// COSC 1320 Summer 2015

// Name: Adrian Davila

// Programming Assignment 4

// This is my own work; I will not post

#include <iostream>

#include <string>

#include "Video.h"

#include "Customer.h"

#include "Controller.h"

#include "View.h"

using namespace std;

int main(){

Controller cont;

View view;

int option = 0;

cont.readFile();

while(option != 6){

view.showMenu();

cin >> option;

switch(option){

case 1:

cont.displayInventory();

break;

case 2:

cont.searchInventory();

break;

case 3:

cont.displayCustomers();

break;

case 4:

cont.displayCustomerInfo();

break;

case 5:

cont.displayCustomerRecord();

break;

case 6:

cont.writeFile();

break;

default:

cout << "Input not recognized" << endl;

break;

}

}

return 0;

}

// COSC 1320 Summer 2015

// Name: Adrian Davila

// Programming Assignment 4

// This is my own work; I will not post

#include <iostream>

#include <string>

#include "Video.h"

#include "Customer.h"

#include "Controller.h"

#include "View.h"

using namespace std;

int main(){

Controller cont;

View view;

int option = 0;

cont.readFile();

while(option != 6){

view.showMenu();

cin >> option;

switch(option){

case 1:

cont.displayInventory();

break;

case 2:

cont.searchInventory();

break;

case 3:

cont.displayCustomers();

break;

case 4:

cont.displayCustomerInfo();

break;

case 5:

cont.displayCustomerRecord();

break;

case 6:

cont.writeFile();

break;

default:

cout << "Input not recognized" << endl;

break;

}

}

return 0;

}

// COSC 1320 Summer 2015

// Name: Adrian Davila

// Programming Assignment 4

// This is my own work; I will not post

#include <iostream>

#include <string>

#include "Video.h"

#include "Customer.h"

#include "Controller.h"

#include "View.h"

using namespace std;

int main(){

Controller cont;

View view;

int option = 0;

cont.readFile();

while(option != 6){

view.showMenu();

cin >> option;

switch(option){

case 1:

cont.displayInventory();

break;

case 2:

cont.searchInventory();

break;

case 3:

cont.displayCustomers();

break;

case 4:

cont.displayCustomerInfo();

break;

case 5:

cont.displayCustomerRecord();

break;

case 6:

cont.writeFile();

break;

default:

cout << "Input not recognized" << endl;

break;

}

}

return 0;

}

// COSC 1320 Summer 2015

// Name: Adrian

// Programming Assignment 4

// This is my own work; I will not post

#pragma once

#include <iostream>

using namespace std;

class View

{

public:

View(void);

~View(void);

void showMenu();

};

// COSC 1320 Summer 2015

// Name: Adrian

// Programming Assignment 4

// This is my own work; I will not post

#pragma once

#include <iostream>

#include <string>

#include "View.h"

using namespace std;

View::View(void)

{

}

View::~View(void)

{

}

void View::showMenu(){

const int menuLength = 6;

string menu[menuLength] = { "Show the inventory" ,

"Search the inventory" ,

"Display the customers",

"Show a customer's info" ,

"Show a customer's records",

"Save and exit"};

cout << endl;

for(int i = 0; i < menuLength; i++){

cout << (i+1) << " " << menu[i] << endl;

}

cout << endl;

}

// COSC 1320 Summer 2015

// Name: Adrian Davila

// Programming Assignment 4

// This is my own work; I will not post

#pragma once

#include <string>

using namespace std;

const int MAX = 5;

class Customer{

private:

string fName;

string lName;

int id;

int amountRented;

string \*currentRentals[MAX];

public:

Customer(void);

~Customer(void);

Customer(string , string , int);

string getFirstName();

string getLastName();

int getID();

void setFirstName(string);

void setLastName(string);

void setID(int);

void addRental(string);

void deleteRental(string);

string getRentals();

int getRentalCount();

string toString();

};

// COSC 1320 Summer 2015

// Name: Adrian Davila

// Programming Assignment 4

// This is my own work; I will not post

#include "Customer.h"

#include <sstream>

#include <iostream>

using namespace std;

Customer::Customer(void){

fName = "No given name";

lName = "No surname";

id = -1;

amountRented = 0;

