Completion Date : 18/10/2020

Assigment 01

(31139)

Title : Use of collations and generics :

Problem statement : Design a system with the help of advance data structure in Java and enhance the system using collections and generics.

Objective : To understand and implement java generics and collections.

Outcomes : We will understand the basics of java generics and collections.

Software and hardware :
Eclipse IDE, Fedora OS/Windows 10, Intel i5 processor, 4 GB RAM.

Theory :

1) Collections in Java.

- Collections is a framework that provides an architecture to store and manipulatek group of objects.
- all the operations that you perform on a data such as searching, sorting, etc. Can be performed by Collections.

a] Java ArrayList class : (31139)

- It uses a dynmmic array for storing the elements. It inherits list class.
- It can contain duplicate entries.
- Allows random access

- methods :
  i) void add (int index, object elemet)
  - insert at specific index
  ii) void clear () - remove all elements
  iii) boolean addAll (collection c) - append at end
  iv) object clone ()

e.g. ArrayList <string> al = new ArrayList <string>

Application : A widely used data str. since it is dynamic, resizable and randomly accessible.

b] Java LinkedList class :

- Use doubly linked list to store elements.
- methods
  i) void add(int index, object elements)
  ii) void addFirst (object o)
  iii) void addLast (object o)
  iv) int size()
  v) boolean contains (object o)

Application : can be used to implement stack, queue c] Java Hashset class: (31139)

- Used to create a collection that uses hashtable for storage.

i) contains unique elements.
ii) storage by hashing.

-methods
i) void clear()
ii) boolean contains (object o)
iii) boolean add (object o)
iv) boolean isEmpty()

e.g. HashSet <String> set = new HashSet<String>();

Applications : Hashset is exclusively used since lookup time is less.

d] Java Queue interface:

- order element in FIFO manner
- methods
  i) boolean add (object)
  ii) object remove() - removes head
  iii) object element(). -retrieve

- Appliction - Transversal

(311 39)

ii) Generics in Java:

Advantages:

1) Type - safety - We can hold only a single type of objects in generics.

2) Type casting is not required

3) Compile-time Checking.

- Syntax: class or interface <type>

  e.g: Arraylist <String>

- Generics class : A class that can refer to any type is known as generic class.

e.g.
```
class myGen <T>
    T obj;
    void add ( T obj)
        {
            this.obj = obj;
        }
    T.get ( ) {
        return obj; }
```

T indicates that it can be any type.

Generic methods also exist.

- Features of Java :

(31139)

i) Object oriented : In Java, everything is an object. It can be easily extended.

ii) Platform department :
Unlike many other programming language including C & C++. When JAVA is completed, it is not compiled into platform specific machine.

iii) Simple : Java is designed to be easy to learn

iv) Secure : Java enables to develop virus-free, tamperfree systems.

v) Portable : Being architectural-neutral and having no implementation dependent aspects of the specification makes Java ~~porta~~ portable.

Collection (interface)

List (interface)	Set (interface)	Queue (interface)	map
ArrayList (class)	Hashset (class)	PoiorityQueue (class)	HashMap (class)
LinkedList (class)	LinkedHashset (class)	Deque (interface)	LinkedHashMap (class)
Vector	TreeSet (class)	ArrayDequeue (class	TreeMap. (class)

प्र.क्र.
Q. No.

Test Case: (31139)

