

Programming Assignment 2

Submission Date : 26.03.2021

Due Date : 16.04.2021 (23:59)

Programming Languages: Java

Subject : Inheritance, Access Modifiers

1 Introduction

Object-oriented programming has advantages such as modeling problems with less complexity and more code reuse. In this experiment, you will observe these advantages by using inheritance mechanism which is an important property of object-oriented programming. By the help of this experiment, you will learn the concept of inheritance, relationships among classes by using object references, control of multiple instances of classes, access modifiers in Java.

2 Useful Information

Under this section you will find some introductory information about the OOP concepts you might use in the project. Please make your own research for more information.

2.1 Inheritance

Object-oriented programming (OOP) covers software in terms similar to those that people use to describe real-world objects. It takes advantage of class relationships, where objects of a certain class, such as a class of vehicles, have the same characteristics cars, trucks, little red wagons and roller skates have much in common. Inheritance is one of important property of OOP. OOP takes advantage of inheritance relationships, where new classes of objects are derived by absorbing characteristics of existing classes and adding unique characteristics of their own. In Java, a class (called the derived class or subclass) extends from another class (called the base class or super class).

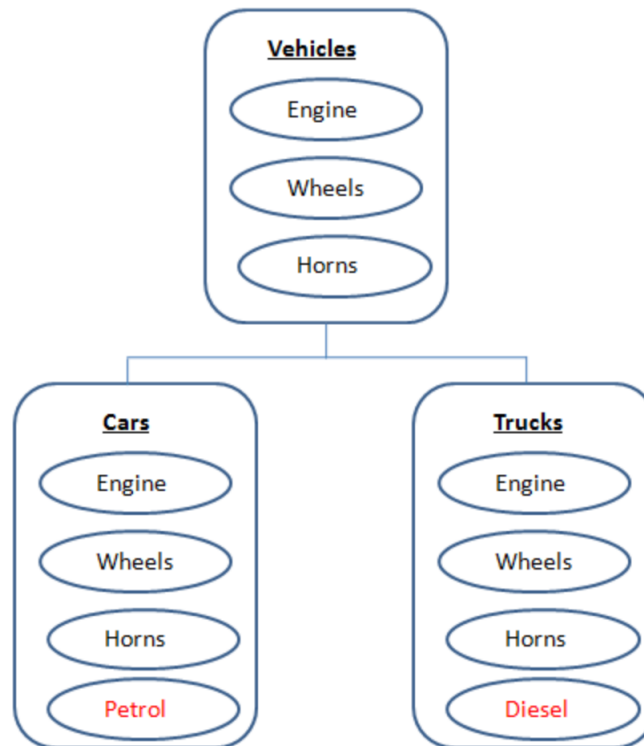


Figure 1: A hierarchy of Vehicle class

In Figure 1, vehicle class hierarchy is seen. Vehicle class is super class of all other classes. Derived classes which are Cars and Trucks have certain common properties; all have engine, wheels, horns etc. Thus they can be grouped under a Class called Vehicles. Apart from sharing these common features, each Derived Class has its own particular features - Cars use petrol while Trucks use diesel.

2.1.1 Method Overriding

When a class extend another class, the subclass can use the super class' methods. However sometimes the subclass should change the behavior of a method which provided by super class. The method implementation in the subclass overrides (replaces) the implementation in the super class. The method in the subclass and its corresponding method in the super class method have the same name, parameters and the same return type. That is called method overriding.

2.2 Method Overloading

Method overloading is another important concept of OOP. When programmers need more than one method with the same functionality, they don't have to declare new methods with different names for each one. By using method overloading feature, they declare each methods with same name but with different signatures (different argument list, argument types or

orders). *System.out.println()* is an example of overloading method in Java. This method takes float, int, double or String as arguments.

2.3 Access Modifiers

In Java, there are three access modifiers which provide access levels for classes and members of classes: private (visible to the class only), protected (visible to the package and all subclasses) and public (visible to the everywhere). The default is visible to the package.

3 Experiment

This section contains four subsections. The first section introduces the problem that you need to solve and gives its details. In the second subsection, the content of reports are described. The constraints are given in the third subsection. Information about the submission is given in the last subsection.

