//Marim Elhanafy 201803468

//Hissa Al-Mohannadi 201801523

//Hagar Elsayed 201805123

//project c++ Fall2019 : Phase1

//B55 : Instructor: Aws Fida El-Din

#include <iostream>

#include <string>

#include <fstream>

#include <iomanip>

using namespace std;

void drawLine(char, int);

int choices();

void addItem(string[], int[], int[], int[], int[], int[], double[], int, int);

void listItems(string[], int[], int[], int[], int[], int[], double[], int);

void itemsQuantityLessFive(string[], int[], int[], int[], int[], int[], double[], int);

void itemsLessGivenPrice(string[], int[], int[], int[], int[], int[], double[], int);

void expiredItems(string[], int[], int[], int[], int[], int[], double[], int);

void stockStatistics(string[], int[], int[], int[], int[], int[], double[], int);

void updateStock(string[], int[], int[], int[], int[], int[], double[], int);

void deleteItem(string[], int[], int[], int[], int[], int[], double[], int);

void exit();

int main()

{

char repeat;

do

{

const int SIZE = 110;

string ItemName, itemName[SIZE];

int sizeArray, Quantity, ItemNo, AisleNo, ExpMonth, ExpDay, quantity[SIZE], itemNo[SIZE] = { 0 }, aisleNo[SIZE], expMonth[SIZE], expDay[SIZE];

double Price, price[SIZE];

fstream file;

file.open("inventory.txt", ios::in | ios::out);

if (file.fail()) //Testing for File Open Errors

{

cout << "File open failure!\n";

return 0;

}

cout << "Enter Array Size (Number of Items in your file) : ";

cin >> sizeArray;

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

{

file >> ExpDay >> ExpMonth >> AisleNo >> ItemNo >> Price >> Quantity;

getline(file, ItemName);

itemName[i] = ItemName;

quantity[i] = Quantity;

itemNo[i] = ItemNo;

aisleNo[i] = AisleNo;

expMonth[i] = ExpMonth;

expDay[i] = ExpDay;

price[i] = Price;

}

for (int i = sizeArray; i < SIZE; i++)

{

itemName[i] = '0';

quantity[i] = 0;

itemNo[i] = 0;

aisleNo[i] = 0;

expMonth[i] = 0;

expDay[i] = 0;

price[i] = 0;

}

switch (choices())

{

case 1: //add new item : asks the user to input data of a new item and then add the data to the items file

{

addItem(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, sizeArray, SIZE);

break;

}

case 2: //List the current inventory in a tabular format all the information about each item

{

listItems(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

break;

}

case 3: //All items whose quantity field < 5

{

itemsQuantityLessFive(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

break;

}

case 4: //All items whose price is less than a value provided by the user

{

itemsLessGivenPrice(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

break;

}

case 5: //Show All expired items information

{

expiredItems(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

break;

}

case 6: //Stock statistics: the average price, the most expensive item, the least expensive item

{

stockStatistics(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

break;

}

case 7: //Update Stock:

{

updateStock(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

break;

}

case 8: //Delete Item:

{

deleteItem(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

break;

}

case 9: //exit the menu

{

exit();

break;

}

default: cout << "Please enter valid number\n"; //check validation

}

file.close();

cout << "Do you want to repeat the menu? (Y/N) : ";

cin >> repeat;

} while (repeat == 'Y' || repeat == 'y'); //ask user if he want yo repeat the menu

return 0;

}

void drawLine(char ch, int num) //function to draw a line

{

for (int m = 0; m < num; m++) cout << ch;

cout << endl;

}

int choices() //function display choices and return the user choice

{

int choice;

cout << "Hypermarket Managment System \n" << endl;

cout << "\t\t[1] Add Item\n";

cout << "\t\t[2] List Items\n";

cout << "\t\t[3] Item whose quantity < 5\n";

cout << "\t\t[4] Item with less price than a given price\n";

cout << "\t\t[5] Expired items\n";

cout << "\t\t[6] Stock statistics\n";

cout << "\t\t[7] Update stock\n";

cout << "\t\t[8] Delete Item\n\n";

cout << "\t\t[9] Exit\n";

cout << "\tEnter your choice: ";

cin >> choice;

cout << endl;

return choice;

}

void addItem(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int sizeArray, int SIZE)

{

string ItemName;

int position, Quantity, ItemNo=0, AisleNo, ExpMonth, ExpDay=0;

double Price;

char val;

cout << "Stock Control - Add Item: \n";

drawLine('=', 25);

cout << "Enter the item name: ";

cin >> ItemName;

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

{

if (ItemName == itemName[i])