1. Select Genre: 1. Horror 2. Comedy 3. Crime 4. Adventure 5 Drama
i/p. ~~Horror~~ 1.
o/p. Genre chosen: Horror.

2. Recomend
o/p Get Out

3. List Movis
o/p 1. A quit place : 7.5
2. Get Out : 7.7
3. Alien : 8.4
4. World War Z : 7.0
5. The Thing : 8.1

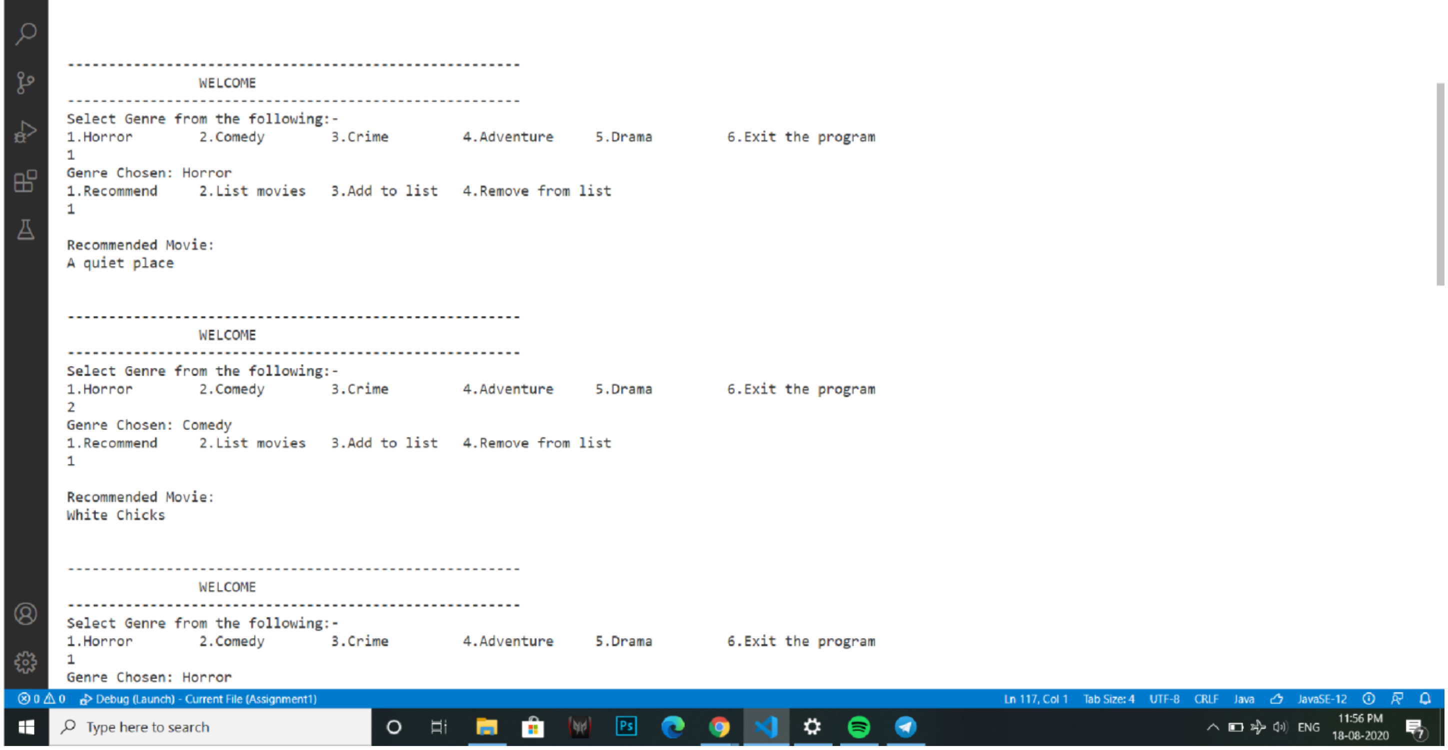
4. Add to list
i/p movie Name : IT
Rating : 8.8
o/p. movie Added

5. Remove from list
i/p Enter the indx of the movie to be removed : 6
o/p The movie to be removed is IT : 8.8
movie Removed.