3.1 Problem Definition

In this experiment, you are supposed to develop a simple Movie Database System similar to IMDB. You are responsible for using inheritance mechanism and access modifiers in Java programming language. The system will process several data input files and will generate results of commands which will be read from a command input file. All input files will be error free only syntactically. The requirements and rules for the system are given below:

- The system includes information about people and films.
- Each Person has name, surname, country and a unique id. A person in the system could be either Artist or User.
- Each User has a unique id, name, surname and country.
- There are three kinds of Artist: Performer, Director and Writer. Each Director has a unique id, name, surname, country and agent where he/she works. Each Writer has a unique id, name, surname, country and writing style/type.
- There are also three types of Performers which are Actor, ChildActor and StuntPerformer. Each Actor has a unique id, name, surname, country and height. Each ChildActor has a unique id, name, surname and country. Each StuntPerformer has a unique id, name, surname and country and height.
- There are four types of films in the system: Feature Film, Short Film, Documentary and TV Series. Each film (Feature Film, Short Film, Documentary and TV Series) has a rating score which calculated from users' average rating scores for that film.
- A unique film id, film title, language, runtime, country, directors of a film and cast (actors and actresses of a movie) are common in all film types.
- Feature Films have a release date, budget, writers of movie and film genre in addition to the common data.
- A Short Film has a release date, writers and genre in addition to the common data. A Short Film' runtime should be less (or equal) than 40 min.

- Documentaries have only a release date in addition to the common film data.
- TV Series have start date and end date of series, number of seasons, number of episodes, genre of series and writers in addition to the common film data.
- A film may have more than one directors, writers, performers and genres in this system. A comma will be used to separate these data.

3.1.1 The Way of Execution

The program will be executed with four command line arguments:

```
<people_file><films_file><commands_file><output_file>
```

Usage example:

```
>javac Main.java  
>java Main people.txt films.txt commands.txt output.txt
```

There are three types of data input files and one output file. All the file names will be taken as program arguments. The format of each file is given below.

3.1.2 People File

There are five different recording samples in this file. These are:

```
DIRECTOR:<tab><ID><tab><NAME><tab><SURNAME><tab><COUNTRY><tab><AGENT>  
WRITER:<tab><ID><tab><NAME><tab><SURNAME><tab><COUNTRY><tab><TYPE>  
ACTOR:<tab><ID><tab><NAME><tab><SURNAME><tab><COUNTRY><tab><HEIGHT>  
CHILDACTOR:<tab><ID><tab><NAME><tab><SURNAME><tab><COUNTRY><tab><AGE>  
STUNTPERFORMER:<tab><ID><tab><NAME><tab><SURNAME><tab><COUNTRY><tab><HEIGHT><tab>  
<ACTOR1ID,...,ACTORnID>  
USER:<tab><ID><tab><NAME><tab><SURNAME><tab><COUNTRY>
```

An exemplar people file is shown in Figure 2. As shown in Figure 2, the person type is specified at the beginning of each line, and then attributes are given according to the person type, separated by tab characters.

```
1 Actor: 352 Alessio Boni Italy 187
2 Actor: 353 Luigi Cascio Italy 175
3 ChildActor: 354 Noah Schnapp USA 17
4 ChildActor: 355 Jacob Tremblay Canada 15
5 ChildActor: 356 Millie Brown Spain 17
6 Director: 357 Marco Giordana Italy Tullio
7 Writer: 358 Sandro Petraglia Italy Drama
8 Writer: 359 Stefano Rulli Italy Romance
9 StuntPerformer: 360 Dar Robinson UK 175 353,386,393
10 Director: 361 Ramin Bahrani USA Ramin
11 Writer: 362 Jenni Jenkins USA Drama
12 Actor: 363 Werner Herzog Germany 185
13 ChildActor: 364 Jaeden Martell Philadelphia 18
14 Director: 366 Martin Dennis UK Perkins
15 Writer: 367 Steven Moffat UK Comedy
16 Actor: 368 Jack Davenport UK 188
17 Actor: 369 Ben Miles UK 179
18 Actor: 370 Richard Coyle UK 183
19 ChildActor: 371 Madison Reyes NewYork 16
20 ChildActor: 372 Isla Johnston UK 14
21 ChildActor: 373 Sunny Pawar Indian 13
22 StuntPerformer: 374 Hal Needham USA 180 404,369
23 Director: 375 Emad Burnat Palestine Burnat
24 Director: 376 Guy Davidi Israel Burnat
```

Figure 2: Sample people file

3.1.3 Films File

Since there are four different film types in this system, there are also four different record samples in this file.

