Exercise - 9– Collection Framework

1. Using Collection framework, create a doubly linked list of integers and perform the

following operations.

a. Insert element on both sides

b. Delete element on both sides

c. Insert an element at a particular position

d. Delete a particular element

e. Search for a particular element

f. Display list in forward order and backward order

g. Sort the elements in LinkedList

h. Replace one element in the list with another list

i. Remove duplicate elements

import java.util.\*;

class main

{

public static void main(String[] args)

{

int a,b,i=0,c;

Scanner in =new Scanner(System.in);

LinkedList<Integer> doubly =new LinkedList<Integer>();

while(i<10){

System.out.print("Enter two elements for insert : ");

b=in.nextInt();

doubly.add(b);

i++;

}

System.out.println("Doubly LinkedList : "+doubly);

System.out.println(" a. Insert element on both sides");

System.out.print("Enter two elements for insert on both sides : ");

a=in.nextInt();

b=in.nextInt();

doubly.addFirst(a);

doubly.addLast(b);

System.out.println("After inserted on both sides ,Doubly LinkedList : "+doubly);

System.out.println(" b. Delete element on both sides");

doubly.removeFirst();

doubly.removeLast();

System.out.println("After deleted on both sides ,Doubly LinkedList : "+doubly);

System.out.println(" c. Insert an element at a particular position");

System.out.print("Enter the element and position to insert : ");

a=in.nextInt();

b=in.nextInt();

doubly.add(b,a);

System.out.println("After inserted in "+ b+"th position ,Doubly LinkedList : "+doubly);

System.out.println(" d. Delete a particular element");

System.out.print("Enter the element to delete : ");

a=in.nextInt();

b=doubly.indexOf(a);

System.out.println(b);

doubly.remove(b);

System.out.println("After deleted "+a+" ,Doubly LinkedList : "+doubly);

System.out.println(" f.Search for a particular element");

System.out.print("Enter the element to search : ");

a=in.nextInt();

if(doubly.contains(a)) System.out.println("\n"+a+" is present");

else System.out.println("\n"+a+" is not present");

System.out.println("Display list in forward order and backward order");

System.out.println("Display list in forward order ");

ListIterator<Integer> litr = doubly.listIterator();

while(litr.hasNext())

System.out.print(litr.next()+" ");

System.out.println("\n\*\*\*\*\*");

System.out.println("\nDisplay list in backward order ");

while(litr.hasPrevious ())

System.out.print(litr.previous()+" ");

System.out.println("\n\*\*\*\*\*");

System.out.println(" g. Sort the elements in LinkedList");

Collections.sort(doubly);

System.out.println("After sort the doubly linkedlist : "+doubly);

System.out.println(" h. Replace one element in the list with another list");

System.out.print("Enter the list element and replace element : ");

a=in.nextInt();

b=in.nextInt();

c=doubly.indexOf(a);

doubly.set(c,b);

System.out.println("After replace doubly linkedlist : "+doubly);

System.out.println(" i. Remove duplicate elements");

LinkedHashSet<Integer> arr2 = new LinkedHashSet<Integer>(doubly);

LinkedList<Integer> arr3 = new LinkedList<Integer>(arr2);

System.out.println("After removal duplicate elements, doubly linkedlist : "+arr3);

}

}

/\*

C:\Users\BALAJI\Desktop>java main

Enter two elements for insert : 9

Enter two elements for insert : 7

Enter two elements for insert : 6

Enter two elements for insert : 5

Enter two elements for insert : 4

Enter two elements for insert : 3

Enter two elements for insert : 4

Enter two elements for insert : 1

Enter two elements for insert : 2

Enter two elements for insert : 8

Doubly LinkedList : [9, 7, 6, 5, 4, 3, 4, 1, 2, 8]

a. Insert element on both sides

Enter two elements for insert on both sides : 11 10

After inserted on both sides ,Doubly LinkedList : [11, 9, 7, 6, 5, 4, 3, 4, 1, 2, 8, 10]

b. Delete element on both sides

After deleted on both sides ,Doubly LinkedList : [9, 7, 6, 5, 4, 3, 4, 1, 2, 8]

c. Insert an element at a particular position

Enter the element and position to insert : 0 1

After inserted in 1th position ,Doubly LinkedList : [9, 0, 7, 6, 5, 4, 3, 4, 1, 2, 8]

d. Delete a particular element

Enter the element to delete : 9

0

After deleted 9 ,Doubly LinkedList : [0, 7, 6, 5, 4, 3, 4, 1, 2, 8]

f.Search for a particular element

Enter the element to search : 2

2 is present

Display list in forward order and backward order

Display list in forward order

0 7 6 5 4 3 4 1 2 8

\*\*\*\*\*

Display list in backward order

8 2 1 4 3 4 5 6 7 0

\*\*\*\*\*

g. Sort the elements in LinkedList

After sort the doubly linkedlist : [0, 1, 2, 3, 4, 4, 5, 6, 7, 8]

h. Replace one element in the list with another list

Enter the list element and replace element : 4 5

After replace doubly linkedlist : [0, 1, 2, 3, 5, 4, 5, 6, 7, 8]

i. Remove duplicate elements

After removal duplicate elements, doubly linkedlist : [0, 1, 2, 3, 5, 4, 6, 7, 8]

\*/

2. Create a class “Movie” with name, actor, year, genre and rating as data members. Add

functions to get inputs and display. Write a program to create a Movie Linked Queue

using collection framework. Perform the following operations:

a) Add a Movie at last

b) Remove a Movie from front

c) Display the Movie details in queue

d) Display the movies based on rating score

e) Display the movies based on recent release

import java.util.\*;

class Movie

{

String name,actor,genre;double rating,year;

public Movie(String name,String actor,double year,String genre,double rating)

