**OBJECT ORIENTED PROGRAMMING**

**FINAL PROJECT**

**ABDULLAH BASHIR – 180437**

**MUHAMMAD OZAIR MALIK – 190798**

**MUHAMMAD UMAIR – 190814**

**OFFICE MANAGEMENT AND**

**SURVEILLANCE SYSTEM**

#include<iostream>

#include<string>

#include <Windows.h>

#include<conio.h>

#include<iomanip>

using namespace std;

#define max 20

class employee

{

public:

char name[30];

long int jic;

char jp[20];

int ye;

int age;

};

int x;

employee pqr[max],tempemp[max];

class admin

{

private:

int account\_balance;

int monthly\_expenses;

int orderrecived;

int orderdelevierd;

int profit;

public:

void setdata(int x,int y,int z,int w,int d){

account\_balance=x;

monthly\_expenses=y;

orderdelevierd=w;

orderrecived=z;

profit=d;

}

int getaccountbalance()

{

return account\_balance;

}

int getmonthlyexpense()

{

return monthly\_expenses;

}

int getorderdelevired()

{

return orderdelevierd;

}

int getorderrecived()

{

return orderrecived;

}

int getprofit()

{

return profit;

}

void display()

{ system("cls");

cout<<"\n\n\n\n\n\n\n\n"<<endl;

cout<<"\n\n\t\tCurrent account Balance of the company is "<<setw(7)<<account\_balance<<endl;

cout<<"\n\n\t\tCurrent Monthly expense of the company is "<<setw(7)<<monthly\_expenses<<endl;

cout<<"\n\n\t\tCurrent order recived of the company is "<<setw(7)<<orderrecived<<endl;

cout<<"\n\n\t\tCurrent order deleivered of the company is "<<setw(7)<<orderdelevierd<<endl;

cout<<"\n\n\t\tCurrent profit of the company is "<<setw(7)<<profit<<endl;

}

};

class store : public admin

{

private:

int storeitem;

int statics;

public:

store()

{

}

void setData(int n)

{

storeitem=n;

}

bool material()

{ cout<<endl;

cout<<endl;

int c;

cout<<" \n\n\t\tPress 1 If you are having greater than 25% of manufacturing material\n\n\t\tPress 2 if you having less than 25%of the manufacting material"<<endl;

cin>>c;

if(c==1)

{

cout<<"\n\n\t\tThat's Good .Continue your work"<<endl;

return true;

}

else if(c==2)

{

cout<<"\n\n\t\tYour report has been recived and Order team is on the way to take order from you"<<endl;

return false;

}

else

cout<<"\n\n\t\tMake sure you are entering correct option"<<endl<<"please pay attention "<<endl;

}

bool order()

{cout<<endl;

cout<<endl;

int a;

cout<<"\n\n\t\tPress 1 if There all the order are completed and each and everthing is okay";

cin>>a;

if(a==1)

{

cout<<"\n\n\t\tWe are sastified from your performances"<<endl;

}

if(a!=1)

{

cout<<"\n\n\t\tkindly list out which order are pending and problem you are facing and report it to admin"<<endl;

return false;

}

else

return true;

}

};

class transport : public admin

{

private:

string Driver\_name;

int id;

int vechile;

int inspection;

int dutyhours;

public:

transport()

{

};

void setdata(string n,int i,int l,int q,int t)

{

Driver\_name=n;

id=i;

vechile=l;

inspection=q;

dutyhours=t ;

}

string getOName()

{

return Driver\_name;

}

int getvechile()

{

return vechile;

}

int getinspection()

{ if(inspection>12)

{

cout<<"\n\n\t\tThe car are not more to be used for driving they need to be inspected"<<endl;

cout<<"\n\n\t\t Our inspection team is on the way"<<endl;

}

else

{return inspection;}

}

int getOID()

{

return id;

}

int getdutyhours()

{ if (dutyhours>12)

{cout<<"\n\n\t\tlife is precious.No more driiving \n\n\t\t shift reffered to second driver"<<endl;}

return dutyhours;

}

void display()