FeatureFilm:<tab><ID><tab><TITLE><tab><LANGUAGE><tab><DIRECTOR1ID,...,DIRECTORnID>
<tab><LENGTH><tab><COUNTRY><tab><PERFORMER1ID,...,PERFORMERnID><tab><GENRE1,...,
GENREn><tab><RELEASEDATE><tab><WRITER1ID,...,WRITERnID><tab><BUDGET>

ShortFilm:<tab><ID><tab><TITLE><tab><LANGUAGE><tab><DIRECTOR1ID,...,DIRECTORnID>
<tab><LENGTH><tab><COUNTRY><tab><PERFORMER1ID,...,PERFORMERnID><tab><GENRE1,...,
GENREn><tab><RELEASEDATE><tab><WRITER1ID,...,WRITERnID>

Documentary:<tab><ID><tab><TITLE><tab><LANGUAGE><tab><DIRECTOR1ID,...,<DIRECTORnID>
<tab><LENGTH><tab><COUNTRY><tab><PERFORMER1ID,...,PERFORMERnID><tab><RELEASEDATE>

TVSeries:<tab><ID><tab><TITLE><tab><LANGUAGE><tab><DIRECTOR1ID,...,DIRECTORnID>
<tab><LENGTH><tab><COUNTRY><tab><PERFORMER1ID,...,PERFORMERnID><tab><GENRE1,...,
GENREn><tab><WRITER1ID,...,WRITERnID><tab><STARTDATE><tab><ENDDATE><tab><SEASONS>
<tab><EPISODES>

An exemplar film file is given in Figure 3. In this file, the film type is specified at the beginning of each line, and then attributes are given according to the film type, separated by tab characters.

1	FeatureFilm:	100	La meglio gioventu	Italian	357	180	Italy	352,353,354,355,356	Drama,Romance	22.06.2003	358,359	240000		
2	ShortFilm:	101	Plastic Bag	English	361	18	USA	363,364	Drama	07.09.2009	362			
3	TVSeries:	102	Coupling	English	366	30	UK	368,369,370,371,372,373	Comedy	367	12.05.2000	14.06.2004	4	28
4	Documentary:	103	5 Broken Cameras	Hebrew	375,376	94	Palestine	377,378,379		20.02.2013				
5	FeatureFilm:	104	Pulp Fiction	English	381	154	USA	383,384,385,386,387,388	Crime,Thriller	14.04.1994	382	8000000		
6	FeatureFilm:	105	Rear Window	English	390	112	USA	393,394,395	Mystery,Thriller	01.04.1956	391,392	1000000		
7	TVSeries:	106	Lost	English	397,398	42	USA	404,405,406,407,408,409,410	Adventure,Drama,Fantasy	399,400,401,402,403	22.09.2004	23.05.2010	6	121
8	Documentary:	107	Gelibolu	Turkish	412	114	Turkey	413,414,415		18.03.2005				
9	TVSeries:	108	Friends	English	417	22	USA	421,422,423,424,425,426	Comedy,Romance	418,419,420	22.09.1994	06.05.2004	10	236
10	FeatureFilm:	109	Before Sunrise	English	428	105	USA	430,431	Drama,Romance	19.05.1995	429	2500000		
11	FeatureFilm:	110	Annie Hall	English	433	93	USA	435,436	Comedy,Drama,Romance	20.04.1977	434	4000000		
12	FeatureFilm:	111	Biyeolhan Geori	Korean	438	141	South Korea	440,441,442	Action,Crime,Thriller	15.06.2006	439	4700000		
13	FeatureFilm:	112	Silver Linings Playbook	English	444	122	USA	446,447,448,449	Comedy,Drama,Romance	04.01.2013	445	21000000		
14	FeatureFilm:	113	Amelie	French	451	122	France	453,454	Comedy,Romance	25.04.2001	452	10000000		
15	FeatureFilm:	114	The Godfather	English	456	175	USA	458,459,460,461	Crime,Drama	05.10.1973	457	6000000		

Figure 3: Sample films file

3.1.4 Commands File

All data input files (people and films) will be processed according to the commands which will be given in a commands file. The command file contains 12 types of commands whose definitions and formats are given below.

