

Main.Java

```
package JDBC;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        MovieDAO mdao=new MovieDAO();
        UserDAO udao=new UserDAO();
        ReviewDAO redao=new ReviewDAO();
        RatingDAO radao=new RatingDAO();
        java.util.List<Movie> mvs=mdao.listMovie();

        while (true) {
            System.out.println("\npls enter your option");
            System.out.println("1. View, 2. edit, -1. Exit");
            int opt = sc.nextInt();
            if (opt == 1) {
                System.out.println("\nYou have the following options. Please enter
your choice.");
                System.out.println("1. view list of movies from a genre\n"
                    +"2. view list of top 10 movies overall\n"
                    +"3. view list of top 10 movies from a genre\n"
                    +"4. search for a movie by name \n"
                    +"5. view all movies");
                int opt1=sc.nextInt();

                //view list of movies from a genre
                if(opt1==1){
                    System.out.println("pls enter genre: ");
                    String genre=sc.next();
                    mdao.titleByGenre(genre);
                }

                //view list of top 10 movies overall
                else if(opt1==2){
                    mdao.getTopMovies();
                }
                //view list of top 10 movies from a genre
                else if(opt1==3){
                    System.out.println("pls enter genre: ");
```

```

        String genre=sc.next();
        mdao.getTopMoviesByGenre(genre);
    }
    //search for a movie by name
    else if(opt1==4){
        System.out.println("pls enter movie name:");
        String title=sc.next();
        //mdao.movieDetails(title);
        java.util.List<Movie> mv=mdao.listMovie(title);
        System.out.println(mv);
    }
    //view all movies
    else if(opt1==5){
        mvs=mdao.listMovie();
        System.out.println(mvs);
    }
    else System.out.println("pls enter a num between 1-5");
}

```

```

else if (opt == 2) {
    int uid=0;
    System.out.println("\nTo edit u need to be a user.\n 1. signin or
2.signup");

    int sign=sc.nextInt();
    if(sign==1){
        System.out.println("enter email-id: ");
        String email=sc.next();
        uid=udao.getUserId(email);
    }
    else if(sign==2){
        System.out.println("enter id: ");
        uid=sc.nextInt();
        System.out.println("enter email-id: ");
        String email=sc.next();
        System.out.println("enter name: ");
        String name=sc.next();
        User u=new User(uid,name,email);
        System.out.println(udao.usave(u)+" records inserted");
    }

    System.out.println("\n1.add a new movie entry\n"
        + "2.edit an existing movie entry\n"
        + "3.add review to a movie\n"

```

```

        + "4.rate a movie");
int eopt=sc.nextInt();
if(eopt==1){ //add a new movie entry
    String title,summary,cast,genre,review;
    sc.nextLine();
    System.out.println("pls enter movie title:");
    title=sc.nextLine();
    System.out.println("pls enter summary: ");
    summary=sc.nextLine();
    System.out.println("pls enter cast:");
    cast=sc.nextLine();
    System.out.println("pls enter genre:");
    genre=sc.nextLine();
    float rate=0;
    Movie m=new Movie(title,summary,cast,genre,rate,uid);
    System.out.println(mdao.insert(m)+" movie record inserted
successfully");
}
else if(eopt==2){ //edit an existing movie entry
    sc.nextLine();
    String title,summary,cast,genre;
    System.out.println("pls enter movie title:");
    title=sc.nextLine();
    System.out.println("pls enter summary: ");
    summary=sc.nextLine();
    System.out.println("pls enter cast:");
    cast=sc.nextLine();
    System.out.println("pls enter genre:");
    genre=sc.nextLine();
    float rate=0;
    Movie m=new Movie(title,summary,cast,genre,rate,uid);
    System.out.println(mdao.update(m)+" record updated");
//
System.out.println(mdao.updateMovie(title,summary,cast,genre,uid)+" record is updated");
}
else if(eopt==3){//add review to a movie
    sc.nextLine();
    System.out.println("pls enter Movie name:");
    String movie = sc.nextLine();
    System.out.print("Add Review: ");
    String review = sc.nextLine();
    System.out.println(redao.addReview(movie,review,uid)+"
review added");
}

