NEW HORIZON
COLLEGE OF ENGINEERING
New Horizon Knowledge Park, Ring Road, Marathalli
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

"DEPARTMENT STORE MANAGEMENT SYSTEM"

A MINI PROJECT REPORT

Submitted by

SUCHALA K L [1NH19IS408-T]

Under the guidance of

Ms. Swathi. B

Sr. Assistant Professor, ISE, NHCE

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

INFORMATION SCIENCE AND ENGINEERING

FOR

COURSE NAME: MINI PROJECT

COURSE CODE: 19ISE391

NEW HORIZON
COLLEGE OF ENGINEERING
New Horizon Knowledge Park, Ring Road, Marathalli
Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade, Accredited by NBA

CERTIFICATE

Certified that the project work entitled **DEPARTMENT STORE MANAGEMENT SYSYEM** carried out by Ms. **SUCHALA K L**, bearing USN:**1NH19IS408-T,** a bonafide student of 3rd semester in partial fulfillment for the award of Bachelor of Engineering in Information Science & Engineering of the Visveswaraiah Technological University, Belagavi during the year 2019-20. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated. The project report has been approved as it satisfies the academic requirements in respect of Mini Project work prescribed for the said Degree.

Name & Signature of Guide	Name & Signature of HOD	Name & Signature of Principal
Ms. Swathi. B	Dr. Anandhii R J	Dr. Manjunatha

Examiners :

	Name	Signature
1.	……	…….
2.	……	……

ABSTRACT

This **mini project in C Department Store Management System** is a simple console built application without graphic. In this project, you can manage a typical 'fashion wear' department store. You can add goods, edit goods, search, delete and display the goods.

File handling has been used to record the information (rate, quantity, name and code) of the added goods. You can search the goods by rate, code or quantity. And, similar goes for display; you can display the items by quantity, rate or code.

ACKNOWLEDGEMENT

Any project is a task of great enormity and it cannot be accomplished by an individual without support and guidance. I am grateful to a number of individuals whose professional guidance and encouragement has made this project completion a reality.

I have a great pleasure in expressing my deep sense of gratitude to the beloved Chairman **Dr. Mohan Manghnani** for having provided me with a great infrastructure and well-furnished labs.

I take this opportunity to express my profound gratitude to the Principal **Dr.Manjunatha** for his constant support and management.

I am grateful to **Dr. R J Anandhi** , Professor and Head of Department of ISE, New Horizon College of Engineering, Bengaluru for her strong enforcement on perfection and quality during the course of my project work.

I would like to express my thanks to the guide **Ms. Swathi. B** , Senior Assistant Professor, Department of ISE, New Horizon College of Engineering, Bengaluru who has always guided me in detailed technical aspects throughout my project.

I would like to mention special thanks to all the Teaching and Non-Teaching staff members of Information Science and Engineering Department, New Horizon College of Engineering, Bengaluru for their invaluable support and guidance.

SUCHALA K L

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

A mercantile establishment retail store sales outlet may be a retail establishment giving a good vary of trade goods in several product classes referred to as "departments". In trendy major cities, the sales outlet created a dramatic look within the middle of the nineteenth century, and for good reshaped searching habits, and therefore the definition of service and luxury.

Today, departments usually embody the following: vesture, furniture, home appliances, toys, cosmetics, housewares, gardening, toiletries, sports equipment, hump yourself, paint, and hardware. to boot, alternative lines of merchandise like food, books, jewelry, natural philosophy, stationery, equipment, baby merchandise, and merchandise for pets ar typically enclosed. Customers usually verify close to the front of the shop, though some stores embody sales counters among every department. Some stores are one amongst several among a bigger distributor, whereas others are freelance retailers. within the Nineteen Seventies, they came beneath serious pressure from discounters, and have come back beneath even heavier pressure from e-commerce sites since 2010. Big-box stores, hypermarkets, and discount stores ar such as historical malls.

Department stores are usually classified in step with the types of product they carry and therefore the costs they charge; typical classes embody discount, general merchandise, fashion or haute couture, and specialty. several supply further services, together with wrapping, alterations, delivery, and private searching.
The development of malls was connected to the expansion within the nineteenth century of enormous population centers, transportation, and therefore the harnessing of electricity for power and lighting. The Bon Marche in Paris, that began as atiny low search within the early nineteenth century, is wide thought of the primary sales outlet. John Wanamaker carried the construct to the u. s. in 1875 by getting a rail-freight depot in his native City of Brotherly Love and populating it with a set of specialty retailers. Among his innovations were the introduction of value tags and therefore the development of aggressive advertising programs for his growing chain of stores.

1.1 PURPOSE:

Departmental store is a place where we get all our daily use products. These products are the basic requirements such as clothes, utensils, grocery, toys, watches etc. From the administrator point of view, the management of these products is not an easy job. Since they need to keep track of the product, their various suppliers, the customer details, their employees etc. All these jobs are done mostly manually or using some old software.

1.2 PROBLEM DEFINITION:

In the departmental stores most of the work is done manually by maintaining registers.
This involves many limitations such as:

- Increases the paper work
- The time to access the data increases
- Chances of information leakage or loss of information increases
- Maintaining records is not integrated
- Duplication of data

They also use DOS based system with monotonous Look and duplication of data having many gaps and inefficiency in forecasting due to batch process or lack of information due to either manual systems or old software. So, there is a lack of tools for manipulation of data.

1.3 OBJECTIVES:

The objectives of the project is to create a system that:

- Provides security to the data
- According to the user, provides that relevant data
- Handles the inventory, accounting and employee information
- Auto Checking the stocks, performence of the employee
- Less prone to errors
- Efficient data storage that will reduce the redundancy of data
- Less chances of information leakage

1.4 METHODOLOGY:

In this project I have used basic C concepts. The main aim is to billing the product by online.
In the departmental stores most of the work is done manually by maintaining registers.
Department stores are usually classified in step with the types of product they carry and therefore they charge. The development of malls was connected to the expansion within the nineteenth century of enormous population centers, transportation and therefore the hamessing of electricity for power and lighting.

CHAPTER 2

SOFTWARE

2.1 System Requirements Specification :

2.1.1 Hardware Requirements:

Processor: Code blocks
RAM:
Hard Disk:
Input Devices: Keyword (standard 102 keys) and Mouse
Output Device: Monitor

2.1.2 Software Requirements:

Operating System: Windows 8

2.2 About Software used:

Code::Blocks is a free, open-source cross-platform IDE that supports multiple compilers including GCC,Clang and Visual C++. It is developed in C++ using wxWidgets as the GUI toolkit. Using plugin architecture, its capabilities and features are defined by the provided plugins. Currently, Code::Blocks is an oriented towards C, C++ and Fortran. It has the custom build system and optional make support.

Features of Code::Blocks:

Compilers

Code::Blocks supports multiple compilers, including GCC, MinGW, Digital Mars, Microsoft Visual C++, Borland C++, LLVM Clang, Watcom, LCC and the Intel C++ compiler. Although the IDE was designed for the C++ language, there is some support for other languages, including Fortran and D. A plug-in system is included to support an other programming languages.

Code editor

The IDE features an syntax highlighting and code folding (through its Scintilla editor component), C++ code completion, a class browser, a hex editor and many other utilities. Opened files are organized into tabs. The code editor supports a font and font size selection and personalized syntax highlighting of colours.

Debugger

The Code::Blocks debugger has a full breakpoint support. It will also allows the user to debug their program by having access to an local function symbol and argument display, the user-defined watches, call stack, disassembly, custom memory dump, thread switching, a CPU registers and GNU Debugger Interface.

GUI designer

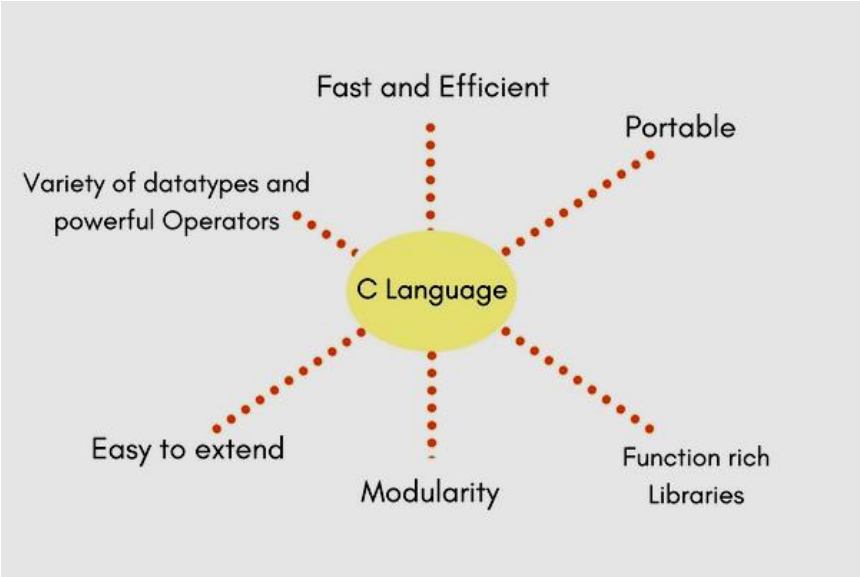
As of version 13.12 Code::Blocks comes with an GUI designer called a wxSmith. It is a derivative port of wxWidgets version 2.9.4 To make a complete wxWidgets application, the appropriate wxWidgets SDK must be installed.

User migration

Some of the Code::Blocks features are targeted at the users migrating from the other IDE's – these are include Dev-C++, Microsoft Visual C++ project import (MSVC 7 & 10), and the Dev-C++ Devpak support.

Project files and build system

Code::Blocks is uses a custom build system, which stores its information in a XML-based project files. It can optionally use external makefiles, which simplifies interfacing with a projects using the GNU or qmake build systems.



CHAPTER 3

ARCHITECTURE OF EXECUTION

3.1 System Analysis

The part is elaborated appraisal of existing system. This appraisal includes however the system works & what it will. It additionally embody searching for in additional detail what are the issues with system & what user needs from a replacement system or any new changes in system. The output of this page ends up in model of system. The model describes the system operate & information & system data flow. The part additionally contains the detail set of user demand & this demand ar wont to set objectives for a replacement system.

3.2 Current System :

A emporium retail store sales outlet outlet could be a retail establishment that focuses on mercantilism a large vary of product while not one predominant merchandise line. A grocery store could be a store that sells a spread of various product like attire, furniture, appliances, natural philosophy, and in addition choose alternative lines of product like paint, hardware, toiletries, cosmetics, equipment, jewellery, toys, and sports equipment garments, fashion accessories, cd's, dvd's, books, eatables, etc. Such a market carries sizable amount of transactions on a each day like sale of varied product, shopping for these product from suppliers, recognizing merchandise that square measure ought to be ordered, creating payments to the provider etc. until recently, all such info was hold on within the sort of paper. thus it becomes tough to manage such Brobdingnagian amounts of data by the present manual system. conjointly information} hold on paper isn't reliable and retrieval of any data becomes inconvenient and time intense. This results in wastage of a great deal of your time, effort and area.

3.3 Proposed system:

The proposed system may be included following features:

- Creating an database for the stores containing the information present with them on the paper in an existing system.
- Access to the database will be based on a logon-id and password. Different person will have a different login-id and to distinct set of access rights.
- The developed system will also print bills and reports and also maintain various transactions of the super-market.
- It will have an comfortable and user friendly GUI.
- And Also the system is intended to take very few inputs from the user.

3.4 Advantages of Proposed System:

- User friendly, accurate and robust systems.
- Store various product information, it's supplies information & their sales.
- Handles order placing and delivery.
- Store information about delivery of an orders & their payments.
- Manage stock of all the products.
- Production of a bills.
- Security of a data.
- Integration of an all the functions in to one system.
- Remove the redundancy of data.
- Remove the inconsistency of data.

3.5 Requirement Gathering:

The works begins with the information of gathering. Developer and client meet and outline the objectives for the code, determine no matter necessities are known , and description areas wherever more definition is necessary. In our code development method, we tend to gathered info from the assorted developers and users to grasp concerning the present system. this can facilitate North American nation to develop a planned system.

Analysis: once gathering all the necessities from the user, we'd like to distribute the assorted necessities that we've got already collected. we tend to discard the surplus necessities & solely think about the relevant necessities. conjointly some necessities that haven't been thought of however are greatly required conjointly have to be compelled to be discovered to avoid changes within the project within the later stages.

