Book.cpp

2 constructors -

- Default constructor
    - Set isbn, title, author, and publisher equal to empty strings
- Second constructor
    - Set isbn equal to what the user passes in as parameters
    - Set title equal to what the user passes in as parameters
    - Set author equal to what the user passes in as parameters
    - Set publisher equal to what the user passes in as parameters

Include destructor

Operation overload of =

- Setting a Book object equal to another Book means that all of the attributes of the Book should be the same as the attributes of the Book it is set equal to
- The address to the book should be passed as the parameter

Operation overload of <<

- Prints out the attribute values of the book the operation is performed on

Cashier.cpp

Constructor with a pointer to InventoryDatabase object as a parameter

- Should set the pInventoryDatabase pointer attribute equal to the parameter
- Should set the unique pointer cart attribute to a list of InventoryBooks of size 100
- Should set cart size to 0
- Should set the unique pointer inventory attribute by calling the getInventoryArray function in InventoryDatabase

Add Book to Cart function

- Should have a string as parameter (the isbn number of the book)
- Should go through the inventory array using a for loop
    - If the isbn of the book is equal to the string argument, should add the book to cart by increasing quantity
    - Should decrease the quantity in the copy of the inventory array in Inventory Database

Remove Book from Cart function

- Should have a string as parameter (the isbn number of the book)
- Should go through the inventory array using a for loop

    - If the isbn of the book is equal to the string argument, should remove the book from cart by decreasing quantity
    - Should increase the quantity in the copy of the inventory array in Inventory Database

Get Book Cart Index
- Go through the cart using a for loop
- If the isbn number passed as a parameter matches the isbn of the book, pass i (the counter in the for loop)

Book price
- Isbn in the string parameter
- If getBookIndexByIsbn returns -1 when isbn is passed as the argument, then -1 is returned
    - If it doesn't pass -1, the price is returned by finding the book in the inventory by isbn and multiplying its wholesale number by sales tax

Subtotal
- Function returns subtotal of books in the cart by looping through the cart and multiplying the quantity of the book times its retail price

Total Price of Cart
- Does the same thing as the above function but multiplies the subtotal by the sales tax and adds it to the subtotal

Checkout
- Reduces quantity in inventory by the amount of books that are in the cart
- Loops through the books in the cart and subtracts this value from the inventory array
- Does this in the copy of the array and the inventory array in the InventoryDatabase by calling addToBookQuantityByIsbn

getCart
- Make a copyCartArray unique pointer of cart size
- Go through the cart array and assign all the attribute values of each book to the books in the copyCartArray
- Return the copyCartArray

printCart
- Loop through the cart array
- Print out the isbn, title, quantity, and retail

printCartForReciept
- Loop through the cart using a for loop
- Print the attribute values of the book, formatting the way you desire

- Go to the next line at the end of the loop

clearCart
- Set the cart size to 0
- Reassign cart to a unique pointer cart attribute to a list of InventoryBooks of size 100 (All attribute values reset)

InventoryBook.cpp

3 constructors -
- Default constructor
    - Set isbn, title, author, and publisher, and addDate equal to empty strings
    - Set quantity, wholesale, and retail to 0
- Second constructor
    - The address to an InventoryBook should be the parameter
    - Set the attributes (isbn, title, author, and publisher, addDate, quantity, wholesale, and retail) equal to the attributes of the InventoryBook in the parameter
- Third constructor
    - 5 strings and 3 ints should be in the parameter
    - Set the attributes isbn, title, author, and publisher, and addDate equal to the string values
    - Set the quantity, wholesale, and retail equal to the int values

Operation overload of =
- Setting an InventoryBook object equal to another InventoryBook means that all of the attributes of the InventoryBook should be the same as the attributes of the InventoryBook it is set equal to

Operation overload of +, -, +=, -=, ++, —
- These operations are performed on the quantity attribute of the book

Operation overload of <<
- Prints out the attribute values of the book the operation is performed on

InventoryDatabase.cpp

parseString
- Use the delimiter to parse the string and then return this string

fileToString
- Create a input file object with the string path as the argument (the path string should be a parameter of the function)

