#include<iostream>

#include<string>

#include<fstream>

using namespace std;

#define Max 100

static int s = 0;//货物的总量

class Order;

class Deliver;

class Customer

{

protected:

string name[Max];

string address[Max];

int contact[Max];

int paytype[Max];

public:

friend void getorder(Customer& c, Order& o, Deliver& d);

friend void customercheckorder(Customer& c, Order& o, Deliver& d);

friend void deliverpersoncheckorder(Customer& c, Order& o, Deliver& d);

friend void cancelorder(Customer& c, Order& o, Deliver& d);

friend void modifyorder(Customer& c, Order& o, Deliver& d);

friend void adddeliverperson(Customer& c, Order& o, Deliver& d);

friend void modifypackagestate(Customer& c, Order& o, Deliver& d);

};

class Order

{

protected:

int ordertime[Max];

int gettime, delivertime[Max];

string getaddress[Max];

string deliveraddress[Max];

int weight[Max];

int delivertype[Max], servicetype[Max];

int paytype[Max];

int paytime[Max];

int money[Max];

public:

friend void getorder(Customer& c, Order& o, Deliver& d);

friend void customercheckorder(Customer& c, Order& o, Deliver& d);

friend void deliverpersoncheckorder(Customer& c, Order& o, Deliver& d);

friend void cancelorder(Customer& c, Order& o, Deliver& d);

friend void modifyorder(Customer& c, Order& o, Deliver& d);

friend void adddeliverperson(Customer& c, Order& o, Deliver& d);

friend void modifypackagestate(Customer& c, Order& o, Deliver& d);

void calculatemoney()

{

for (int i = 0; i < s; i++)

{

if (weight[i] <= 1)

money[i] = 22;

else

{

money[i] = 22 + 14 \* (weight[i] - 1);

}

}

}

};

class Deliver

{

protected:

int servicetype[Max];

int gettime[Max];

string deliverperson[Max];

string packagestate[Max];

int delivertime[Max];

public:

friend void getorder(Customer& c, Order& o, Deliver& d);

friend void customercheckorder(Customer& c, Order& o, Deliver& d);

friend void deliverpersoncheckorder(Customer& c, Order& o, Deliver& d);

friend void cancelorder(Customer& c, Order& o, Deliver& d);

friend void modifyorder(Customer& c, Order& o, Deliver& d);

friend void adddeliverperson(Customer& c, Order& o, Deliver& d);

friend void modifypackagestate(Customer& c, Order& o, Deliver& d);

};

Customer c;

Order o;

Deliver d;

void menu();

void customermenu()

{

system("cls");

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "欢迎使用物流系统！" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

ofstream outfile("1.txt", ios::out);//初始化文件，清空存储的内容

if (!outfile)

{

cout << "打开文件失败！" << endl;

exit(1);

}

outfile.close();

int n;

bool b = true;

system("pause");

system("cls");

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*功能序号\*\*\*\*\*\*\*" << endl;

cout << "1.查询货物的信息" << endl;

cout << "2.添加货物信息" << endl;//添加信息

cout << "3.删除货物信息" << endl;//删除信息

cout << "4.修改信息" << endl;

cout << "5.退出登录" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

do

{

cin >> n;

switch (n)

{

case 1: customercheckorder(c,o,d);

break;

case 2: getorder(c,o,d);

break;

case 3: cancelorder(c,o,d);

break;

case 4: modifyorder(c,o,d);

break;

case 5: menu();

break;

default:cout << "请输入标准的功能序号" << endl;

customermenu();

}

} while (b);

}

void getorder(Customer& c, Order& o, Deliver& d)

{

system("cls");

ofstream outfile("1.txt", ios::app);

if (!outfile)

{

cerr << "打开失败" << endl;

exit(1);

}

outfile.close();

cout << "请输入姓名" << endl;

cin >> c.name[s];

outfile << c.name[s] << " ";

cout << "请输入联系方式" << endl;

cin >> c.contact[s];

outfile << c.contact[s] << " ";

cout << "请输入取货地址" << endl;

cin >> o.getaddress[s];

outfile << o.getaddress[s] << " ";

cout << "请输入送货地址" << endl;

cin >> o.deliveraddress[s];

outfile << o.deliveraddress[s] << " ";

cout << "请输入货物重量" << endl;

cin >> o.weight[s];

outfile << o.weight[s] << " ";

cout << "请输入支付方式" << endl;

cout << "（1、现金支付 2、按月支付）";

cin >> o.paytype[s];

outfile << o.paytype[s] << " ";

d.packagestate[s] = "准备就绪";

outfile << d.packagestate[s] << " ";

s++;

outfile.close();

}

void customercheckorder(Customer& c, Order& o, Deliver& d)

{

system("cls");

ifstream infile("1.txt", ios::in);

if (!infile)

{

cout << "打开文件失败" << endl;

exit(1);

}

cout << "姓名 " << "联系方式 " << "取货地址 " << "送货地址 " << "货物重量 " << "包裹状态 " << endl;

for (int j = 0; j < s; j++)