{

char CH;

cout << "The Item is already there!! ";

cout << "Do you want to update it? (Y/N) ";

cin >> CH;

while (CH != 'Y' && CH != 'y' && CH != 'N' && CH != 'n')

{

cout << "Enter (Y/N)!";

cin >> CH;

}

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

{

updateStock(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, sizeArray);

}

else cout << "The Item not saved. Thank you!";

}

}

cout << "Enter the item Expiry Month: ";

cin >> ExpMonth;

while (ExpMonth <= 0 || ExpMonth > 12)

{

cout << "Invalid month!! Please Enter a number betweem 1 and 12 : ";

cin >> ExpMonth;

}

if (ExpMonth == 12 || ExpMonth == 10 || ExpMonth == 8 || ExpMonth == 7 || ExpMonth == 5 || ExpMonth == 3 || ExpMonth == 1)

{

cout << "Enter the item Expiry Day: ";

cin >> ExpDay;

while (ExpDay < 0 || ExpDay>31)

{

cout << "Invalid Day!! Please Enter a number between 1 and 31 : ";

cin >> ExpDay;

}

}

else if (ExpMonth == 11 || ExpMonth == 9 || ExpMonth == 6 || ExpMonth == 4)

{

cout << "Enter the item Expiry Day: ";

cin >> ExpDay;

while (ExpDay < 0 || ExpDay>30)

{

cout << "Invalid Day!! Please Enter a number between 1 and 30 : ";

cin >> ExpDay;

}

}

else if (ExpMonth == 2)

{

cout << "Enter the item Expiry Day: ";

cin >> ExpDay;

while (ExpDay < 0 || ExpDay>29)

{

cout << "Invalid Day!! Please Enter a number between 1 and 29 : ";

cin >> ExpDay;

}

}

cout << "Enter the item Aisle No: ";

cin >> AisleNo;

while (AisleNo < 0)

{

cout << "Invalid Aisle Numbet !! Please Enter a positive number : ";

cin >> AisleNo;

}

cout << "Enter the item price: ";

cin >> Price;

while (Price < 0)

{

cout << "Invalid Price!! Please Enter a positive number : ";

cin >> Price;

}

cout << "Enter the quantity to be added: ";

cin >> Quantity;

while (Quantity < 0)

{

cout << "Invalid Quantity!! Please Enter a positive number : ";

cin >> Quantity;

}

cout << "Do you want to save the item? (Y/N) ";

cin >> val;

while (val != 'Y' && val != 'y' && val != 'N' && val != 'n')

{

cout << "Enter (Y/N)!";

cin >> val;

}

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

{

cout << "At which position (Enter index number) ? ";

cin >> position;

for (int i = sizeArray; i > position; i--) // creating a space at the required position

{

itemName[i] = itemName[i - 1];

itemNo[i] = ++itemNo[i - 1];

expDay[i] = expDay[i - 1];

expMonth[i] = expMonth[i - 1];

aisleNo[i] = aisleNo[i - 1];

price[i] = price[i - 1];

quantity[i] = quantity[i - 1];

}

itemName[position] = ItemName;

itemNo[position] = position;

expDay[position] = ExpDay;

expMonth[position] = ExpMonth;

aisleNo[position] = AisleNo;

price[position] = Price;

quantity[position] = Quantity;

cout << "The Item saved successfully.\n";

fstream file;

file.open("inventory.txt", ios::in | ios::out);

if (file.fail()) //Testing for File Open Errors

{

cout << "File open failure!\n";

EXIT\_SUCCESS;

}

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

{

file << expDay[i] << "\t" << expMonth[i] << "\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << itemName[i] << endl;

}

file.close();

}

else cout << "The Item not saved. Thank you!";

}

void listItems(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int SIZE)

{

cout << "Stock Control - List Items: \n";

drawLine('=', 30);

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tQuantity\tItemName\n";

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

{

cout << expDay[i] << "\t\t" << expMonth[i] << "\t\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

}

}

void itemsQuantityLessFive(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int SIZE)

{

cout << "Stock Control - Items with Quantity Less than Five: \n";

drawLine('=', 48);

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tQuantity\tItemName\n";

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

{

if (quantity[i] < 5 && quantity[i] != 0)

{

cout << expDay[i] << "\t\t" << expMonth[i] << "\t\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

}

}

}

void itemsLessGivenPrice(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int SIZE)

{

cout << "Stock Control - Items that Less than a Given Price: \n";

drawLine('=', 48);

double userPrice;

cout << "Enter a price : ";

cin >> userPrice;

while (userPrice < 0)

{

cout << "Invalid Price!! Please Enter a positive number : ";

cin >> userPrice;