3.6 Design:

The step focuses on four attributes of the program

- Data structure
- Software Architecture
- Procedural Detail
- Interface Characterization

CHAPTER 4

FLOW CHARTS

4.1 WHAT IS FLOWCHART?

A flow chart is a graphical representation of the steps. It was originated from technology as a tool for representing algorithms and programming logic however had extended to use all told alternative kinds of processes. Nowadays, flowcharts play a very necessary role in displaying data and aiding reasoning. they assist North American country visualize complicated processes, or build specific the structure of issues and tasks. A flow sheet also can be accustomed outline a method or project to be enforced.

4.2 FLOWCHART SYMBOLS:

Different flowchart the shapes have a different conventional meanings. The meanings of some of the more common pages are as follows:

Terminator

The terminator symbol is represents an starting or ending point of the system.

Process

A box indicates some of the particular operation.

Document

This represents a printout, such as a document or an report.

Decision

A diamond represents an decision or branching point. Lines coming out from the diamond it will indicates the different possible situations, leading to the different sub-processes.

Data

It represents a information entering or leaving from the system.an input is might be an order from the customer. Output can be a product to be delivered.

On-Page Reference

This symbol will contains a letter inside, it indicates that the flow continues on a matching symbol containing the same letter somewhere else on the same page.

Off-Page Reference

This symbol would contain a letter inside. It indicates that the flow continues on matching symbol containing the same letter somewhere else on a different page.

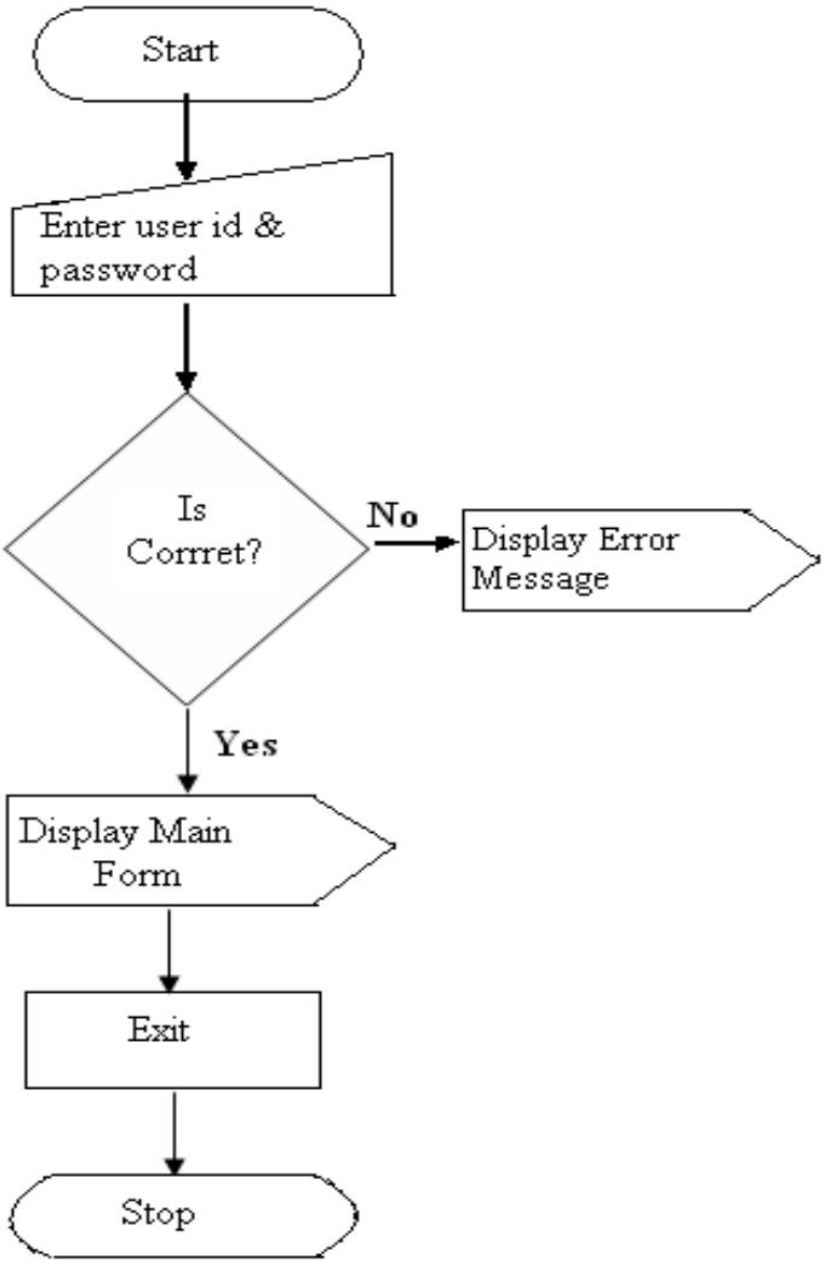
Delay or Bottlene

Identifies a dealy or a bottle neck.

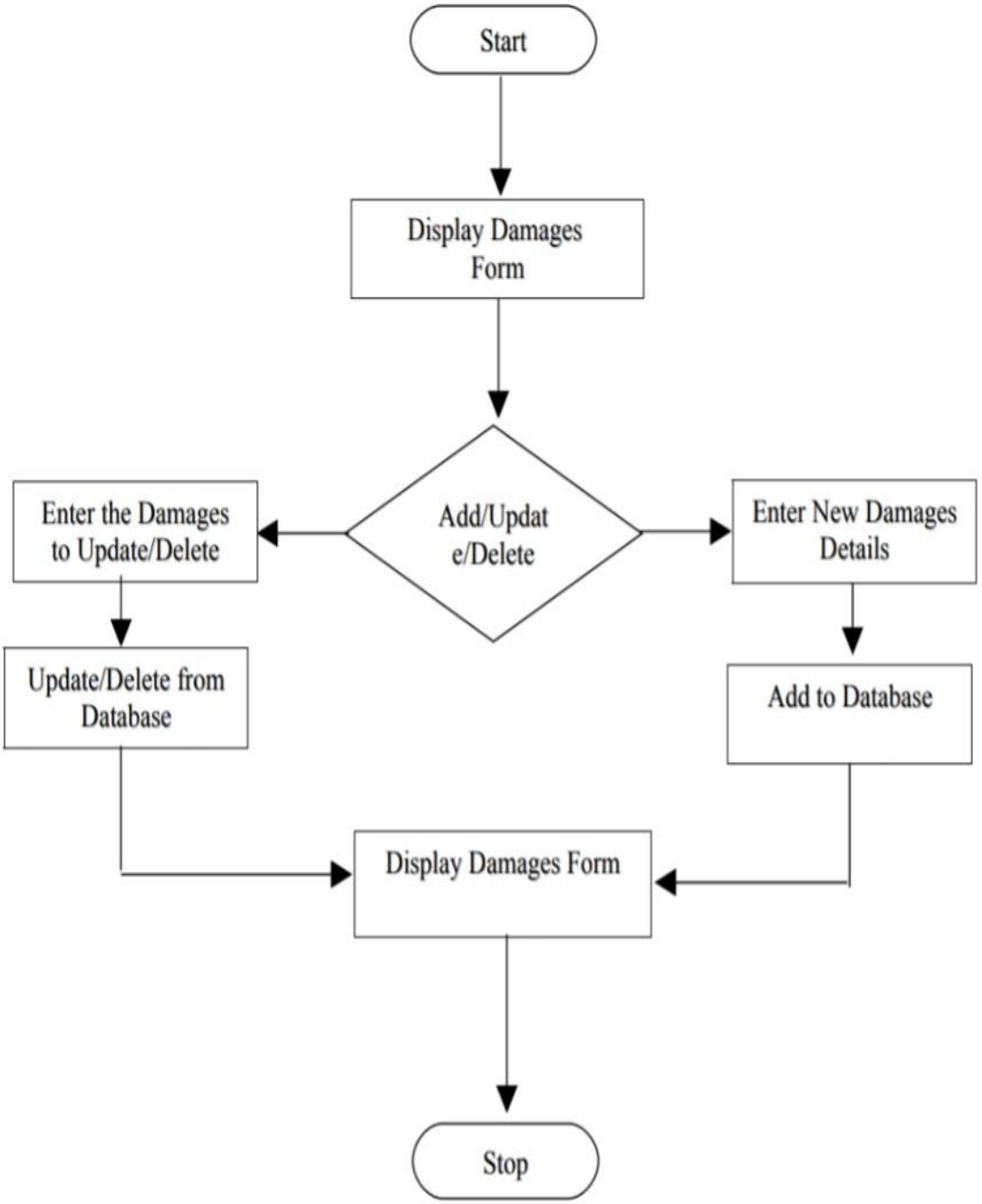
Flow

Lines represents the flow of the sequence and direction of a process.

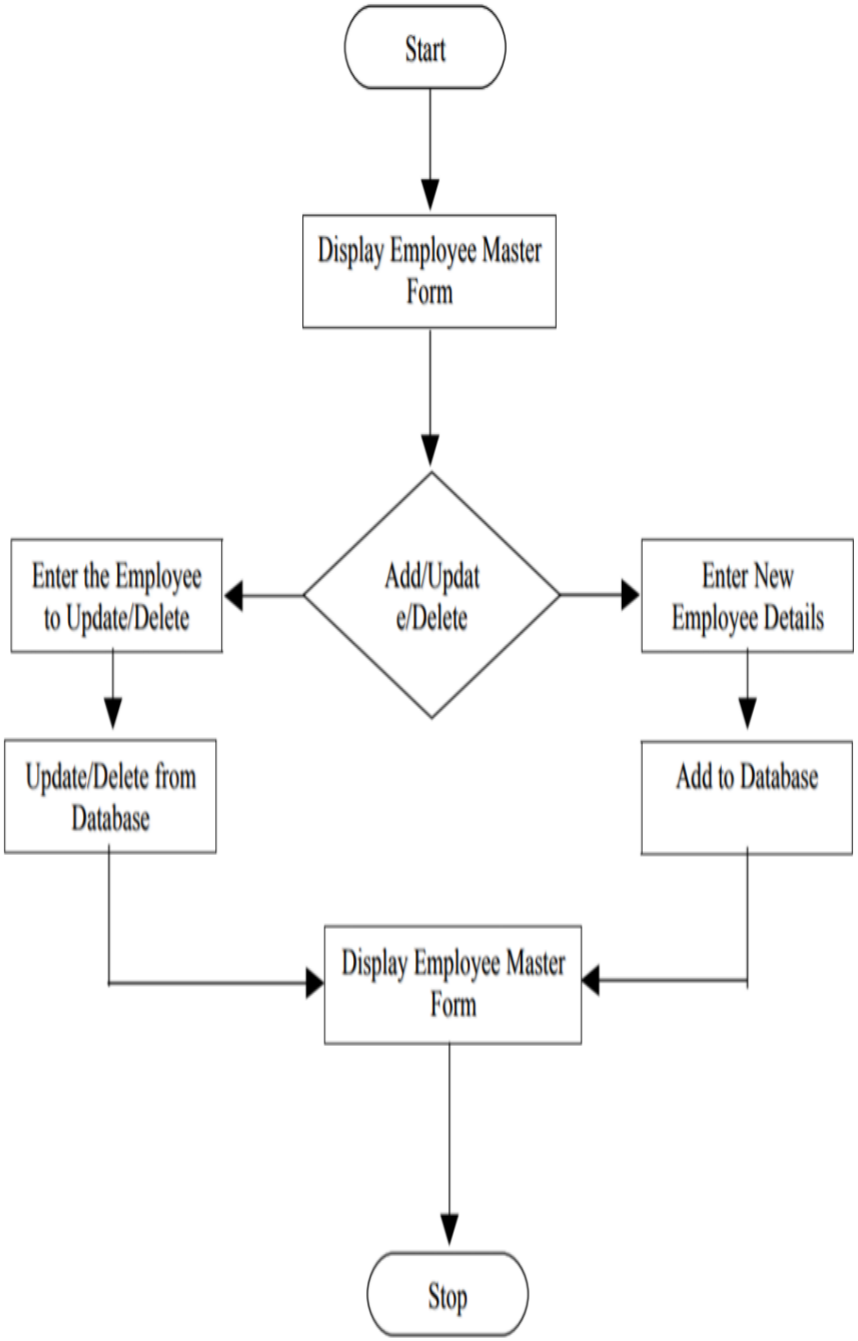
FLOW CHART FOR LOGIN:



