q1)(STRING operations on arraylist)

import java.util.\*;

public class Arraylist

{

static Scanner input=new Scanner(System.in);

public static void main(String[] args)

{

int choice;

ArrayList<String> arr=new ArrayList<String>();

while(true)

{

System.out.print("1 --append\n2 --replace\n3 --display\n4 --insert\n5 --starts with\n6 --Substring\n7 --sort \n8 --remove\n9 --remove duplicates\n10 --index\n-1 --stop\nenter your choice : ");

choice=input.nextInt();

if(choice==1)

append(arr);

else if(choice==2)

replace(arr);

else if(choice==3)

System.out.println("arr : "+arr);

else if(choice==4)

insert(arr);

else if(choice==5)

startswith(arr);

else if(choice==6)

substring(arr);

else if(choice==7)

sort(arr);

else if(choice==8)

remove(arr);

else if(choice==9)

removeduplicates(arr);

else if(choice==10)

index(arr);

else if(choice==-1)

break;

}

}

static void append(ArrayList arr)

{

String appstring;

System.out.print("enter String to append : ");

appstring=input.next();

arr.add(appstring);

}

static void replace(ArrayList arr)

{

String newst;

int position;

System.out.print("enter String to insert at particular position : ");

newst=input.next();

System.out.print("enter position : ");

position=input.nextInt();

arr.set(position,newst);

}

static void insert(ArrayList arr)

{

String newst;

int position;

System.out.print("enter String to insert at particular position : ");

newst=input.next();

System.out.print("enter position : ");

position=input.nextInt();

arr.add(position,newst);

}

static void startswith(ArrayList arr)

{

char x;

System.out.print("enter character : ");

x=input.next().charAt(0);

for(int i=0;i<arr.size();i++)

{

//System.out.print(" "+i+": ");

String newst=(String)(arr.get(i));

if(newst.charAt(0)==x)

System.out.print(newst+" ");

}

System.out.println();

}

static void substring(ArrayList arr)

{

String sub,listele;

System.out.print("enter sub string : ");

sub=input.next();

for(int i=0;i<arr.size();i++)

{

listele=(String)arr.get(i);

if(listele.contains(sub))

System.out.print(listele+" ");

}

}

static void sort(ArrayList arr)

{

Collections.sort(arr);

System.out.println(arr);

}

static void remove(ArrayList arr)

{

String temp;

System.out.print("enter name to remove : ");

temp=input.next();

arr.remove(temp);

}

static void removeduplicates(ArrayList arr)

{

ArrayList<String> newlist=new ArrayList<String>();

for(int i=0;i<arr.size();i++)

{ String element= (String) arr.get(i);

if(!newlist.contains(element))

{

newlist.add(element);

}

}

System.out.println("with out duplicates : "+newlist);

}

static void index(ArrayList arr)

{

int i;

String element;

System.out.print("enter element to find it's index : ");

element=input.next();

System.out.println("index is "+arr.indexOf(element));

}

}

/\*

SAMPLE I/O:

aswin@aswin-pavilion-15:~/Desktop/sem3/java/lab/ex8$ java Arraylist

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 1

enter String to append : java

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 1

enter String to append : python

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 1

enter String to append : c

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 1

enter String to append : r

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 1

enter String to append : c++

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 4

enter String to insert at particular position : 1

enter position : 1

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 3

arr : [java, 1, python, c, r, c++]

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : 5

enter character : ja

java

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

-1 --stop

enter your choice : -1

(recomplied)

aswin@aswin-pavilion-15:~/Desktop/sem3/java/lab/ex8$ java Arraylist

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

10 --index

-1 --stop

enter your choice : 1

enter String to append : aswin

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

10 --index

-1 --stop

enter your choice : 10

enter element to find it's index : aswin

index is 0

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

10 --index

-1 --stop

enter your choice : 10

enter element to find it's index : as

index is -1

1 --append

2 --replace

3 --display

4 --insert

5 --starts with

6 --Substring

7 --sort

8 --remove

9 --remove duplicates

10 --index

-1 --stop

\*/

=================================================================================

Q2)(INT arraylist)

import java.util.\*;

class intArraylist

{

static Scanner input=new Scanner(System.in);

public static void main(String[] args)

{

int choice;

ArrayList<Integer>intlist1=new ArrayList<Integer>();

ArrayList<Integer>intlist2=new ArrayList<Integer>();