```

```

        else if(eopt==4){//rate a movie
            sc.nextLine();
            System.out.println("pls enter movie name:");
            String movie=sc.nextLine();
            System.out.println("pls enter rating:");
            int rating=sc.nextInt();
            System.out.println(radao.addRating(movie,rating,uid)+"
rating added");
        }
        else System.out.println("enter a num between 1-4");
    }
    else if (opt == -1)break;
    else System.out.println("pls enter from {1,2,-1}");
}
}
}

```

Movie.Java

```

package JDBC;

public class Movie {
    int movie_id;
    String title;
    String summary;
    String cast;
    String genre;
    float avgRate;
    int user_id;
    public int getMovie_id() {
        return movie_id;
    }
    public void setMovie_id(int movie_id) {
        this.movie_id = movie_id;
    }
    public String getTitle() {
        return title;
    }
    public void setTitle(String title) {
        this.title = title;
    }
}

```

```

    }
    public String getSummary() {
        return summary;
    }
    public void setSummary(String summary) {
        this.summary = summary;
    }
    public String getCast() {
        return cast;
    }
    public void setCast(String cast) {
        this.cast = cast;
    }
    public String getGenre() {
        return genre;
    }
    public void setGenre(String genre) {
        this.genre = genre;
    }
    public int getUser_id() {
        return user_id;
    }
    public void setUser_id(int user_id) {
        this.user_id = user_id;
    }
}

    public float getAvgRate() {
        return avgRate;
    }
    public void setAvgRate(float avgRate) {
        this.avgRate = avgRate;
    }
}
@Override
public String toString() {
    return "\nMovie [movie_id=" + movie_id + ", title=" + title
        + ", summary=" + summary + ", cast=" + cast + ", genre="
        + genre + ", avgRate=" + avgRate + ", user_id=" + user_id + "];"
}

public Movie() {}
public Movie(int movie_id, String title, String summary, String cast,
    String genre, float avgRate, int user_id) {
    super();
}

```

```

        this.movie_id = movie_id;
        this.title = title;
        this.summary = summary;
        this.cast = cast;
        this.genre = genre;
        this.avgRate = avgRate;
        this.user_id = user_id;
    }
    public Movie(String title, String summary, String cast, String genre,
                  float avgRate, int user_id) {
        super();
        this.title = title;
        this.summary = summary;
        this.cast = cast;
        this.genre = genre;
        this.avgRate = avgRate;
        this.user_id = user_id;
    }
}

```

User.Java

```

package JDBC;

public class User {
    int user_id;
    String name;
    String email;

    public int getUser_id() {
        return user_id;
    }
    public void setUser_id(int user_id) {
        this.user_id = user_id;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}

```

```

    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }

    @Override
    public String toString() {
        return "User [user_id=" + user_id + ", name=" + name + ", email="
            + email + "]";
    }

    public User() {}
    public User(int user_id, String name, String email) {
        this.user_id = user_id;
        this.name = name;
        this.email = email;
    }
}

```

Review.Java

```
package JDBC;
```

```

public class Review {
    int review_id;
    int movie_id;
    String review;
    int user_id;

    public int getUser_id() {
        return user_id;
    }
    public void setUser_id(int user_id) {
        this.user_id = user_id;
    }
    public int getReview_id() {
        return review_id;
    }
    public void setReview_id(int review_id) {
        this.review_id = review_id;
    }
}

```

```

    public int getMovie_id() {
        return movie_id;
    }
    public void setMovie_id(int movie_id) {
        this.movie_id = movie_id;
    }
    public String getReview() {
        return review;
    }
    public void setReview(String review) {
        this.review = review;
    }

    @Override
    public String toString() {
        return "Review [review_id=" + review_id + ", movie_id=" + movie_id
            + ", review=" + review + ", user_id=" + user_id + "]";
    }

    public Review(int review_id, int movie_id, String review, int user_id) {
        super();
        this.review_id = review_id;
        this.movie_id = movie_id;
        this.review = review;
        this.user_id = user_id;
    }
    public Review() {}
}