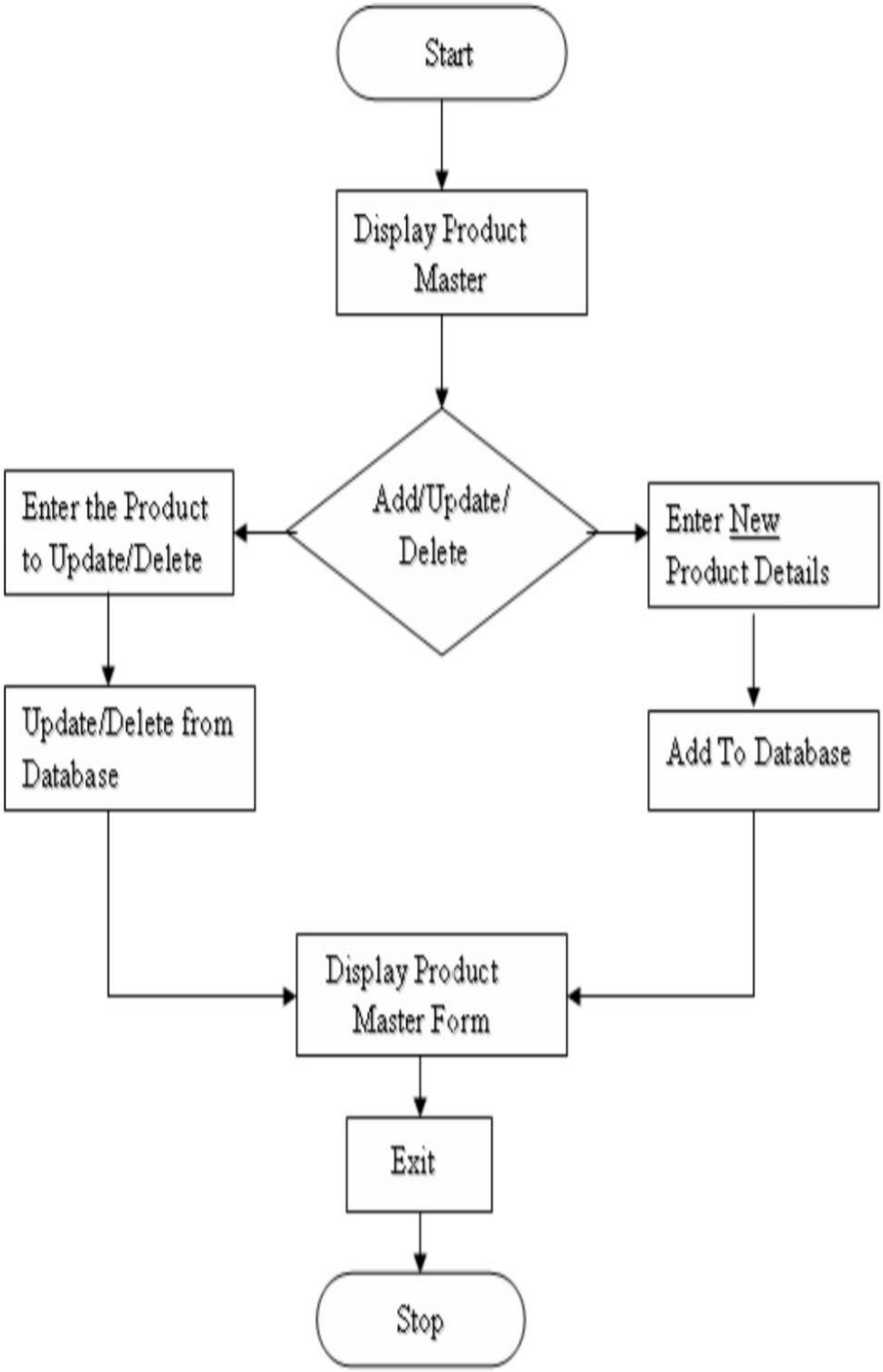
Flowchart for Add/Update/Delete Damages:



Flow chart for Add/Delete/Update Employee Information:



Flow chart for Add/Delete/Update Product Information:



When to Draw Flowchart?

- It helps to clarify complex processes.
- It identifies steps that do not add value to the internal or external customer, including delays; needless storage and transportation; unnecessary work, duplication, and added expense; breakdowns in communication.
- It helps team members gain a shared understanding of the process and use this knowledge to collect data, identify problems, focus discussions, and identify resources.
- It serves as a basis for designing new processes.

APPENDIX A

PSUEDO CODE

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
#include<ctype.h>
#include<windows.h>

#define ANS fifteen
#define ACS four
COORD coord= ;
void gotoxy(int x,int y)

void c_code(char[]);
int check(char[]);

typedef struct
 rec;
rec item;

/*declaration of show functions*/
void curser(int);
void dbill();
void d_mainmenu();
void display(rec *,int,int);
void window(int,int,int,int);
void dis_con();
void d_search();
void highlight(int,int);

/*declaration of main menu functions*/
void bill() ;
void edit();void add();
void del();
void exit();

/*declaration of display submenu functions*/
void d_code();
void d_rate();
void d_quan();
void d_all();

/*start of main*/
int main()
```

```
void d_mainmenu()
;
   system("cls");
//textbackground(11);
//textcolor(0);
   window(25,50,20,32);
   gotoxy(33,18);
   printf("SEARCH MENU");
   for (i=0; i<=3; i++)
   void curser(int no)
{
   int count=1;
   char ch='0';
   gotoxy(30,23);
   while(1)
   {
      switch(ch)
      {
      case 80:
         count++;
         if (count==no+1) count=1;
         break;
      case 72:
         count--;
         if(count==0) count=no;
         break;
      }
      highlight(no,count);
      ch=getch();
      if(ch=='\r')
      {
         if(no==7)
         {
            if (count==1) bill() ;
            else if(count==2) add();
            else if(count==3) edit();
            else if (count==4) d_all();
            else if (count==5) d_search();
            else if (count==6) del();
            else   exit(0);
         }
         if(no==4)
         {
            if (count==1) d_code();
            else if (count==2)d_rate();
            else if (count==3) d_quan();
```

```
            else d_mainmenu();
          }
       }
    }
}

void highlight(int no,int count)
{
    if (no==4)
    {
       //textbackground(11);
       //textcolor(0);
       gotoxy(30,23);
       printf("   By Code        ");
       gotoxy(30,24);
       printf("   By Rate        ");
       gotoxy(30,25);
       printf("   By Quantity     ");
       gotoxy(30,26);
       printf("   Back to main menu");
       //textcolor(0);
       //textbackground(2);
       switch (count)
       {
       Case 1:
          gotoxy(30,23);
          printf(" - By Code        ");
          break;
       case 2:
          gotoxy(30,24);
          printf(" - By Rate        ");
          break;
        case 3:
          gotoxy(30,25);
          printf(" - By Quantity     ");
          break;
        case 4:
          gotoxy(30,26);
          printf(" - Back to main menu");
          break;
       }
    }

    if(no==7)
    {
       //textbackground(11);
       //textcolor(0);
```

```
gotoxy (30,23);
printf("  Calculate Bill ");
gotoxy (30,24);
printf("  Add Goods     ");
gotoxy (30,25);
printf("  Edit Goods    ");
gotoxy (30,26);
printf("  Display All   ");
gotoxy (30,27);
printf("  Search        ");
gotoxy (30,28);
printf("  Delete Goods  ");
gotoxy (30,29);
printf("  Exit          ");
//textcolor(0);
//textbackground(2);
switch(count)
{
case 1:
   gotoxy (30,23);
   printf(" - Calculate Bill ");
   break;
case 2:
   gotoxy (30,24);
   printf(" - Add Goods      ");
   break;
case 3:
   gotoxy (30,25);
   printf(" - Edit Goods     ");
   break;
case 4:
   gotoxy (30,26);
   printf(" - Display All    ");
   break;
case 5:
   gotoxy (30,27);
   printf(" - Search         ");
   break;
case 6:
   gotoxy (30,28);
   printf(" - Delete Goods   ");
   break;
case 7:
   gotoxy (30,29);
   printf(" - Exit           ");
   break;
}
```

```
    }
}

void bill()
{
    char x[4]= {0};
    int j=29,q=0,       =0,i=1;
    float total=0,gtotal=0;
    FILE *file;
    file=fopen("record.txt","r+b");
    rewind(file);
    system("cls");
    dbill();
    gotoxy(26,15);
    printf("enter \"end\" to finish input");
    while(1)
    {
        gotoxy(25,18);
        printf("                ");
        gotoxy(25,19);
        printf("                ");
        gotoxy(25,18);
        printf("enter item code:");
        scanf("%s",x);
        if(strcmp(x,"end")==0)
            break;
        gotoxy(25,19);
        printf("enter quantity:");
        scanf("%d",&q);
        rewind(file);
        while(fread(&item,sizeof(item),1,file))
        {
            if((strcmp(item.code,x)==0))
            {
                total=item.rate*q;
                gotoxy(11,j);
                printf("%4d",i);
                printf("%9s",item.name);
                printf("%13d",q);
                printf("%15.2f",item.rate);
                printf("%13.2f",total);
                gtotal=gtotal+total;
                size=sizeof(item);
                item.quantity=item.quantity-q;
                j+=2;
                i++;
                fseek(file,-size,SEEK_CUR);
```

```
            fwrite(&item,sizeof(item),1,file);
            break;
          }
        }
    }
    if(gtotal!=0)
    {
      gotoxy(30,j+5);
      printf("TOTAL AMOUNT = NRs. %6.2f",gtotal);
    }
    fclose(file);
    getch();
    d_mainmenu();
}
/*function to display  bill window*/
void  dbill()
{
    int  i;
    gotoxy(20,10);
//;
    for (i=1; i<=10; i++)
      printf("*");
    printf(" * FASHION WEAR * ");
    for  (i=1; i<=10; i++)
      printf("*");
    printf("\n\n");
    gotoxy(30,11);
    printf("Departmental  Store");
//textcolor(1);
    gotoxy(32,25);
    printf("CUSTOMER'S BILL") ;
//textcolor(8);
    gotoxy(13,27);
    printf("SN.  Item Name     Quantity      Rate        Total");

}
/*function to add records*/
void add ()
{
    FILE *file;
    char y[ACS],x[12];
    system("cls");
//textbackground(11);
//textcolor(0);
    gotoxy(25,25);
    printf("Enter new record(Y/N)?");
    while(toupper(getche())=='Y')
```

```
    {
      system("cls");
      file=fopen("record.txt","ab");
      c_code(y);
      strcpy(item.code,y);
      gotoxy(22,28);
      printf("Enter rate of the item:");
      scanf("%f",&item.rate);
      gotoxy(22,30);
      printf("Enter quantity of the item:");
      scanf("%d",&item.quantity);
      gotoxy(22,32);
      printf("Enter name of the item:");
      scanf("%s",item.name);
      fseek(file,0,SEEK_END);
      fwrite(&item,sizeof(item),1,file);
      fclose(file);
      gotoxy(22,34);
      printf("Enter new record(Y/N)?");

    }
    d_mainmenu();
}

/*function to check  availability  of code*/
void c_code(char y[])
{
    int flag;
    FILE *file;
    file=fopen("record.txt","rb");
    while(1)
    {
      system("cls");
      window(20,58,23,36);
      gotoxy(32,18);
      printf(" ADD ARTICLES ")  ;
      flag=1;
      rewind    (file);
      gotoxy(22,25);
      printf("Enter new code of the article:");
      scanf(" %[^\n]",y);
      while(fread(&item,sizeof(item),1,file)==1)
      {
        if (strcmp(y,item.code)==0)
        {
          flag=0;
          gotoxy(26,30);
```

```
            printf("code already exists");
            gotoxy(29,32);
            printf("enter  again");
            getch  ();
            break;
          }
      }
      if (flag==1)
         break;
   }
}

/*function for  editing*/
void   edit()
{
   int flag=0,choice;
   char   x[ACS],y[ACS];
   FILE * file;
   int size;
   system("cls");
//textcolor(0);
//textbackground(11);
   window(20,63,20,46);
   gotoxy(35,18);
   printf("EDIT RECORDS");
   ;
   gotoxy(25,23);
   printf("enter item code: ");
   scanf("%s",x);
   flag=check(x);
   if(flag==0)
   {
      file=fopen("record.txt","r+b")  ;
      rewind(file) ;
      while (fread(&item  ,sizeof (item),1,file))
      {
         if(strcmp(item. code  ,x)==0)
         {
            //textcolor(0);
            gotoxy(25,27);
            printf("name      = %s",item.name);
            gotoxy(25,28);
            printf("code      = %s",item.code);
            gotoxy(25,29);
            printf("rate      = %g",item.rate);
            gotoxy(25,30);
            printf("quantity  = %d",item.quantity);
```