System.out.println("for list 1");

input(intlist1);

System.out.println("for list 2");

input(intlist2);

System.out.print("elements in list 1 : ");

output\_fun(intlist1);

System.out.print("elements in list 2 : ");

output\_fun(intlist2);

while(true)

{

System.out.print("1 --union\n2 intersection\n3 --comapare\n4 --merge\n-1 --stop\nenter your choice : ");

choice=input.nextInt();

if(choice==-1)

break;

else if(choice==1)

{

ArrayList<Integer>U=union(intlist1,intlist2);

System.out.println(U); }

else if(choice==2)

{

if(intlist1.size()>=intlist2.size())

intersection(intlist1,intlist2);

else

intersection(intlist2,intlist1);

}

else if(choice==4)

merge(intlist1,intlist2);

else if(choice==3)

compare(intlist1,intlist2);

}

}

static void input(ArrayList<Integer> list)

{

Integer newint;

while(true)

{

System.out.print("enter Integer to add in list(-1122 to stop) : ");

newint=input.nextInt();

if(newint!=-1122)

list.add(newint);

else

break;

}

}

static void output\_fun(ArrayList<Integer> list)

{

System.out.println(list);

}

static ArrayList<Integer> union(ArrayList<Integer> list1,ArrayList<Integer> list2)

{

ArrayList<Integer> U=new ArrayList<Integer>();

for(Integer temp : list1)

if(list2.contains(temp)&&!U.contains(temp))

U.add(temp);

//System.out.println("union is : "+U);

return U;

}

static void intersection(ArrayList<Integer> list1,ArrayList<Integer> list2)

{

ArrayList<Integer> U=new ArrayList<Integer>();

for(Integer temp : list1)

if(!list2.contains(temp)&&!U.contains(temp))

U.add(temp);

System.out.println("union is : "+U);

}

static void merge(ArrayList<Integer> list1,ArrayList<Integer> list2)

{

ArrayList<Integer> U=new ArrayList<Integer>();

U.addAll(list1);

for(Integer temp : list2)

if(!U.contains(temp))

U.add(temp);

System.out.println("mered list is : "+U);

}

static void compare(ArrayList<Integer> list1,ArrayList<Integer> list2)

{

ArrayList<Integer>U=union(list1,list2);

if(U.size()==list1.size()&&U.size()==list2.size())

System.out.println("they are same ");

else

System.out.println("they aren't same ");

}

}

/\*

SAMPLE i/o:

aswin@aswin-pavilion-15:~/Desktop/sem3/java/lab/ex8$ javac intArraylist.java

aswin@aswin-pavilion-15:~/Desktop/sem3/java/lab/ex8$ java intArraylist

for list 1

enter Integer to add in list(-1122 to stop) : 1

enter Integer to add in list(-1122 to stop) : 2

enter Integer to add in list(-1122 to stop) : 3

enter Integer to add in list(-1122 to stop) : 4

enter Integer to add in list(-1122 to stop) : 5

enter Integer to add in list(-1122 to stop) : 6

enter Integer to add in list(-1122 to stop) : 7

enter Integer to add in list(-1122 to stop) : -1122

for list 2

enter Integer to add in list(-1122 to stop) : 1

enter Integer to add in list(-1122 to stop) : 2

enter Integer to add in list(-1122 to stop) : 8

enter Integer to add in list(-1122 to stop) : 9

enter Integer to add in list(-1122 to stop) : 5

enter Integer to add in list(-1122 to stop) : 6

enter Integer to add in list(-1122 to stop) : 3

enter Integer to add in list(-1122 to stop) : 4

enter Integer to add in list(-1122 to stop) : -1122

1 --union

2 intersection

3 --comapare

4 --merge

-1 --stop

enter your choice : 1

[1, 2, 3, 4, 5, 6]

1 --union

2 intersection

3 --comapare

4 --merge

-1 --stop

enter your choice : 2

union is : [8, 9]

1 --union

2 intersection

3 --comapare

4 --merge

-1 --stop

enter your choice : 3

they aren't same

1 --union

2 intersection

3 --comapare

4 --merge

-1 --stop

enter your choice : 4

mered list is : [1, 2, 3, 4, 5, 6, 7, 8, 9]

1 --union

2 intersection

3 --comapare

4 --merge

-1 --stop

enter your choice :-1

\*/

=================================================================================

+++end+++