Project7\_1:

import java.util.\*;

public class Project7\_1 {

public static void main(String[] args){

Scanner read = new Scanner(System.in);

linkedList setA = new linkedList();

linkedList setB = new linkedList();

linkedList setAUB = new linkedList();

enterInt(setA, "setA", read);

display(setA, "setA=");

enterInt(setB, "setB", read);

display(setB, "setB=");

Union(setA,setB,setAUB);

display(setAUB, "setAUB=");

}

public static void enterInt(linkedList set, String setName, Scanner read){

System.out.println("Enter the number of " + setName + ": ");

System.out.print(" Enter a group of positive integer number with -1 at the end: ");

int num = read.nextInt();

while(num != -1){

set.insertOrdered(num);

num = read.nextInt();

}

String temp = read.nextLine();

}

public static void Union(linkedList setA, linkedList setB, linkedList setAUB){

node p = setA.ordered;

node q = setB.ordered;

while(p!=null && q!=null){

if(p.info<q.info ){

setAUB.insertOrdered(p.info);

p = p.next;

}else if(p.info>q.info){

setAUB.insertOrdered(q.info);

q=q.next;

}else{

setAUB.insertOrdered(p.info);

p=p.next;

q=q.next;

}

}

while(p!= null){

setAUB.insertOrdered(p.info);

p = p.next;

}

while(q!= null){

setAUB.insertOrdered(q.info);

q = q.next;

}

}

public static void display(linkedList set, String setName){

System.out.print(setName);

set.display(set.ordered);

}

}

output:

Enter the number of setA:

Enter a group of positive integer number with -1 at the end: 1 2 3 7 8 23 12 -1

setA=1->2->3->7->8->12->23->null

Enter the number of setB:

Enter a group of positive integer number with -1 at the end: 23 12 34 2 8 -1

setB=2->8->12->23->34->null

setAUB=1->2->3->7->8->12->23->34->null

Project7\_2:

import java.util.\*;

import java.io.\*;

public class Project7\_2 {

public static void main(String[] args){

Scanner read = new Scanner(System.in);

Hashing accounts = new Hashing();

char Continue =' ';

int choice;

try {

copydata("accounts", accounts);

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

Menu();

do{

System.out.print("Please enter your choice(1-4): ");

choice = read.nextInt();

Process(choice, accounts);

System.out.print("CONTINUE(y/n)?");

Continue = read.next().charAt(0);

String temp = read.nextLine();

}while(Continue == 'y' || Continue == 'Y');

}

public static void copydata(String fName, Hashing accounts) throws Exception{

BufferedReader read = new BufferedReader(new FileReader(fName));

String line;

while((line = read.readLine())!=null){

String[] token=line.split(" ", 3);

accounts.insertHash(Integer.parseInt(token[0]), token[1], Double.parseDouble(token[2]));

}

}

public static void Menu(){

System.out.println("------------Menu-------------"

+ "\n1. Show my balance"

+ "\n2. Deposit in my account "

+ "\n3. Withdraw from my account "

+ "\n4. Show me all\n");

}

public static void Process(int choice, Hashing accounts){

switch(choice){

case 1: Show(accounts); break;

case 2: Deposit(accounts); break;

case 3: Withdraw(accounts); break;

case 4: accounts.display(); break;

default: break;

}

}

public static acctnode FindAcctId(Hashing accounts){

System.out.print("Enter your account number: ");

Scanner read = new Scanner(System.in);

int acctId = read.nextInt();

acctnode p = accounts.H[acctId%5];

while(p.id != acctId){

p=p.next;

}

return p;

}

public static void Show(Hashing accounts){

acctnode p = FindAcctId(accounts);

System.out.println(p.name + ", your balance is " + p.balance);

}

public static void Deposit(Hashing accounts){

acctnode p = FindAcctId(accounts);

System.out.print(p.name + ", how much would you like to deposit?");

Scanner read = new Scanner(System.in);

double trans = read.nextDouble();

p.balance += trans;

System.out.println("Your new balance is " + p.balance);

}

public static void Withdraw(Hashing accounts){

acctnode p = FindAcctId(accounts);

System.out.print(p.name + ", how much would you like to withdraw?");

Scanner read = new Scanner(System.in);

double trans = read.nextDouble();

p.balance -= trans;

System.out.println("Your new balance is " + p.balance);

}

}

output:

------------Menu-------------

1. Show my balance

2. Deposit in my account

3. Withdraw from my account

4. Show me all

Please enter your choice(1-4): 4

Acct[0]->170,Bill,170.3->110,John,110.1->null

Acct[1]->null

Acct[2]->122,Adam,120.6->null

Acct[3]->103,Mary,100.2->null

Acct[4]->194,Tina,190.5->154,George,150.4->null

CONTINUE(y/n)?y

Please enter your choice(1-4): 1

Enter your account number: 110

John, your balance is 110.1

CONTINUE(y/n)?y

Please enter your choice(1-4): 2

Enter your account number: 110

John, how much would you like to deposit?20

Your new balance is 130.1

CONTINUE(y/n)?y

Please enter your choice(1-4): 3

Enter your account number: 110

John, how much would you like to withdraw?30

Your new balance is 100.1

CONTINUE(y/n)?n