{ system("cls");

cout<<"\n\n\n\n";

cout<<"\n\n\t\tDriver name : :"<<getOName()<<endl;

cout<<"\n\n\t\tDriver ID : :"<<getOID()<<endl;

cout<<"\n\n\t\tTotal vechiles :"<<getvechile()<<endl;

cout<<"\n\n\t\tinspected months for vechile is :"<<getinspection()<<endl;

cout<<"\n\n\t\tDuty hours are :"<<getdutyhours()<<endl;

}

};

class Security

{

private:

int guards;

string name;

int id;

int duty\_hours;

int gate\_number;

protected:

string name2;

string vehicle;

string timein;

string IDcardNo;

public:

Security()

{

}

void guest\_input()

{

cout<<"Enter your name : ";

cin>>name2;

cout<<"Enter vehicle registration number : ";

cin>>vehicle;

cout<<"Enter your ID card number : ";

cin>>IDcardNo;

cout<<"Enter going in time : ";

cin>>timein;

}

void setguarddata(int g,string n, int i, int d, int gn)

{

guards=g;

name=n;

id=i;

duty\_hours=d;

gate\_number=gn;

}

int getguards()

{

return guards;

}

string getName()

{

return name;

}

int getID()

{

return id;

}

int getHours()

{

return duty\_hours;

}

int getGateNumber()

{

return gate\_number;

}

void display()

{

cout<<"Gurad on Duty Are : "<<getguards()<<endl;

cout<<"Name : "<<getName()<<endl;

cout<<"ID Number : "<<getID()<<endl;

cout<<"Duty Hours : "<<getHours()<<endl;

cout<<"Currently on gate "<<getGateNumber()<<endl;

}

void guestslip()

{

cout<<"Name : "<<name2<<endl;

cout<<"Vehicle number : "<<vehicle<<endl;

cout<<"ID Card Number : "<<IDcardNo<<endl;

cout<<"Entering Time : "<<timein<<endl;

cout<<"Allowed by Guard : "<<getName()<<endl;

}

};

class CCTV : public Security

{

private:

int Number\_of\_Cameras;

public:

CCTV()

{

}

void setData(int n)

{

Number\_of\_Cameras=n;

}

bool Cameras()

{

int c;

cout<<" Press 1 if all Cameras are working\n Press 2 if any of the Cameras is not working\n";

cin>>c;

if(c==1)

{

cout<<"OKAY Good... Now keep a close eye and report in case of any unusual activity\n";

return true;

}

else if(c==2)

{

cout<<"Report has been sent to admin. Wait for further orders\n";

return false;

}

else

cout<<"You are not entering the appropriate number. Kindly recheck\n";

}

bool activity()

{

int a;

cout<<"Press 1 if there is any unusual movement\n";

cin>>a;

if(a==1)

{

cout<<"Report sent to admin. Keep close eye until further orders\n";

return false;

}

else

return true;

}

};

class Patrol : public Security

{

private:

string officer\_name;

int id;

int guard;

public:

Patrol()

{

}

void setdata(string n,int i,int g)

{

officer\_name=n;

id=i;

guard=g;

}

string getOName()

{

return officer\_name;

}

int getOID()

{

return id;

}

int getguards()

{

return guard;

}

void display()

{

cout<<"Patrolling officer name : "<<getOName()<<endl;

cout<<"Officer ID : "<<getOID()<<endl;

cout<<"Guards on Duty : "<<getguards()+4;

cout<<"Guards on patrol : "<<getguards()<<endl;

}

};

void adminportal(admin a,transport t,store s)

{

int j;

cout<<"\n\n\n\n\n\n\n\n\n\n\n \t\t\t|============ WELCOME TO Admin Section ============|";

\_getch();

do

{

system("cls");

cout<<"\n\n\t\t 1. TO VIEW STATISTICS\n\n\t\t 2. STORE MENU\n\n\t\t 3. TRANSPORT MENU";

cout<<"\n\n\t\t 4. To Return To Main Menu"<<endl;

cout<<"\n\n=============================================================\n";

cout<<"\n\nEnter Your Choice:";

cin>>j;

system("cls");

switch(j){

case 1:

cout<<"Welcome To view statics"<<endl;

cout<<"PLEASE WAIT WHILE WE ARE GETTING RECORD READY FOR YOU"<<endl;

cout<<endl;

\_getch();

system("cls");

a.display();