1. A user can rate a film so that film will be saved to his/her rate list. Rating score must be between 1 and 10 integers.

RATE<tab><USERID><tab><FILMID><tab><RATINGPOINT>

2. It's possible to add a new Feature Film to the system.

ADD<tab>**FEATUREFILM**<tab><ID><tab><TITLE><tab><LANGUAGE><tab><DIRECTOR1ID,...,DIRECTORnID><tab><LENGTH><tab><COUNTRY><tab><PERFORMER1ID,...,PERFORMERnID><tab><GENRE1,...,GENREn><tab><RELEASEDATE><tab><WRITER1ID,...,WRITERnID><tab><BUDGET>

3. Details of a film are displayed by using below command.

VIEWFILM<tab><FILMID>

4. A user can list all films which he/she rated so far.

LIST<tab>**USER**<tab><USERID><tab>**RATES**

5. A user can edit a film which he/she rated before.

EDIT<tab>**RATE**<tab><USERID><tab><FILMID><tab><NEWRATINGPOINT>

6. A user can remove one of his/her rated films.

REMOVE<tab>**RATE**<tab><USERID><tab><FILMID>

7. List all the TV Series in the system.

LIST<tab>**FILM**<tab>**SERIES**

8. List all the films from a specified country.

LIST<tab>**FILMS**<tab>**BY**<tab>**COUNTRY**<tab><COUNTRY>

9. List all the films released before a specified year.

LIST<tab>**FEATUREFILMS**<tab>**BEFORE**<tab><YEAR>

10. List all the films released after a specified year.

LIST<tab>FEATUREFILMS<tab>AFTER<tab><YEAR>

11. List all the films in descending order and categorized according to film rating degrees.

LIST<tab>FILMS<tab>BY<tab>RATE<tab>DEGREE

12. List all the artists from a specified country and display in a categorized order.

LIST<tab>ARTISTS<tab>FROM<tab><COUNTRY>

```
1 RATE      470 113 9
2 RATE      470 113 9
3 RATE      454 113 8
4 ADD FEATUREFILM 115 Fight_Club English 463 139 USA 466,467,468 Drama 10.12.1999 464,465 63000000
5 ADD FEATUREFILM 115 Night_Club English 463 139 USA 466,467,468 Drama 10.12.1999 464,465 63000000
6 ADD FEATUREFILM 116 Right_Club English 463 139 USA 466,467,999 Drama 10.12.1999 464,465 63000000
7 RATE      475 115 10
8 RATE      477 115 9
9 RATE      480 111 9
10 RATE     477 111 7
11 RATE     476 114 10
12 RATE     474 114 10
13 RATE     473 114 9
14 RATE     472 112 8
15 RATE     472 103 8
16 RATE     472 101 6
17 RATE     471 102 10
18 RATE     479 102 9
19 RATE     478 100 9
20 RATE     478 109 9
21 RATE     476 104 9
22 RATE     473 106 9
23 RATE     472 106 8
24 RATE     471 106 7
25 RATE     478 106 10
26 VIEWFILM    115
27 VIEWFILM    101
28 RATE      470 115 9
29 RATE      479 115 8
30 RATE      478 115 8
31 RATE      470 108 8
32 VIEWFILM    115
33 LIST      USER      470 RATES
34 LIST      USER      476 RATES
35 LIST      USER      444 RATES
36 EDIT      RATE      470 115 10
37 EDIT      RATE      470 109 10
38 VIEWFILM    103
39 LIST      ARTISTS FROM Turkey
40 EDIT      RATE      470 222 10
41 LIST      FEATUREFILMS AFTER 1995
42 EDIT      RATE      475 109 10
43 VIEWFILM    108
44 RATE      470 100 10
45 RATE      473 113 8
46 REMOVE    RATE      470 108
47 VIEWFILM    108
48 REMOVE    RATE      470 109
49 LIST      USER      470 RATES
50 LIST      FILM      SERIES
51 VIEWFILM    106
52 LIST      FILMS BY COUNTRY Japan
53 LIST      FILMS BY COUNTRY USA
54 LIST      FEATUREFILMS BEFORE 1995
55 LIST      FEATUREFILMS BEFORE 1920
56 LIST      FILMS BY RATE DEGREE
```