- Reserve space in the string to store the entire file
- Store the entire file in the string by iterating from beginning to end

bookToString
- Turn the book into a string using the delimiter and return this string

inventoryArrayToString
- Create a string variable named bookString
- Loop through each book in the string
- Call the function bookToString and pass the current book in the inventory array
    - Append what the function returns to bookString
- After the loop is finished, return bookString

getNumBooksInString
- Using a while loop and the book delimiter, get the number of books that are in the inventory string

saveInventoryArrayToFile
- Convert the inventory array to a string
- Save it to a file by creating an ofstream object (output file) and writing into the output file
- Close the output file at the end

buildInventoryArray
- Use a for loop to loop through each index in the inventory array
- Use the parse string function to set the title, isbn, author, publisher, addDate, quantity, wholesale, and retail of each book in the inventory array

getInventoryArray
- Make a copyInventoryArray unique pointer of cart size
- Go through the inventory array and assign all the attribute values of each book to the books in the copyInventoryArray
- Return the copyInventoryArray

getInventoryArraySize
- Returns the inventory size attribute

set Book Isbn By Isbn
- The isbn string (isbn) of the book should be the parameter as well as the isbn that the user wants to change the book's isbn to (edit)
- Loop through the inventory array
- If the isbn passed as the argument is equal to the isbn of the book, set the isbn of the book to edit

setBookTitleByIsbn

- The isbn string (isbn) of the book should be the parameter as well as the title that the user wants to change
- Loop through the inventory array
- If the isbn passed as the argument is equal to the isbn of the book, set the title of the book to title passed in the parameter

setBookAuthorByIsbn

- The isbn string (isbn) of the book should be the parameter as well as the author that the user wants to change
- Loop through the inventory array
- If the isbn passed as the argument is equal to the isbn of the book, set the author of the book to author passed in the parameter

setBookPublisherByIsbn

- The isbn string (isbn) of the book should be the parameter as well as the publisher that the user wants to change
- Loop through the inventory array
- If the isbn passed as the argument is equal to the isbn of the book, set the publisher of the book to publisher passed in the parameter

setBookAddDateByIsbn, setBookQuantityByIsbn, setBookWholesaleByIsbn, setBookRetailByIsbn, addToBookQuantityByIsbn should have the same psuedocode as above except with alternate parameters depending on the function

getBookByIsbn

- The isbn on the book should be the parameter
- Loop through the inventory array
- If the isbn passed as the argument matches the isbn of the book, return that book

getBookIndexByIsbn

- The isbn on the book should be the parameter
- Loop through the inventory array
- If the isbn passed as the argument matches the isbn of the book, return the index of that book by returning the counter of the for loop

addBookToArray

- Create a new inventory array and make the size the old size + 1
- Copy the old books into the array
- Copy the new book into the array
- The new book should be passed as the parameter
- Swap the old array and the new array, then set the array size variable to the right value
- Save changes to file

remove Book From Array

- Make a new array to hold the old books, minus the one we are removing
- Keep track of whether we have passed the book to remove or not
- Copy the old books into the new array, except the one we are removing
- Swap the old array and the new array, then set the array size variable to the right value
- Save changes to file

2 constructors

- Default constructor
    - Should set inventoryFilePath to an empty string as well as inventoryString to an empty string
- 2nd constructor
    - Should set inventoryFilePath to the string named path (should be a parameter) and should set inventoryString to an empty string

Include the destructor, and call saveInventoryArrayToFile in the destructor

Report.cpp

selectionSortQunatity

- A pointer to Inventory Book and an int named numBooks should be the parameters
- Iterate through and find the book with the largest quantity
- Swap the book with the largest quantity with the index farthest to the left that is unsorted
- Keep doing this till the entire list is sorted (this can be done using 2 for loops)

selectionSortCost

- A pointer to Inventory Book and an int named numBooks should be the parameters
- Iterate through and find the book with the largest cost
- Swap the book with the largest quantity with the index farthest to the left that is unsorted
- Keep doing this till the entire list is sorted (this can be done using 2 for loops)

selectionSortAge

- A pointer to Inventory Book and an int named numBooks should be the parameters
- Iterate through and find the book with the largest age
- Swap the book with the largest quantity with the index farthest to the left that is unsorted
- Keep doing this till the entire list is sorted (this can be done using 2 for loops)

converyDateToInt

- Use the function stoi to convert the date to an integer value and return this integer value

getBookTotalWholesale

- The parameter should be an InventoryBook object

- Multiply its quantity times wholesale and return this number

getBookTotalRetail

- The parameter should be an InventoryBook object
- Multiply its quantity times retail and return this number