```
gotoxy(25,32);;
printf("Do you want to edit this record?(y/n):");
fflush(file);
if(toupper(getche())=='Y')
{
   //textcolor(0);
   gotoxy(25,34);
   printf("1- edit name ");
   gotoxy(25,35);
   printf("2- edit code ");
   gotoxy(25,36);
   printf("3- edit rate ");
   gotoxy(25,37);
   printf("4- edit quantity ");
   gotoxy(25,39);  ;
   printf(" enter your   choice(1, 2, 3, 4) ");
   scanf("%d",&choice);
   switch(choice)
   {
   case 1:
      system("cls");
      window(23,48,20,40);
      gotoxy(35,18);
      printf("EDIT RECORDS");
      gotoxy(25,24);
      printf(" enter new name: ");
      scanf("%s",item.name);
      size=sizeof(item);
      fseek(file,-size,SEEK_CUR);
      fwrite(&item,sizeof(item),1,file);
      break;
   case 2:
      system("cls");
      window(23,65,20,40);
      gotoxy(35,18);
      printf("EDIT RECORDS");
      gotoxy(25,24);
      c_code(y);
      strcpy(item.code,y);
      size=sizeof(item);
      fseek(file,-size,SEEK_CUR);
      fwrite(&item,sizeof(item),1,file);
      break;
   case 3:
      system("cls");
      window(23,65,20,40);
      gotoxy(35,18);
```

```
                printf("EDIT RECORDS");
                gotoxy(25,24);
                printf(" enter new rate: ");
                scanf("%f",&item.rate);
                size=sizeof(item);
                fseek(file,-size,SEEK_CUR);
                fwrite(&item,sizeof(item),1,file);
                break;
             case 4:
                system("cls") ;
                window(23,65,20,40) ;
                gotoxy(35,18) ;
                printf("EDIT RECORDS" );
                gotoxy(25,24) ;
                printf(" enter new quantity: ") ;
                scanf("%d",&item.quantity) ;
                size=sizeof(item) ;
                fseek(file,-size,1) ;
                fwrite(&item,sizeof(item),1,file) ;
                break ;
             }
             gotoxy(27,30) ;
             printf("--- item edited---") ;
             break  ;
          }
        }
      }
   }
   if (flag==1)
   {
      gotoxy(32,30) ;
      printf("Item does not exist.") ;
      gotoxy(36,32) ;
      printf("TRY ABGAIN") ;
   }
   getch();
   fclose(file) ;
   d_mainmenu() ;
}

/*function to   display all records*/
void d_all()
{
   int i,j=1 ;
   FILE *file ;
   dis_con() ;
   file=fopen("record.txt","rb") ;
```

```
   rewind(file) ;
   i=26 ;
   fflush(file)  ;
   while(fread(&item,sizeof(item),1,file))
   {
      display(&item,i,j) ;
      i++ ;
      j++ ;
      if ((j%20)==0)
      {
         gotoxy(27,47);/*textcolor(0)*/ ;
         printf("Press any key to see more.....  ");
         getch() ;
         system("cls") ;
         dis_con()  ;
         i=26 ;
         continue;
      }
   }
   getch() ;
   if (i==26)
   {
      gotoxy(24,30) ;
      printf("-- no articles found --") ;
   }
   getch() ;
   fclose(file) ;
   d_mainmenu() ;
}

/*function to display by    quantity*/
void d_quan()
{
   int i,j=1 ;
   int a,b ;
   FILE *file ;
   dis_con() ;
   file=fopen("record.txt","rb") ;
   rewind(file) ;
   i=26  ;
   gotoxy(16,20) ;
   printf("Enter lower range  ");
   scanf("%d",&a) ;
   gotoxy(16,21) ;
   printf("Enter upper range:") ;
   scanf("%d",&b) ;
   fflush(file) ;
```

```
   while(fread(&item,sizeof(item),1,file))
   {
      if((item.quantity>=a)&&(item.quantity<=b))
      {
         display(&item,i,j) ;
         i++ ;
         j++ ;
         if ((j%20)==0)
         {
            gotoxy(27,47) ;
            printf("Press any key to see more.....") ;
            getch() ;
            system("cls") ;
            dis_con() ;
            i=26 ;
            continue ;
         }
      }
   }
   getch() ;
   if (i==26)
   {
      gotoxy(28,30) ;
      printf(" No items found.") ;
   }
   getch() ;
   d_search( ) ;
   fclose(file) ;
}

/*function to   display by rate*/
void d_rate()
{
   int i,j=1 ;
   float a,b ;
   FILE *file ;
   dis_con() ;
   file=fopen("record.txt","rb") ;
   rewind(file) ;
   i=26  ;
   gotoxy(16,20) ;
   printf("enter lower range: ") ;
   scanf("%f",&a) ;
   gotoxy(16,21) ;
   printf("enter upper range: ") ;
   scanf("%f",&b) ;
   fflush(file)  ;
```

```
   while(fread(&item,sizeof(item),1,file))
   {
      if((item.rate>=a)&&(item.rate<=b))
      {
         display(&item,i,j) ;
         i++ ;
         j++ ;
         if ((j%20)==0)
         {
            gotoxy(27,47) ;
            printf("press any key to see more.....") ;
            getch() ;
            system("cls") ;
            dis_con() ;
            i=26 ;
            continue ;
         }
      }
   }
   getch() ;
   if (i==26)
   {
      gotoxy(28,30) ;
      printf(" no item found ") ;
   }
   getch() ;
   fclose(file) ;
   d_search() ;
}

/*function to   display by code*/
void d_code()
{
   int i,j=1 ;
   char x[4]= {0} ;
   FILE *file ;
   dis_con() ;
   file=fopen("record.txt","rb") ;
   rewind(file) ;
   i=26 ;
   gotoxy(16,20) ;
   printf("enter item code: ") ;
   scanf("%s",x) ;
   fflush(file) ;
   while(fread(&item,sizeof(item),1,file))
   {
      if((strcmp(item.code,x)==0))
```

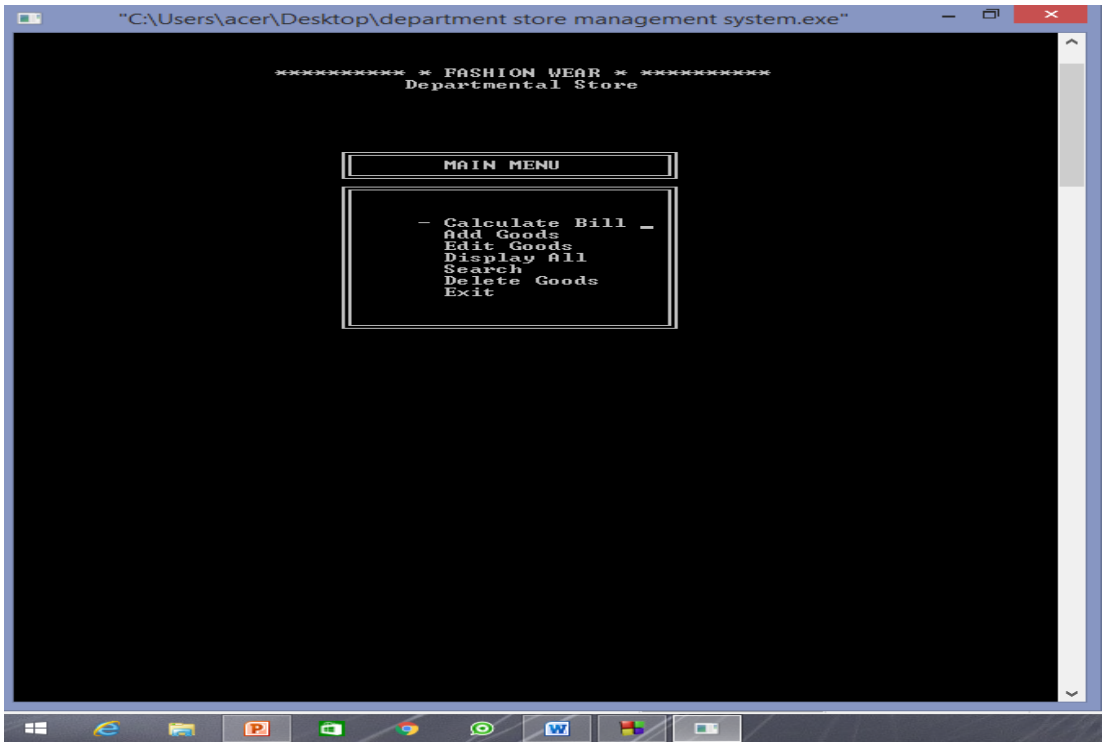
```
      {
         display(&item,i,j) ;
         i++ ;
         j++ ;
         break ;
      }
   }
   if (i==26)
   {
      gotoxy(28,30) ;
      printf("no item found") ;
   }
   getch() ;
   fclose(file) ;
   d_search() ;
}

/*function to    display window for item display*/
void dis_con( )
{
   int i;
   system("cls") ;
   gotoxy(20,10) ;
   ;
   for (i=1; i<=10; i++)
      printf("*") ;
   printf(" * FASHION WEAR * ") ;
   for (i=1; i<=10; i++)
      printf("*") ;
   printf("\n\n") ;
   gotoxy(30,11) ;
   printf("Departmental Store") ;
//textcolor(1) ;
   gotoxy(32,17) ;
   printf("RECORDS") ;
//textcolor(8)  ;
   gotoxy(18,23)  ;
   printf ("SN   Item Name   Item Code     Rate    Quantity")  ;
}

/*function to display in screen*/
void display(rec *item,int i,int j )
{
   gotoxy(16,i);//textcolor(13) ;
   printf("%4d",j) ;
   printf("%9s",item->name) ;
   printf("%12s",item->code) ;
```

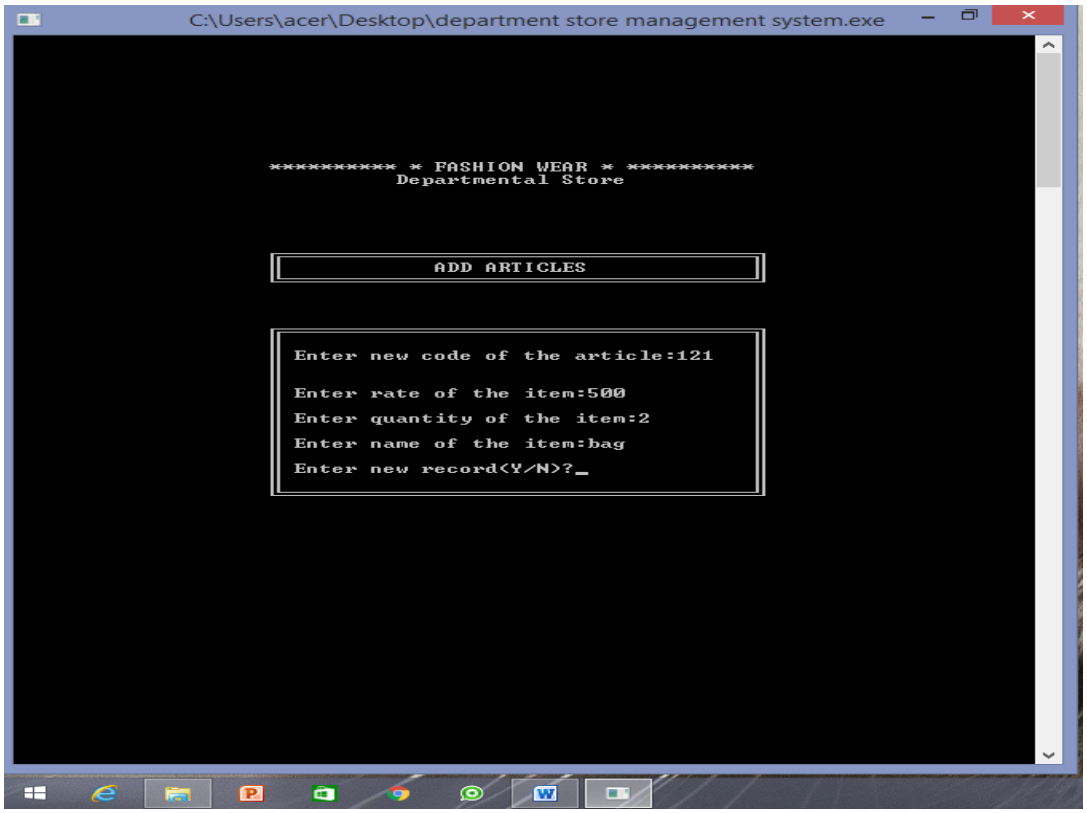
```
   printf("%14.2f",item->rate) ;
   printf("%11d",item->quantity) ;
}