{

this.name=name;

this.actor=actor;

this.year=year;

this.genre=genre;

this.rating=rating;

}

public String toString()

{

String msg= "\nNAME : "+name+"\nACTOR : "+actor+"\nYEAR : "+year+"\nGENRE : "+genre+"\nRATING : "+rating;

return msg;

}

public static void main(String[] args) {

int n,j=0;

Scanner in =new Scanner(System.in);

Queue<Movie> m =new LinkedList<Movie>();

System.out.print("Enter number of entries : ");

n=in.nextInt();String name\_1,actor\_1,genre\_1;double year\_1,rating\_1;

while(j<n)

{

System.out.println("Enter name,actor,year,genre,rating......");

name\_1=in.next();

actor\_1=in.next();

year\_1=in.nextDouble();

genre\_1=in.next();

rating\_1=in.nextDouble();

m.add(new Movie(name\_1,actor\_1,year\_1,genre\_1,rating\_1));

j++;

}

System.out.println("Display the Movie details in queue");

for(Movie s : m)

System.out.println(s);

System.out.println("Display the Movie details in queue,after removal a Movie from front");

m.remove();

for(Movie s1 : m)

System.out.println(s1);

LinkedList<Movie> m1 =new LinkedList<Movie>(m);

int i=0;

while(i<n)

{

Collections.sort(m1,new RatingComparator());

i++;

}

System.out.println("Based on rating.....");

for(Movie s2 : m1)

System.out.println(s2);

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

LinkedList<Movie> m2 =new LinkedList<Movie>(m);

i=0;

while(i<n)

{

Collections.sort(m2,new ReleaseComparator());

i++;

}

System.out.println("Based on recent release.....");

for(Movie s3 : m2)

System.out.println(s3);

}

}

class RatingComparator implements Comparator<Movie>

{

public int compare(Movie s1, Movie s2)

{

if (s1.rating < s2.rating)

return 1;

return -1;

}

}

class ReleaseComparator implements Comparator<Movie>

{

public int compare(Movie s1, Movie s2)

{

if (s1.year < s2.year)

return 1;

return -1;

}

}

/\*C:\Users\BALAJI\Desktop>javac last1.java

C:\Users\BALAJI\Desktop>java Movie

Enter number of entries : 5

Enter name,actor,year,genre,rating......

SURA VIJAY 2010 COMEDY 1.5

Enter name,actor,year,genre,rating......

AYAN SURYA 2012 ACTION 9.5

Enter name,actor,year,genre,rating......

NGK SURYA 2019 MYSTERY 9.4

Enter name,actor,year,genre,rating......

VIKRAM\_VEDHA VIJAY\_SETHUPATHI 2017 DRAMA 8.7

Enter name,actor,year,genre,rating......

INCEPTION LEONARDO\_DICAPRIO 2010 THIRLLER/SCI-FI 10.0

Display the Movie details in queue

NAME : SURA

ACTOR : VIJAY

YEAR : 2010

GENRE : COMEDY

RATING : 1.5

NAME : AYAN

ACTOR : SURYA

YEAR : 2012

GENRE : ACTION

RATING : 9.5

NAME : NGK

ACTOR : SURYA

YEAR : 2019

GENRE : MYSTERY

RATING : 9.4

NAME : VIKRAM\_VEDHA

ACTOR : VIJAY\_SETHUPATHI

YEAR : 2017

GENRE : DRAMA

RATING : 8.7

NAME : INCEPTION

ACTOR : LEONARDO\_DICAPRIO

YEAR : 2010

GENRE : THIRLLER/SCI-FI

RATING : 10.0

Display the Movie details in queue,after removal a Movie from front

NAME : AYAN

ACTOR : SURYA

YEAR : 2012

GENRE : ACTION

RATING : 9.5

NAME : NGK

ACTOR : SURYA

YEAR : 2019

GENRE : MYSTERY

RATING : 9.4

NAME : VIKRAM\_VEDHA

ACTOR : VIJAY\_SETHUPATHI

YEAR : 2017

GENRE : DRAMA

RATING : 8.7

NAME : INCEPTION

ACTOR : LEONARDO\_DICAPRIO

YEAR : 2010

GENRE : THIRLLER/SCI-FI

RATING : 10.0

Based on rating.....

NAME : INCEPTION

ACTOR : LEONARDO\_DICAPRIO

YEAR : 2010

GENRE : THIRLLER/SCI-FI

RATING : 10.0

NAME : AYAN

ACTOR : SURYA

YEAR : 2012

GENRE : ACTION

RATING : 9.5

NAME : NGK

ACTOR : SURYA

YEAR : 2019

GENRE : MYSTERY

RATING : 9.4

NAME : VIKRAM\_VEDHA

ACTOR : VIJAY\_SETHUPATHI

YEAR : 2017

GENRE : DRAMA

RATING : 8.7

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Based on recent release.....

NAME : NGK

ACTOR : SURYA

YEAR : 2019

GENRE : MYSTERY

RATING : 9.4

NAME : VIKRAM\_VEDHA

ACTOR : VIJAY\_SETHUPATHI

YEAR : 2017

GENRE : DRAMA

RATING : 8.7

NAME : AYAN

ACTOR : SURYA

YEAR : 2012

GENRE : ACTION

RATING : 9.5

NAME : INCEPTION

ACTOR : LEONARDO\_DICAPRIO

YEAR : 2010

GENRE : THIRLLER/SCI-FI

RATING : 10.0

C:\Users\BALAJI\Desktop>\*/