}

cout << "Items with less price than " << userPrice << " are : \n";

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tQuantity\tItemName\n";

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

{

if (price[i] < userPrice && price[i] != 0)

{

cout << expDay[i] << "\t\t" << expMonth[i] << "\t\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

}

}

}

void expiredItems(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int SIZE)

{

cout << "Stock Control - Expired Items: \n";

drawLine('=', 35);

int userDay=0, userMonth;

cout << "Enter the expire Month : ";

cin >> userMonth;

while (userMonth <= 0 || userMonth > 12)

{

cout << "Invalid month!! Please Enter a number betweem 1 and 12 : ";

cin >> userMonth;

}

if (userMonth == 12 || userMonth == 10 || userMonth == 8 || userMonth == 7 || userMonth == 5 || userMonth == 3 || userMonth == 1)

{

cout << "Enter the item Expiry Day: ";

cin >> userDay;

while (userDay < 0 || userDay>31)

{

cout << "Invalid Day!! Please Enter a number between 1 and 31 : ";

cin >> userDay;

}

}

else if (userMonth == 11 || userMonth == 9 || userMonth == 6 || userMonth == 4)

{

cout << "Enter the item Expiry Day: ";

cin >> userDay;

while (userDay < 0 || userDay>30)

{

cout << "Invalid Day!! Please Enter a number between 1 and 30 : ";

cin >> userDay;

}

}

else if (userMonth == 2)

{

cout << "Enter the item Expiry Day: ";

cin >> userDay;

while (userDay < 0 || userDay>29)

{

cout << "Invalid Day!! Please Enter a number between 1 and 29 : ";

cin >> userDay;

}

}

cout << "Expired Items are : \n";

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tQuantity\tItemName\n";

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

{

if (expMonth[i] < userMonth && expMonth[i] != 0)

{

cout << expDay[i] << "\t\t" << expMonth[i] << "\t\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

}

else if (expMonth[i] == userMonth)

{

if (expDay[i] < userDay)

{

cout << expDay[i] << "\t\t" << expMonth[i] << "\t\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

}

}

}

}

void stockStatistics(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int sizeArray)

{

cout << "Stock Control - Stock Statistics: \n";

drawLine('=', 40);

string maxItemName, minItemName;

int maxQuantity=0, maxItemNo=0, maxAisleNo=0, maxExpMonth=0, maxExpDay=0, minQuantity, minItemNo, minAisleNo, minExpMonth, minExpDay;

double maxPrice=0, minPrice, sum = 0, max = price[0], min = price[0], average;

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

{

if (price[i] != 0)

{

sum += price[i];

if (max <= price[i])

{

max = price[i];

maxExpDay = expDay[i];

maxExpMonth = expMonth[i];

maxAisleNo = aisleNo[i];

maxItemNo = itemNo[i];

maxPrice = price[i];

maxQuantity = quantity[i];

maxItemName = itemName[i];

}

else if (min == 0) min = price[i];

else if (min >= price[i])

{

min = price[i];

minExpDay = expDay[i];

minExpMonth = expMonth[i];

minAisleNo = aisleNo[i];

minItemNo = itemNo[i];

minPrice = price[i];

minQuantity = quantity[i];

minItemName = itemName[i];

}

}

}

average = sum / sizeArray;

cout << "Average price is " << average << " QR \n";

cout << "The maximum price is " << max << " QR \n";

cout << "The minimum price is " << min << " QR \n";

cout << "The most expensive item is : \n";

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tQuantity\t\tItemName\n";

cout << maxExpDay << "\t\t" << maxExpMonth << "\t\t" << maxAisleNo << "\t" << maxItemNo << "\t" << maxPrice << "\t" << maxQuantity << "\t\t" << maxItemName << endl;

cout << "The least expensive item is : \n";

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tQuantity\t\tItemName\n";

cout << minExpDay << "\t\t" << minExpMonth << "\t\t" << minAisleNo << "\t" << minItemNo << "\t" << minPrice << "\t" << minQuantity << "\t\t" << minItemName << endl;

}

void updateStock(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int SIZE)

{

cout << "Stock Control - Update Stock: \n";

drawLine('=', 40);

int ItemNo, Quantity, choice;

char val;

fstream file;

file.open("inventory.txt", ios::in | ios::out);

if (file.fail()) //Testing for File Open Errors

{

cout << "File open failure!\n";

EXIT\_SUCCESS;

}

cout << "Enter the item number : ";

cin >> ItemNo;

while (ItemNo < 0 || ItemNo > SIZE)

