Zachary Prince, Christopher Yu, Roark Burney, Abi Kasraie

Serendipity BookSellers

Chrysanthemum Industries

RAD(Requirements Analysis Documentation)

Purpose:

The purpose of this application is to read from an inventory data file and allow the user to be capable of efficiently sifting through it, modify it, as well as use our application in store purposes. The application will offer an intuitive and well-designed user experience. Essentially, this application is aimed to be a useful tool for amateur book sellers to get into the business.

Outline:

The application will be organized into three modules. Each of these three modules will focus on a subset of features which will offer a focused UI so actions can be well recorded and mistakes can be prevented. In addition, the application will allow for under the hood changes from module to module and overall will offer the necessary features required to run a book store.

Under-the-Hood:

The application will utilize an excel .csv file which is easily maintained and used by laymen to manually store book data. The application will then run the .csv into a vector of book classes (in this case "pink")upon start-up from which the modules will then inherit and manipulate data from. The book class offers attributes for each piece of information that is required about a book.

Cashier Module:

The Cashier module is where the user will be handling the purchasing of books. This module will offer the basic functionality of purchasing books, reviewing checked out books, and then printing out a receipt as well as reporting the balance due.

In the engine underneath, the module takes in pink as an input parameter. In order to purchase books, the module prompts the user for a string input and the module then stores the input. Depending on the user's input the module then proceeds to offer further instruction until a book is successfully purchased. A limitation found during the process was case in-sensitivity for user input when doing string comparisons to search via title. The module then proceeds to record the information parameters of purchased books that the user may view anytime and when finished, may print a receipt.

Essentially the module accepts the book class vector and pulls information for comparison and then modifies the original book class vector when finished.

Inventory Database Module:

The inventory database module will allow the user to directly modify and save to the original inventory.csv file. The user can add books, remove books, and edit currently existing book's information.

In the engine underneath, the module prompts a variety of choices to the user. If the user chooses to add a book, the module prompts the user for information for each attribute of a book (I.e. Author, publisher, etc.) and then adds them into an incremented spot in the vector. To delete a book, the user will be offered the full selection of books and upon viewing the desired deletion target, will simply only have to enter in the integer number corresponding to the book that the user wishes to delete. The module will then delete the information of the book from the vector and decrement the size by however many books are deleted.

So essentially the inventory module accepts the book object vector and refers to it to add, delete, and view it.

Report Module:

The report module will allow the user to look at the inventory list without being able to make changes and sort by various parameters. These parameters include quantity of books, cost of books, and age of books.

The module for viewing the inventory simply runs through and spits out the content of the inventory.csv vector to the user. To sort through the various parameters, Booleans are used in order to re-organize the vector for the inventory.csv to the user's wishes. The module steps through each object and compares the current and the previous.

Basically the module accepts the book object vector and sorts the vector and displays it to the user.

Pseudocode

```
FOR
DATE.H
{
  define class date
  {
    attributes: day,month,year, all as ints
    methods:
  }
}
FOR BOOK.H
{
  include date header file
  define class book
  {
    static int to keep track of how many book objects were created
    attributes: strings: title,author,publisher
                ints: ISBN, quantity on hand
                floats: wholesale and retail price
                object of date class
    constructor: increment n when called; accept parameters for title, author, etc.
  }
  initialze static int to zero
  define constructor (i.e. a setter)
  define void function save, to be used for saving data to a file
  {
    Takes three parameters: vector of books, size of vector, file path.
    Open file for output; write each attribute from each object of the vector
    to the file.
    Close the file.
  }
}
FOR SETTING.H
{
  include date and book header file
  define three boolean functions for sorting algorithms
  {
    One is for sorting by quantity, one is by cost, and the other is by age.
    All three take two parameters; two constant book objects (by reference).
    Quantity: return true if the first book's quantity > second book's quantity.
    Cost: return true if the first book's price > second book's price
    Age:
    {
      Declare boolean variable and set it equal to false
      Compare the two book object's years; if unequal, return true
      If not
      {
        Compare their months; if unequal, return true
        If not
        {
          compare their days; if unequal, return true
        }
      }
    }
  }
}
FOR MAIN.CPP
{
  include: iostream, iomanip, fstream, string, vector, algorithm, book.h, date.h, sorting.h
  Filepath for file to read from and write to: "Inventory.csv"
  Declare void function report_message()
  {
    Print list of a given book's attributes to screen
  }

  MAIN
  {
  Display welcome message
  Open file using default filepath
  Create new book object for each line of data in file
  Read through file, adding the appropriate attributes to each object
  Tuck each object in one vector of book objects
  WHILE(TRUE)
  {
    Give user 5 options:
    {
      1: launch cashier module
      2: launch inventory module
      3: launch report module
      4: exit program
      5: credits
    }
    if one, call cashier module function
    if two, call inventory module function, passing vector of books as a parameter (by ref.)
    if three, call report module function, passing vector of books as a parameter (by ref.)
    if four, display exit message
    if five, display credits

  INVENTORY MODULE FUNCTION
  {
    Display menu of options to user;
    1: add book
    {
      prompt user for information for book
      create object with the given information
      push_back object into vector of books
```

```
}
2: remove book
{
  display to user a list of books, listing only their title
  give user three choices: "delete, next, exit"
  if exit
    break
  if delete
    prompt user for integer matching up to title of book to delete
    use vector of books[user's integer].erase to erase the object from the vector
    let user know book has been deleted
    return to main menu
}
3: edit book
{
        prompt the user for a search string
        call search function:
          {
  convert all uppercase characters in search string to lowercase equivalent
  find the equivalent integer of search string (if applicable)
            for all books in the inventory:
                        {
                                    check if the converted search string exists in the converted strings in title, author, or publisher
                                    if found
                                                add index of the book object in inventory to the array of found books
                                    else
                                                check if the numeric form of search string matches any of the numeric values
                                                              (ISBN, year, month, day, or integer parts of prices)
                                                if found
                                                            add index of the book object in inventory to the array of found books
              }
            show all books that meet the search criteria
            prompt the user to choose the desired book for editing
            accept the item number of the book
              {
                        show the details of the book in an itemized format
                        prompt the user to choose any fields for editing
                        accept the new information for each field and update
                        show all details with updated information
              }
            return to Edit menu
        }
    }
{
prompt the user for a search string
call search function:
{
convert all uppercase characters in search string to lowercase equivalent
find the equivalent integer of search string (if applicable)
for all books in the inventory:
{
check if the converted search string exists in the converted strings in title, author, or publisher
if found
add index of the book object in inventory to the array of found books
else
check if the numeric form of search string matches any of the numeric values
(ISBN, year, month, day, or integer parts of prices)
if found
add index of the book object in inventory to the array of found books
}
show all the details of objects in array of found books
prompt the user to choose the desired book for editing
{
show the details of the book in an itemized format
prompt the user to choose any fields for editing
accept the new information for each field
show all details with updated information
return to Edit menu
}
   4: save
     call save function in book.h
   5: return to main menu
     break;
  }
  REPORTS MODULE FUNCTION
{
Display menu to user; 1 lists, 2 lists by quantity, 3 sorts by cost, 4 sorts by age

If (1)
 {
cout book details for each
 element in the book vector
 }
 If (2)
{

sort(book.begin,book.end,sorting algorithm for quantity) cout book details for each element in
book vector (Same for 3 and 4; the only thing that changes is the algorithm)
```