{

infile >> c.name[j];

cout << c.name[j] << " ";

infile >> c.contact[j];

cout << c.contact[j] << " ";

infile >> o.getaddress[j];

cout << o.getaddress[j] << " ";

infile >> o.deliveraddress[j];

cout << o.deliveraddress[j] << endl;

infile >> o.weight[j];

cout << o.weight[j] << endl;

infile >> d.packagestate[j];

cout << d.packagestate[j] << endl;

}

infile.close();

customermenu();

}

void cancelorder(Customer& c, Order& o, Deliver& d)

{

system("cls");

int k;

ifstream infile("1.txt", ios::in);

if (!infile)

{

cout << "打开文件失败" << endl;

exit(1);

}

cout << "姓名 " << "联系方式 " << "取货地址 " << "送货地址 " << "货物重量 " << "包裹状态 " << endl;

for (int j = 0; j < s; j++)

{

infile >> c.name[j];

cout << c.name[j] << " ";

infile >> c.contact[j];

cout << c.contact[j] << " ";

infile >> o.getaddress[j];

cout << o.getaddress[j] << " ";

infile >> o.deliveraddress[j];

cout << o.deliveraddress[j] << endl;

infile >> o.weight[j];

cout << o.weight[j] << endl;

infile >> d.packagestate[j];

cout << d.packagestate[j] << endl;

}

infile.close();

ofstream outfile("1.txt", ios::out);

if (!outfile)

{

cerr << "打开失败" << endl;

exit(1);

}

cout << "你想删除第几个" << endl;

cin >> k;

for (int i = 0; i < s; i++)

{

if (i != k - 1)

{

outfile << c.name[i] << " ";

outfile << c.contact[i] << " ";

outfile << o.getaddress[i] << " ";

outfile << o.deliveraddress[i] << " ";

outfile << o.weight[i] << " ";

outfile << d.packagestate[i] << " ";

}

}

s--;

outfile.close();

customermenu();

}

void modifyorder(Customer& c, Order& o, Deliver& d)

{

system("cls");

int n1;

int z;

string name;

int contact;

string getaddress;

string deliveraddress;

int weight;

ifstream infile("1.txt", ios::in);

if (!infile)

{

cerr << "打开失败" << endl;

exit(1);

}

cout << "姓名 " << "联系方式 " << "取货地址 " << "送货地址 " << "货物重量 " << "包裹状态 " << endl;

for (int j = 0; j < s; j++)

{

infile >> c.name[j];

cout << c.name[j] << " ";

infile >> c.contact[j];

cout << c.contact[j] << " ";

infile >> o.getaddress[j];

cout << o.getaddress[j] << " ";

infile >> o.deliveraddress[j];

cout << o.deliveraddress[j] << endl;

infile >> o.weight[j];

cout << o.weight[j] << endl;

}

infile.close();

cout << "请输入要修改第几个" << endl;

cin >> n1;

for (int i = 0; i <= s; i++)

{

if (n1 - 1 == i)

{

cout << "1.修改姓名" << endl;

cout << "2.修改联系方式" << endl;

cout << "3.修改取货地址" << endl;

cout << "4.修改送货地址" << endl;

cout << "5.修改货物重量" << endl;

cout << "请输入序号" << endl;

cin >> z;

switch (z)

{

case 1: cout << "请输入要修改的姓名" << endl;

cin >> name;

c.name[i] = name;

cout << "修改成功!" << endl;

break;

case 2: cout << "请输入要修改的联系方式" << endl;

cin >> contact;

c.contact[i] = contact;

cout << "修改成功!" << endl;

break;

case 3:cout << "请输入要修改的取货地址" << endl;

cin >> getaddress;

o.getaddress[i] = getaddress;

cout << "修改成功!" << endl;

break;

case 4:cout << "请输入修改的送货地址" << endl;

cin >> deliveraddress;

o.deliveraddress[i] = deliveraddress;

cout << "修改成功!" << endl;

break;

default:

cout << "请输入修改的货物重量" << endl;

cin >> weight;

o.weight[i] = weight;

cout << "修改成功!" << endl;

break;

}

break;

}

}

ofstream outfile("1.txt", ios::out);

if (!outfile)

{

cerr << "打开失败" << endl;

exit(1);

}

for (int j = 0; j < s; j++)

{

outfile << c.name[j] << " ";

outfile << c.contact[j] << " ";

outfile << o.getaddress[j] << " ";

outfile << o.deliveraddress[j] << " ";

outfile << o.weight[j] << " ";

}

outfile.close();

customermenu();

}

void deliverpersonmenu()

{

system("cls");

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "欢迎使用物流系统！" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

ofstream outfile("2.txt", ios::out);//初始化文件，清空存储的内容

if (!outfile)

{

cout << "打开文件失败！" << endl;

exit(2);

}

outfile.close();

int n;

bool b = true;

system("pause");

system("cls");

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*功能序号\*\*\*\*\*\*\*" << endl;

cout << "1.查看订单" << endl;

cout << "2.添加取货人" << endl;

cout << "3.更新包裹状态" << endl;

cout << "4.退出登录" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

do

{

cin >> n;

switch (n)

