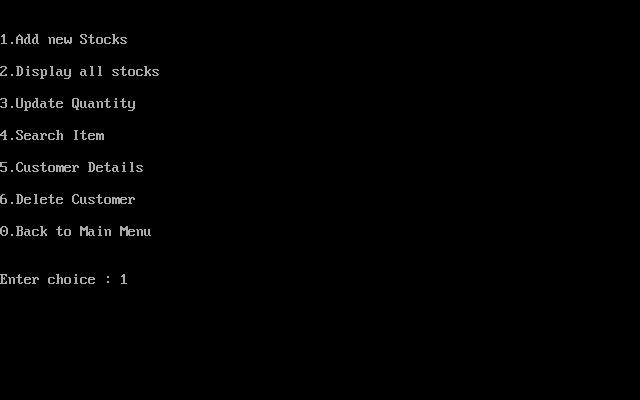
}

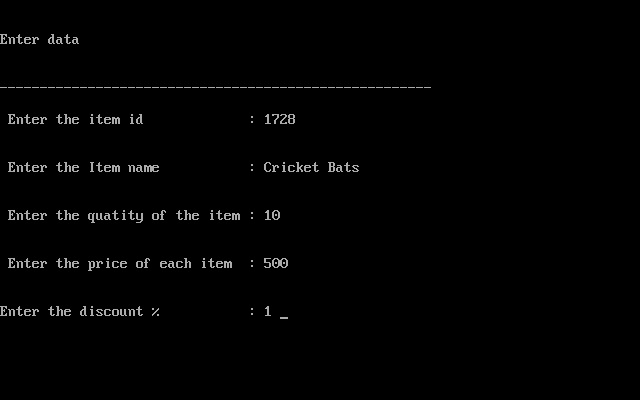
**OUTPUT**

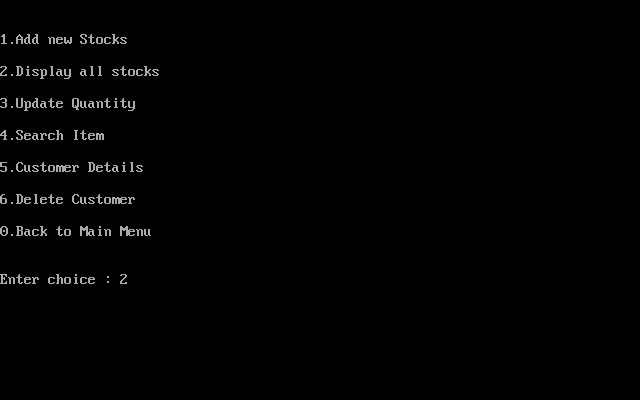


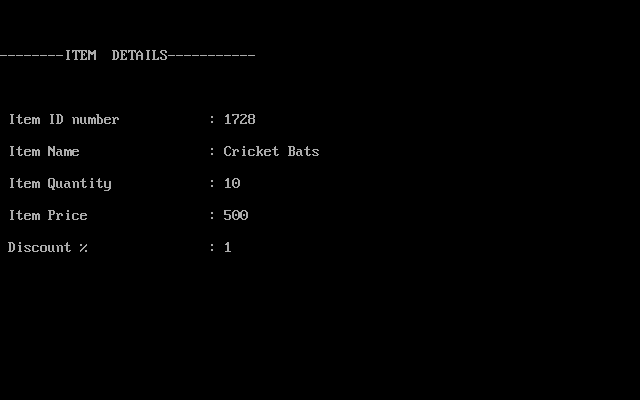


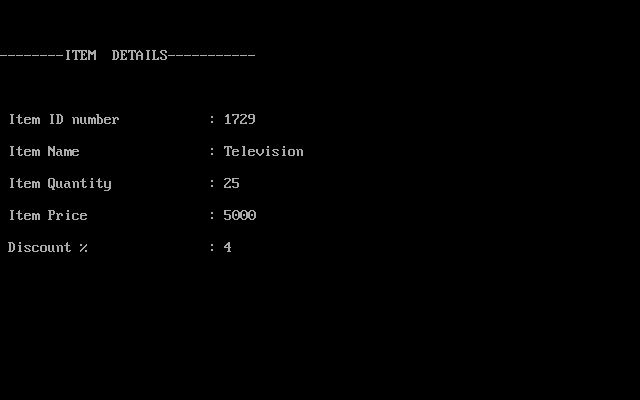


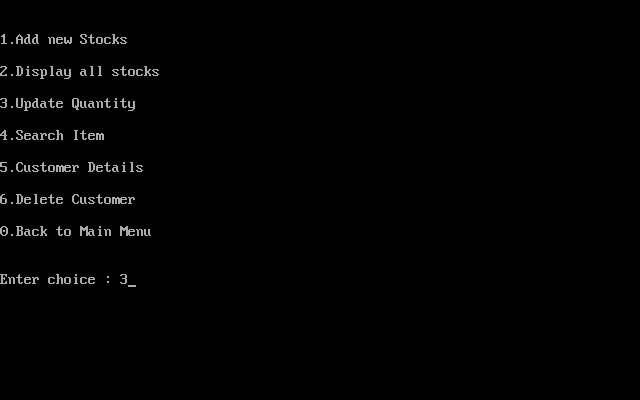


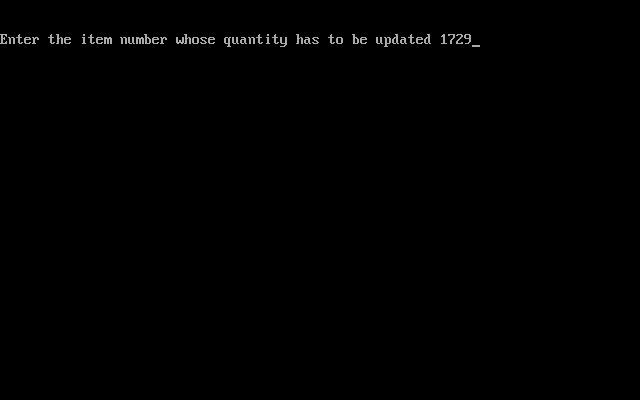




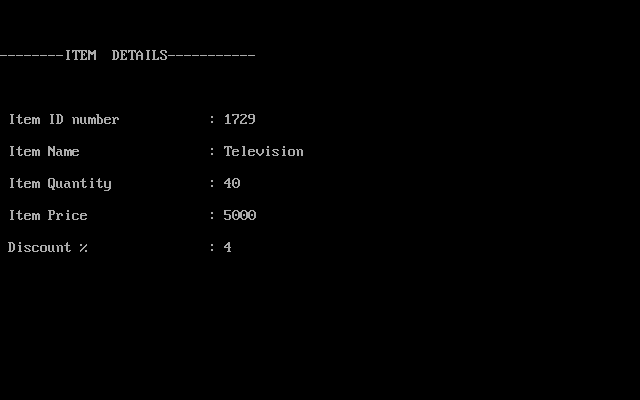