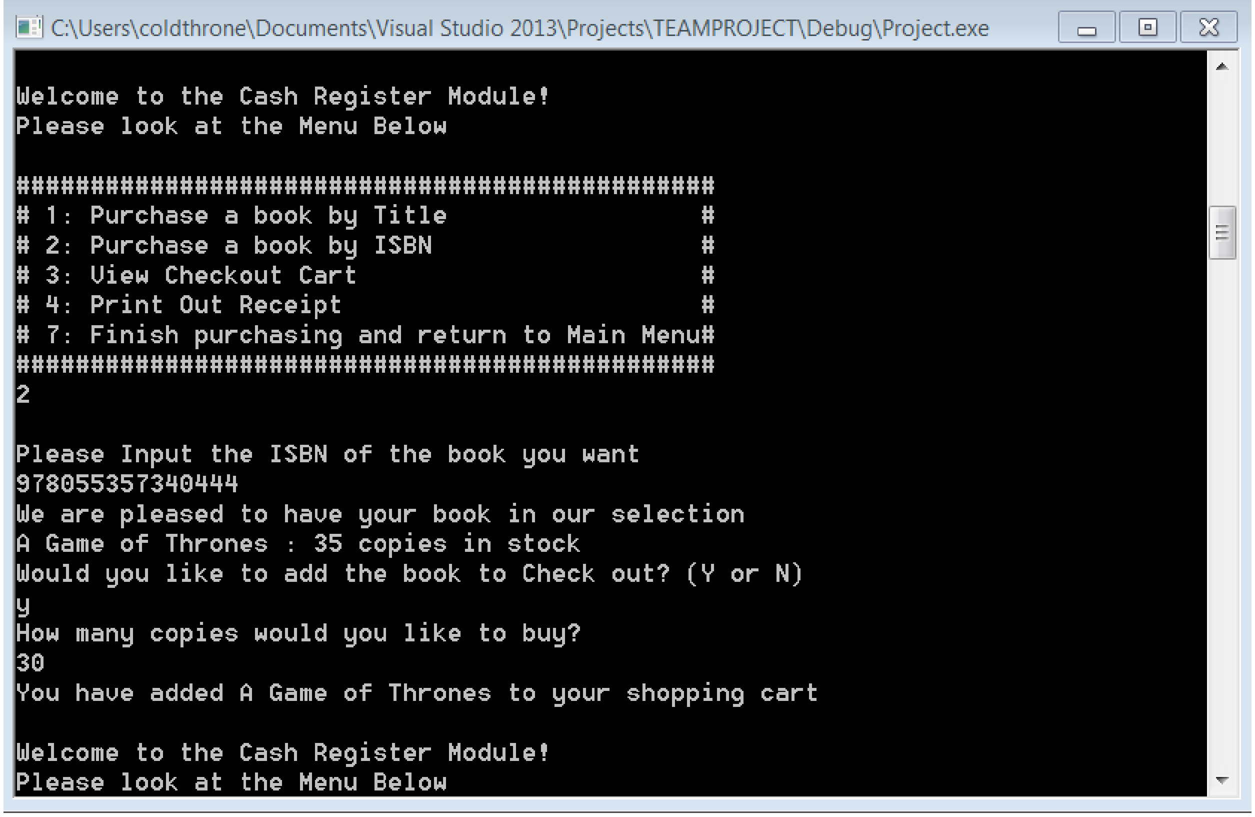
```
}

FOR CASHIER MODULE FUNCTION

{

1.
        Initialize variables:
output file receipt
int choice for navigating menus
int titlecounter for navigating purchased books
int pquantity for purchasing books

string titlechoice for user input of book title's
string cochoice for confirmation page
string isbnchoice for user input of isbn's

title,isbn, quantity, and price arrays for the receipt

2.
        Start the Looping Menu
Display Menu
1. Search by title
2. Purchase by ISBN
3. View purchased items
4. Print out Receipt
7. Go back to main menu

1/2
{
-Bool exist=false;
-prompt user for title/ISBN of desired book and store in string

-loop through the vector of books comparing strings
if found, check for quantity, if 0 end, if greater, continue

-Prompt user if they would like to add to shopping cart

-Prompt User for number of copies.
-Compare desired number of copies with actual quantity

-if less then, stop and re-prompt
- if equal to or greater than, decrease object.quantity by
user input and then continue. else, stop.
-When book is successfully purchased, store the purchased book's
details in respective information arrays for receipt output.

-Reset all navigation to 0
}
3
{
        Check out cart
Loop through stored book arrays and display their information
}
4
{
        Receipt
Same as above except create an ofstream object and report
to that
}
```

UML

Book Class
Private:
Public: -Title : String -Author : String -Publisher : String -ISBN :String -Quantity : int -ourPrice : float -theirPrice : float +read(ifstream object) : Void; +remove(int allocate, int location) : Void; +save(int, string) : Void;

