JAVA PROGRAMMING

(340)

REGIONAL – 2019

**Production Portion:**

Program 1: Movie Ratings (255 points)

***TOTAL POINTS***  *(255 points)*

**Graders: Please double check and verify all scores and answer keys!**

Property of Business Professionals of America.

May be reproduced only for use in the Business Professionals of America

*Workplace Skills Assessment Program* competition.

You will have ninety (90) minutes to complete your work.

Your name and/or school name should *not* appear on work you submit for grading.

1. Create a folder on the flash drive provided using your contestant number as the name of the folder.
2. Copy your entire solution/project into this folder.
3. Submit your entire solution/project so that the graders may open your project to review the source code.
4. Ensure that the files required to run your program are present and will execute on the flash drive provided.

\*Note that the flash drive letter may *not* be the same when the program is graded as it was when you created the program.

\*It is recommended that you use relative paths rather than absolute paths to ensure that the program will run regardless of the flash drive letter.

The graders will *not* compile or alter your source code to correct for this.   
Submissions that do *not* contain source code will *not* be graded.

**Assumptions to make when taking this assessment:**

* The user will not enter negative numbers as input.
* Movie reviews are rated from 1 to 100.

**Development Standards:**

* Your Code must use a consistent variable naming convention.
* All subroutines (if any), functions (if any), and methods (if any) must be documented with comments explaining the purpose of the method, the input parameters (if any), and the output (if any).
* If you create a class, then you must use Javadoc comments.

***Note to Graders:***

* Output will vary for critic rating and overall movie rating due to the random double generated for the critic rating.
* Error message may differ from sample output

**Test Case #1**

How many movies are you rating?

One

Incorrect data has been entered

**Test Case #2**

How many movies are you rating?

3

Please enter ratings from the movie review website for movie #1

75

99

45.7

Incorrect data has been entered

**Test Case #3**

How many movies are you rating?

1

Please enter ratings from the movie review website for movie #1

75

99

10

Please enter ratings from the focus group for movie #1

84.5

92.3

Ratings for movie #1

Average website rating: 61.33

Average focus group rating: 88.40

Average movie critic rating: 91.60

Overall movie rating: 84.59

**Test Case #4**

How many movies are you rating?

3

Please enter ratings from the movie review website for movie #1

75

99

10

Please enter ratings from the focus group for movie #1

84.5

92.3

Ratings for movie #1

Average website rating: 61.33

Average focus group rating: 88.40

Average movie critic rating: 91.60

Overall movie rating: 84.59

Please enter ratings from the movie review website for movie #2

85

90

20

Please enter ratings from the focus group for movie #2

65.7

74.3

Ratings for movie #2

Average website rating: 65.00

Average focus group rating: 70.00

Average movie critic rating: 69.10

Overall movie rating: 68.55

Please enter ratings from the movie review website for movie #3

55

75

35

Please enter ratings from the focus group for movie #3

72.8

83.6

Ratings for movie #3

Average website rating: 55.00

Average focus group rating: 78.20

Average movie critic rating: 85.15

Overall movie rating: 77.03

|  |  |  |
| --- | --- | --- |
| **Solution and Project** |  | |
| The project is present on the flash drive |  | 10 points |
| The projects main class is named **Movie** |  | 10 points |
| **Program Execution** |  |  |
| The program runs from the USB flash drive |  | 15 points |
| If the program does *not* execute, then the remaining items in this section receive a score of zero. | | |
| The program displays an error message if the user enters invalid data |  | 20 points |
| The program calculates the correct percentages for all averages |  | 10 points |
| The program displays the correct movie number during input |  | 10 points |
| The program displays the correct movie number during output |  | 10 points |
| The program displays the average website rating to two decimals |  | 20 points |
| The program displays the average focus group rating to two decimals |  | 20 points |
| The program displays the average movie critic rating to two decimals |  | 20 points |
| The program calculates the correct weighted percentage for each category | \_\_\_\_\_\_ | 30 points |
|  |  |  |
| The program uses the Random method to originate the critic rating |  | 30 points |
|  |  |  |
| **Source Code Review** |  |  |
|  |  |  |
| The source code is properly commented |  |  |
| A comment containing the contestant number is present |  | 10 points |
| Methods and code sections are commented |  | 20 points |
| Code uses try... catch for exception handling |  | 10 points |
| Code uses a consistent variable naming convention |  | 10 points |
|  |  |  |
| **Total Points =\_\_\_\_ / 255 points** | | |

**Suggested Solution**

/\*\*

\* BPA Java Programming Contest : Movie

\*

\* @author - contestant number goes here

\*

\* @version January 2018

\*

\*/

import java.util.concurrent.ThreadLocalRandom;

import java.util.Scanner;

public class Movie {

public static void main(String[] args) {

// Declare Scanner and Variables

Scanner scan = new Scanner(System.in);

int online1, online2, online3;

double focus1, focus2;

double criticAverage, onlineAverage, focusAverage, overallRating;

int movies;

int i = 1;

try {

System.out.println(“How many movies are you rating?”);

movies = scan.nextInt();

while (i <= movies)

{

// Input ratings

System.out.println(“Please enter ratings from the movie review website for movie #” + i);

online1 = scan.nextInt();

online2 = scan.nextInt();

online3 = scan.nextInt();

System.out.println(“Please enter ratings from the focus group for movie #” + i);

focus1 = scan.nextDouble();

focus2 = scan.nextDouble();

criticAverage = 1.0 \* ThreadLocalRandom.current().nextDouble(1, 100);

// Calculate Average

onlineAverage = 1.0 \* (online1 + online2 + online3) / 3;

focusAverage = (focus1 + focus2) / 2;

overallRating = (0.2 \* onlineAverage) + (0.3 \* focusAverage) + (0.5 \* criticAverage);

// Print Results

System.out.println(“Ratings for movie #” + i);

System.out.format(“Average website rating: “ + “%.2f%n”, onlineAverage);

System.out.format(“Average focus group rating: “ + “%.2f%n”, focusAverage);

System.out.format(“Average movie critic rating: “ + “%.2f%n”, criticAverage);

System.out.format(“Overall movie rating: “ + %.2f%n”, overallRating);

System.out.println();

i++;

}

} catch (Exception e) {

System.out.println(“Incorrect data has been entered.”);

}

}

}