Main class

package Project;

import java.util.ArrayList;

import java.util.Dictionary;

import java.util.Hashtable;

import java.time.LocalDate;

import java.time.chrono.ChronoLocalDate;

import java.time.format.DateTimeFormatter;

import java.util.List;

import java.util.Scanner;

import java.lang.\*;

import java.util.StringTokenizer;

interface LibraryManagmentSystem

{

String Library\_name="Amrita Library";

static int number\_of\_books=1;

int number\_of\_shelves=2000;

}

abstract class person

{

String name;

String phone;

int age;

char Gender;

int ID;

String Mail\_id;

String password;

public int bookTakenID;

String address1;

abstract void assignID(int id);

abstract boolean validate(int id,String name);

abstract int searchperson(int id9);

abstract void setanyfine();

abstract boolean returnFine();

abstract void revanyfine();

abstract void getDetails(int id1);

abstract void setbookTakenId(int id44);

}

class customer extends person

{

int RequestID;

boolean anyfine;

public int bookTakenID;

int fine;

customer()

{

Scanner s1 = new Scanner(System.in);

while(true)

{

try

{

System.out.println("enter the name:");

this.name=s1.next();

break;

}

catch(Exception e)

{

System.out.println(e);

}

}

while(true)

{

System.out.println("enter the phone:");

phone=s1.next();

try

{

int a=phone.length();

if(a==10)

{

break;

}

else

{

throw new IllegalAccessException(" Not valid Phoneno");

}

}

catch(IllegalAccessException e)

{

System.out.println(e);

}

}

System.out.println("enter the age:");

this.age=s1.nextInt();

System.out.println("enter the gender");

this.Gender=s1.next().charAt(0);

}

boolean returnFine()

{

return anyfine;

}

void setanyfine()

{

anyfine=true;

}

void revanyfine()

{

anyfine=false;

}

void assignID(int id)

{

this.ID=id;

System.out.println("the id assigned to you is:"+ID);

}

boolean validate(int id1,String name1)

{

if(this.ID==id1 && name.equals(name1))

{

return true;

}

else

{

return false;

}

}

int searchperson(int id9)

{

if(ID==id9)

{

return 1;

}

return 0;

}

void setbookTakenId(int id44)

{

bookTakenID=id44;

}

void getDetails(int id1) {

// TODO Auto-generated method stub

if(ID==id1)

{

System.out.println("name: "+this.name);

System.out.println("phone: "+this.phone);

System.out.println("age: "+this.age);

System.out.println("gender: "+this.Gender);

System.out.println("book id which he issued:"+this.bookTakenID);

}

}

}

class book implements LibraryManagmentSystem

{

int ID;

String aurthor;

public boolean isReserved=false;

String name;

public boolean isIssued=false;

LocalDate issueDate;

LocalDate returnDate;

LocalDate schedule\_returndate;

int price;

String BookType;

int fine;

public void funscheduledate(int n)

{

String tempdate1=issueDate.toString();

StringTokenizer st = new StringTokenizer(tempdate1,"-");

int dd1[]=new int[3];

int i=2;

while(st.hasMoreTokens())

{

dd1[i] = Integer.parseInt(st.nextToken());

i--;

}

if(dd1[0]+n>31)

{

dd1[1]=dd1[1]+1;

if(dd1[1]>=12)

{

dd1[1]=01;

dd1[2]=dd1[2]+1;

dd1[0]=dd1[0]+n-31;

}

}

else

{

dd1[0]+=n;

}

DateTimeFormatter formatter = DateTimeFormatter.ofPattern("yyyy-MM-dd");

String s;

if(dd1[1]<10 && dd1[0]<10)

{

s=dd1[2]+"-0"+dd1[1]+"-0"+dd1[0];

}

else if(dd1[1]<10 && dd1[1]>10)

{

s=dd1[2]+"-0"+dd1[1]+"-"+dd1[0];

}

else if(dd1[1]>10 && dd1[0]<10)

{

s=dd1[2]+"-"+dd1[1]+"-0"+dd1[0];

}

else

{

s=dd1[2]+"-"+dd1[1]+"-"+dd1[0];

}

System.out.println(s);

//convert String to LocalDate

schedule\_returndate = LocalDate.parse(s, formatter);