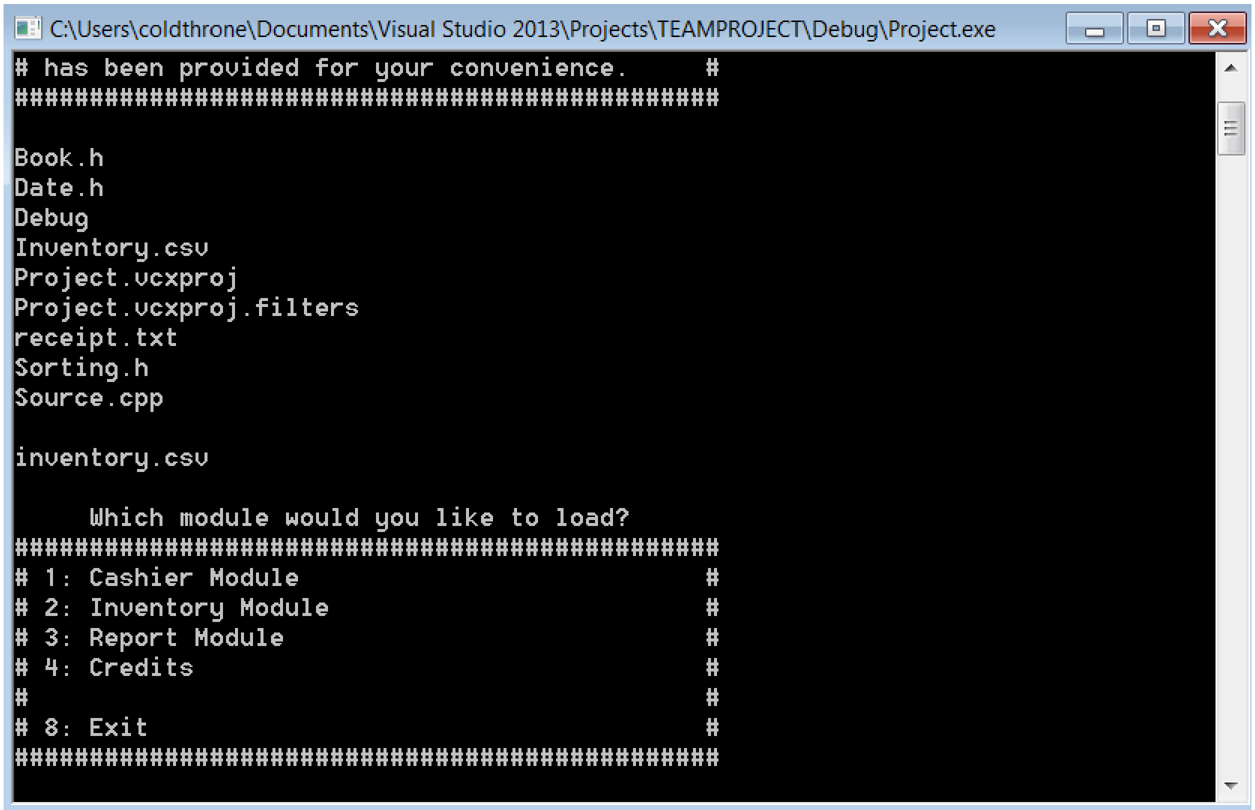
Date Class
Private:
Public: -Day : int -Month : int -Year : int