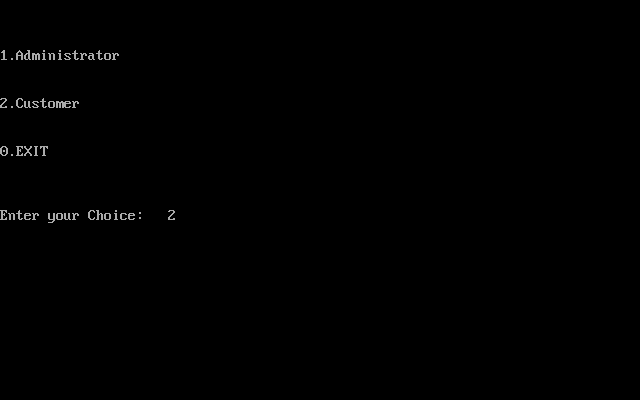


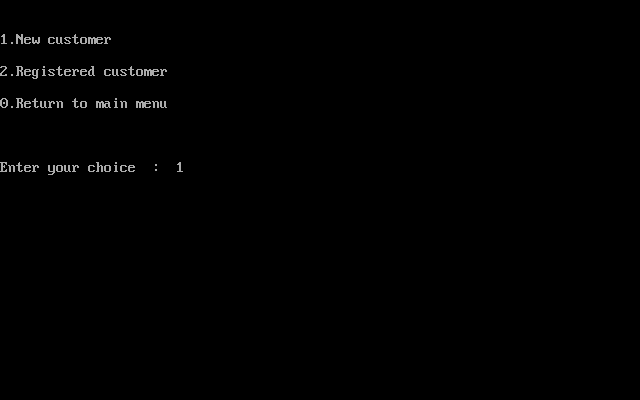


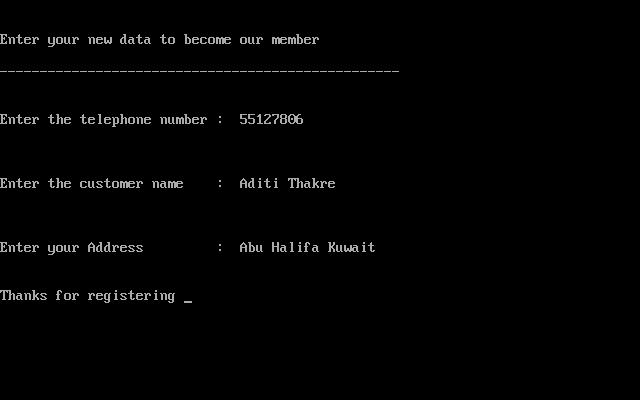






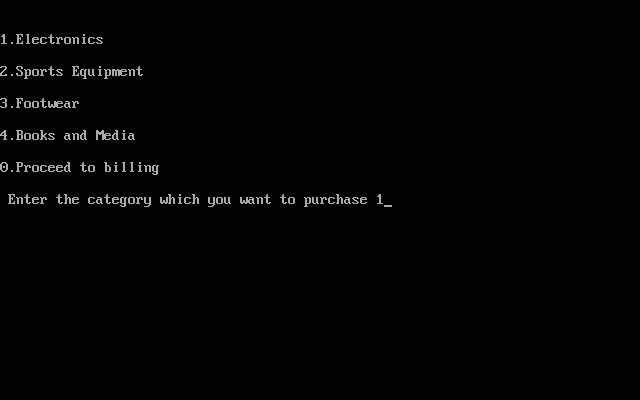


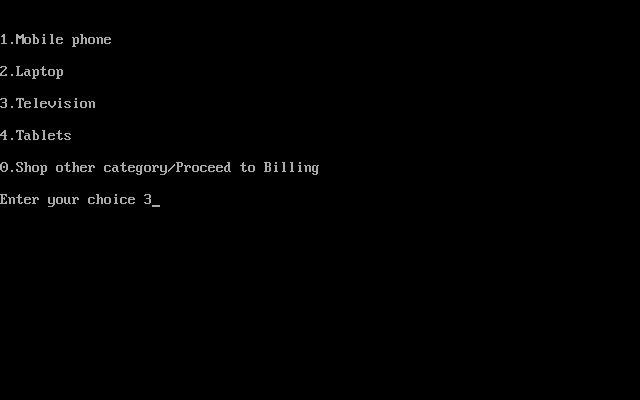


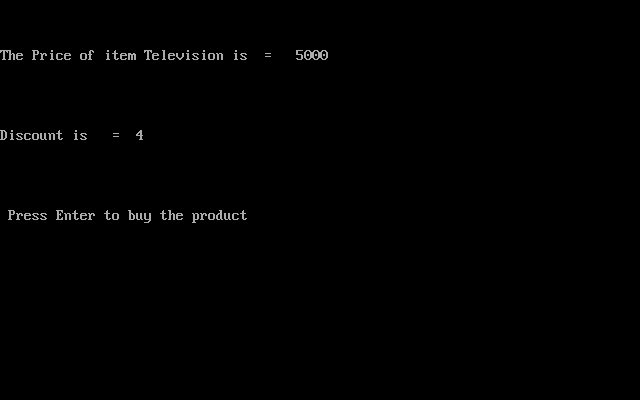




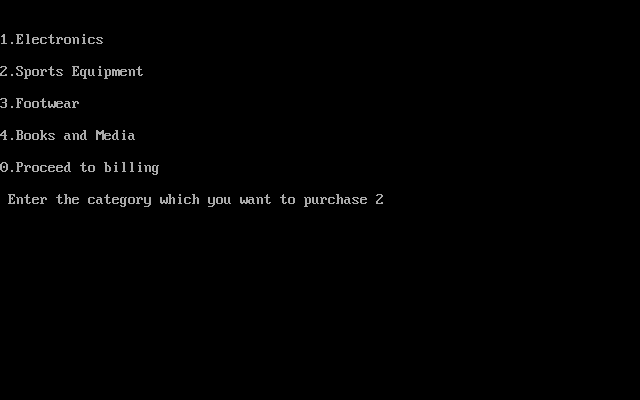


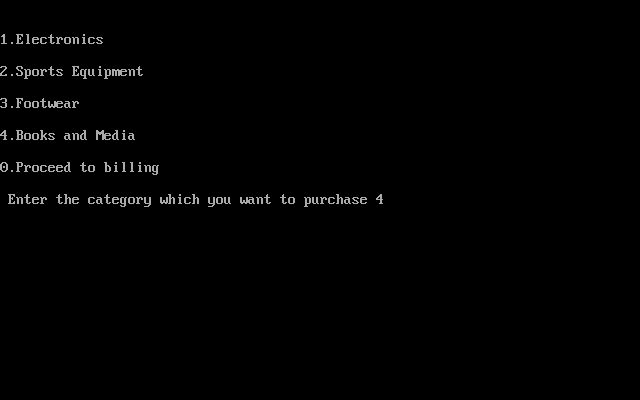