}

public void reservebook\_code(LocalDate resd1,LocalDate returd1,int resbookid,List <Integer> list1,List <Integer> list2,Dictionary dict,book[] b1,int id,Dictionary dict1,List <Integer> list3)

{

Scanner s1111 =new Scanner(System.in);

String tempdate1=resd1.toString();

String tempdate2=returd1.toString();

StringTokenizer st = new StringTokenizer(tempdate1,"-");

StringTokenizer st1 = new StringTokenizer(tempdate2,"-");

int d1[]=new int[3];

int d2[]=new int[3];

int i=2;

while(st.hasMoreTokens())

{

d1[i] = Integer.parseInt(st.nextToken());

i--;

}

i=2;

while(st1.hasMoreTokens())

{

d2[i] = Integer.parseInt(st1.nextToken());

i--;

}

if(tempdate1.equals(tempdate2))

{

System.out.println("You Can Reserve the book");

int frbc;

System.out.println("Enter 1 if u surely want to reserve the book once u reserve u cant cancel");

frbc=s1111.nextInt();

if(frbc==1)

{

System.out.println("Your book is sucessfully reserved");

// list2.add(resbookid);

int sri=0;

for(int z=0;z<book.number\_of\_books;z++)

{

sri=b1[z].search\_id(resbookid);

}

dict.put(sri, resd1);

dict1.put(sri,id);

list3.add(resbookid);

System.out.println(list3);

}

else

{

System.out.println("looks like u dont want to reserve the book! but the book is availbale");

}

}

else if(d2[0]-d1[0]>0 && d1[1]==d2[1] && d1[2]==d2[2])

{

int rn = d2[0]-d1[0];

System.out.println("looks like the book is not available you need to wait for" +rn+"days");

}

else if (d2[1]-d1[1]>=1 && d1[2]==d2[2])

{

int n = Math.abs(d2[1]-d1[1]);

int rn = 31+d2[0]-d1[0]+n\*31;

System.out.println("looks like the book is not available you need to wait for" +rn+"days");

}

else if(d2[2]>d1[2] )

{

int rn = d2[0] + 31-d1[0];

System.out.println("looks like the book is not available you need to wait for" +rn+"days");

}

else

{

System.out.println("The book you want is available here");

System.out.println("Enter 1 if u surely want to reserve the book once u reserve u cant cancel");

int frbc=s1111.nextInt();

if(frbc==1)

{

System.out.println("Your book is sucessfully reserved but there may be a case where customers who issued didnt return a book");

// list2.add(resbookid);

System.out.println(list3);

int sri=0;

for(int z=0;z<book.number\_of\_books;z++)

{

sri=b1[z].search\_id(resbookid);

}

dict.put(sri, resd1);

dict1.put(sri,id);

list3.add(sri);

System.out.println(list3);

}

else

{

System.out.println("looks like u dont want to reserve the book! but the book is availbale");

}

}

}

LocalDate get\_schedule\_date()

{

return schedule\_returndate;

}

int getid()

{

return ID;//

}

book()

{

Scanner io=new Scanner(System.in);

System.out.println("here comes the book details");

System.out.println("enter the ID:");

ID = io.nextInt();

System.out.println("enter the aurthor:");

aurthor = io.next();

System.out.println("enter the book name:");

name = io.next();

System.out.println("enter the price:");

price=io.nextInt();

System.out.println("enter the booktype:");

BookType=io.next();

}

int computefine(LocalDate datem,LocalDate daten,int n)

{

String tempdate1=datem.toString();

String tempdate2=daten.toString();

StringTokenizer st = new StringTokenizer(tempdate1,"-");

StringTokenizer st1 = new StringTokenizer(tempdate2,"-");

int d1[]=new int[3];

int d2[]=new int[3];

int i=2;

while(st.hasMoreTokens())

{

d1[i] = Integer.parseInt(st.nextToken());

i--;

}

i=2;

while(st1.hasMoreTokens())

{

d2[i] = Integer.parseInt(st1.nextToken());

i--;

}

if(d1[1]==1 || d1[1]==3 || d1[1]==5 || d1[1]==7 || d1[1]==8 || d1[1]==10 || d1[1]==12)

{

if((d1[1]==d2[1]))

{

if(d2[0]-d1[0]>n)

{

fine = fine + ((d2[0]-d1[0])-n)\*10;