```

Rating.Java

```

package JDBC;

public class Rating {
    int rating_id;
    int movie_id;
    int rating;
    int user_id;
    public int getRating_id() {
        return rating_id;
    }
    public void setRating_id(int rating_id) {
        this.rating_id = rating_id;
    }
}

```



```

    }
    public int getMovie_id() {
        return movie_id;
    }
    public void setMovie_id(int movie_id) {
        this.movie_id = movie_id;
    }
    public int getRating() {
        return rating;
    }
    public void setRating(int rating) {
        this.rating = rating;
    }
    public int getUser_id() {
        return user_id;
    }
    public void setUser_id(int user_id) {
        this.user_id = user_id;
    }
    @Override
    public String toString() {
        return "Rating [rating_id=" + rating_id + ", movie_id=" + movie_id
            + ", rating=" + rating + ", user_id=" + user_id + "]";
    }

    public Rating() {}
    public Rating(int rating_id, int movie_id, int rating, int user_id) {
        this.rating_id = rating_id;
        this.movie_id = movie_id;
        this.rating = rating;
        this.user_id = user_id;
    }

}

```

MovieDAO.Java

```

package JDBC;
import java.sql.*;
import java.util.*;

```

```

public class MovieDAO {
    Connection con;
    Statement stmt;
    public MovieDAO(){
        con=null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/movieReview", "root",
"@pikachu");
            stmt = con.createStatement();
        }
        catch(Exception e)
        {}
    }

    public MovieDAO(String driver, String URL, String user, String pass){
        con=null;
        try {
            Class.forName(driver);
            con = DriverManager.getConnection(
                URL, user, pass);
            stmt = con.createStatement();
        }
        catch(Exception e)
        {}
    }

    public void titleByGenre(String g){
        try {
            ResultSet rs = stmt.executeQuery("select title from movie where
genre='"+g+"'");
            System.out.println("Movies in "+g+" genre: ");
            while(rs.next()){
                System.out.println(rs.getString(1));
            }
        } catch (SQLException e1) {
        }
    }

    public void getTopMovies(){
        try {

```

```

        ResultSet rs = stmt.executeQuery("select title from movie order by
average_rating desc limit 10");
        System.out.println("Top 10 movie overall: ");
        while(rs.next()){
            System.out.println(rs.getString(1));
        }
    } catch (SQLException e1) {
    }

}

public void getTopMoviesByGenre(String g){
    try {
        ResultSet rs = stmt.executeQuery("select title from movie where
genre='"+g+"'order by average_rating desc limit 10");
        System.out.println("Top 10 movie in '"+g+"' genre: ");
        while(rs.next()){
            System.out.println(rs.getString(1));
        }
    } catch (SQLException e1) {
    }

}

public List<Movie> listMovie(String s){
    List<Movie> mvs=new ArrayList<>();
    try {
        ResultSet rs = stmt.executeQuery("select movie_id from movie where
title='"+s+"'");
        rs.next();
        int mid=rs.getInt(1);
        rs = stmt.executeQuery("select * from movie where movie_id="+mid+"");
        while (rs.next())
            mvs.add(new
Movie(rs.getInt(1),rs.getString(2),rs.getString(3),rs.getString(4),rs.getString(5),rs.getFloat(6),rs.g
etInt(7)));
    } catch (Exception e) {}

    return mvs;
}

public List<Movie> listMovie(){
    List<Movie> mvs=new ArrayList<>();
    try {