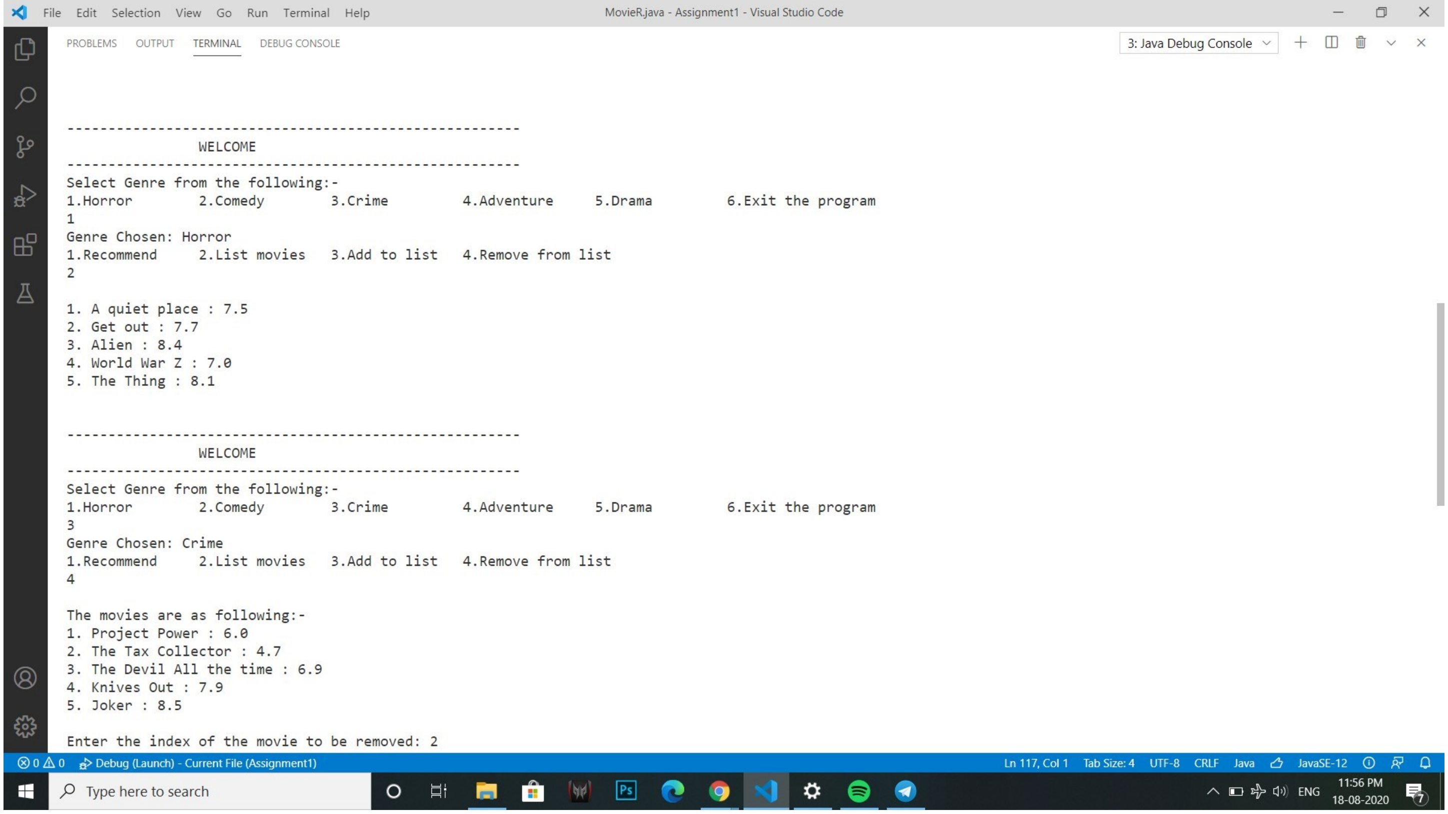
```
import java.util.*;
public class MovieR {
public static Genre horror = new Genre("Horror");
public static Genre comedy = new Genre("Comedy");
public static Genre crime = new Genre("Crime");
public static Genre adventure = new
Genre("Adventure");
public static Genre drama = new Genre("Drama");
public static void main(String[] args) {
        List<Genre> genres = new LinkedList<Genre>();
        genres.add(horror); genres.add(comedy);
genres.add(crime); genres.add(adventure);
genres.add(drama);
        dummyClass dc = new dummyClass();
        dc.addDummyData();
        Scanner input = new Scanner(System.in);
        while(true){
                int genre_no;
                System.out.println("\n-----------------------
---------------------------\n\t\tWELCOME\n-------------------
-------------------------------");
                System.out.println("Select Genre from
the following:-
\n1.Horror\t2.Comedy\t3.Crime\t\t4.Adventure\t5.Drama\t
\t6.Exit the program");
                genre_no = input.nextInt();
                if(genre_no==6){
                        input.close();
                        break;
                }
                System.out.println("Genre Chosen:
"+genres.get(genre_no-1).getName());
                Genre chosen_genre =
genres.get(genre_no-1);
                int choice;

        System.out.println("1.Recommend\t2.List
movies\t3.Add to list\t4.Remove from list");
                choice = input.nextInt();
                int listsize =
chosen_genre.movielist.size();
                System.out.println("");
                switch (choice) {
                        case 1:
                        Random rand = new Random();
                        int index = rand.nextInt(listsize);

        chosen_genre.recommendMovie(index);
                        break;
                        case 2:
                        chosen_genre.printMovies();
                        break;
                        case 3:
                        System.out.println("Enter Movie
Name: ");
                        Scanner s = new
Scanner(System.in);
                        String name = s.nextLine();
                        System.out.print("Enter Rating:
");
```

```
                        double rating =
input.nextDouble();
                        chosen_genre.addMovie(name,
rating);
                        System.out.println("Movie
Added");
                        break;
                        case 4:
                        System.out.println("The movies
are as following:-");
                        chosen_genre.printMovies();
                        System.out.print("Enter the
index of the movie to be removed: ");
                        int movieindex = input.nextInt();
                        if(movieindex>listsize ||