}

}

else if((d1[1]!=d2[1]))

{

if(d2[1]-d1[1]==1)

{

int days=((31-d1[0])+d2[0]);

fine=(days-n)\*10;

}

if(d2[1]-d1[1]>1 && d1[1]!=1)

{

int days=((31-d1[0])+d2[0])+(d2[1]-d1[1]-1)\*30;

fine=(days-n)\*10;

}

else if(d1[1]==1 && d2[1]-d1[1]>1)

{

int days=((31-d1[0])+d2[0])+((d2[1]-d1[1]-2)\*31 + 28);

fine=(days-n)\*10;

}

}

}

if(d1[1]==4 || d1[1]==6 || d1[1]==9 || d1[1]==11)

{

if(d1[1]==d2[1])

{

if(d2[0]-d1[0]>n)

{

fine= fine + ((d2[0]-d1[0])-n)\*10;

}

}

else if((d1[1]!=d2[1]))

{

if(d2[1]-d1[1]==1)

{

int days=((30-d1[0])+d2[0]);

fine=(days-n)\*10;

}

if(d2[1]-d1[1]>1)

{

int days=((30-d1[0])+d2[0])+(d2[1]-d1[1]-1)\*30;

fine=(days-n)\*10;

}

}

}

//

//end of compute fine;

System.out.println(fine);

return fine;

}

int search(int name1)

{

if(ID==name1 && this.isIssued==false)

{

return 1;

}

return 0;

}

int search\_id(int id)

{

if(ID==id)

{

return 1;

}

return 0;

}

LocalDate getIssueDate()

{

int flag=1;

while(flag==1)

{

Scanner io=new Scanner(System.in);

try

{

String date=io.next();

issueDate=LocalDate.parse(date);

flag=0;

}

catch(Exception e)

{

System.out.println("please enter a correct date");

}

}

return issueDate;

}

LocalDate getIssueDate1()

{

return issueDate;

}

LocalDate getreturnDate()

{

Scanner io=new Scanner(System.in);

String date=io.next();

returnDate = LocalDate.parse(date);

return returnDate;

}

void search\_book(String searchbook)

{

if(name.equals(searchbook))

{

System.out.println("Name of the author :" +aurthor);

System.out.println("NAME OF THE BOOK:" +name);

System.out.println("book price RS:" +price);

System.out.println("book is issued:" +isIssued);

}

}

void search\_type(String searchtype)

{

if(BookType.equalsIgnoreCase(searchtype))

{

System.out.println("Name of the author :" +aurthor);

System.out.println("NAME OF THE BOOK:" +name);

System.out.println("book price RS:" +price);

System.out.println("book is issued:" +isIssued);

}

}

static void payfine(person [] p1,List <Integer> list1,List <Integer> list2,book [] b1,int n,int nc,List <Integer> list3,LocalDate date,Dictionary dict,Dictionary dict1)

{

outerloop1:

while(true)

{

System.out.println(date);

System.out.println("please enter the customer id who wants to return");

Scanner s10=new Scanner(System.in);

int id5=s10.nextInt();

if(list1.contains(id5))

{

Scanner s11=new Scanner(System.in);

System.out.println("please enter the book id customer wants to return");

int val=list1.indexOf(id5);

int id6=s10.nextInt();

if(list2.contains(id6))

{

int val1=list2.indexOf(id6);

if(val==val1)

{

for(int i=0;i<book.number\_of\_books;i++)

{

if(b1[i].search\_id(id6)==1)

{

System.out.println("enter the return date");

LocalDate returndate=b1[i].getreturnDate();

LocalDate idate=b1[i].getIssueDate1();

if(date!=null)

{

if(returndate.compareTo(date)>0 && date!=null)

{

System.out.println("book reservation is cancelled");

dict.remove(b1[i].ID);

dict1.remove(b1[i].ID);

int index= list3.indexOf(b1[i].ID);

list3.remove(index);

}

}

int fine1=b1[i].computefine(idate,returndate,n);

if(fine1==0)

{

b1[i].isIssued=false;

list1.remove(val);

list2.remove(val1);

System.out.println("your book is succesfully returned");

break outerloop1;

}

else if(fine1!=0)

{

System.out.println("please enter 1 to pay your fine");

int choice8=s10.nextInt();

if(choice8==1)