```

```

        ResultSet rs = stmt.executeQuery("select * from movie");
        while (rs.next())
            mvs.add(new
Movie(rs.getInt(1),rs.getString(2),rs.getString(3),rs.getString(4),rs.getString(5),rs.getFloat(6),rs.g
etInt(7)));
    } catch (Exception e) {}

    return mvs;
}

public int insert(Movie m){
    int s=0;
    try {
        s = stmt.executeUpdate("insert into
movie(title,summary,cast,genre,average_rating,user_id) values
("+m.getTitle()+",""+m.getSummary()+",""+m.getCast()+",""+m.getGenre()+",""+m.getAvgRate()
+","+m.getUser_id()+")");

    } catch (Exception e) {}
    return s;
}

public int update(Movie m){
    int s=0;
    try {
        ResultSet rs = stmt.executeQuery("select movie_id from movie where
title="+m.getTitle()+""");
        rs.next();
        int mid=rs.getInt(1);
        s = stmt.executeUpdate("upadte movie set
summary="+m.getSummary()+",cast="+m.getCast()+",genre="+m.getGenre()+"" where
tilte="+m.getTitle()+""");

    } catch (Exception e) {}
    return s;
}

public int updateMovie(String title,String summary,String cast,String genre,int uid){
    int s=0;
    try {
        ResultSet rs = stmt.executeQuery("select movie_id from movie where
title="+title+""");

        rs.next();
        int mid=rs.getInt(1);

```

```

        s = stmt.executeUpdate("upadte movie set
summary='"+summary+"',cast='"+cast+"',genre='"+genre+"' where movie_id='"+mid+"'");

    } catch (Exception e) {}
    return s;
}

public void finalise(){
    try {
        con.close();
    } catch (SQLException e) {}
}
}

```

UserDAO.Java

```

package JDBC;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.List;

public class UserDAO {
    Connection con;
    Statement stmt;
    public UserDAO(){
        con=null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/movieReview", "root",
"@pikachu");
            stmt = con.createStatement();
        }
        catch(Exception e)
        {}
    }
}

```

```

public UserDAO(String driver, String URL, String user, String pass){
    con=null;
    try {
        Class.forName(driver);
        con = DriverManager.getConnection(
            URL, user, pass);
        stmt = con.createStatement();
    }
    catch(Exception e)
    {}
}

public int usave(User u)
{
    int s=0;
    try {
        s = stmt.executeUpdate
            ("insert into users values("+u.getUser_id()+",
"+u.getName() + ", "+u.getEmail()+ ")");
    } catch (SQLException e1) {
    }
    return s;
}

public int getUserId(String email){
    int id=0;
    try {
        ResultSet rs = stmt.executeQuery("select user_id from users where
email='"+email+"'");
        rs.next();
        id=rs.getInt(1);
    } catch (Exception e) {}
    return id;
}

public List<User> listUser(){
    List<User> users=new ArrayList<>();
    try {
        ResultSet rs = stmt.executeQuery("select * from user");
        while (rs.next())
            users.add(new User(rs.getInt(1),rs.getString(2),rs.getString(3)));
    } catch (Exception e) {}

    return users;
}

```

```

    }

    public void finalise(){
        try {
            con.close();
        } catch (SQLException e) {}
    }
}

```

ReviewDAO.Java

```

package JDBC;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class ReviewDAO {
    Connection con;
    Statement stmt;
    public ReviewDAO(){
        con=null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/movieReview", "root",
"@pikachu");
            stmt = con.createStatement();
        }
        catch(Exception e)
        {}
    }
    public ReviewDAO(String driver, String URL, String user, String pass){
        con=null;
        try {
            Class.forName(driver);
            con = DriverManager.getConnection(
                URL, user, pass);
            stmt = con.createStatement();
        }
        catch(Exception e)
    }
}

```

```

        {}
    }
    public int addReview(String movie,String review,int uid){
        int s=0;
        try {
            ResultSet rs = stmt.executeQuery("select movie_id from movie where
title='"+movie+"'");
            rs.next();
            int mid=rs.getInt(1);
            s = stmt.executeUpdate("insert into review(movie_id,review,user_id)
values ('"+mid+"','"+review+"','"+uid+"')");
        } catch (Exception e) {}
        return s;
    }
    public void finalise(){
        try {
            con.close();
        } catch (SQLException e) {}
    }
}

```

RatingDAO.Java

```

package JDBC;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class RatingDAO {
    Connection con;
    Statement stmt;
    public RatingDAO(){
        con=null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/movieReview", "root",
"@pikachu");
            stmt = con.createStatement();
        }
    }
}