/*function to delete records*/
void del()
{
   int flag ;
   char x[ANS] ;
   FILE *file,*file1 ;
   system("cls") ;
//textbackground(11) ;
//textcolor(0) ;
   window(23,51,25,34) ;
   gotoxy(29,18) ;
   printf("DELETE ARTICLES")   ;
   gotoxy(27,27) ;
   printf("enter item code: ") ;
   scanf("%s",x) ;
   flag=check(x) ;
   if(flag==0)
   {
      file1=fopen("record1.txt","ab") ;
      file=fopen("record.txt","rb") ;
      rewind(file) ;
      while (fread(&item,sizeof (item),1,file))
      {
         if(strcmp(item.code,x)!=0)
            fwrite(&item,sizeof(item),1,file1) ;
      }
      gotoxy(27,29) ;
      printf("---item deleted---") ;
      remove("record.txt") ;
      rename("record1.txt","record.txt") ;
   }
   if (flag==1)
   {
      gotoxy(25,29) ;
      printf("---item does not exist---") ;
      gotoxy(30,31) ;
      printf("TRY AGAIN") ;
   }
   fclose(file1) ;
   fclose(file) ;
   getch() ;
   d_mainmenu() ;
}
```

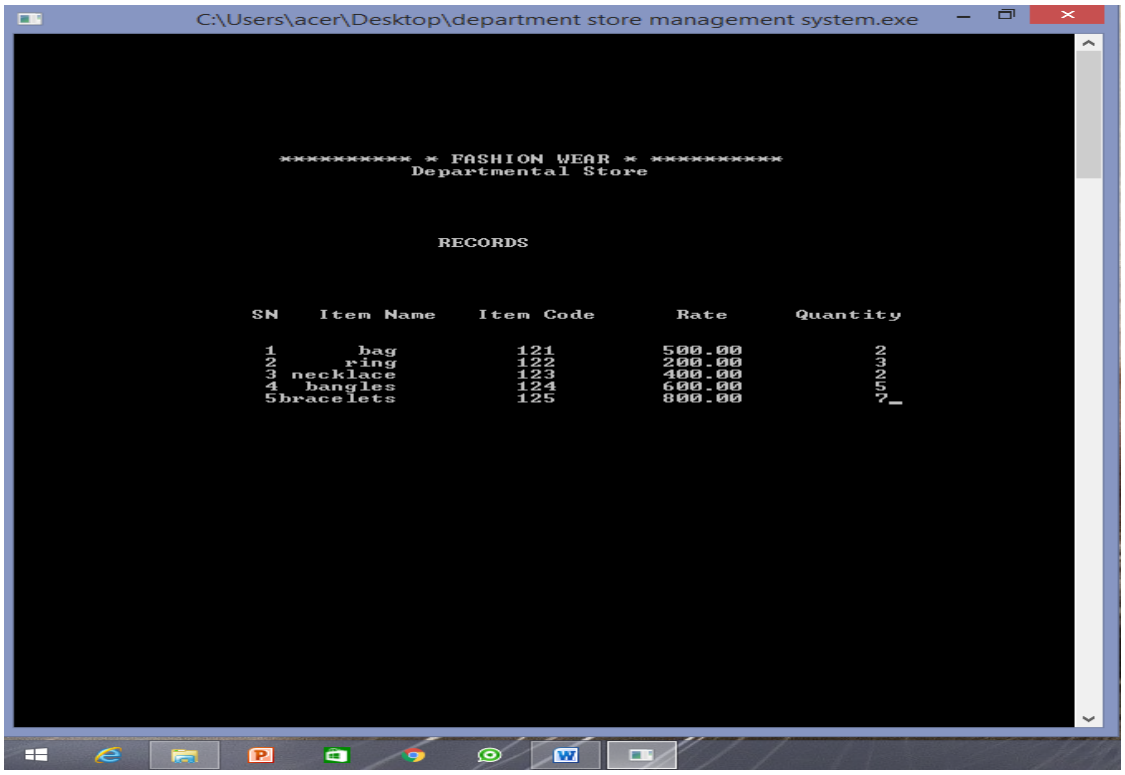
```
/*function to check validity of code while editing and deleting*/
int check(char x[ANS])
{
   FILE *file ;
   int flag=1 ;
   file=fopen("record.txt","rb") ;
   rewind(file) ;
   while (fread(&item, size of (item),1,file))
   {
      if(strcmp(item.code,x)==0)
      {
         flag=0 ;
         break ;
      }
   }
   fclose(file) ;
   return flag ;
}