{

cout << "Invalid Item Number!! Please enter a Valid number : ";

cin >> ItemNo;

}

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

{

if (itemNo[i] == ItemNo)

{

cout << "Enter the quantity : ";

cin >> Quantity;

while (Quantity < 0)

{

cout << "Invalid Quantity!! Please enter a positive number : ";

cin >> Quantity;

}

cout << "Type of operation, enter 1 to add or 2 to remove : ";

cin >> choice;

while (choice != 1 && choice != 2)

{

cout << "Invalid choice!! Please enter 1 to add or 2 to remove : ";

cin >> choice;

}

if (choice == 1)

{

quantity[i] += Quantity;

cout << Quantity << " Items are added to Item No " << ItemNo << ". The current total is " << quantity[i] << endl;

for (int j = 0; j < SIZE; i++)

{

file << expDay[j] << "\t" << expMonth[j] << "\t" << aisleNo[j] << "\t" << itemNo[j] << "\t" << price[j] << "\t" << quantity[j] << "\t" << itemName[j] << endl;

break;

}

}

else if (choice == 2)

{

if (Quantity < quantity[i])

{

cout << "You do not have enough quantity for this item.\n";

}

else

{

quantity[i] -= Quantity;

if (quantity[i] > 0)

{

cout << Quantity << " Items are removed from Item No " << ItemNo << ". The current total is " << quantity[i] << endl;

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tquantity\tItemName\n";

cout << expDay[i] << "\t" << expMonth[i] << "\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

for (int j = 0; j < SIZE; i++)

{

file << expDay[j] << "\t" << expMonth[j] << "\t" << aisleNo[j] << "\t" << itemNo[j] << "\t" << price[j] << "\t" << quantity[j] << "\t" << itemName[j] << endl;

break;

}

}

else if (quantity[i] == 0)

{

cout << Quantity << " Items are removed from Item No " << ItemNo << ". The current total is " << quantity[i] << endl;

cout << "ExpDateDay\tExpDateMonth\tAisleNo\tItemNo\tprice\tquantity\tItemName\n";

cout << expDay[i] << "\t" << expMonth[i] << "\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

cout << "(Quantity is zero) If you want to remove the item, go for Delete item.";

for (int j = 0; j < SIZE; i++)

{

file << expDay[j] << "\t" << expMonth[j] << "\t" << aisleNo[j] << "\t" << itemNo[j] << "\t" << price[j] << "\t" << quantity[j] << "\t" << itemName[j] << endl;

break;

/\*

do you want to delete?

cin >> val;

while (val != 'Y' && val != 'y' && val != 'N' && val != 'n')

{

cout << "Enter (Y/N)!";

cin >> val;

}

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

{

deleteItem(itemName, quantity, itemNo, aisleNo, expMonth, expDay, price, SIZE);

}

else

{

\*/

}

}

}

}

}

}

file.close();

}

void deleteItem(string itemName[], int quantity[], int itemNo[], int aisleNo[], int expMonth[], int expDay[], double price[], int SIZE)

{

int ItemNo;

char val;

cout << "Stock Control - Delete Item: \n";

drawLine('=', 40);

cout << "Enter the item number : ";

cin >> ItemNo;

while (ItemNo < 0)

{

cout << "Invalid Item Number!! Please enter a positive number : ";

cin >> ItemNo;

}

if (ItemNo > SIZE)

{

cout << "The Item does not exist!!" << endl;

}

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

{

if (ItemNo == itemNo[i])

{

itemName[i] = '-1';

quantity[i] = -1;

itemNo[i] = -1;

aisleNo[i] = -1;

expMonth[i] = -1;

expDay[i] = -1;

price[i] = -1;

}

//for (int i = ItemNo; i < SIZE; i++) // creating a space at the required position

//{

// itemName[i] = itemName[i + 1];

// itemNo[i] = --itemNo[i + 1];

// expDay[i] = expDay[i + 1];

// expMonth[i] = expMonth[i + 1];

// aisleNo[i] = aisleNo[i + 1];

// price[i] = price[i + 1];

// quantity[i] = quantity[i + 1];

//}

}

fstream file;

file.open("inventory.txt", ios::in | ios::out);

if (file.fail()) //Testing for File Open Errors

{

cout << "File open failure!\n";

EXIT\_SUCCESS;

}

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

{

if (itemNo[i] != -1)

{

file << expDay[i] << "\t" << expMonth[i] << "\t" << aisleNo[i] << "\t" << itemNo[i] << "\t" << price[i] << "\t" << quantity[i] << "\t" << itemName[i] << endl;

}

}

file.close();

}

void exit()

{

cout << "Bye, see you again later!";

}