Figure 4: Sample commands file

3.1.5 Output File

The output of the commands will be printed to the specified output file. Each command's output will include the command itself as read from the command file and the result (error message if necessary) of its execution. The general format of the output file is shown below:

```
<COMMAND>
<NEWLINE>
<RESULT>
<NEWLINE>
<----->
```

Detailed format of <RESULT> (mentioned above in the general format) output for each command type is given below (WS represents Whitespace).

1. *Film rated successfully*

Film type:< WS><FILMTYPE>

Film title:< WS><TITLE>

If there is not any user or film with specified ID the <RESULT> should be as follows:

Command Failed

User ID:< WS><USERID>

Film ID:< WS><FILMID>

If the specified film was already rated by the given user, then there should be a warning message as follows:

This film was earlier rated

2. *FeatureFilm added successfully*

Film ID:< WS><FILMID>

Film title:< WS><TITLE>

If there is already a film with specified <FILMID> or if there is not any specified director, writer or performer in the system, the <RESULT> should be as follows:

Command Failed

Film ID:< WS><FILMID>

Film title:< WS><TITLE>

3. If specified film is Feature Film or Short Film the result will be as follows:

<TITLE>< WS>(<RELEASEDATE>)

<GENRE>

Writers:< WS><NAME>< WS><SURNAME>

Directors:< WS><NAME>< WS><SURNAME>

Stars:< WS><NAME>< WS><SURNAME>

<RATINGS>/10 from<VOTECOUNT>users

If specified film is Documentary; since a documentary doesn't have writers and genre in the system, the result will be as follows:

<TITLE><WS>(<RELEASEDATE>)
Directors:<WS><NAME><WS><SURNAME>
Stars:<WS><NAME><WS><SURNAME>
<RATINGS>/10 from<VOTECOUNT>users

If specified film is TV Series the result will be as follows:

<TITLE><WS>(<STARTDATE>-<ENDDATE>)
<SEASONS><WS>seasons,<WS><EPISODES><WS>episodes
<GENRE>
Writers:<WS><NAME><WS><SURNAME>
Directors:<WS><NAME><WS><SURNAME>
Stars:<WS><NAME><WS><SURNAME>
<RATINGS>/10 from<VOTECOUNT>users

If there is not any film with specified <FILMID> the <RESULT> should be as follows:

Command Failed
Film ID:<WS><FILMID>

If there is not any rating votes for that film, then below warning message should be printed:

Awaiting for votes

4. <TITLE>:<WS> <RATINGSCORE>

If there is not any ratings of the specified user, then a warning message will be printed to the output file as follows:

There is not any ratings so far

If there is not any user with specified <USERID> the <RESULT> should be as follows:

Command Failed
User ID:<WS><USERID>

5. *New ratings done successfully*

Film title:<WS><TITLE>
Your rating:<WS><NEWRATINGSCORE>

If there is not any user or film with specified IDs and if the user has no rating score for the specified film, then the <RESULT> should be as follows:

Command Failed
User ID:<WS><USERID>
Film ID:<WS><FILMID>

6. *Your film rating was removed successfully*

Film title:<WS><TITLE>

If there is not any user or film with specified IDs and if the user has no rating score for the specified film, then the <RESULT> should be as follows:

Command Failed

User ID:<WS><USERID>

Film ID:<WS><FILMID>

7. *<TITLE><WS>(<STARTDATE>-<ENDDATE>)
<SEASONS><WS>seasons and<WS><EPISODES><WS>episodes*

If there is not any TV Series in the system, then a warning message will be printed to the output file as follows:

No result

8. *Film title:<WS><TITLE>
<LENGTH><WS>min
Language:<WS><LANGUAGE>*

If there is not any film for the specified country in the system, then a warning message will be printed to the output file as follows:

No result

9. *Film title:<TITLE><WS>(<RELEASEDATE>)
<LENGTH><WS>min
Language:<WS><LANGUAGE>*

If there is not any film released before the specified date in the system, then a warning message will be printed to the output file as follows:

No result

10. *Film title:<TITLE><WS>(<RELEASEDATE>)
<LENGTH><WS>min
Language:<WS><LANGUAGE>*

If there is not any film released after specified date in the system, then a warning message will be printed to the output file as follows:

No result

11. *FeatureFilm:
<TITLE><WS>(<RELEASEDATE>) Ratings:<WS><RATINGS>/10 from<VOTECOUNT>users*

*ShortFilm:
<TITLE><WS>(<RELEASEDATE>) Ratings:<WS><RATINGS>/10 from<VOTECOUNT>users*

*Documentary:<TITLE><WS>(<RELEASEDATE>) Ratings:<WS><RATINGS>/10
from<VOTECOUNT>users*

*TVSeries:
<TITLE><WS>(<STARTDATE>-<ENDDATE>) Ratings:<WS><RATINGS>/10
from<VOTECOUNT>users*

All the results should be printed in descending order.

If there is not any result for a category, then a warning message will be printed to the output file for that category as follows:

No result

12. *Directors:*

<NAME><WS><SURNAME><WS><AGENT>

Writers:

<NAME><WS><SURNAME><WS><TYPE>

Actors:

<NAME><WS><SURNAME><WS><HEIGHT><WS>cm

ChildActors:

<NAME><WS><SURNAME><WS><AGE>

StuntPerformers:

<NAME><WS><SURNAME><WS><HEIGHT><WS>cm

If there is not any result for a category, then a warning message will be printed to the output file for that category as follows:

No result

According to these definitions a sample output file is given in Figure 5 and Figure 6. This output file is not complete. The rest will be provided to you as sample input and output on the Piazza page. Further examples which give more details will be provided at the course's Piazza page.

1 RATE 470 113 9	40 -----
2	41 RATE 475 115 10
3 Film rated successfully	42
4 Film type: FeatureFilm	43 Film rated successfully
5 Film title: Amelie	44 Film type: FeatureFilm
6	45 Film title: Fight_Club
7	46
8 RATE 470 113 9	47 -----
9	48 RATE 477 115 9
10 This film was earlier rated	49
11	50 Film rated successfully
12	51 Film type: FeatureFilm
13 RATE 454 113 8	52 Film title: Fight_Club
14	53
15 Command Failed	54
16 User ID: 454	55 RATE 480 111 9
17 Film ID: 113	56
18	57 Film rated successfully
19	58 Film type: FeatureFilm
20 ADD FEATUREFILM 115 Fight_Club English 463 139 USA 466,467 Drama 10.12.1999 464,465 63000000	59 Film title: Biyeolhan_Geori
21	60
22 FeatureFilm added successfully	61 -----
23 Film ID: 115	62 RATE 477 111 7
24 Film title: Fight_Club	63
25	64 Film rated successfully
26	65 Film type: FeatureFilm
27 ADD FEATUREFILM 115 Night_Club English 463 139 USA 466,467,468 Drama 10.12.1999 464,465 63000000	66 Film title: Biyeolhan_Geori
28	67
29 Command Failed	68
30 Film ID: 115	69 RATE 476 114 10
31 Film title: Night_Club	70
32	71 Film rated successfully
33	72 Film type: FeatureFilm
34 ADD FEATUREFILM 116 Right_Club English 463 139 USA 466,467,999 Drama 10.12.1999 464,465 63000000	73 Film title: The_Godfather
35	74
36 Command Failed	75 -----
37 Film ID: 116	76 RATE 474 114 10
38 Film title: Right_Club	77
39	78 Film rated successfully
	79 Film type: FeatureFilm
	80 Film title: The_Godfather
	...