movieindex< 0){

        System.out.println("Enter valid index");
                                // continue;
                        }

        chosen_genre.removeMovie(movieindex-1);
                        break;
                        default:
                        System.out.println("Enter valid
Input");
                        break;
                }
        }
        input.close();
}
};


class Genre{
String name;
List<Movie> movielist = new Vector<Movie>();
Genre(){}
Genre(String Name){
        name = Name;
}
public String getName(){
        return name;
}

public String addMovie(String name, double rating){
        movielist.add(new Movie(name,rating));
        return movielist.get(movielist.size()-1).toString();

}
public void removeMovie(int index){
        System.out.println("The movie to be removed is:
"+movielist.get(index).toString());
        movielist.remove((index));
        System.out.println("Movie removed\n");
}
public void recommendMovie(int index){
        System.out.println("Recommended Movie: \n"+
movielist.get(index).getMovieName()+"\n");
}
public void printMovies(){
        for(int i = 0; i < movielist.size(); i++){
                System.out.println((i+1)+". "
+(movielist.get(i).toString()));
        }
        System.out.println("");
}

};



class Movie{
String name;
double rating;
Movie(String Name, double Rating){
        name = Name;
        rating = Rating;
}
public String getMovieName(){return name;}
public double getRating(){return rating;}
public String toString() {
        return (name + " : " + rating);
}

};



class dummyClass{
public void addDummyData(){
        MovieR.horror.addMovie("A quiet place",7.5);
        MovieR.horror.addMovie("Get out",7.7);
        MovieR.horror.addMovie("Alien",8.4);
        MovieR.horror.addMovie("World War Z",7);
        MovieR.horror.addMovie("The Thing",8.1);

        MovieR.comedy.addMovie("Step Brothers",6.9);
        MovieR.comedy.addMovie("White Chicks",5.6);
        MovieR.comedy.addMovie("The Hot chick",5.5);
        MovieR.comedy.addMovie("The Hangover",7.7);
        MovieR.comedy.addMovie("Horrible
Bosses",6.9);

        MovieR.crime.addMovie("Project Power",6.0);
        MovieR.crime.addMovie("The Tax
Collector",4.7);
        MovieR.crime.addMovie("The Devil All the
time",6.9);
        MovieR.crime.addMovie("Knives Out",7.9);
        MovieR.crime.addMovie("Joker",8.5);

        MovieR.adventure.addMovie("Mulan",6.9);
        MovieR.adventure.addMovie("The Old
Guard",6.7);
        MovieR.adventure.addMovie("Dune",2020);
        MovieR.adventure.addMovie("Jurassic
Park",8.1);
        MovieR.adventure.addMovie("Wonder Woman
1984",6.9);

        MovieR.drama.addMovie("Titanic",7.8);
        MovieR.drama.addMovie("Tristan & Isolde",6.8);
```

