RIPHAH INTERNATIONAL UNIVERSITY, GULBERG GREEN CAMPUS



**SCD Lab**

**Submitted to: Ms. Shazwa Tun Naeem**

**Submitted by: Ayesha Mukhtar Asad–44587**

#include <iostream>

#include <vector>

#include <string>

#include <iomanip>

using namespace std;

// info expert data

class Book {

public:

string title;

double price;

Book(string title, double price) {

this->title = title;

this->price = price;

}

};

// Creator creation

class OrderItem {

public:

Book book;

int quantity;

OrderItem(Book book, int quantity) : book(book), quantity(quantity) {}

// Marking this method as const to allow it to be called on const objects

double getItemTotal() const {

return book.price \* quantity;

}

};

// information expert

class Order {

public:

vector<OrderItem> items;

void addItem(Book book, int quantity) {

OrderItem orderItem(book, quantity);

items.push\_back(orderItem);

}

void getOrderSummary() const { // Marked this method as const since it doesn't modify the object

double totalPrice = 0;

cout << "\nOrder Summary:\n";

for (const auto& item : items) {

double itemTotal = item.getItemTotal(); // Calling the const method

cout << "Book: " << item.book.title

<< ", Quantity: " << item.quantity

<< ", Price: $" << fixed << setprecision(2) << itemTotal << endl;

totalPrice += itemTotal;

}

cout << "\nTotal Price: $" << fixed << setprecision(2) << totalPrice << endl;

}

};

// Controller

class BookshopController {

public:

vector<Book> books;

Order order;

void addBookToCatalog(string title, double price) {

Book book(title, price);

books.push\_back(book);

}

void displayBooks() {

cout << "\nAvailable Books:\n";

for (size\_t i = 0; i < books.size(); ++i) {

cout << i + 1 << ". " << books[i].title << " - $" << fixed << setprecision(2) << books[i].price << endl;

}

}

void handleOrder() {

while (true) {

displayBooks();

cout << "\nEnter the book number to add to your order (or 'done' to finish): ";

string input;

getline(cin, input);

if (input == "done") {

break;

}

try {

int bookIndex = stoi(input) - 1;

if (bookIndex >= 0 && bookIndex < books.size()) {

cout << "Enter quantity for " << books[bookIndex].title << ": ";

int quantity;

cin >> quantity;

cin.ignore(); // To consume the newline left by cin

if (quantity > 0) {

order.addItem(books[bookIndex], quantity);

} else {

cout << "Quantity must be greater than 0. Try again." << endl;

}

} else {

cout << "Invalid book number. Try again." << endl;

}

} catch (invalid\_argument& e) {

cout << "Invalid input. Please enter a valid number." << endl;

}

}

order.getOrderSummary();

}

};

// Main function

int main() {

BookshopController controller;

// Adding books to the catalog

controller.addBookToCatalog("Herry ", 10.0);

controller.addBookToCatalog("Time Machine", 15.5);

controller.addBookToCatalog("Oliver Twist", 7.75);

cout << "\nWelcome to the Bookshop!" << endl;

controller.handleOrder();

return 0;

}