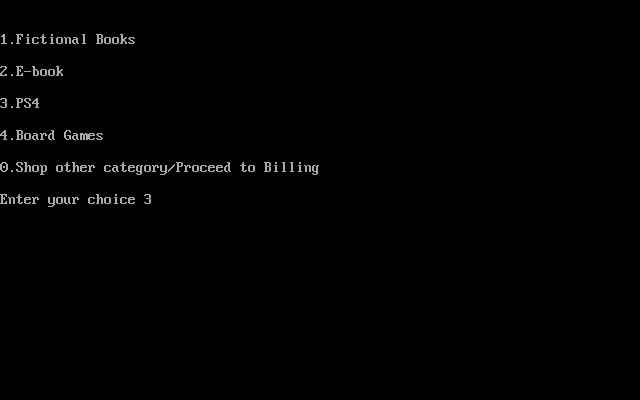


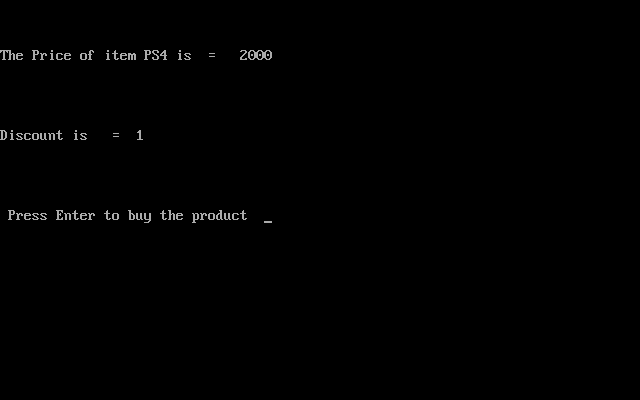


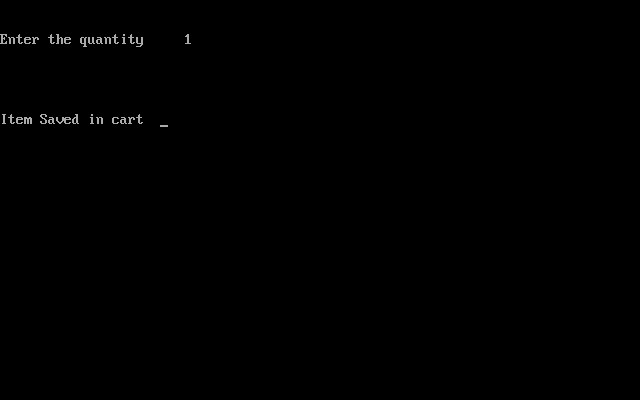


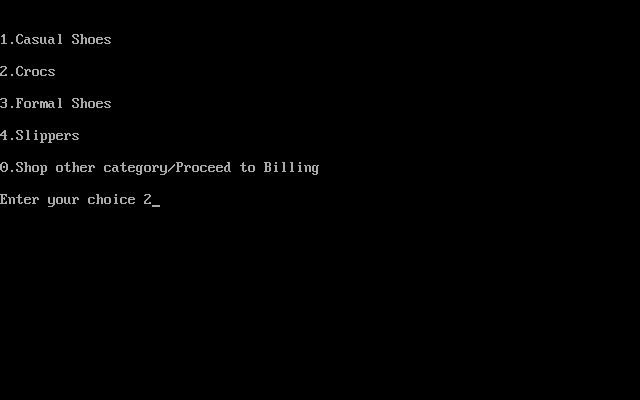


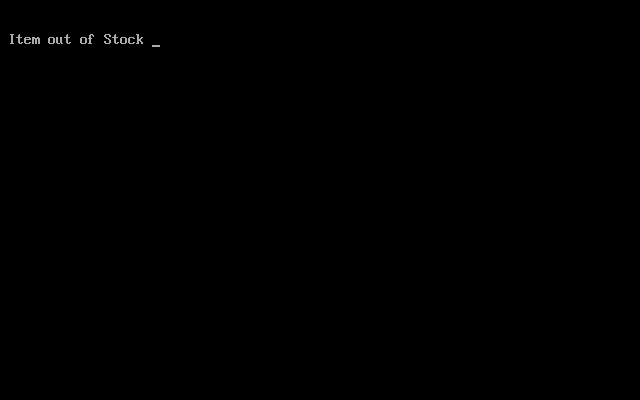


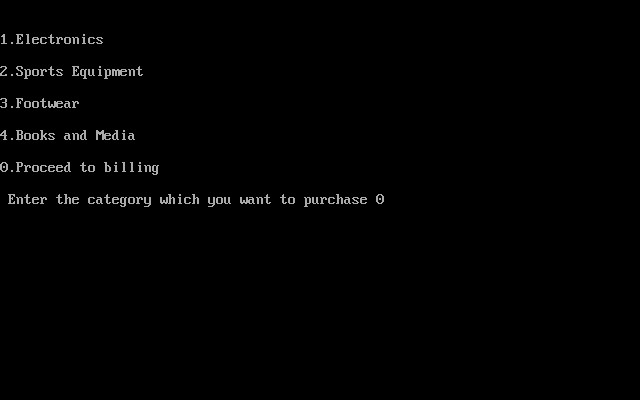




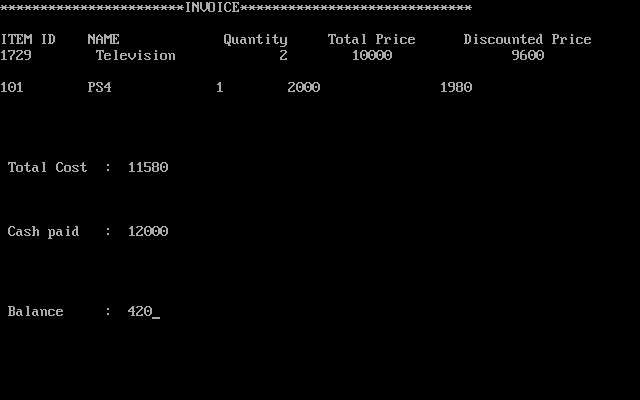


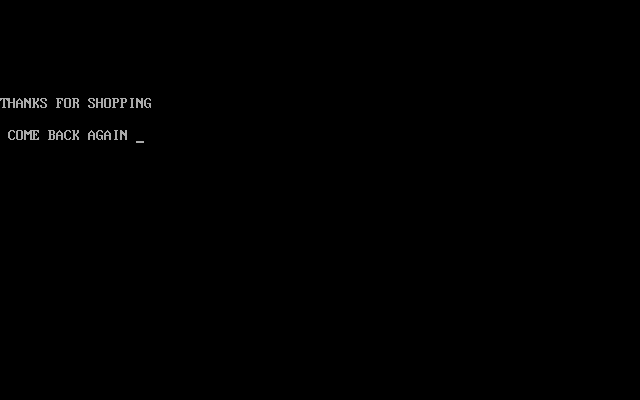












CONCLUSION

The program developed helps us in organising the management of the stocks effectively. The complexities in modifications of the existing items are achieved through simple interfaces like addition, deletion and modification. It can be used by both Customers and Administrators. It allows a customer to buy item from the Stock and Displays the Bill at the end of the Shopping. It has a proper relationship, example when an item is bought its quantity is decremented in the data of available stock. Through this program all the various types of data can be stored effectively and makes it easily accessible through user interface.

BIBLIOGRAPHY

* CLASS XII – COMPUTER SCIENCE BY SUMITA ARORA