/*function to display box*/
void window(int a,int b,int c,int d)
{
   int i ;
   system("cls") ;
   gotoxy(20,10) ;
//textcolor(1) ;
   for (i=1; i<=10; i++)
      printf("*") ;
   printf(" * FASHION WEAR * ") ;
   for (i=1; i<=10; i++)
      printf("*") ;
   printf("\n\n") ;
   gotoxy(30,11) ;
   printf("Departmental Store") ;
//textcolor(4) ;
   for (i=a; i<=b; i++)
   {
      gotoxy(i,17) ;
      printf("\xcd") ;
      gotoxy(i,19) ;
      printf("\xcd") ;
      gotoxy(i,c) ;
      printf("\xcd") ;
      gotoxy(i,d) ;
      printf("\xcd") ;
   }
```

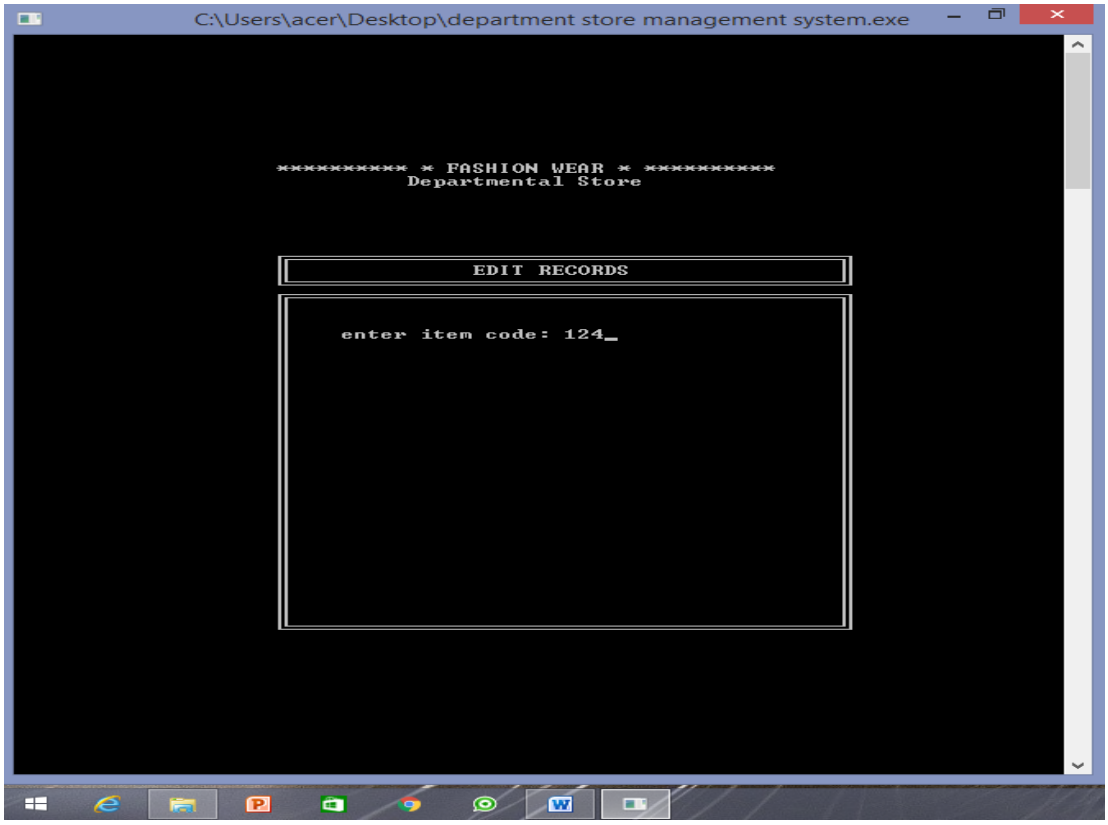
```
   gotoxy(a,17) ;
   printf("\xc9") ;
   gotoxy(a,18) ;
   printf("\xba") ;
   gotoxy(a,19) ;
   printf("\xc8") ;
   gotoxy(b,17) ;
   printf("\xbb") ;
   gotoxy(b,18) ;
   printf("\xba") ;
   gotoxy(b,19) ;
   printf("\xbc") ;
//textcolor(4) ;
   for(i=c; i<=d; i++)
   {
      gotoxy(a,i) ;
      printf("\xba") ;
      gotoxy(b,i) ;
      printf("\xba") ;
   }
   gotoxy(a,c) ;
   printf("\xc9") ;
   gotoxy(a,d) ;
   printf("\xc8") ;
   gotoxy(b,c) ;
   printf("\xbb") ;
   gotoxy(b,d) ;
   printf("\xbc") ;
//textbackground(11) ;
//textcolor(0) ;
}
```

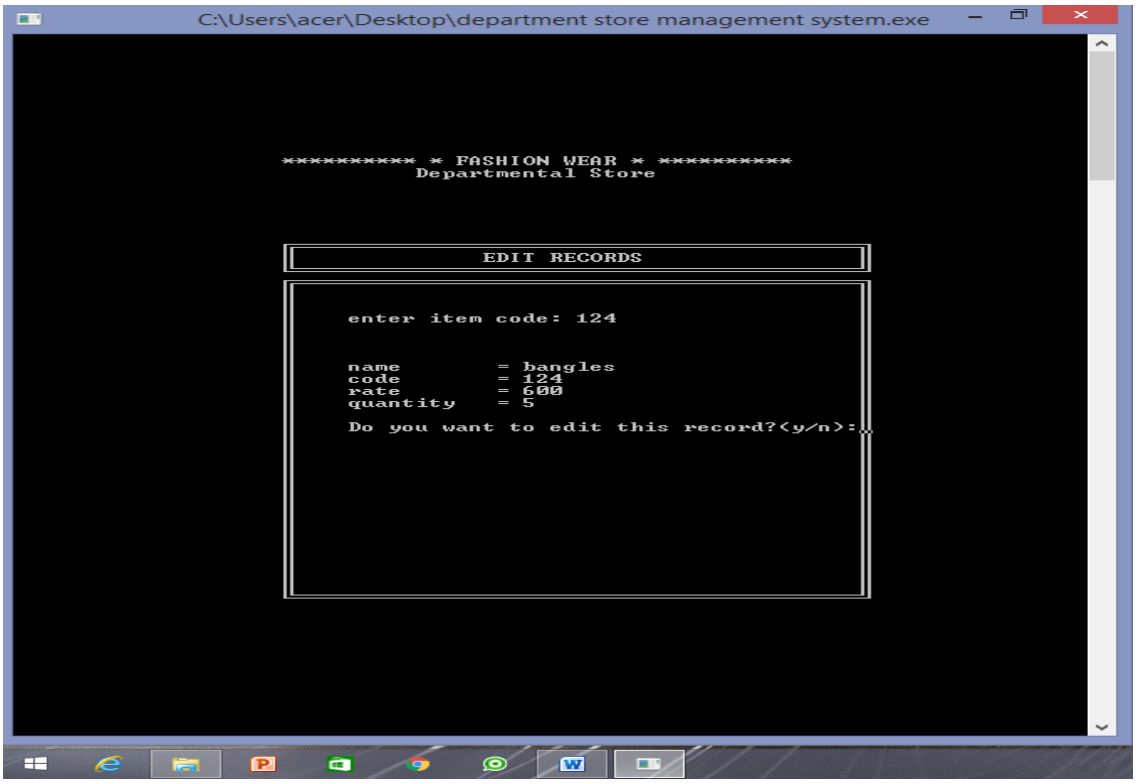
OUTPUT

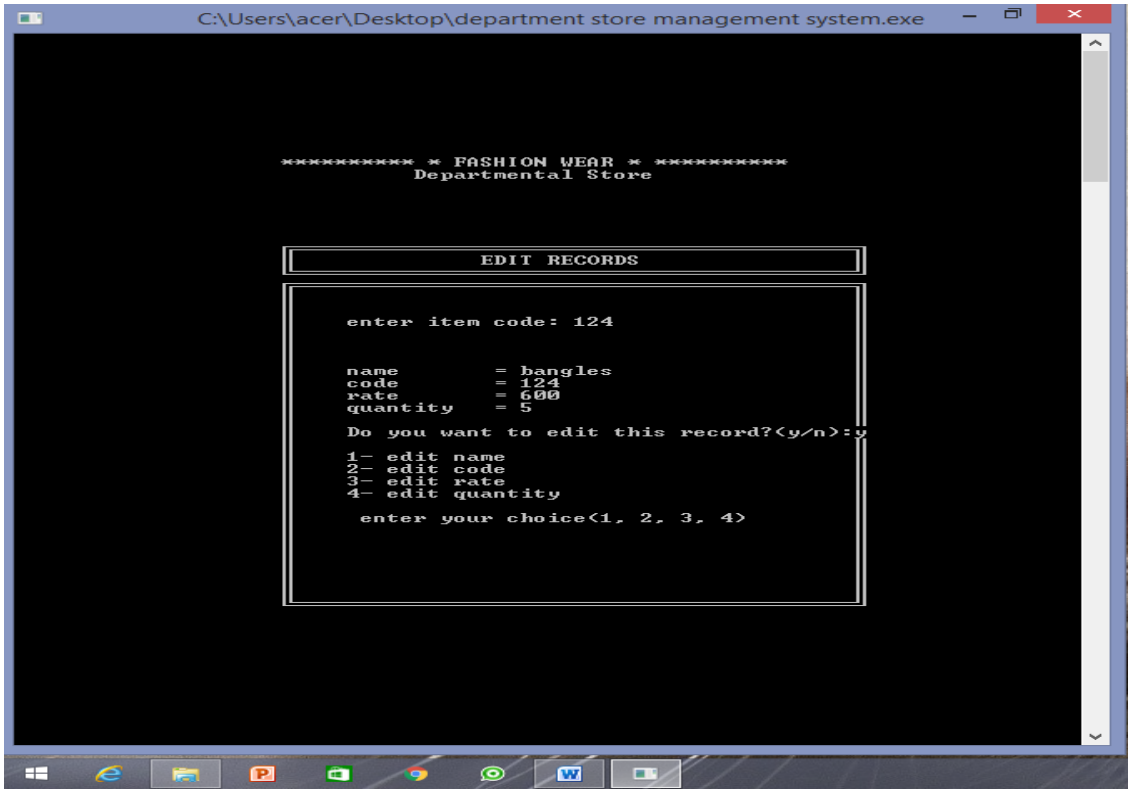

"C:\Users\acer\Desktop\department store management system.exe"
********** * FASHION WEAR * **********
Departmental Store
MAIN MENU
- Calculate Bill _
Add Goods
Edit Goods
Display All
Search
Delete Goods
Exit

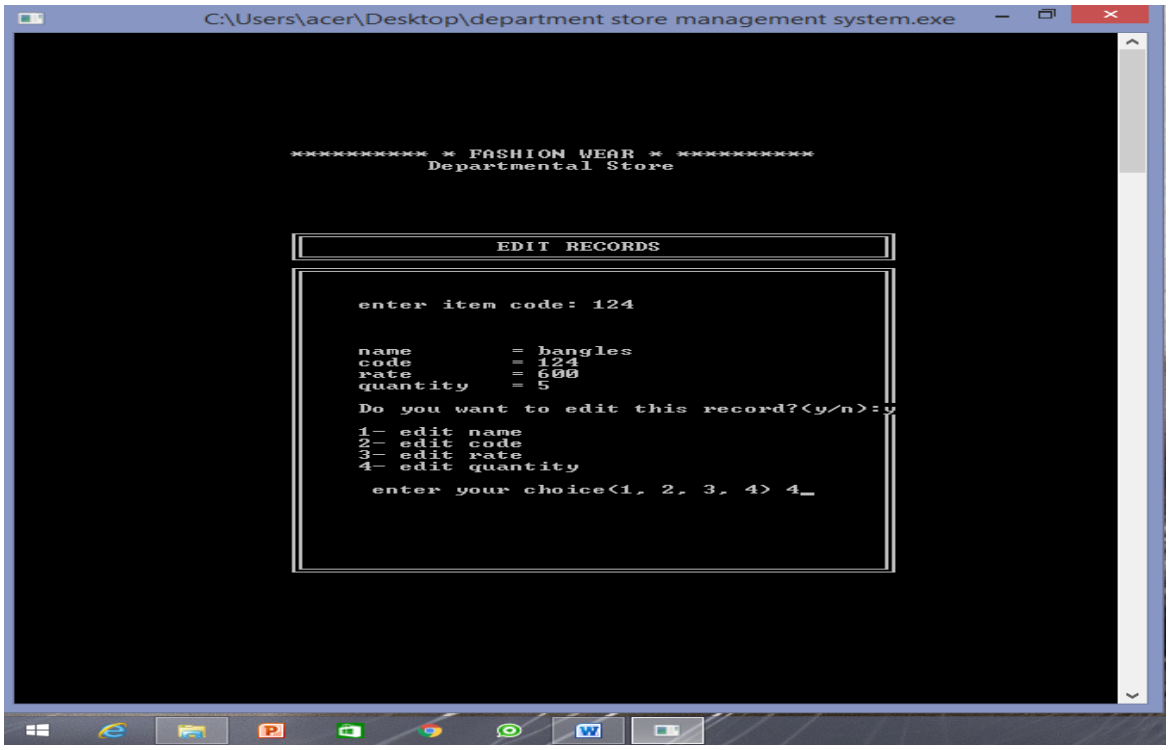

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
ADD ARTICLES
Enter new code of the article:121
Enter rate of the item:500
Enter quantity of the item:2
Enter name of the item:bag
Enter new record(Y/N)?_

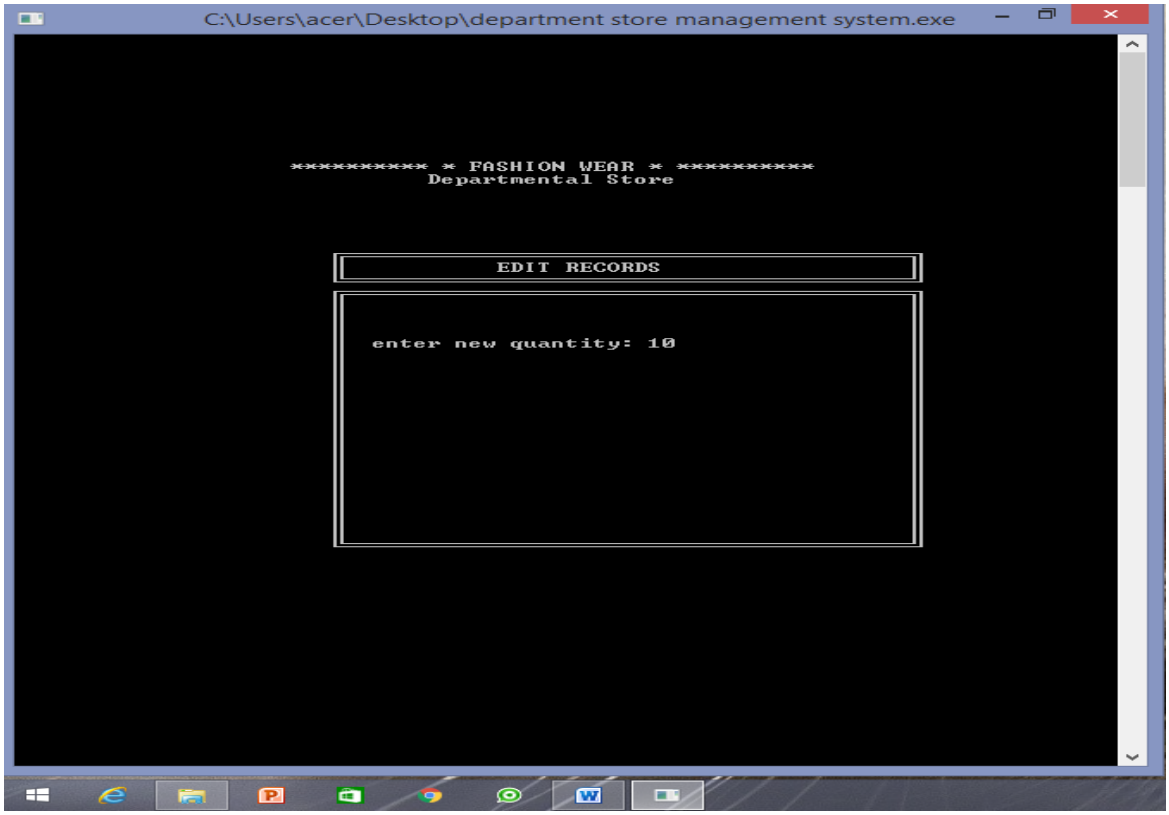

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