Pictures

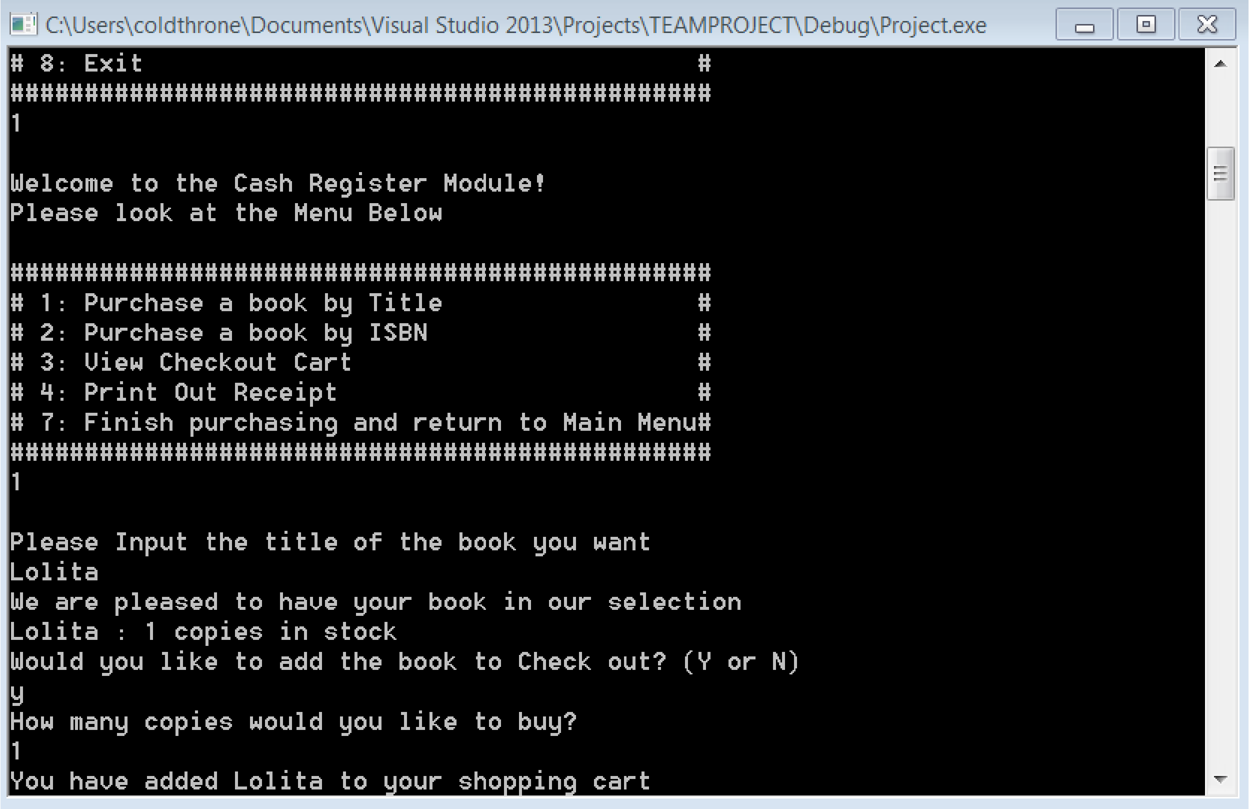


C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

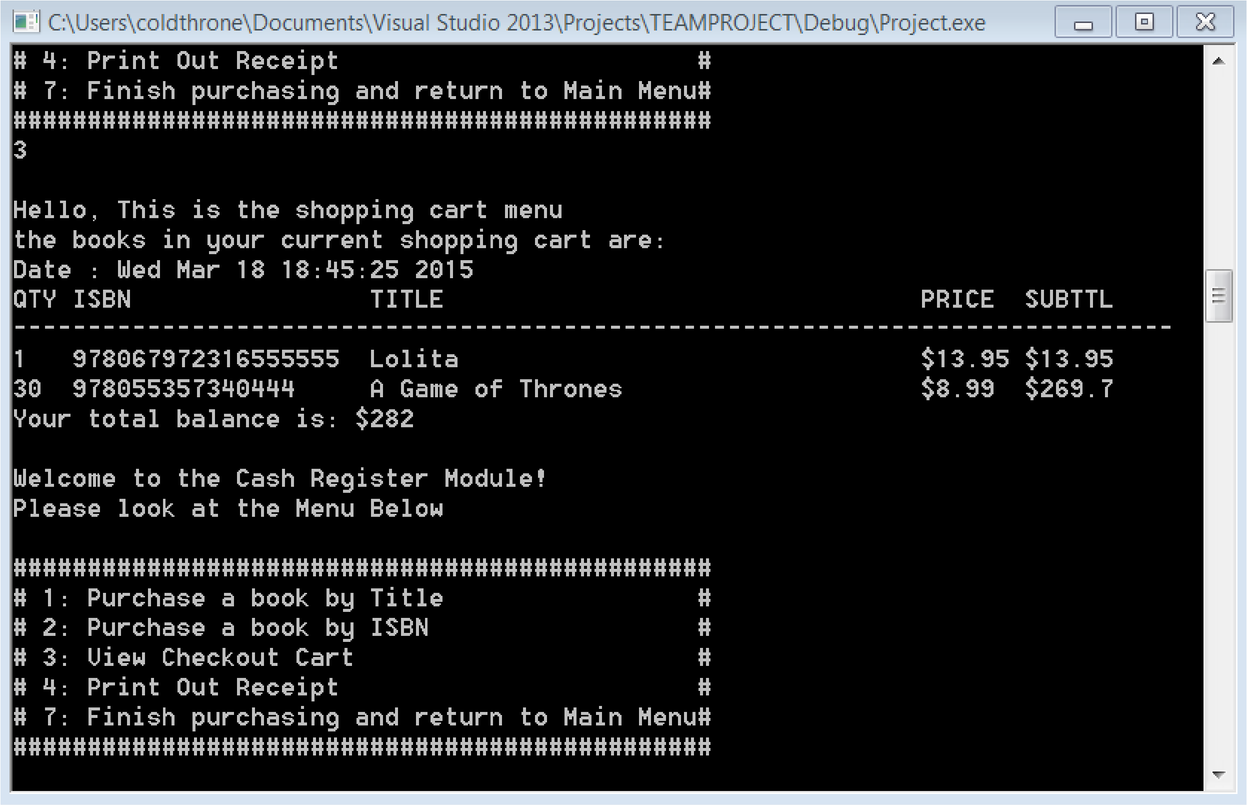
```

Welcome to the Cash Register Module!
Please look at the Menu Below

################################################
# 1: Purchase a book by Title                  #
# 2: Purchase a book by ISBN                   #
# 3: View Checkout Cart                        #
# 4: Print Out Receipt                         #
# 7: Finish purchasing and return to Main Menu#
################################################
2

Please Input the ISBN of the book you want
978055357340444
We are pleased to have your book in our selection
A Game of Thrones : 35 copies in stock
Would you like to add the book to Check out? (Y or N)
y
How many copies would you like to buy?
30
You have added A Game of Thrones to your shopping cart

Welcome to the Cash Register Module!
Please look at the Menu Below
```


C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

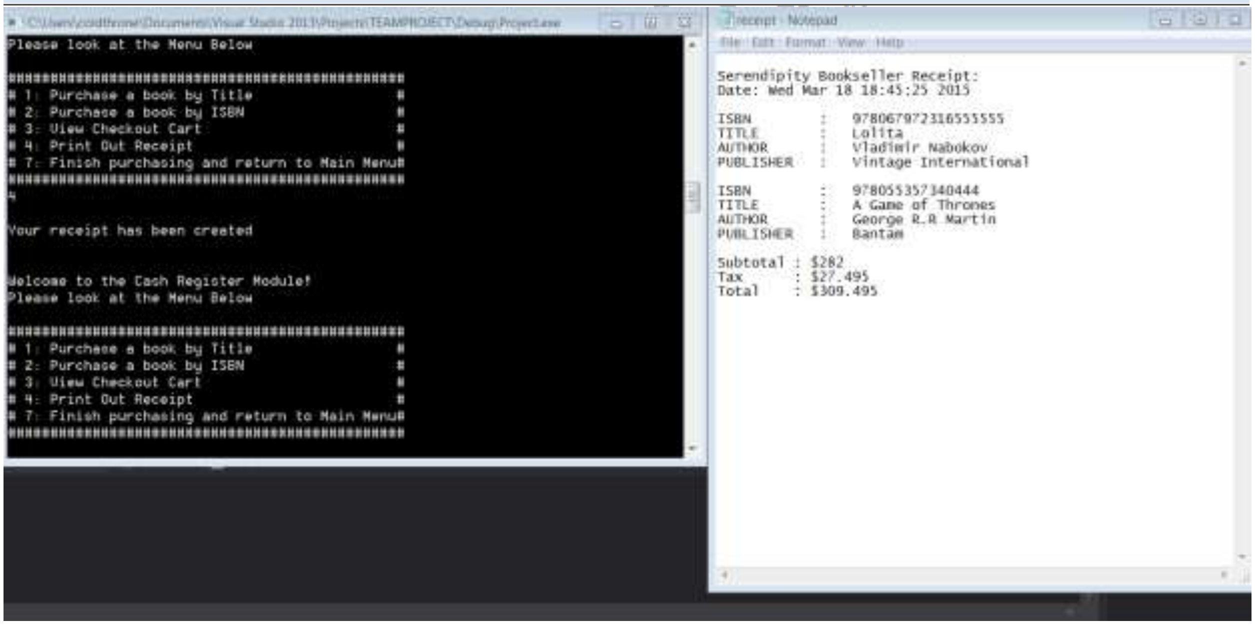
```
# has been provided for your convenience.     #
###############################################

Book.h
Date.h
Debug
Inventory.csv
Project.vcxproj
Project.vcxproj.filters
receipt.txt
Sorting.h
Source.cpp

inventory.csv

     Which module would you like to load?
###############################################
# 1: Cashier Module                           #
# 2: Inventory Module                         #
# 3: Report Module                            #
# 4: Credits                                  #
#                                             #
# 8: Exit                                     #
###############################################
```