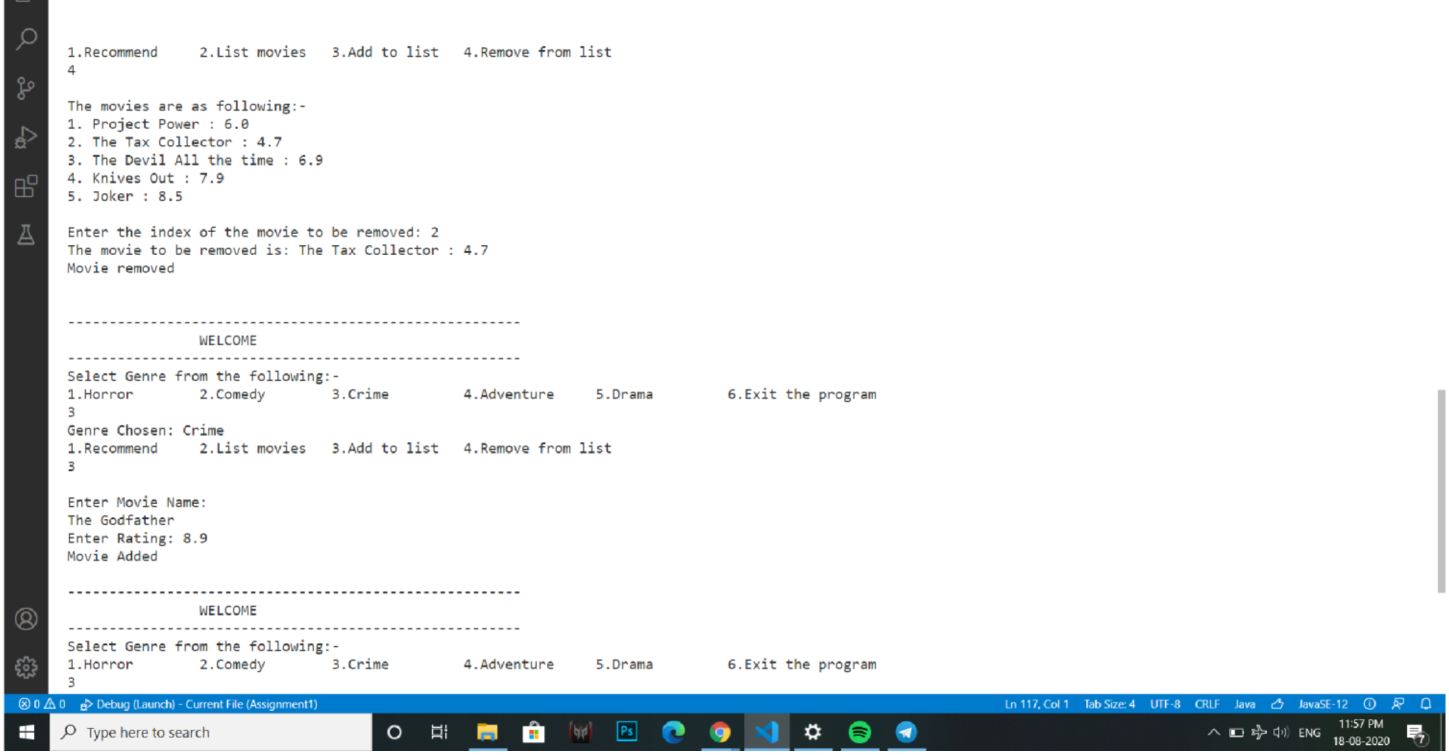
```
            MovieR.drama.addMovie("Dangerous
Beauty",7.2);
            MovieR.drama.addMovie("Head in the
clouds",6.6);
            MovieR.drama.addMovie("Captain Corelli's
Mandolin",5.9);


}
}
```