\_getch();

break;

case 2:

cout<<"Welcome To Store menu "<<endl;

cout<<"PLEASE WAIT WHILE WE ARE GETTING Things READY FOR YOU"<<endl;

s.setData(34);

\_getch();

system("cls");

s.material();

\_getch();

system("cls");

s.order();

\_getch();

break;

case 3:

cout<<"Welcome To Transport menu"<<endl;

cout<<"PLEASE WAIT WHILE WE ARE GETTING Things READY FOR YOU"<<endl;

t.setdata("Ramzan",1234,6,3,14);

\_getch();

system("cls");

t.display();

\_getch();

break;

}

}

while(j!=4);

}

void menu()

{

cout<<"\n\n\n\n\n\n\n\n\n\n\n \t\t\t|============ WELCOME TO Employee Record Management Section ============|";

\_getch();

system("cls");

cout<<"\n\n\t\t 1. TO Create all Employee Data\n\n\t\t 2. To See Employee Data \n\n\t\t 3. To Add New Employee Data\n\n\t\t";

cout<<" 4. TO Delete Employee Data\n\n\t\t 5. To Update Employee Data\n\n\t\t 6. To Search data of Employee\n\n\t\t";

cout<<" 7. To Quit the Employee Management system\n\n\t\t "<<endl;

}

void newentry()

{

system("cls");

cout<<" \*\*\*WELCOME TO EMPLOYEE ENTRY TABLE\*\*\*\* ";

cout<<endl;

cout<<endl;

cout<<"The Maxium Number of Employee in the Company are 15"<<endl;

cout<<endl;

cout<<"Enter the Number of the Employee you wanted to add"<<endl;

cin>>x;

cout<<endl;

cout<<"Enter The Below Information of the Employees"<<endl;

for(int i=0;i<=x-1;i++)

{

cout<<"Enter the data for"<<i+1<<"Employee"<<endl;

cout<<endl<<endl;

cout<<"Name :";

cin.getline(pqr[i].name,30);

cin.getline(pqr[i].name,30);

cout<<"Job Identity Code : ";

cin>>pqr[i].jic;

cout<<"Job Post :";

cin>>pqr[i].jp;

cout<<"Years of Experience : ";

cin>>pqr[i].ye;

cout<<"Age : ";

cin>>pqr[i].age;

cout<<endl<<endl<<endl;

}

cout<<"going to main menu";

Sleep(500);

}

void list()

{

system("cls");

cout<<" \*\*\*Employee Record Table\*\*"<<endl;

cout<<" Name Job Identity Code Job Post Years(EXP) Age "<<endl;

cout<<" ---------------------------------------------------------------- "<<endl;

for(int i=0;i<=x-1;i++)

{

cout<<setw(15)<<pqr[i].name;

cout<<"\t\t";

cout<<setw(4)<<pqr[i].jic;

cout<<"\t\t ";

cout<<setw(6)<<pqr[i].jp;

cout<<"\t\t";

cout<<setw(4)<<pqr[i].ye;

cout<<"\t ";

cout<<setw(2)<<pqr[i].age;

cout<<"\t";

cout<<endl;

}

cout<<"going to main menu";

\_getch();

}

void insert()

{

system("cls");

int i=x;

x+=1;

cout<<"Insert New Record";

cout<<endl;

cout<<endl;

cout<<"You Are Entering Data For "<<x<<"Employee "<<endl;

cout<<"Name :";

cin>>pqr[i].name;

cout<<"Job Identity Code : ";

cin>>pqr[i].jic;

cout<<"Job Post :";

cin>>pqr[i].jp;

cout<<"Years of Experience : ";

cin>>pqr[i].ye;

cout<<"Age : ";

cin>>pqr[i].age;

cout<<endl<<endl;

cout<<"going to main menu";

Sleep(500);

}

void deletes()

{

system("cls");

int jic;

int check=0;

cout<<"You Want Delete An Employee Record"<<endl;

cout<<endl;

cout<<"Enter An JobCode To Delete That Entry ";

cin>>jic;

int i;

for(i=0;i<=x-1;i++)

{

if(pqr[i].jic==jic)

{

check=i;

}

}

for(i=0;i<=x-1;i++)

{

if(i==check)

{

continue;

}

else

{

if(i>check)

{

tempemp[i-1]=pqr[i];

}

else

{

tempemp[i]=pqr[i];