{

b1[i].isIssued=false;

list1.remove(val);

list2.remove(val1);

System.out.println("your book is succesfully returned");

p1[i].bookTakenID=0;

break outerloop1;

}

else

{

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

{

if(p1[j].searchperson(id5)==1);

{

p1[j].setanyfine();

System.out.println("you cannot issue a book until u pay the fine");

break outerloop1;

}

}

}

}

///

}

else

{

System.out.println("please give the crct details");

}

}

///for loop end

}

else

{

System.out.println("please give the crct details");

}

//val1==val2 if end

}

else

{

System.out.println("please give the crct details");

}

//list2.contains(id6) end

}

else

{

System.out.println("you did not take a book i guess");

}

//list2.id5 end

}

}

static void payfine1(int trump,List <Integer> list1,List <Integer> list2,person [] p1,book [] b1,int nc)

{

int val1=list1.indexOf(trump);

int val2=val1;

int bid=list2.get(val2);

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

{

if(p1[i].searchperson(trump)==1)

{

p1[i].revanyfine();

list1.remove(val1);

p1[i].bookTakenID=0;

}

}

for(int i=0;i<book.number\_of\_books;i++)

{

if(b1[i].search\_id(bid)==1)

{

b1[i].isIssued=false;

list2.remove(val1);

p1[i].bookTakenID=0;

}

}

System.out.println("book succesfully returned");

}

}

//class reservation extends book

//{

// String date;

//}

public class MainClass {

public static void main(String[] args) {

// TODO Auto-generated method stub

int n=0;

LocalDate resdate = null;

Dictionary dict = new Hashtable();

Dictionary dict1 = new Hashtable();

List<Integer> list1=new ArrayList<Integer>();

List<Integer> list2=new ArrayList<Integer>();

List<Integer> list3=new ArrayList<Integer>();

Scanner s1= new Scanner(System.in);

System.out.println("Enter the number of customers");

int nc=s1.nextInt();

//output message

person p1[]=new customer[nc];

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

{

p1[i]=new customer();

double rand=(Math.random())\*10000;

int imp=(int) (rand);

p1[i].assignID(imp);

//SOP

}

//SOP

System.out.println("Enter the number of staff");

int ns=s1.nextInt();

staff.get\_num\_of\_staff(ns);

staff p2[]=new staff[ns];

Thread t1[]=new Thread[ns];

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

{

p2[i]= new staff();

staff.address i1=p2[i].new address();

double rand=(Math.random())\*10000;

int imp=(int) (rand);

p2[i].assignID(imp);

t1[i]=new Thread(p2[i]);

t1[i].start();

}

int m;

//book

//SOP1

book b1[]=new book[book.number\_of\_books];

for(int i=0;i<book.number\_of\_books;i++)

{

b1[i]= new book();

//may change if we use up casting with reservation class

}

outerloopqo:

while(true && ns!=0)

{

int flag=1;

System.out.println("enter ur id and password to login");

Scanner s3000=new Scanner(System.in);

int id1=s3000.nextInt();

String password=s3000.next();

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

{

if(p2[i].validate(id1, password))

{

flag=0;

break outerloopqo;

}

}

if(flag==1)

{

System.out.println("please enter crct credentials");

}

}

while(true)

{

System.out.println("Choose any of the following options");

System.out.println("1)get book details");

System.out.println("2)issue book");

System.out.println("3)return book");

System.out.println("4)check if any fine and pay if any");

System.out.println("5)get staff details");

System.out.println("6)reserve a book");

System.out.println("7)get user details");

System.out.println("8)exit");

int choice3= s1.nextInt();

if(choice3==1)

{

System.out.println("Choose any of the following options");

System.out.println("1)search by bookname");

System.out.println("2)search by type");

int choice4= s1.nextInt();

if(choice4==1)

{

String searchBook=s1.next();

for(int i=0;i<book.number\_of\_books;i++)

{

b1[i].search\_book(searchBook);

}

// display the person id who displayed the book

}

else

{

String searchType=s1.next();

for(int i=0;i<book.number\_of\_books;i++)

{

b1[i].search\_type(searchType);

}

//display the person id who displayed the book

}

}

//end of first if

if(choice3==2)

{

int flag=0;

System.out.println("Enter the bookid2"