C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

```
# 8: Exit                                     #
###############################################
1

Welcome to the Cash Register Module!
Please look at the Menu Below

###############################################
# 1: Purchase a book by Title                 #
# 2: Purchase a book by ISBN                  #
# 3: View Checkout Cart                       #
# 4: Print Out Receipt                        #
# 7: Finish purchasing and return to Main Menu#
###############################################
1

Please Input the title of the book you want
Lolita
We are pleased to have your book in our selection
Lolita : 1 copies in stock
Would you like to add the book to Check out? (Y or N)
y
How many copies would you like to buy?
1
You have added Lolita to your shopping cart
```

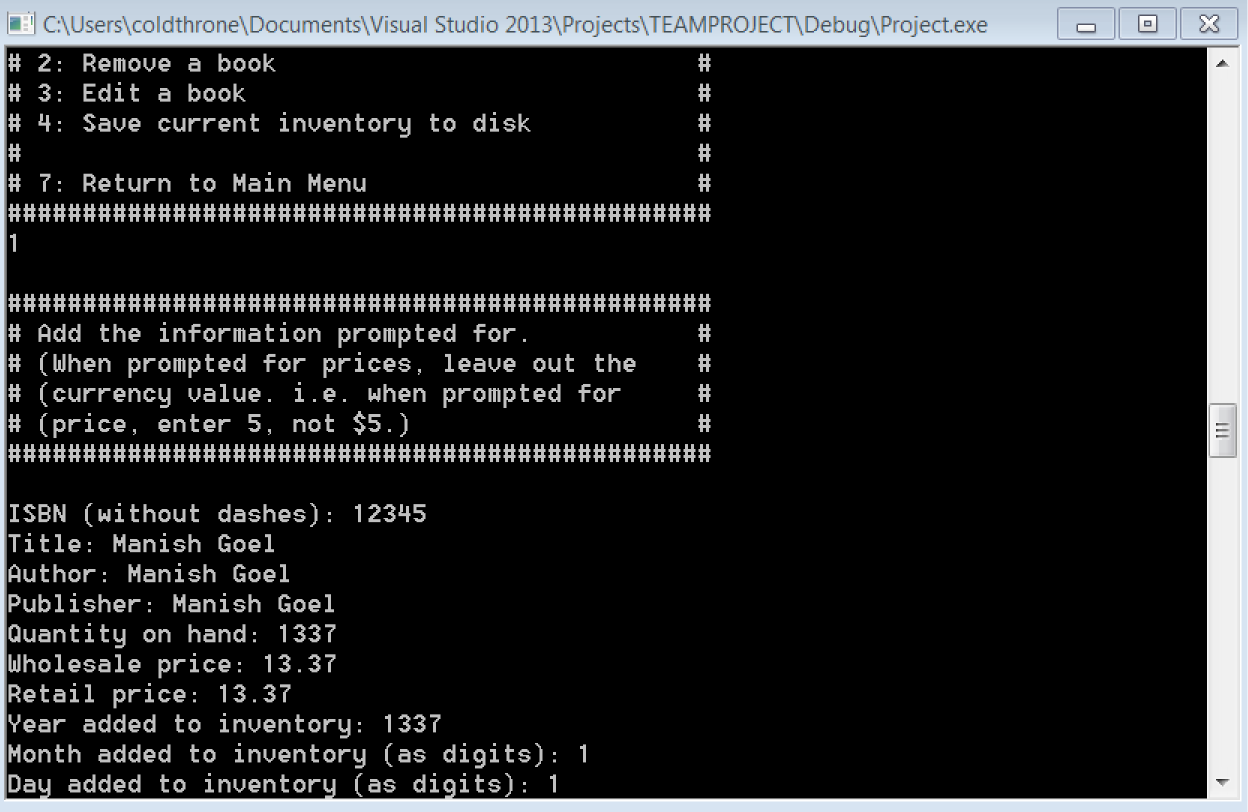

C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

```
# 4: Print Out Receipt                         #
# 7: Finish purchasing and return to Main Menu#
###############################################
3

Hello, This is the shopping cart menu
the books in your current shopping cart are:
Date : Wed Mar 18 18:45:25 2015
QTY ISBN                   TITLE                                    PRICE  SUBTTL
------------------------------------------------------------------------------
1   978067972316555555  Lolita                                   $13.95 $13.95
30  978055357340444     A Game of Thrones                        $8.99  $269.7
Your total balance is: $282

Welcome to the Cash Register Module!
Please look at the Menu Below

###############################################
# 1: Purchase a book by Title                  #
# 2: Purchase a book by ISBN                   #
# 3: View Checkout Cart                        #
# 4: Print Out Receipt                         #
# 7: Finish purchasing and return to Main Menu#
###############################################
```


C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe
receipt - Notepad
File Edit Format View Help