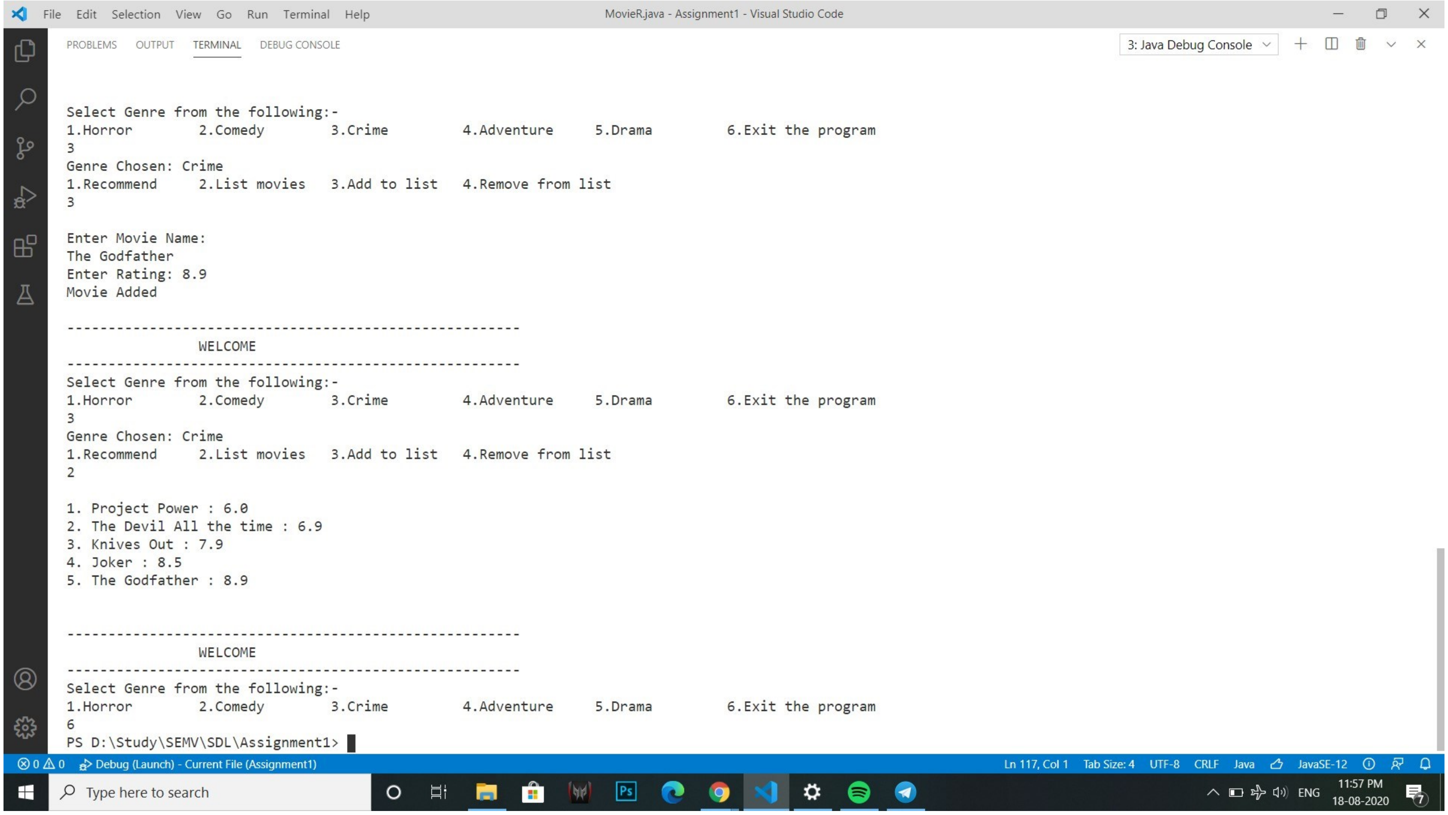


```
--------------------------------------------------------
                WELCOME
--------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
1
Genre Chosen: Horror
1.Recommend     2.List movies   3.Add to list   4.Remove from list
1

Recommended Movie:
A quiet place


--------------------------------------------------------
                WELCOME
--------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
2
Genre Chosen: Comedy
1.Recommend     2.List movies   3.Add to list   4.Remove from list
1

Recommended Movie:
White Chicks


--------------------------------------------------------
                WELCOME
--------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
1
Genre Chosen: Horror
```



```
--------------------------------------------------------
                WELCOME
--------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
1
Genre Chosen: Horror
1.Recommend     2.List movies   3.Add to list   4.Remove from list
2

1. A quiet place : 7.5
2. Get out : 7.7
3. Alien : 8.4
4. World War Z : 7.0
5. The Thing : 8.1


--------------------------------------------------------
                WELCOME
--------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
3
Genre Chosen: Crime
1.Recommend     2.List movies   3.Add to list   4.Remove from list
4

The movies are as following:-
1. Project Power : 6.0
2. The Tax Collector : 4.7
3. The Devil All the time : 6.9
4. Knives Out : 7.9
5. Joker : 8.5

Enter the index of the movie to be removed: 2
```


```
1.Recommend     2.List movies   3.Add to list   4.Remove from list
4

The movies are as following:-
1. Project Power : 6.0
2. The Tax Collector : 4.7
3. The Devil All the time : 6.9
4. Knives Out : 7.9
5. Joker : 8.5

Enter the index of the movie to be removed: 2
The movie to be removed is: The Tax Collector : 4.7
Movie removed


-------------------------------------------------------
                WELCOME
-------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
3
Genre Chosen: Crime
1.Recommend     2.List movies   3.Add to list   4.Remove from list
3

Enter Movie Name:
The Godfather
Enter Rating: 8.9
Movie Added

-------------------------------------------------------
                WELCOME
-------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
3
```



```
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
3
Genre Chosen: Crime
1.Recommend     2.List movies   3.Add to list   4.Remove from list
3

Enter Movie Name:
The Godfather
Enter Rating: 8.9
Movie Added

-------------------------------------------------------
                WELCOME
-------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
3
Genre Chosen: Crime
1.Recommend     2.List movies   3.Add to list   4.Remove from list
2

1. Project Power : 6.0
2. The Devil All the time : 6.9
3. Knives Out : 7.9
4. Joker : 8.5
5. The Godfather : 8.9


-------------------------------------------------------
                WELCOME
-------------------------------------------------------
Select Genre from the following:-
1.Horror        2.Comedy        3.Crime         4.Adventure     5.Drama         6.Exit the program
6
PS D:\Study\SEMV\SDL\Assignment1> 
```