}

}

}

x--;

for(i=0;i<=x-1;i++)

{

pqr[i]=tempemp[i];

}

cout<<"Record Deleted sucessfully"<<endl;

\_getch();

system("cls");

}

void edit()

{

system("cls");

int jobcode;

cout<<"You Want to Update an Employee Entry Record ";

cout<<endl<<endl;

int i;

void editmenu();

void editname(int);

void editcode(int);

void editjp(int);

void editexp(int);

void editage(int);

char option;

cout<<"Enter An jobcode To Edit An Entry---- ";

cin>>jobcode;

editmenu();

for(i=0;i<=x-1;i++)

{

if(pqr[i].jic==jobcode)

{

while((option=cin.get())!='q')

{

switch(option)

{

case 'N':

editname(i);

break;

case 'C':

editcode(i);

break;

case 'J':

editjp(i);

break;

case 'E':

editexp(i);

break;

case 'A':

editage(i);

break;

}

editmenu();

}

}

}

cout<<"Record Updated SucessFully"<<endl;

\_getch();

system("cls");

}

void editmenu()

{

system("cls");

cout<<" N=> To Edit Name "<<endl;

cout<<" C=> To Edit Job Identity Code "<<endl;

cout<<" J=> To Edit Job Post"<<endl;

cout<<" E=> To Edit Experience "<<endl;

cout<<" A=> To Edit Age "<<endl;

cout<<" Q=> To QUIT "<<endl;

cout<<" Select What You Want to Update and Enter Correct Option : "<<endl;

}

void editname(int i)

{

cout<<"Enter New Name : "<<endl;

cin>>pqr[i].name;

}

void editcode(int i)

{

cout<<"Enter New Job Job Identity Code- :"<<endl;

cin>>pqr[i].jic;

}

void editjp(int i)

{

cout<<"enter new Job Post : "<<endl;

cin>>pqr[i].jp;

}

void editexp(int i)

{

cout<<"Enter new Years of Experience : "<<endl;

cin>>pqr[i].ye;

}

void editage(int i)

{

cout<<"Enter new Age : "<<endl;

cin>>pqr[i].age;

}

void search()

{

system("cls");

cout<<"Welcome To Search Of Employee ";

cout<<endl<<endl;

int jobcode;

cout<<"You Can Search Only By Jobcode Of An Employee"<<endl<<endl;

cout<<"Enter Job Identity Code Of An Employee"<<endl;

cin>>jobcode;

for(int i=0;i<=x-1;i++)

{

if(pqr[i].jic==jobcode)

{

cout<<" \*\*\*Employee Record Table\*\*"<<endl;

cout<<" Name Job Identity Code Job Post Years(EXP) Age "<<endl;

cout<<" ---------------------------------------------------------------- "<<endl;

cout<<pqr[i].name;

cout<<"\t\t";

cout<<setw(4)<<pqr[i].jic;

cout<<"\t\t ";

cout<<setw(6)<<pqr[i].jp;

cout<<"\t\t";

cout<<setw(4)<<pqr[i].ye;

cout<<"\t ";

cout<<setw(2)<<pqr[i].age;

cout<<"\t";

cout<<endl;

}

}

cout<<"going to main menu";

\_getch();

}

void portalemployee()

{

int option;

do

{

menu();

cin>>option;

switch(option)

{

case 1:

newentry();

break;

case 2:

list();

break;

case 3:

insert();

break;

case 4:

deletes();

break;

case 5:

edit();

break;

case 6:

search();

break;

}

system("cls");

}

while (option!=7);

}

void adminportal()

{

int option;

do

{

menu();

cin>>option;

switch(option)

{

case 1:

newentry();

break;

case 2:

list();

break;

case 3:

insert();

break;

case 4:

deletes();

break;

case 5:

edit();

break;

case 6:

search();

break;

}

system("cls");

}

while (option!=7);

}

void securityportal(Security sc,CCTV cv,Patrol pt)

{

int j;

cout<<"\n\n\n\n\n\n\n\n\n\n\n \t\t\t|============ WELCOME TO Security Section ============|";