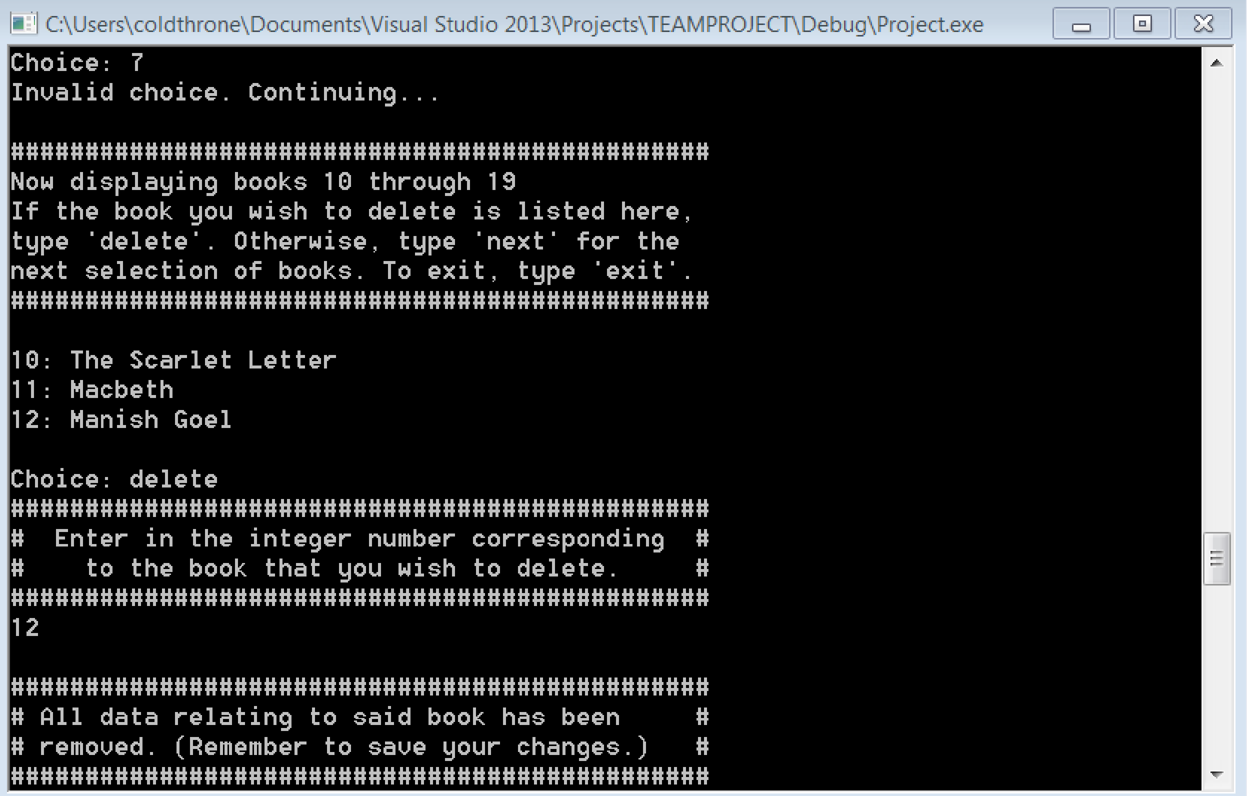
```
Please look at the Menu Below

###############################################
# 1: Purchase a book by Title                  #
# 2: Purchase a book by ISBN                   #
# 3: View Checkout Cart                        #
# 4: Print Out Receipt                         #
# 7: Finish purchasing and return to Main Menu#
###############################################
4

Your receipt has been created


Welcome to the Cash Register Module!
Please look at the Menu Below

###############################################
# 1: Purchase a book by Title                  #
# 2: Purchase a book by ISBN                   #
# 3: View Checkout Cart                        #
# 4: Print Out Receipt                         #
# 7: Finish purchasing and return to Main Menu#
###############################################
```

```
Serendipity Bookseller Receipt:
Date: Wed Mar 18 18:45:25 2015

ISBN         :   978067972316555555
TITLE        :   Lolita
AUTHOR       :   Vladimir Nabokov
PUBLISHER    :   Vintage International

ISBN         :   978055357340444
TITLE        :   A Game of Thrones
AUTHOR       :   George R.R Martin
PUBLISHER    :   Bantam

Subtotal : $282
Tax      : $27.495
Total    : $309.495
```

