Int ordQuantity;

int selAnt;

Int ordPrice;

cout << "Enter antique number:" << endl;

cin >> selAnt;

ofstream outFS;

outFS.open("log2.txt");

if (!outFS.is\_open()) {

cout << "Could not open file log2.txt" << endl;

return 1;

}

cin>>ordQuantity;

ordPrice=antAnt[selAnt].getPrice()

if(ordPrice<budget){

ordPrice=ordQuantity\*antAnt[selAnt].getPrice();

}

else{

ordPrice=budget;

}

outFS << "Sold Tiles for $" << ordPrice<<endl;

budget -= ordPrice;

revenue += ordPrice;

outFS.close();

for (int i = 0; i < 10; i++) {

float newPrice = antAnt[i].getPrice();

antAnt[i].setPrice(newPrice);

}

int selAnt;

cout << "Enter antique number:" << endl;

cin >> selAnt;

if (quantities[selAnt] == 0) {

cout << "Sorry! Antique is out of stock.";

}

if (budget < 0) {

cout << "Insufficient funds.";

}

else {

budget -= 0;

revenue += 0;

quantities[selAnt] -= 1;

cout << "Enjoy your " << "!";

}

cout<<”Total revenue: $”<<revenue;

cout<<”Remaining budget: $”<<budget;

inSS.clear();

inSS >> name;

inSS >> price;

Return inSS

#include<iostream>

#include <iomanip>

#include <string>

#include <fstream>

#include <sstream>

#include "antique.h"

#include "merchant.h"

Merchant::Merchant(Antique antiques[], int quantities[]) { // Default constructor

for (int i = 0; i < 10; i++) {

antAnt[i].setName(antiques[i].getName());

antAnt[i].setPrice(antiques[i].getPrice());

}

revenue = 0;

}

void Merchant::haggle() {

for (int i = 0; i < 10; i++) {

antAnt[i].setPrice(antAnt[i].getPrice()-(antAnt[i].getPrice()/10));

}

}

void Merchant::printMenu() {

string name;

for (int i = 0; i < 10; i++) {

name= antAnt[i].getName();

cout << i + 1 << ") " << antAnt[i].getName() << ": $" << fixed << setprecision(2) << antAnt[i].getPrice() << endl;

}

}

void Merchant::selectAntique(float budget) {

int selAnt;

int ordPrice;

int itemnum;

cout << "Enter item number:" << endl;

cin >> itemnum;

itemnum--;

ofstream outFS;

outFS.open("log2.txt");

if (quantities[itemnum] == 0) {

cout << "Sorry! Antique is out of stock." << endl;

}

else {

ordPrice = antAnt[itemnum].getPrice();

if (ordPrice < budget) {

outFS << "Sold " << antAnt[itemnum].getName() << " for $" << ordPrice << endl;

revenue += ordPrice;

budget -= ordPrice;

quantities[itemnum] -= 1;

cout << "Enjoy your "<< antAnt[itemnum].getName() <<"!";

}

else {

cout << "Insufficient funds." << endl << endl;

}

outFS.close();

}

}

void Merchant::leave(float budget) {

ofstream outFS;

outFS.open("log2.txt");

outFS << "Total revenue: $" << fixed << setprecision(2)<< revenue << endl;

outFS << "Remaining budget: $" << budget << endl;

}

;