+ " the customer want to issue");

int searchBook=s1.nextInt();

System.out.println("Searching for the booking");

outerloop:

for(int i=0;i<book.number\_of\_books;i++)

{

int validity=b1[i].search(searchBook);

if(validity==1)

{

System.out.println("The book is found");

System.out.println("Do you want to take the book enter 1");

int choice5=s1.nextInt();

if(choice5==1)

{

System.out.println("please enter ur id and name to proceed further");

int id1=s1.nextInt();

String name1=s1.next();

int flag4=0;

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

{

if(p1[j].validate(id1, name1)==true)

{

flag4=1;

break;

}

}

if(flag4==1)

{

if(list1.contains(id1))

{

System.out.println("please return book first");

break outerloop;

}

else

{

System.out.println("enter the number of days you want");

n=s1.nextInt();

System.out.println("enter the issue date");

System.out.println(b1[i].getIssueDate());

b1[i].funscheduledate(n);

System.out.println(list3.contains(b1[i].ID));

if(list3.contains(b1[i].ID))

{

int val= (b1[i].issueDate).compareTo((LocalDate) dict.get(b1[i].ID));

System.out.println("hi");

if(val==0 && id1!=(int)dict1.get(b1[i].ID) )

{

System.out.println("the book is already reserved for another person");

}

else if(id1==(int)dict1.get(b1[i].ID))

{

System.out.println("u already reserved the book so can issue it");

b1[i].isIssued=true;

list1.add(id1);

list2.add(b1[i].ID);

for(int mar=0;mar<nc;mar++)

{

if(p1[mar].searchperson(id1)==1);

{

p1[mar].setbookTakenId(b1[i].ID);

}

}

System.out.println("you issued the book");

dict.remove(b1[i].ID);

dict1.remove(b1[i].ID);

int index= list3.indexOf(b1[i].ID);

list3.remove(index);

list1.add(p1[i].ID);

list2.add(b1[i].ID);

b1[i].isReserved=false;

}

else if(val>0 && id1!=(int)dict1.get(b1[i].ID))

{

System.out.println("cannot be reserved as book is already reserved for that person");

}

else if(val<0 && id1!=(int)dict1.get(b1[i].ID))

{

int val1= ((LocalDate)dict.get(b1[i].ID)).compareTo(b1[i].schedule\_returndate);

if(val1<0)

{

System.out.println("you cannot issue the book");

}

else

{

b1[i].isIssued=true;

list1.add(id1);

list2.add(b1[i].ID);

for(int mar=0;mar<nc;mar++)

{

if(p1[mar].searchperson(id1)==1);

{

p1[mar].setbookTakenId(b1[i].ID);

}

}

System.out.println("you issued the book");

}

}

}

else

{

System.out.println("u succesfully issued the book");

list1.add(id1);

list2.add(b1[i].ID);

b1[i].isIssued=true;

}

}

}

else

{

System.out.println("please enter ur crct detials"); //exception

}

}

else

{

break outerloop;

}

flag=1;

}

}

if(flag==0)

{

System.out.println("looks like the book is not found");

}

}

//returning the book

if(choice3==3)

{

book.payfine(p1,list1,list2,b1,n,nc,list3,resdate,dict,dict1);

// end of choice3

}

if(choice3==4)

{

System.out.println("please enter your id if you have any fine");

Scanner s111=new Scanner(System.in);

int id11=s111.nextInt();

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

{

if(p1[i].searchperson(id11)==1)

{

System.out.println(p1[i].returnFine());

if(p1[i].returnFine())

{

System.out.println("enter 1 to clear your fine");

int choice11=s111.nextInt();

if(choice11==1)

{

book.payfine1(id11,list1,list2,p1,b1,nc);

}

else

{

break;

}

}

else

{

break;

}

}

}

}

//end of choice 4

if(choice3==5)

{

System.out.println("enter the 1d of the staff u want to get the details");

Scanner s1111=new Scanner(System.in);

int id1=s1111.nextInt();

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

{

p2[i].getDetails(id1);

}

}

if(choice3==6)

{

//editable

// don't edit others

System.out.println ("enter the customer id to reserve the book");

Scanner sm=new Scanner(System.in);

int id=sm.nextInt();

System.out.println ("enter the date you want to reserve the book");

int resid;