\_getch();

do

{

system("cls");

cout<<"\n\n\t\t 1. TO VIEW Guards Status\n\n\t\t 2. To Go to CCtv Section\n\n\t\t 3.To Patrol section";

cout<<"\n\n\t\t 4. To Return To Main Menu"<<endl;

cout<<"\n\n=============================================================\n";

cout<<"\n\nEnter Your Choice:";

cin>>j;

system("cls");

switch(j)

{

case 1:

cout<<"Welcome To Guards section"<<endl;

cout<<"PLEASE WAIT WHILE WE ARE GETTING RECORD READY FOR YOU"<<endl;

cout<<endl;

\_getch();

system("cls");

sc.setguarddata(1,"Asfand Khan",2345,12,2);

sc.display();

\_getch();

system("cls");

cout<<"Welcome To guest Entry Setup and slip generation Section"<<endl;

sc.guest\_input();

\_getch();

system("cls");

sc.guestslip();

\_getch();

system("cls");

break;

case 2:

cout<<"Welcome To CCTV menu "<<endl;

cout<<"PLEASE WAIT WHILE WE ARE GETTING Things READY FOR YOU"<<endl;

\_getch();

system("cls");

cv.setData(9);

cv.Cameras();

\_getch();

system("cls");

cv.activity();

\_getch();

break;

case 3:

cout<<"Welcome To Security Patrol Section"<<endl;

cout<<"PLEASE WAIT WHILE WE ARE GETTING Things READY FOR YOU"<<endl;

\_getch();

system("cls");

pt.setdata("Ali Ahmed",23456,4);

pt.display();

\_getch();

break;

}

}

while(j!=4);

}

int main()

{

Security sc;

admin a;

transport t;

store s;

CCTV cv;

Patrol pt;

int j;

cout<<"\n\n\n\n\n\n\n\n\n\n\n \t\t\t|============ WELCOME TO OFFICE MANAGEMENT ============|";

\_getch();

system("cls");

do

{

system("cls");

cout<<"\n\t\t Welcome To Main Menu \n";

cout<<"=============================================================";

cout<<"\n\n\t\t 1. TO ADMIN SECTION\n\n\t\t 2. SECURITY SECTION\n\n\t\t 3. EMPLOYEE RECORD SECTION";

cout<<"\n\n\t\t 4. To Exit The Program"<<endl;

cout<<"\n\n=============================================================\n";

cout<<"\n\nEnter Your Choice:";

cin>>j;

system("cls");

if (j==3)

{

string x ="";

char p;

cout << "Enter Secret login Password :\n";

p= \_getch();

while(p != 13)

{

x.push\_back(p);

cout << '\*';

p = \_getch();

}

if(x == "Employee1234")

{

cout<<endl;

cout<<"\n\n\n\n\n";

cout << "\*\*\*\*Access granted\*\*\*\*\n";

\_getch();

system("cls");

portalemployee();

}

else

{

cout << "\*\*\*\*\*Access aborted\*\*\*\*\*\n";

cout<<"---------Authorized Person only-----------\n";

}

}

if(j==1)

{a.setdata(1200000,12304,98,23,200000);

{

string x ="";

char p;

cout << "Enter Secret login Password :\n";

p= \_getch();

while(p != 13)

{

x.push\_back(p);

cout << '\*';

p = \_getch();

}

if(x == "Admin1234")

{

cout<<endl;

cout<<"\n\n\n\n\n";

cout << "\*\*\*\*Access granted\*\*\*\*\n";

\_getch();

system("cls");

adminportal(a,t,s);

}

else

{

cout << "\*\*\*\*\*Access aborted\*\*\*\*\*\n";

cout<<"---------Authorized Person only-----------\n";

}

}

}

if(j==2)

{

string x ="";

char p;

cout << "Enter Secret login Password :\n";

p= \_getch();

while(p != 13)

{

x.push\_back(p);

cout << '\*';

p = \_getch();

}

if(x == "Security1234")

{

cout<<endl;

cout<<"\n\n\n\n\n";

cout << "\*\*\*\*Access granted\*\*\*\*\n";

\_getch();

system("cls");

sc.setguarddata(4,"Asfand Khan",2345,12,2);

securityportal(sc,cv,pt);

}

else

{

cout << "\*\*\*\*\*Access aborted\*\*\*\*\*\n";

cout<<"---------Authorized Person only-----------\n";

}

}

}

while(j!=4);

system("pause");

return 0;

}