Figure 5: Sample output file part1

82	-----	124	-----
83	RATE 473 114 9	125	RATE 478 100 9
84		126	
85	Film rated successfully	127	Film rated successfully
86	Film type: FeatureFilm	128	Film type: FeatureFilm
87	Film title: The_Godfather	129	Film title: La_meglio_gioventu
88		130	
89	-----	131	-----
90	RATE 472 112 8	132	RATE 478 109 9
91		133	
92	Film rated successfully	134	Film rated successfully
93	Film type: FeatureFilm	135	Film type: FeatureFilm
94	Film title: Silver_Linings_Playbook	136	Film title: Before_Sunrise
95		137	
96	-----	138	-----
97	RATE 472 103 8	139	RATE 476 104 9
98		140	
99	Film rated successfully	141	Film rated successfully
100	Film type: Documentary	142	Film type: FeatureFilm
101	Film title: 5_Broken_Cameras	143	Film title: Pulp_Fiction
102		144	
103	-----	145	-----
104	RATE 472 101 6	146	RATE 473 106 9
105		147	
106	Film rated successfully	148	Film rated successfully
107	Film type: ShortFilm	149	Film type: TVSeries
108	Film title: Plastic_Bag	150	Film title: Lost
109		151	
110	-----	152	-----
111	RATE 471 102 10	153	RATE 472 106 8
112		154	
113	Film rated successfully	155	Film rated successfully
114	Film type: TVSeries	156	Film type: TVSeries
115	Film title: Coupling	157	Film title: Lost
116		158	
117	-----	159	-----
118	RATE 479 102 9	160	RATE 471 106 7
119		161	
120	Film rated successfully	162	Film rated successfully
121	Film type: TVSeries	163	Film type: TVSeries
122	Film title: Coupling	164	Film title: Lost

Figure 6: Sample output file part 2

3.2 Report

The structure of report is described below:

- Cover Page
- Class Diagram and Solution, describe details of your solution, stating its advantages and disadvantages technically. Draw class diagram. Show attributes and method names of each class in your diagram.
- Comments, give feedback about problem, problem description, and solution constraints.
- References, give the references you used throughout your work at the end of your report.

3.3 Constraints

1. The methods' and attributes' names should be satisfied the most common naming conventions in Java.
2. You should model entities of the system with classes.
3. Your design will be graded. You are expected to propose a suitable design for the problem.
4. You should use inheritance mechanism and correct access modifiers.
5. All the input files and output file will be taken as command line arguments.

3.4 Submit Format

- File hierarchy must be zipped before submitted (Not .rar, only .zip files are supported by the system)

- $\langle studentid \rangle.zip(example : 1234567.zip)$

- [src]
 - Main.java
 - *.java
- [report]
 - report.pdf
 - *.jpg or *.jpeg

4 Grading Policy

Task	Point
Compiled	10
Report	15
Code design, clean and readable code, comments	15
Output of program	60
Total	100

5 Notes

- The assignment must be original, individual work. Downloaded or modified source codes will be considered as cheating. Also the students who share their works will be punished in the same way.
- We will be using the Measure of Software Similarity (MOSS) to identify cases of possible plagiarism. Our department takes the act of plagiarism very seriously. Those caught plagiarizing (both originators and copiers) will be sanctioned.
- Please do not miss that the name of the input and output files will be fixed. Use the file names as indicated in the leaflet. File names with different names will not be evaluated.
- You can ask your questions through course's piazza group and you are supposed to be aware of everything discussed in the piazza group. General discussion of the problem is allowed, but DO NOT SHARE answers, algorithms, source codes and reports .
- Ignore the cases which are not stated in this assignment and do not ask questions on Piazza for such extreme cases.
- Don't forget to write comments of your codes when necessary.
- The names of classes', attributes' and methods' should obey to Java naming convention.
- Do not submit your project without first compiling it on dev machine.
- Save all work until the assignment is graded.
- Do not miss the deadline. Submission will be end at 16/04/2021 23:59, the system will be open until 23:59:59. The problem about submission after 23:59 will not be considered.

- There will be again 3 days extensions (each day degraded by 10 points) in this project.
- Your grades will be announced within 15 days at the latest. Objections about your grades can be made within specified days given by TAs. Objections made outside of the specified days will not be accepted.