// =======================

// Included: Lab 9 Book.cpp, Book.h, Source.cpp

// =======================

// Lab 9

// =======================

// Christian Falucho

// CMPR 121

// =======================

/\*================ CONTENTS FROM Book.h ===================

#include <iostream>

#include <string>

#ifndef BOOK\_H

#define BOOK\_H

using namespace std;

class Book{

private:

string ISBN;

int year;

double price;

static int bookCount;

public:

Book();

Book(string, int, double);

~Book();

int getCount();

void displayBook() const;

bool operator>(Book&) const;

bool operator==(Book&) const;

bool operator>(double);

double operator+(Book&);

bool operator<(int);

friend ostream& operator<<(ostream&, Book&);

friend istream& operator>>(istream&, Book&);

};

#endif

/\*================ CONTENTS FROM Book.h ===================\*/

/\*================ CONTENTS FROM Book.cpp ===================

#include <iostream>

#include <string>

#include "Book.h"

Book::Book(){

ISBN = "";

year = 0;

price = 00.00;

bookCount++;

}

Book::Book(string ISBN, int year, double price){

this->ISBN = ISBN;

this->year = year;

this->price = price;

bookCount++;

}

Book::~Book(){

}

int Book::bookCount = 0;

void Book::displayBook() const{

cout << "ISBN: " << ISBN << endl;

cout << "Year: " << year << endl;

cout << "Price: " << price << endl;

}

int Book::getCount(){

return bookCount;

}

// Use to compare prices of two Book objects.

bool Book::operator>(Book& book) const{

if (this->price > book.price)

{

return true;

}else{

return false;

}

}

// Use to compare prices of two Book objects.

bool Book::operator==(Book& book) const{

if (this->price == book.price)

{

return true;

}else{

return false;

}

}

// Use to compare an object’s price to 10.00.

bool Book::operator>(double price){

if (this->price > price)

{

return true;

}else{

return false;

}

}

double Book::operator+(Book& book){

return price + book.price;

}

bool Book::operator<(int year){

if(this->year < year){

return true;

}else{

return false;

}

}

ostream& operator<<(ostream& osObject, Book& book){

osObject << "ISBN: " << book.ISBN << endl

<< "Year: " << book.year << endl

<< "Price: " << book.price << endl;

return osObject;

}

istream& operator>>(istream& stream, Book& book)

{

cout << "ISBN: ";

stream >> book.ISBN;

cout << "Year: ";

stream >> book.year;

cout << "Price: ";

stream >> book.price;

return stream;

}

/\*================ CONTENTS FROM Book.cpp ===================

/\*================ CONTENTS FROM Source.cpp ===================

/\*

===========================================================

=== MAIN FUNCTION ENDS ===

===========================================================

\*/

#include <iostream>

#include <string>

#include "Book.h"

using namespace std;

int main()

{

Book b1("0-12345-9", 1990, 12.50);

Book b2("0-54321-9", 2001, 7.75);

Book b3;

double avg;

cout << "Here is book #1:\n";

b1.displayBook();

cout << endl;

cout << "Here is book #2:\n";

b2.displayBook();

cout << "There are " << b1.getCount() << " books.\n\n";

// ------------------------------

if (b1 > b2) // Use this overloaded function: bool operator>();

cout << "Book #1 has a higher price.\n\n";

else

cout << "Book #1 does not have a higher price.\n\n";

// ------------------------------

if (b1 == b2) // Use this overloaded function: bool operator==();

cout << "Same price.\n\n";

else

cout << "Not the same price.\n\n";

// ------------------------------

if (b2 > 10.00) // Use this overloaded function: bool operator>();

cout << "The price is more than $10.00.\n\n";

else

cout << "The price is not more than $10.00.\n\n";

// ---------------

avg = (b1 + b2) / 2.0; // Use this overloaded function: bool operator+();

cout << "The average book price is " << avg << ".\n\n";

// ---------------

if (b1 < 2000) // Use this overloaded function: bool operator<();

cout << "The book was published before 2000.\n\n";

else

cout << "The book was not published before 2000.\n\n";

// ---------------

cout << b1; // Use this: friend ostream& operator<<();

// ---------------

cout << "\nEnter Book #3 information.\n";

cin >> b3; // Use this: friend istream& operator>>();

// (This function is given to you below).

cout << "\nHere is what you entered for Book #3:\n";

cout << b3; // Use this: friend ostream& operator<<();

return 0;

}

/\*

===========================================================

=== MAIN FUNCTION ENDS ===

===========================================================

\*/

/\*================ CONTENTS FROM Source.cpp ===================

/\*

===========================================================

=== CODE OUTPUT ===

===========================================================

\*/

A screenshot of a computer screen

AI-generated content may be incorrect.

/\*

===========================================================

=== CODE OUTPUT ===

===========================================================

\*/