**NAME : MANGESH A. GHADWAJE**

**ROLL NO:24**

**BATCH : B2**

**COURSE: OOPs PRACTICAL**

**Assginment No. 4**

**Problem Statment:** Design a base class with name, date of birth, blood group and another base class consisting of the data members such as height and weight. Design one more base class consisting of the insurance policy number and contact address. The derived class contains the data members’ telephone numbers and driving license number. Write a menu driven program to carry out the following things: i. Build a master table ii. Display iii. Insert a new entry iv. Delete entry v. Edit vi. Search for a record.

#include<iostream>

#include<algorithm>

using namespace std;

const int MAX\_RECORDS=10;

class PersonalDetails

{

public:

string name;

string dob;

string blood;

PersonalDetails(const string& n,const string& d,const string& b):

name(n),dob(d),blood(b){}

};

class Physical

{

public:

float height;

float weight;

Physical(float h,float w):

height(h),weight(w){}

};

class Contact

{

public:

string policy;

string address;

Contact(const string& p,const string& add):

policy(p),address(add){}

};

class PersonalRecord: public PersonalDetails,public Physical,public Contact

{

public:

string telephone;

string licence;

PersonalRecord():

PersonalDetails("","",""),

Physical(0.0,0.0),

Contact("",""),

telephone(""),licence(""){}

PersonalRecord(const string& n,const string& d,const string& b,

float h,float w,

const string& p,const string& add,

const string& tel,const string& lic):

PersonalDetails(n,d,b),Physical(h,w),Contact(p,add),telephone(tel),licence(lic){}

};

PersonalRecord records[MAX\_RECORDS];

int currentRecords=0;

void Menu()

{

if(currentRecords < MAX\_RECORDS)

{

string name,dob,blood,policy,address,telephone,licence;

double height,weight;

cout<<"Enter the Name of Person:";

cin>>name;

cout<<"Enter the date of birth Of Person:\n";

cin>>dob;

cout<<"Enter the Blood Group Of Person:\n";

cin>>blood;

cout<<"Enter the Height Of The Person:\n";

cin>>height;

cout<<"Enter the weight of The Person:\n";

cin>>weight;

cout<<"Enter Insurance policy number Of The Person:\n";

cin>>policy;

cout<<"Enter contact address:";

cin>>address;

cout<<"Enter telephone number:";

cin>>telephone;

cout<<"Enter the Driving Licence Number:";

cin>>licence;

records[currentRecords]=PersonalRecord(name,dob,blood,height,weight,policy,address,telephone,licence);

++currentRecords;

}

else

{

cout<<"Maximum number Rechord reached.\n";

}

}

void DisplayRecord()

{

cout<<"\nDisplaying \n";

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

{

cout<<"Name:"<<records[i].name<<endl;

cout<<"DOB:"<<records[i].dob<<endl;

cout<<"Blood Group:"<<records[i].blood<<endl;

cout<<"Height:"<<records[i].height<<endl;

cout<<"Weight:"<<records[i].weight<<endl;

cout<<"Policy Number:"<<records[i].policy<<endl;

cout<<"Contact Address:"<<records[i].address<<endl;

cout<<"Telephone:"<<records[i].telephone<<endl;

cout<<"Driving Licence:"<<records[i].licence<<endl<<"\n";

}

}

void insertNewEntry()

{

Menu();

}

void deleteEntry()

{

string searchName;

cout<<"Enter the name to Delete:";

cin>>searchName;

auto it=find\_if(records,records+currentRecords,[searchName](const PersonalRecord& record)

{

return record.name==searchName;

});

if(it!=records+currentRecords)

{

copy(it+1,records + currentRecords ,it);

--currentRecords;

cout<<"Record Deleted Successfully.\n";

}

else

{

cout<<"Record Not Found.\n";

}

}

void editRecord()

{

string searchName;

cout<<"Enter the name to Edit:";

cin>>searchName;

auto it=find\_if(records, records+ currentRecords, [searchName](const PersonalRecord& record)

{

return record.name==searchName;

});

if(it!=records + currentRecords)

{

cout<<"Enter the new name:";

cin>>it->name;

cout<<"Enter the new DOB:";

cin>>it->dob;

cout<<"Enter the Blood Group:";

cin>>it->blood;

cout<<"Enter new Height:";

cin>>it->height;

cout<<"Enter new Weight:";

cin>>it->weight;

cout<<"Enter new Policy Number:";

cin>>it->policy;

cout<<"Enter the New Address:";

cin>>it->address;

cout<<"Enter new Telephone:";

cin>>it->telephone;

cout<<"Enter new Licence Number:";

cin>>it->licence;

cout<<"Record Edited Successfully:";

}

else

{

cout<<"Record Not found...";

}

}

void searchRecord()

{

string searchName;

cout<<"Enter name to serach:";

cin>>searchName;

auto it=find\_if(records,records + currentRecords,[searchName](const PersonalRecord& record)

{

return record.name==searchName;

});

if(it!=records+currentRecords)

{

cout<<"Record Found:\n";

cout<<"Name:"<<it->name<<endl;

cout<<"Date of Birth:"<<it->dob<<endl;

cout<<"Blood Group:"<<it->blood<<endl;

cout<<"Height:"<<it->height<<endl;

cout<<"Weight:"<<it->weight<<endl;

cout<<"Policy Number:"<<it->policy<<endl;

cout<<"Contact Address:"<<it->address<<endl;

cout<<"Telephone:"<<it->telephone<<endl;

cout<<"Driving Licence Number:"<<it->licence<<endl;

}

}

int main()

{

int choice;

do{

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

cout<<"\n1.Creat Table";

cout<<"\n2.Display Data";

cout<<"\n3.Insert New Data";

cout<<"\n4.Delete Data";

cout<<"\n5.Edit Data";

cout<<"\n6.Serach Data";

cout<<"\n7.Exit ";

cout<<"\nEnter your choice:";

cin>>choice;

switch(choice)

{

case 1:

Menu();

break;

case 2:

DisplayRecord();

break;

case 3:

insertNewEntry();

break;

case 4:

deleteEntry();

break;

case 5:

editRecord();

break;

case 6:

searchRecord();

break;

case 7:

cout<<"Exit Program.\n";

break;

default:

cout<<" Entered Choice Is Invalid chioce....\n";

}

}while(choice!=7);

return 0;

}