{

case 1: deliverpersoncheckorder(c, o, d);

break;

case 2: adddeliverperson(c, o, d);

break;

case 3: modifypackagestate(c, o, d);

break;

case 4: menu();

break;

default:cout << "请输入标准的功能序号" << endl;

deliverpersonmenu();

}

} while (b);

}

void deliverpersoncheckorder(Customer& c, Order& o, Deliver& d)

{

system("cls");

ifstream infile("2.txt", ios::in);

if (!infile)

{

cout << "打开文件失败" << endl;

exit(2);

}

cout << "姓名 " << "联系方式 " << "取货地址 " << "送货地址 " << "货物重量 " << "包裹状态 " << endl;

for (int j = 0; j < s; j++)

{

infile >> c.name[j];

cout << c.name[j] << " ";

infile >> c.contact[j];

cout << c.contact[j] << " ";

infile >> o.getaddress[j];

cout << o.getaddress[j] << " ";

infile >> o.deliveraddress[j];

cout << o.deliveraddress[j] << endl;

infile >> o.weight[j];

cout << o.weight[j] << endl;

infile >> d.packagestate[j];

cout << d.packagestate[j] << endl;

}

infile.close();

deliverpersonmenu();

}

void adddeliverperson(Customer& c, Order& o, Deliver& d)

{

system("cls");

int n1;

int z;

string deliverperson;

ifstream infile("2.txt", ios::in);

if (!infile)

{

cerr << "打开失败" << endl;

exit(2);

}

cout << "姓名 " << "联系方式 " << "取货地址 " << "送货地址 " << "货物重量 " << "包裹状态 " << endl;

for (int j = 0; j < s; j++)

{

infile >> c.name[j];

cout << c.name[j] << " ";

infile >> c.contact[j];

cout << c.contact[j] << " ";

infile >> o.getaddress[j];

cout << o.getaddress[j] << " ";

infile >> o.deliveraddress[j];

cout << o.deliveraddress[j] << endl;

infile >> o.weight[j];

cout << o.weight[j] << endl;

infile >> d.packagestate[j];

cout << d.packagestate[j] << endl;

}

infile.close();

cout << "请输入要添加第几个" << endl;

cin >> n1;

for (int i = 0; i <= s; i++)

{

if (n1 - 1 == i)

{

cout << "请输入姓名" << endl;

cin >> deliverperson;

d.deliverperson[i] = deliverperson;

cout << "添加成功!" << endl;

}

}

}

void modifypackagestate(Customer& c, Order& o, Deliver& d)

{

system("cls");

int n1;

int z;

ifstream infile("2.txt", ios::in);

if (!infile)

{

cerr << "打开失败" << endl;

exit(2);

}

cout << "姓名 " << "联系方式 " << "取货地址 " << "送货地址 " << "货物重量 " << "包裹状态 " << endl;

for (int j = 0; j < s; j++)

{

infile >> c.name[j];

cout << c.name[j] << " ";

infile >> c.contact[j];

cout << c.contact[j] << " ";

infile >> o.getaddress[j];

cout << o.getaddress[j] << " ";

infile >> o.deliveraddress[j];

cout << o.deliveraddress[j] << endl;

infile >> o.weight[j];

cout << o.weight[j] << endl;

infile >> d.packagestate[j];

cout << d.packagestate[j] << endl;

}

infile.close();

cout << "请输入要更新第几个" << endl;

cin >> n1;

for (int i = 0; i <= s; i++)

{

if (n1 - 1 == i)

{

cout << "1.准备就绪" << endl;

cout << "2.拾起" << endl;

cout << "3.到达仓库" << endl;

cout << "4.外出交货" << endl;

cout << "5.交付" << endl;

cout << "请输入序号" << endl;

cin >> z;

switch (z)

{

case 1:

d.packagestate[i] = "准备就绪";

cout << "修改成功!" << endl;

break;

case 2:

d.packagestate[i] = "拾起";

cout << "修改成功!" << endl;

break;

case 3:

d.packagestate[i] = "到达仓库";

cout << "修改成功!" << endl;

break;

case 4:

d.packagestate[i] = "外出交货";

cout << "修改成功!" << endl;

break;

case 5:

d.packagestate[i] = "交付";

cout << "修改成功!" << endl;

break;

}

break;

}

}

}

void menu()

{

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "欢迎使用物流系统！" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

int n;

bool b = true;

system("pause");

system("cls");

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

cout << "\*\*\*\*\*\*\*登录\*\*\*\*\*\*\*" << endl;

cout << "1.顾客" << endl;

cout << "2.配送员" << endl;

cout << "3.退出程序" << endl;

cout << "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*" << endl;

do

{

cin >> n;

switch (n)

{

case 1: customermenu();

break;

case 2: deliverpersonmenu();

break;

case 3: b = false;

exit(0);

break;

default:cout << "请输入标准的功能序号" << endl;

deliverpersonmenu();

}

} while (b);

}

int main()

{

menu();

return 0;

}