```



```

        catch(Exception e)
        {}
    }
    public RatingDAO(String driver, String URL, String user, String pass){
        con=null;
        try {
            Class.forName(driver);
            con = DriverManager.getConnection(
                URL, user, pass);
            stmt = con.createStatement();
        }
        catch(Exception e)
        {}
    }

    public int addRating(String movie,int rating,int uid){
        int s=0;
        try {
            ResultSet rs = stmt.executeQuery("select movie_id from movie where
title='"+movie+"'");
            rs.next();
            int mid=rs.getInt(1);
            s = stmt.executeUpdate("insert into rating(movie_id,rating,user_id) values
('"+mid+"','"+rating+"','"+uid+"')");

            rs = stmt.executeQuery("select avg(rating) from rating where
movie_id='"+mid+"'");
            rs.next();
            float rate=rs.getFloat(1);
            int s1 = stmt.executeUpdate("update movie set average_rating="+rate+"
where movie_id='"+mid+"'");

        } catch (Exception e) {}
        return s;
    }
    public void finalise(){
        try {
            con.close();
        } catch (SQLException e) {}
    }
}

```

MySQL database=movieReview,tables=users,movie,review,rating

```
create database movieReview;
```

```
use movieReview;
```

```
Create table users(
```

```
    user_id int Auto_increment NOT NULL Primary Key,
```

```
    name varchar(100) Not Null,
```

```
    email varchar(100) Not Null Unique
```

```
);
```

```
Create table movie (
```

```
    movie_id INT Auto_increment NOT NULL Primary Key,
```

```
    title varchar(100) Not Null,
```

```
    summary varchar(500),
```

```
    cast varchar(200),
```

```
    Genre varchar(200),
```

```
    Average_rating float,
```

```
    user_id int, FOREIGN KEY (user_id) REFERENCES users(user_id)
```

```
);
```

```
Create table review (
```

```
    review_id INT Auto_increment NOT NULL Primary Key,
```

```
    movie_id int not Null,
```

```
    Review varchar(500),
```

```
    user_id int,
```

```
    FOREIGN KEY (user_id) REFERENCES users(user_id)
```

```
);
```

```
Create table rating (
```

```
    rating_id INT Auto_increment NOT NULL Primary Key,
```

```
    movie_id int not Null,
```

```
    Rating int ,
```

```
    user_id int,
```

```
    FOREIGN KEY (user_id) REFERENCES users(user_id)
```

```
);
```

```
select * from movie;
```

#	movie_id	title	summary	cast	Genre	user_id	average_rating
1	RRR	blah...	Ramcharan,NTR	Drama	1	4	
4	bahubali	In the kingdom of Mahishmati, Shivudu falls in love with a young warrior woman. While trying to woo her, he learns about the conflict-ridden past of his family and his true legacy.					
		prabhas,anushka,rana,tamannah	action	2	5		
5	KGF	can be anything	yash	action	2	4.5	

6 harry potter it is a series of 7 fantasy novels daniel radcliffe, emma watson
fantasy3 4

select * from users;

#	user_id	name	email
1		varsha	varsha@gmail.com
2		guduri	guduri@til.com
3		devi	devi@google.com

select * from rating;

#	rating_id	movie_id	Rating	user_id
1	1	4	1	
2	4	5	2	
3	5	4	2	
4	5	5	1	
5	6	4	3	

select * from review;

#	review_id	movie_id	Review	user_id
3	4	A visual treat.	2	
4	5	very good	2	
5	1	Just Awesome	1	
6	6	good	3	

Output for all view options:

Loading class `com.mysql.jdbc.Driver`. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver`. The driver is automatically registered via the SPI and manual loading of the driver class is generally unnecessary.

pls enter your option

1. View, 2. edit, -1. Exit

1

You have the following options. Please enter your choice.

1. view list of movies from a genre

2. view list of top 10 movies overall
3. view list of top 10 movies from a genre
4. search for a movie by name
5. view all movies

1

pls enter genre:

action

Movies in action genre:

bahubali

KGF

pls enter your option

1. View, 2. edit, -1. Exit

1

You have the following options. Please enter your choice.