RECORDS
SN Item Name Item Code Rate Quantity
1 bag 121 500.00 2
2 ring 122 200.00 3
3 necklace 123 400.00 3
4 bangles 124 600.00 5
5bracelets 125 800.00 2

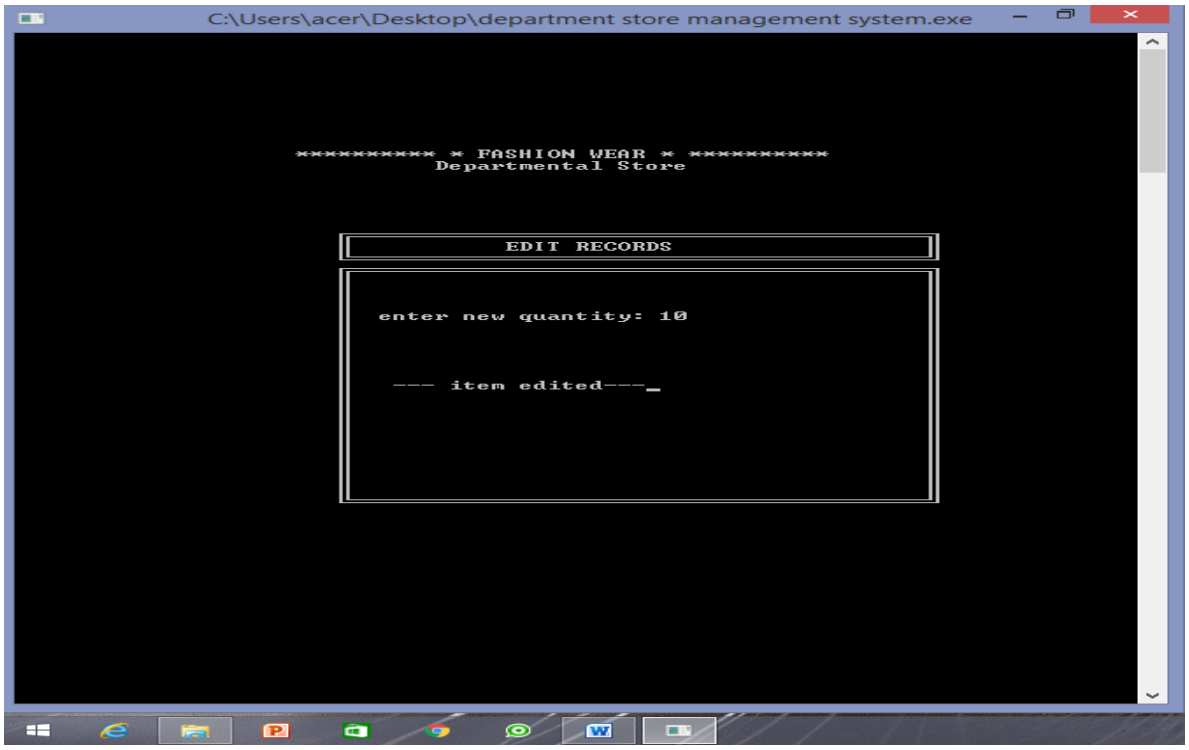

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
EDIT RECORDS
enter item code: 124

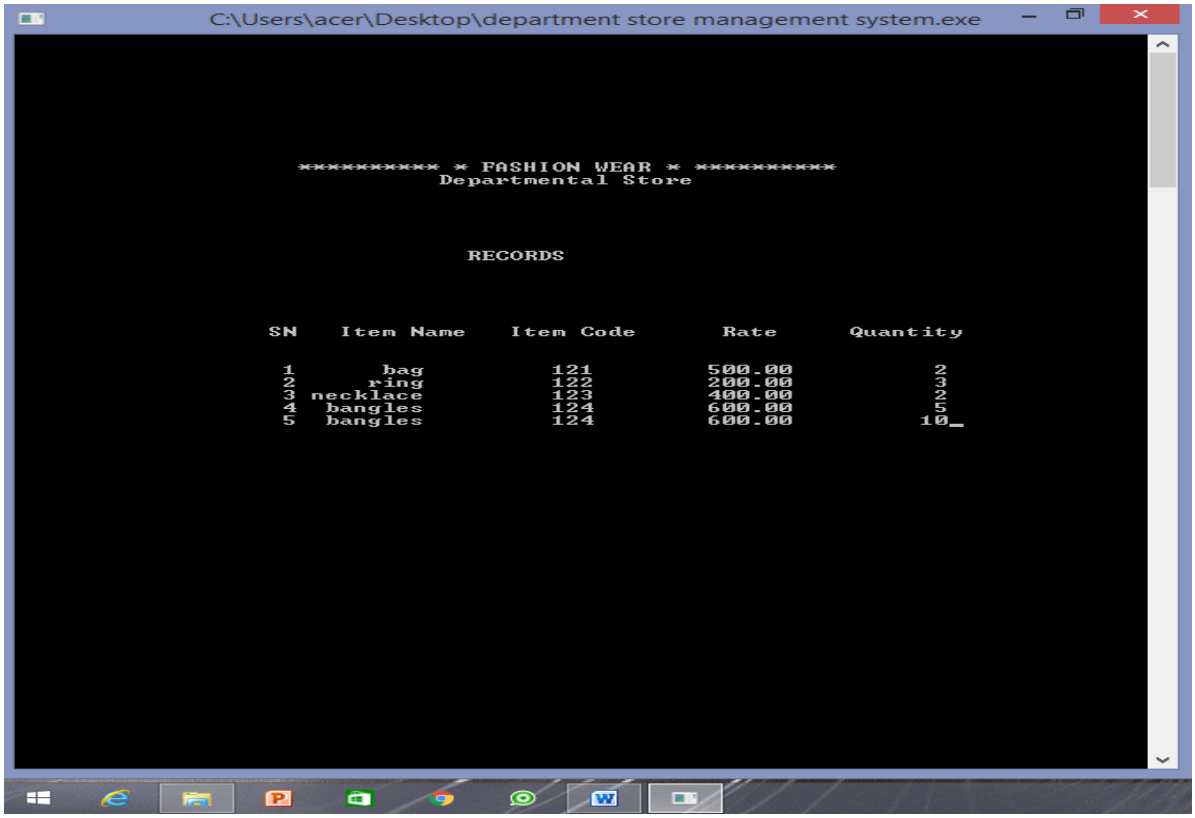

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
EDIT RECORDS
enter item code: 124
name = bangles
code = 124
rate = 600
quantity = 5
Do you want to edit this record?(y/n):

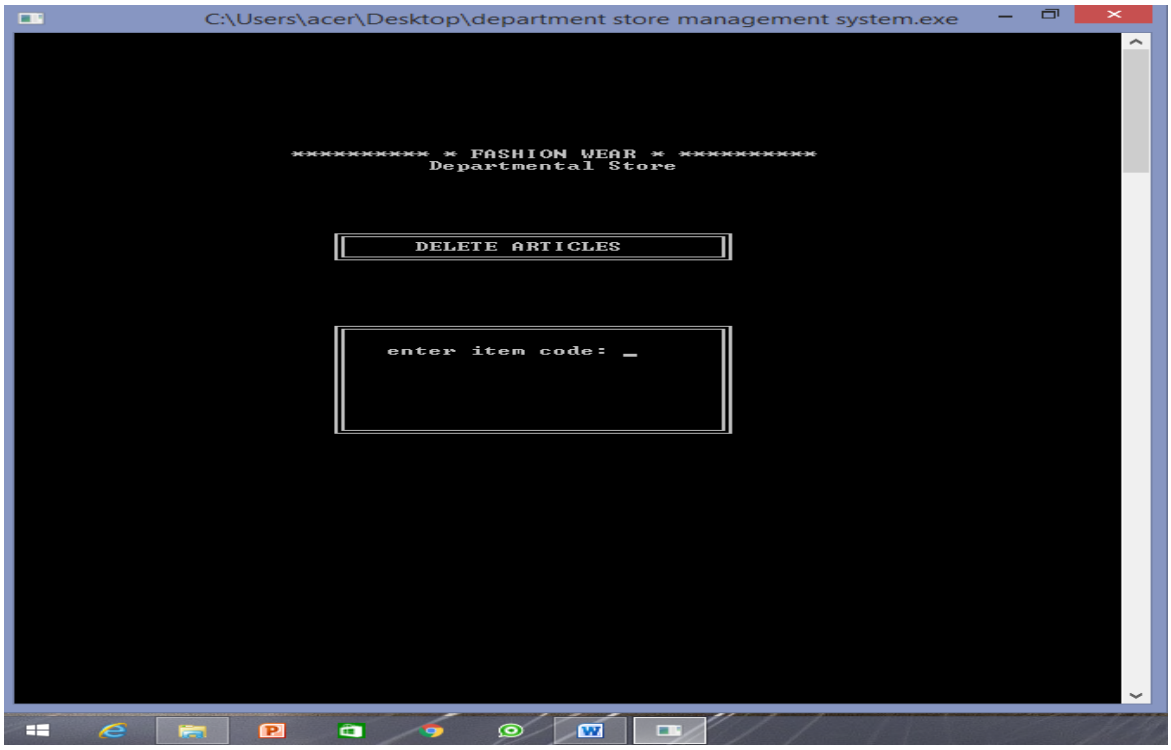

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
EDIT RECORDS
enter item code: 124
name = bangles
code = 124
rate = 600
quantity = 5
Do you want to edit this record?(y/n):y
1- edit name
2- edit code
3- edit rate
4- edit quantity
enter your choice(1, 2, 3, 4)

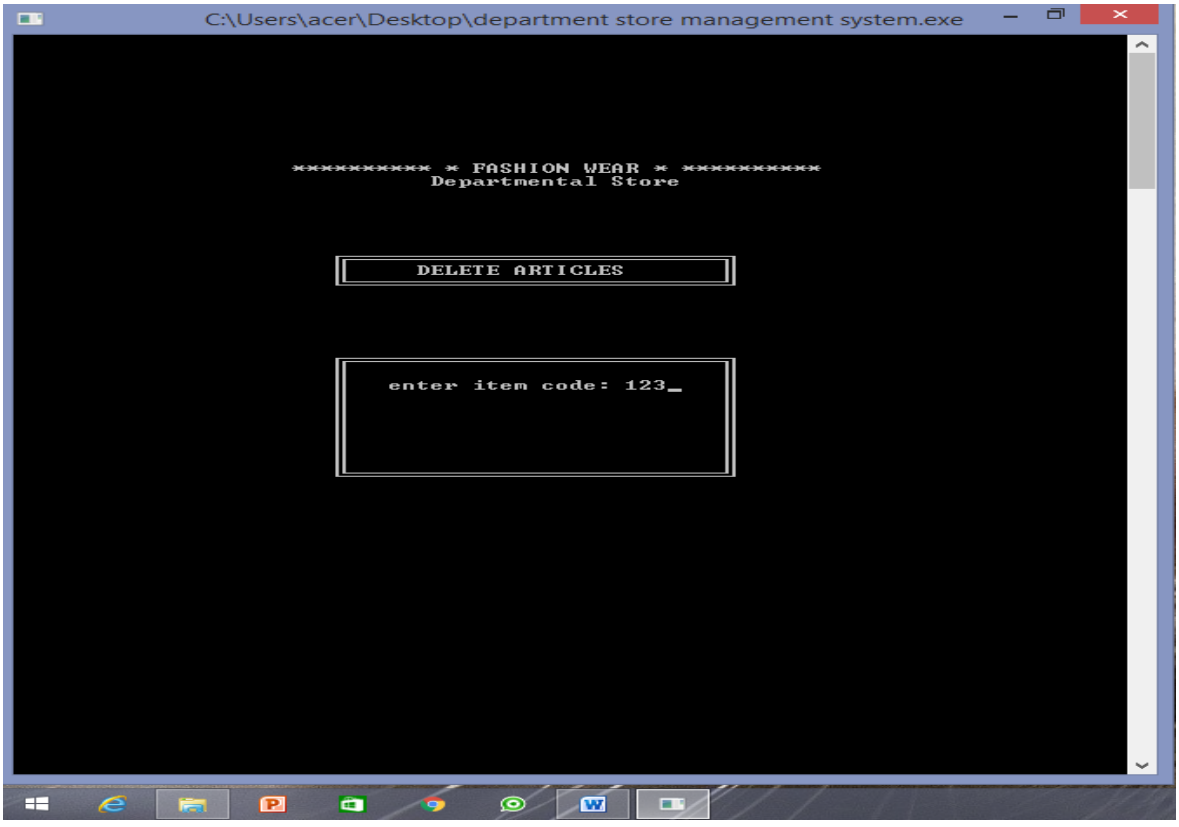

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
EDIT RECORDS
enter item code: 124
name = bangles
code = 124
rate = 600
quantity = 5
Do you want to edit this record?(y/n):y
1- edit name
2- edit code
3- edit rate
4- edit quantity
enter your choice(1, 2, 3, 4) 4

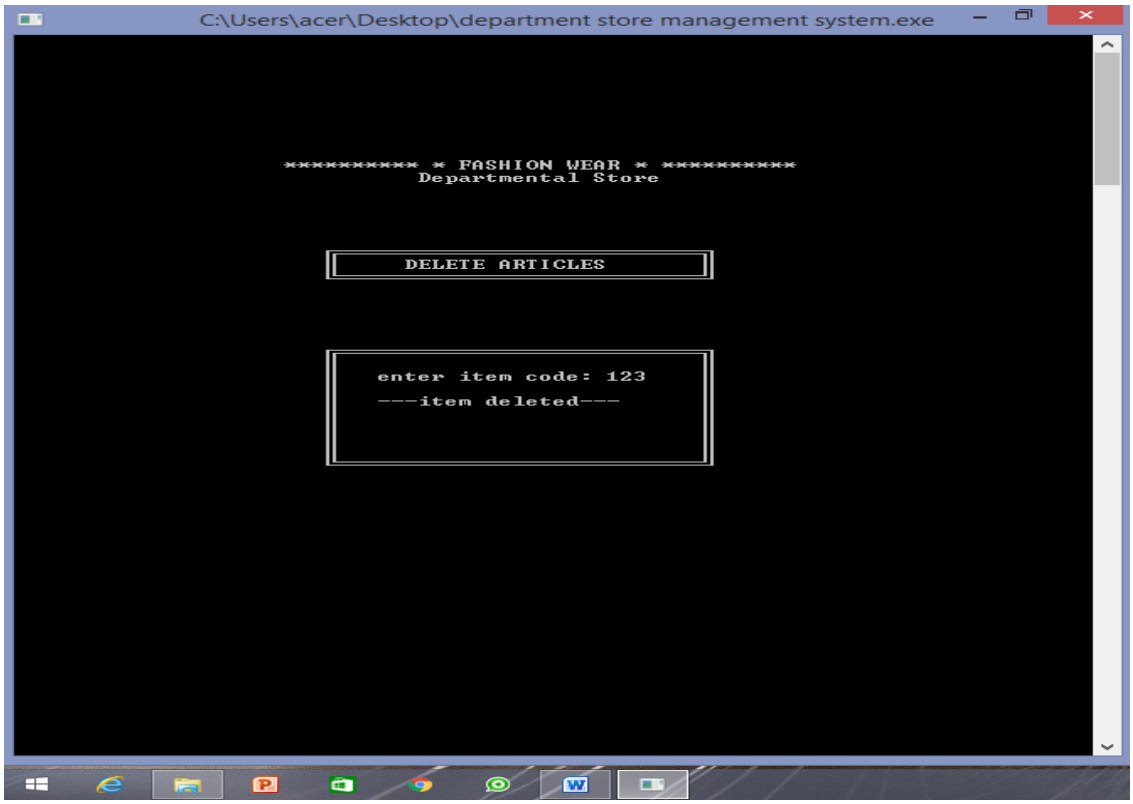

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
EDIT RECORDS
enter new quantity: 10

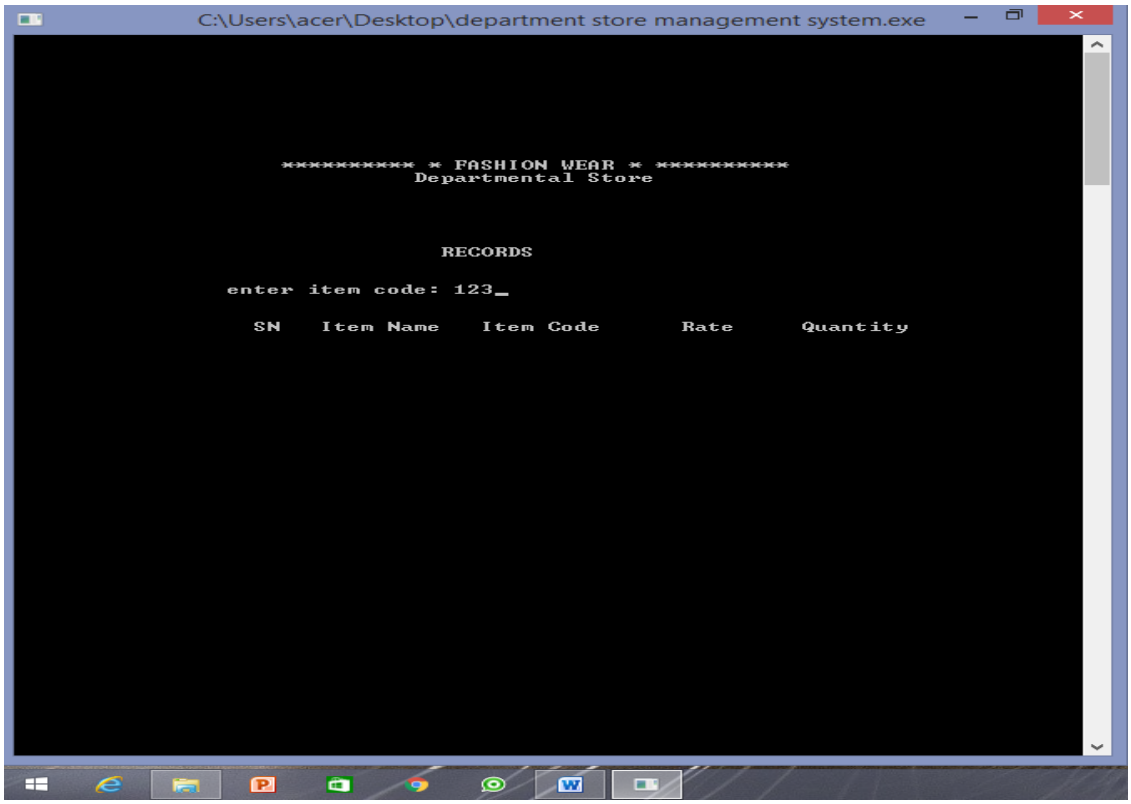

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
EDIT RECORDS
enter new quantity: 10
--- item edited---

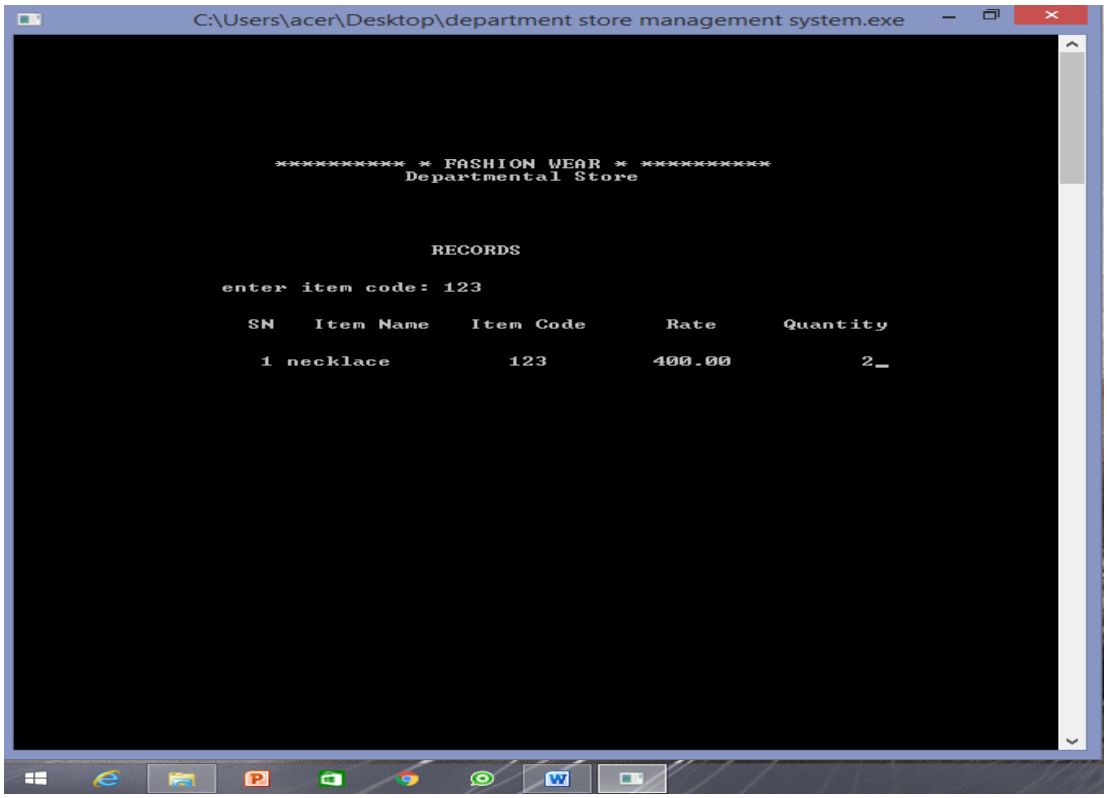

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
RECORDS
SN Item Name Item Code Rate Quantity
1 bag 121 500.00 2
2 ring 122 200.00 3
3 necklace 123 400.00 2
4 bangles 124 600.00 5
5 bangles 124 600.00 10

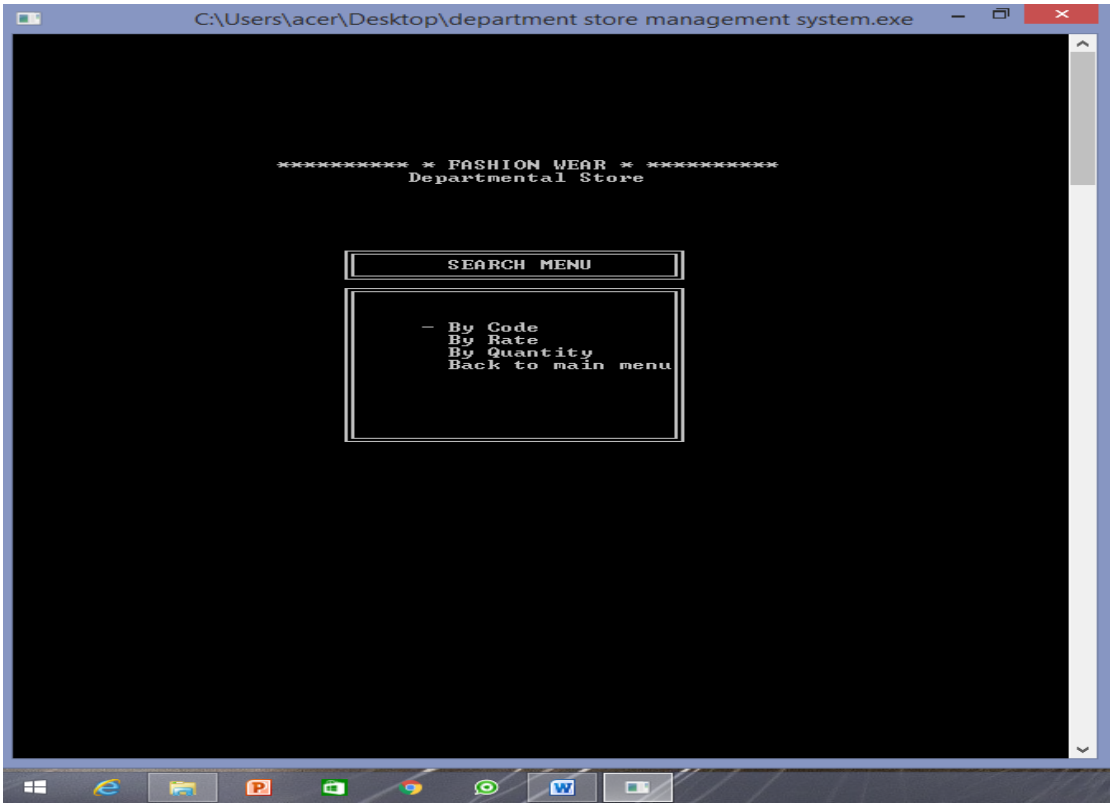

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
DELETE ARTICLES