String a = s1.next();

resdate=LocalDate.parse(a);

System.out.println ("enter the book you want");

String resbook = s1.next();

for(int i=0;i<book.number\_of\_books;i++)

{

if(resbook.equals(b1[i].name))

{

int resbookid = b1[i].getid();

if(list2.contains(resbookid))

{

LocalDate resbook\_returndate=b1[i].get\_schedule\_date();

b1[i].reservebook\_code(resdate,resbook\_returndate,resbookid,list1,list2,dict,b1,id,dict1,list3);

}

else

{

System.out.println("you can reserve the book");

int rbc;

System.out.println("Enter 1 if u want to surely want to reserve the book once u reserve u cant cancel");

rbc=s1.nextInt();

if(rbc==1)

{

int sri=0;

System.out.println("Your book is sucessfully reserved");

b1[i].isReserved=false;

//

for(int z=0;z<book.number\_of\_books;z++)

{

sri=b1[z].search\_id(resbookid);

}

dict.put(sri, resdate);

System.out.println(dict);

dict1.put(sri,id);

System.out.println(dict1);

list3.add(resbookid);

System.out.println(list3);

}

}

}

}

}

if(choice3==7)

{

System.out.println("enter the 1d of the customer u want to get the details");

Scanner s1111=new Scanner(System.in);

int id1=s1111.nextInt();

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

{

p1[i].getDetails(id1);

}

}

if(choice3==8)

{

break;

}

}

}

}

Staff class

**package** Project;

**import** java.util.Scanner;

**public** **class** staff **extends** person **implements** Runnable

{ Scanner s1 = **new** Scanner(System.***in***);

String type\_of\_staff;

**int** salary;

**int** id;

**static** **int** *staffnum*;

String password;

**int** validity;

**public** **static** **void** get\_num\_of\_staff(**int** n)

{

*staffnum*= n;

}

**public** **static** **int** getstaffnum()

{

**return** *staffnum*;

}

**public** staff()

{

System.***out***.println("enter the name:");

**this**.name=s1.next();

**while**(**true**)

{

System.***out***.println("enter the phone:");

phone=s1.next();

**try**

{

**int** a=phone.length();

**if**(a==10)

{

**break**;

}

**else**

{

**throw** **new** IllegalAccessException(" Not valid Phoneno");

}

}

**catch**(IllegalAccessException e)

{

System.***out***.println(e);

}

}

System.***out***.println("enter the age:");

**this**.age=s1.nextInt();

System.***out***.println("enter the gender:");

**this**.Gender=s1.next().charAt(0);

System.***out***.println("enter the type of staff:");

**this**.type\_of\_staff=s1.next();

System.***out***.println("enter the salary:");

**this**.salary=s1.nextInt();

}

**void** getDetails(**int** id1)

{

**if**(**this**.id==id1)

{

System.***out***.println("name: "+**this**.name);

System.***out***.println("phone: "+**this**.phone);

System.***out***.println("age: "+**this**.age);

System.***out***.println("gender: "+**this**.Gender);

System.***out***.println("type\_of\_staff: "+**this**.type\_of\_staff);

System.***out***.println("salary: "+**this**.salary);

}

}

**void** assignID(**int** id)

{

**this**.id=id;

System.***out***.println("the id assigned to you is:"+id);

}

**public** **int** returnid()

{

**return** id;

}

**public** **void** run()

{

password=name+'@'+Gender;

}

@Override

**int** searchperson(**int** id9) {

// **TODO** Auto-generated method stub

**return** 0;

}

@Override

**void** setanyfine() {

// **TODO** Auto-generated method stub

}

@Override

**boolean** returnFine() {

// **TODO** Auto-generated method stub

**return** **false**;

}

@Override

**void** revanyfine() {

// **TODO** Auto-generated method stub

}

@Override

**void** setbookTakenId(**int** id44) {

// **TODO** Auto-generated method stub

}

@Override

**boolean** validate(**int** id, String name) {

// **TODO** Auto-generated method stub

**if**(**this**.id==id && password.equals(name))

{

System.***out***.println("welcome "+**this**.name);

**return** **true**;

}

**return** **false**;

}

**class** address

{

address()

{

System.***out***.println("please enter the address");

Scanner s1=**new** Scanner(System.***in***);

address1=s1.nextLine();

}

}

}