1. view list of movies from a genre
2. view list of top 10 movies overall
3. view list of top 10 movies from a genre
4. search for a movie by name
5. view all movies

2

Top 10 movie overall:

bahubali

KGF

RRR

harry potter

pls enter your option

1. View, 2. edit, -1. Exit

1

You have the following options. Please enter your choice.

1. view list of movies from a genre
2. view list of top 10 movies overall
3. view list of top 10 movies from a genre
4. search for a movie by name
5. view all movies

3

pls enter genre:

fantasy

Top 10 movie in fantasy genre:

harry potter

pls enter your option

1. View, 2. edit, -1. Exit

1

You have the following options. Please enter your choice.

1. view list of movies from a genre
2. view list of top 10 movies overall
3. view list of top 10 movies from a genre
4. search for a movie by name
5. view all movies

4

pls enter movie name:

KGF

[

Movie [movie_id=5, title=KGF, summary=can be anything, cast=yash, genre=action, avgRate=2.0, user_id=4]]

pls enter your option

1. View, 2. edit, -1. Exit

1

You have the following options. Please enter your choice.

1. view list of movies from a genre
2. view list of top 10 movies overall
3. view list of top 10 movies from a genre
4. search for a movie by name
5. view all movies

5

[

Movie [movie_id=1, title=RRR, summary=blah..., cast=Ramcharan,NTR, genre=Drama, avgRate=1.0, user_id=4],

Movie [movie_id=4, title=bahubali, summary=In the kingdom of Mahishmati, Shivudu falls in love with a young warrior woman. While trying to woo her, he learns about the conflict-ridden past of his family and his true legacy., cast=prabhas,anushka,rana,tamannah, genre=action, avgRate=2.0, user_id=5],

Movie [movie_id=5, title=KGF, summary=can be anything, cast=yash, genre=action, avgRate=2.0, user_id=4],

Movie [movie_id=6, title=harry potter, summary=it is a series of 7 fantasy novels, cast=daniel radcliffe, emma watson, genre=fantasy, avgRate=3.0, user_id=4]]

pls enter your option

1. View, 2. edit, -1. Exit

-1

Output for all edit options which will be only available to user:

Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SPI and manual loading of the driver class is generally unnecessary.

pls enter your option

1. View, 2. edit, -1. Exit

2

To edit u need to be a user.

1. signin or 2.signup

1

enter email-id:

guduri@til.com

1.add a new movie entry

2.edit an existing movie entry

3.add review to a movie

4.rate a movie

1

pls enter movie title:

journey 2

pls enter summary:

the mysterious island

pls enter cast:

the rock

pls enter genre:

fantasy

1 movie record inserted successfully

pls enter your option

1. View, 2. edit, -1. Exit

2

To edit u need to be a user.

1. signin or 2.signup

2

enter id:

4

enter email-id:

abcd@gmail.com

enter name:

abcd

1 records inserted

- 1.add a new movie entry
- 2.edit an existing movie entry
- 3.add review to a movie
- 4.rate a movie

2

pls enter movie title:

KGF

pls enter summary:

Rocky, a young man, seeks power and wealth

pls enter cast:

yash

pls enter genre:

action

1 record updated

pls enter your option

1. View, 2. edit, -1. Exit

2

To edit u need to be a user.

1. signin or 2.signup

1

enter email-id:

abcd@gmail.com

- 1.add a new movie entry
- 2.edit an existing movie entry
- 3.add review to a movie
- 4.rate a movie

3

pls enter Movie name:

journey 2

Add Review: **extraordinary**

1 review added

pls enter your option

1. View, 2. edit, -1. Exit

2

To edit u need to be a user.

1. signin or 2.signup

1

enter email-id:

abcd@gmail.com

1.add a new movie entry

2.edit an existing movie entry

3.add review to a movie

4.rate a movie

4

pls enter movie name:

journey 2

pls enter rating:

5

1 rating added

pls enter your option

1. View, 2. edit, -1. Exit

-1