enter item code: _

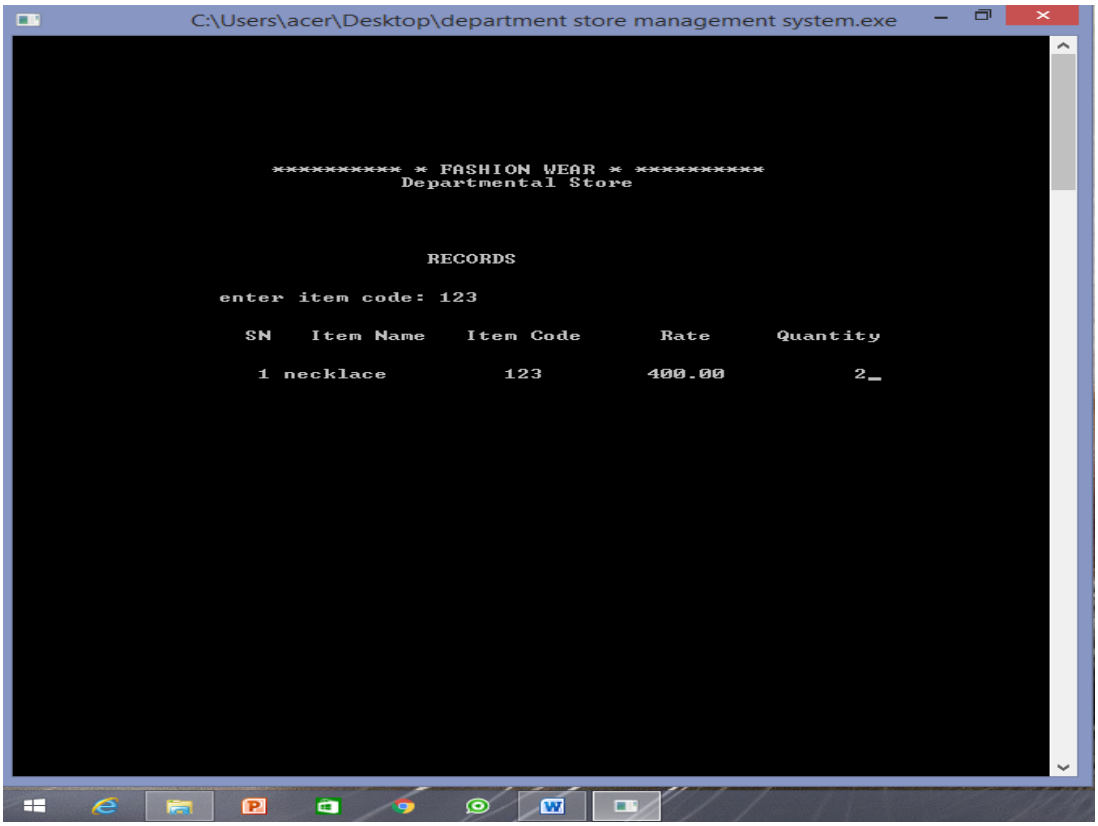

C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
DELETE ARTICLES
enter item code: 123_


C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
DELETE ARTICLES
enter item code: 123
---item deleted---


C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
RECORDS
enter item code: 123
SN Item Name Item Code Rate Quantity


C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
RECORDS
enter item code: 123
SN Item Name Item Code Rate Quantity
1 necklace 123 400.00 2


C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
SEARCH MENU
By Code
By Rate
By Quantity
Back to main menu


C:\Users\acer\Desktop\department store management system.exe
********** * FASHION WEAR * **********
Departmental Store
RECORDS
enter item code: 123
SN Item Name Item Code Rate Quantity
1 necklace 123 400.00 2_

CONCLUSION

Here we have tried to develop a system that will synchronize the data stored at various places to be maintained at a single location with proper accessibility and security. The system will store the various records of different products, distributors and various transactions in a uniform format. Effective management are going to be done to take care of minimum inventory levels of food and alternative essential commodities. The user can get essential info as inconvenience of a particular product or expiration of some merchandise beforehand. User additionally also can even may also may create any updates to antecedently entered info and also delete any obsolete information that isn't needed any longer. Thus, we tend to tried to produce the shop administrator with a straightforward thanks to access his information.

BIBLIOGRAPHY

1. https://www.w3schools.com
2. https://www.github.com
3. https://www.geeksforgeeks.org