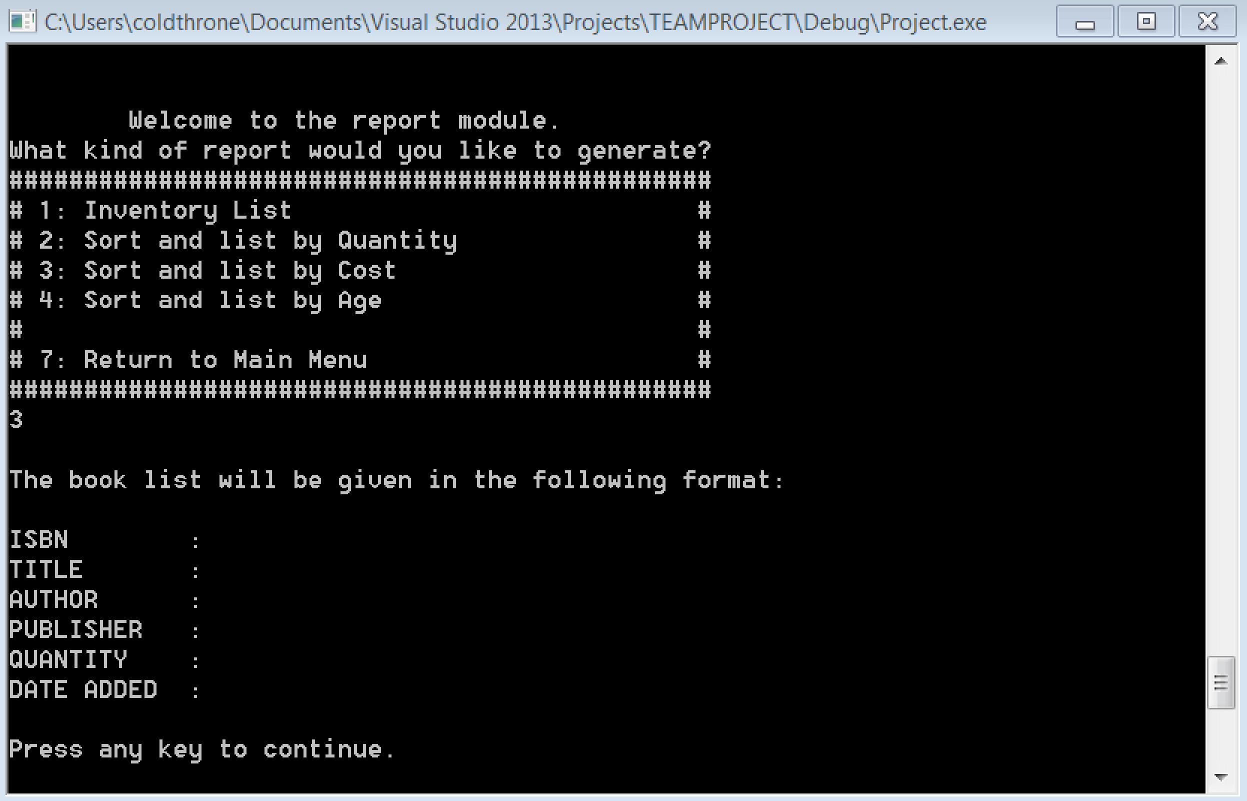


C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

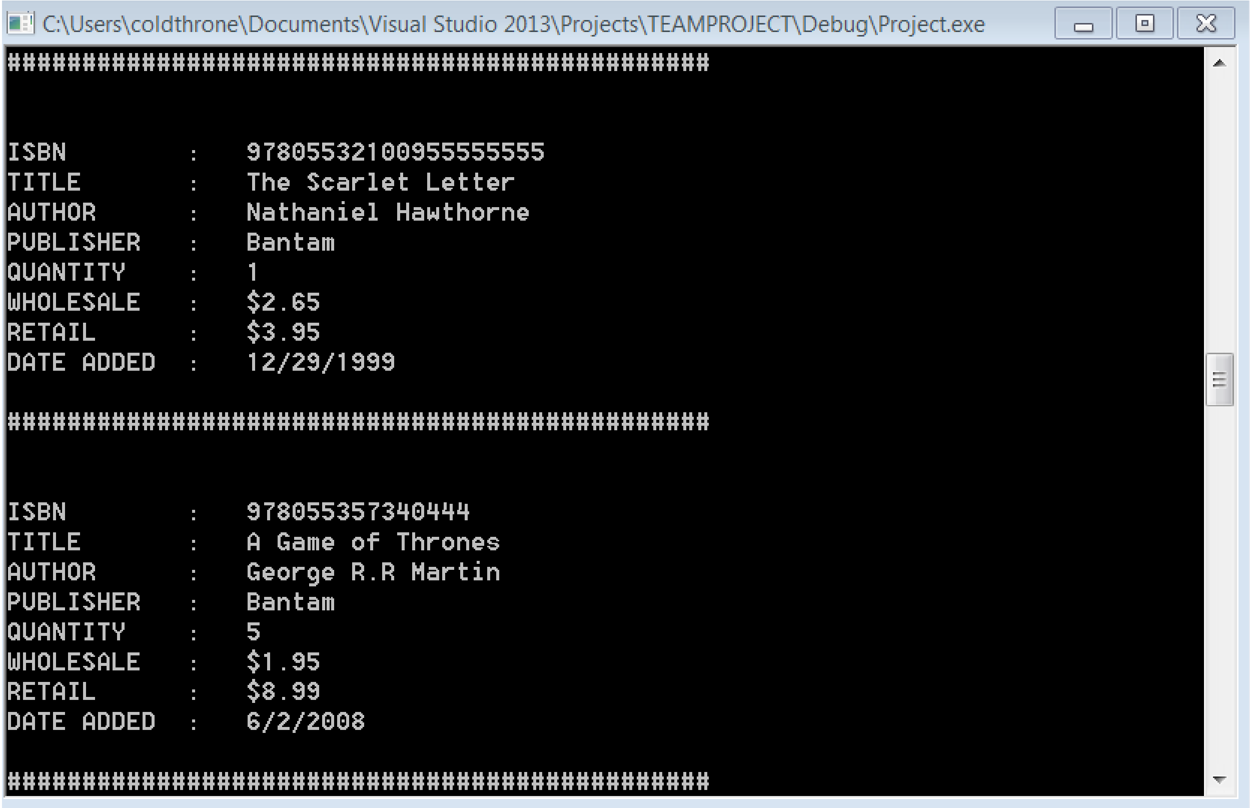
```
# 2: Remove a book                              #
# 3: Edit a book                                #
# 4: Save current inventory to disk             #
#                                               #
# 7: Return to Main Menu                        #
#################################################
1

#################################################
# Add the information prompted for.             #
# (When prompted for prices, leave out the      #
# (currency value. i.e. when prompted for       #
# (price, enter 5, not $5.)                     #
#################################################

ISBN (without dashes): 12345
Title: Manish Goel
Author: Manish Goel
Publisher: Manish Goel
Quantity on hand: 1337
Wholesale price: 13.37
Retail price: 13.37
Year added to inventory: 1337
Month added to inventory (as digits): 1
Day added to inventory (as digits): 1
```


C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

```
Choice: 7
Invalid choice. Continuing...

###############################################
Now displaying books 10 through 19
If the book you wish to delete is listed here,
type 'delete'. Otherwise, type 'next' for the
next selection of books. To exit, type 'exit'.
###############################################

10: The Scarlet Letter
11: Macbeth
12: Manish Goel

Choice: delete
###############################################
#  Enter in the integer number corresponding  #
#    to the book that you wish to delete.     #
###############################################
12

###############################################
# All data relating to said book has been     #
# removed. (Remember to save your changes.)   #
###############################################
```

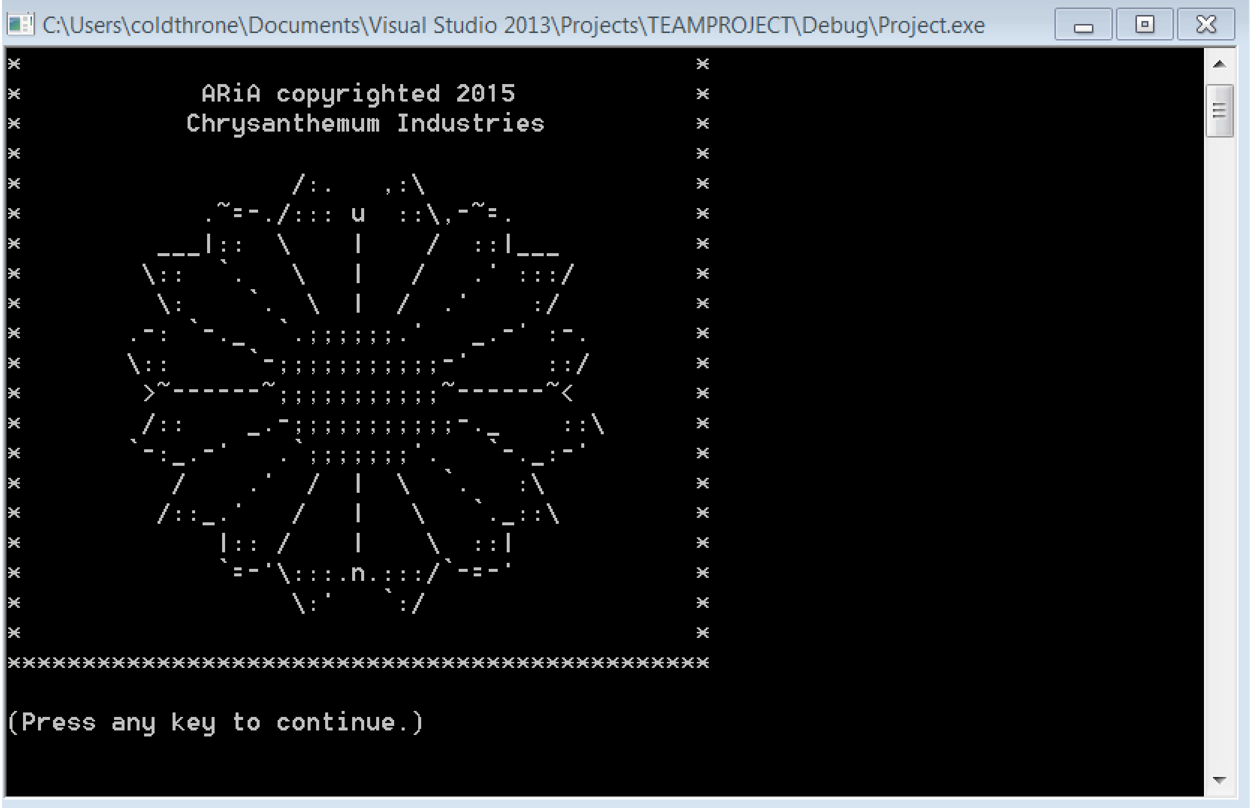

C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

```


        Welcome to the report module.
What kind of report would you like to generate?
###############################################
# 1: Inventory List                           #
# 2: Sort and list by Quantity                #
# 3: Sort and list by Cost                    #
# 4: Sort and list by Age                     #
#                                             #
# 7: Return to Main Menu                      #
###############################################
3

The book list will be given in the following format:

ISBN        :
TITLE       :
AUTHOR      :
PUBLISHER   :
QUANTITY    :
DATE ADDED  :

Press any key to continue.
```