}

Customer::~Customer(void)

{

}

Customer::Customer(string givenName, string surname, int newID){

fName = givenName;

lName = surname;

id = newID;

amountRented = 0;

}

string Customer::getFirstName(){

return fName;

}

string Customer::getLastName(){

return lName;

}

int Customer::getID(){

return id;

}

void Customer::setFirstName(string givenName){

fName = givenName;

}

void Customer::setLastName(string surname){

lName = surname;

}

void Customer::setID(int newID){

id = newID;

}

void Customer::addRental(string title){

if(amountRented == 5){

return;

}

//cout << amountRented << endl;

currentRentals[amountRented] = new string(title);

amountRented++;

}

void Customer::deleteRental(string title){

int h = 0;

for(int i = 0; i < amountRented; i++){

//if the flag is triggered, the array is shifted to the left by one

if(h == 1){

//delete (\*empArr[i]);

currentRentals[i-h] = currentRentals[i];

}

if((\*currentRentals[i]) == title){

//if there is a match, the flag (h) is triggered

h = 1;

cout << "Match found" << endl;

}

if(h == 0 && i == amountRented-1){

cout << "No match found." << endl;

}

}

if(h == 1){

amountRented--;

}

}

string Customer::getRentals(){

if(amountRented == 0){

return "No rentals";

}

string str = "";

for(int i = 0; i < amountRented; i++){

//cout << "hit" << endl;

//cout << currentRentals[i] << endl;

if(i > 0){

str += ",";

}

str += \*currentRentals[i];

}

return str;

}

int Customer::getRentalCount(){

return amountRented;

}

string Customer::toString(){

ostringstream oss;

oss << id;

string str = "";

str += fName + ",";

str += lName + ",";

str += oss.str();

str += "\n";

return str;

}

// COSC 1320 Summer 2015

// Name: Adrian Davila

// Programming Assignment 4

// This is my own work; I will not post

#pragma once

#include <string>

using namespace std;

class Video{

private:

string title;

string star1;

string star2;

string producer;

string director;

string prodCo;

int copies;

public:

Video(void);

Video(string , string , string , string , string , string , int);

~Video(void);

string getTitle();

string getStar1();

string getStar2();

string getProducer();

string getDirector();

string getCompany();

int getQuantity();

void setTitle(string);

void setStar1(string);

void setStar2(string);

void setProducer(string);

void setDirector(string);

void setCompany(string);

void setQuantity(int);

void incrementCopies();

void decrementCopies();

string toString();

};

// COSC 1320 Summer 2015

// Name: Adrian Davila

// Programming Assignment 4

// This is my own work; I will not post

#include <sstream>

#include "Video.h"

Video::Video(void)

{

}

Video::Video(string theTitle, string theStar1, string theStar2, string theProd, string theDir, string theCo, int amount){

title = theTitle;

star1 = theStar1;

star2 = theStar2;

producer = theProd;

director = theDir;

prodCo = theCo;

copies = amount;

}

Video::~Video(void)

{

}

string Video::getTitle(){

return title;

}

string Video::getStar1(){

return star1;

}

string Video::getStar2(){

return star2;

}

string Video::getProducer(){

return producer;

}

string Video::getDirector(){

return director;

}

string Video::getCompany(){

return prodCo;

}

int Video::getQuantity(){

return copies;

}

void Video::setTitle(string newTitle){

title = newTitle;

}

void Video::setStar1(string newStar){

star1 = newStar;

}

void Video::setStar2(string newStar){

star2 = newStar;

}

void Video::setProducer(string newPro){

producer = newPro;

}

void Video::setDirector(string newDir){

director = newDir;

}

void Video::setCompany(string newCo){

prodCo = newCo;

}

void Video::setQuantity(int amount){

copies = amount;

}

void Video::incrementCopies(){

copies++;

}

void Video::decrementCopies(){

copies--;

}

string Video::toString(){

ostringstream oss;

oss << copies;

string str = "";

str += title;

str += "\n";

str += star1;

str += "\n";

str += star2;

str += "\n";

str += producer;

str += "\n";

str += director;

str += "\n";

str += prodCo;

str += "\n";

str += oss.str();

str += "\n";

return str;

}