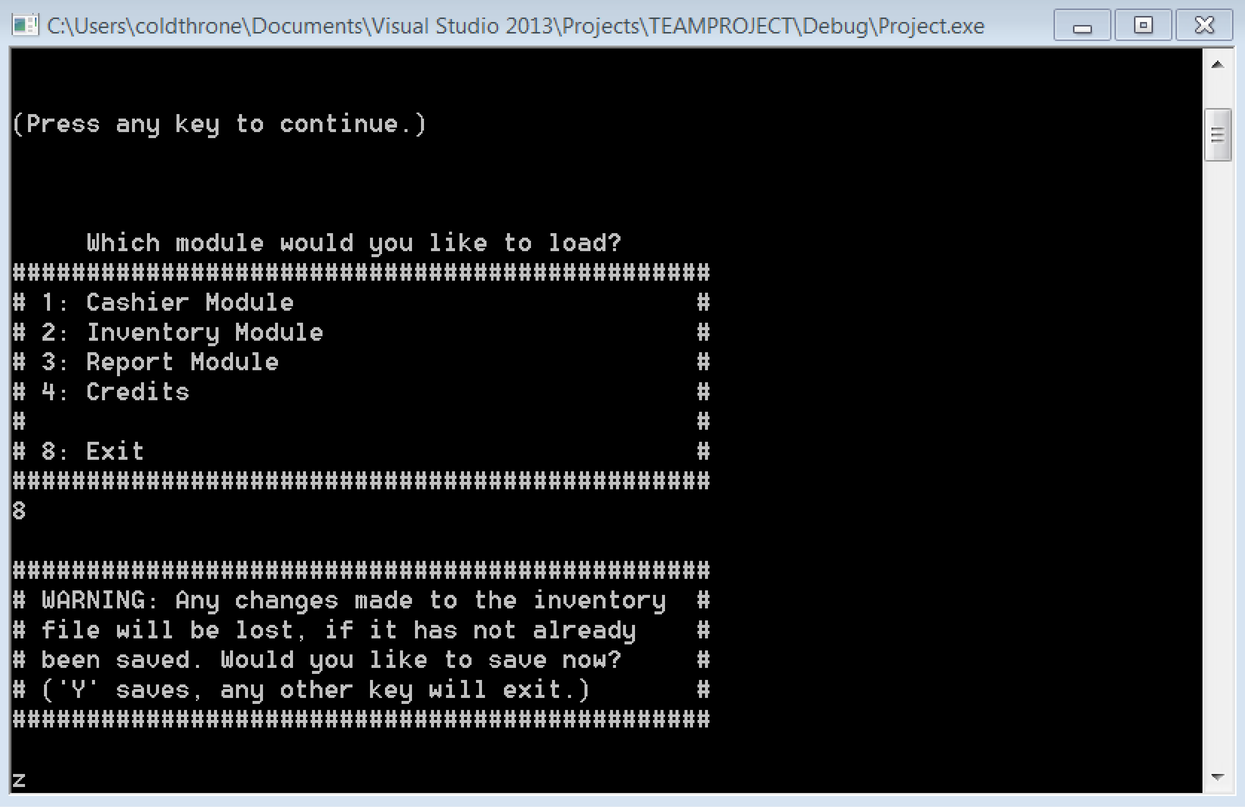
C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

```
###############################################


ISBN        :   97805532100955555555
TITLE       :   The Scarlet Letter
AUTHOR      :   Nathaniel Hawthorne
PUBLISHER   :   Bantam
QUANTITY    :   1
WHOLESALE   :   $2.65
RETAIL      :   $3.95
DATE ADDED  :   12/29/1999

###############################################


ISBN        :   978055357340444
TITLE       :   A Game of Thrones
AUTHOR      :   George R.R Martin
PUBLISHER   :   Bantam
QUANTITY    :   5
WHOLESALE   :   $1.95
RETAIL      :   $8.99
DATE ADDED  :   6/2/2008

###############################################
```



C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

```
*                                             *
*            ARiA copyrighted 2015            *
*           Chrysanthemum Industries          *
*                                             *
*                  /:.   ,:\                  *
*            .~=-./::: u  ::\,-~=.            *
*         ___|::  \    |    /  ::|___         *
*        \::  `.   \   |   /   .' :::/        *
*         \:    `.  \  |  /  .'    :/         *
*       .-: `-._  `.;;;;;;.'   _.-' :-.       *
*       \::     `-;;;;;;;;;;;-'     ::/       *
*        >~------~;;;;;;;;;;;~------~<        *
*        /::    _.-;;;;;;;;;;;-._    ::\      *
*       `-:_.-'   .`;;;;;;;'.   `-._:-'       *
*          /    .'  /  |  \  `.   :\          *
*         /::_.'   /   |   \   `._::\         *
*             |:: /    |    \  ::|            *
*             `=-'\:::.n.:::/`-=-'            *
*                  \:'   `:/                  *
*                                             *
***********************************************

(Press any key to continue.)
```


C:\Users\coldthrone\Documents\Visual Studio 2013\Projects\TEAMPROJECT\Debug\Project.exe

```

(Press any key to continue.)



     Which module would you like to load?
################################################
# 1: Cashier Module                            #
# 2: Inventory Module                          #
# 3: Report Module                             #
# 4: Credits                                   #
#                                              #
# 8: Exit                                      #
################################################
8

################################################
# WARNING: Any changes made to the inventory   #
# file will be lost, if it has not already     #
# been saved. Would you like to save now?      #
# ('Y' saves, any other key will exit.)        #
################################################

z
```