Master of Science in Software Management Program
Summer Java Assignment (Last updated July 2, 2018)
Due: July 16, 2018 at 11:59 pm Pacific Daylight Time
Send your submissions and/or questions to Karimulla Shaikh at karim@cmu.edu
Please check the following site periodically for any updates/corrections on this assignment.
http://www.andrew.cmu.edu/user/karim/mssm2018/

Ever wonder how Netflix decides what movies to recommend for you? Or how Amazon recommends books? We can get a feel for how recommendations work by building a simplified system of our own!

In this assignment, you will demonstrate your Java programming skills by creating a simple recommender system. You will work with movies and movie ratings, but the principles involved can be adapted to books, restaurants, and more. You will write a program to determine which movies should be recommended to a user based on the ratings of several movies by people of similar age.

You will be provided two input files and you will generate an output file after a series of tasks on the input files.

Input File 1: The main input data is in a CSV file and is available at the link where you downloaded this assignment. The first five lines are shown below as an illustration. 5 indicates the best rating and 1 indicates the worst rating.

```
UserID,UserName,UserAge,MovieID,MovieName,Rating
1,Adam,15,,"1,dinosaur planet",3
2,Amir,20,,"2,isle of man tt 2004 review",2
3,Brad,25,,"3,character",1
4,Krishna,35,,"4,paula abdul's get up & dance",4
5,Sangy,21,,"5,the rise and fall of ecw",5
```

Input File 2: You're also given the following input CSV file which lists some new users, their age and the number of movies they want recommended to them. The file also has a question mark (?) which you will replace with movie recommendations.

```
UserName,UserAge,NoOfMoviesToRecommend,Movies
Xin,20,4,?
Erin,24,3,?
Sri,37,2,?
Nick,18,4,?
Vicky,20,2,?
```

Output File: Your final output file will look like this (after replacing the question mark with the names of movies you are recommending to them).

```
UserName,UserAge,NoOfMoviesToRecommend,Movies
Xin,20,4,"dinosaur planet,character,rocky,die hard"
Erin,24,3,"dinosaur,rocky,transformers"
Sri,37,2,"planet of the apes,jurassic park"
Nick,18,4,"rocky,jurassic park,die hard,avengers"
Vicky,20,2,"coco,finding nemo"
```

To accomplish this assignment, you will go through a series of five tasks. If you accomplish the five tasks successfully, you will arrive at the final output file above.

Task 1 - Separate Movie IDs and Movie names.

In the provided input file, you will notice that MovieID column is empty. Unfortunately, Movie IDs and Movie Names are combined into a single column **MovieName**. In this task, you will separate these two parts into into corresponding columns -> MovieID and MovieName. You will create a new CSV file with the MovieID column filled out and with the MovieName column containing only the name of the movie. This task will test your file and string manipulation skills.

So the first few lines of your new CSV file will look like this:

```
UserID,UserName,UserAge,MovieID,MovieName,Rating
1,Adam,15,1,,”dinosaur planet",3
2,Amir,20,2,"isle of man tt 2004 review",2
3,Brad,25,3,”character",1
4,Krishna,35,4,"paula abdul's get up & dance",4
5,Sangy,21,5,”the rise and fall of ecw",5
```

Task 2 - String Capitalization - Capitalizing first letter of every word in the movie names.

Capitalize first letter of every word in MovieName column.
e.g. : dinosaur planet -> Dinosaur Planet
You will create a new CSV file in this format.

The first two lines of your new CSV file will look like this:

```
UserID,UserName,UserAge,MovieID,MovieName,Rating
1,Adam,15,1,,”Dinosaur Planet",3
```

Task 3 - Read in from your new CSV file from Task 2 and parse data into lists and maps

You can choose any type of maps and lists that you like. Below is an example of how you can create one but you are welcome to choose your own as long as you can accomplish the objectives of remaining tasks (4 and 5). You may want to read those two tasks before you make a decision on what maps and lists to use.

As an illustration, you can create a map with the key as the age and the value as another map based on ratings. The inner map will use the rating as the key and the list of movies as the value. For example

```
34 => { 1 => [“Dinosaur Planet”, “Rocky”], 2 => [“Coco”], 3 -> [“Twins”,”Cars”]}
```

indicates that users of age 34 have given Dinosaur Planet and Rocky a rating of 1, Coco a rating of 2 and Twins and Cars a rating of 3. You will have ratings from 1 to 5 in your actual map. Your outer map will have many entries - one for each age.

Once you have these maps and lists, you should be able to easily lookup and identify the best rated movies for each age.

Task 4 - Find the recommended movies for a given age from best to worst ratings

You will write a method that takes an age and goes through the maps/lists you created in the previous task to return (and display) the best recommended movies for that age. You should also take an additional parameter, which is the maximum number of movies to be recommended/displayed for that age. You should display no more than the requested number of movies.

Task 5 - Recommend movies to users in the second input file.

Now process the second input file given to you by us which contains a list of new users with their ages as well as the number of recommendations they requested. Call the method above repeatedly for each line in the file to get the best recommendations for a user based on their age. Create the final output file so that it looks as below. (You can create a new output file - you don't need to physically replace the question marks in the input file)

```
UserName,UserAge,NoOfMoviesToRecommend,Movies
Xin,20,4,"dinosaur planet,character,rocky,die hard"
Erin,24,3,"dinosaur,rocky,transformers"
Sri,37,2,"planet of the apes,jurassic park"
Nick,18,4,"rocky,jurassic park,die hard,avengers"
Vicky,20,2,"coco,finding nemo"
```

Submission

You will submit a Zip file containing the following.

1. Your source code (either single or multiple files)
2. The final output file you generated

Please send your submission to the email address in the header of this document with the subject line: "Submission for Summer Java Assignment"

***IMPORTANT**: Please check the following site periodically (every 2-3 days) for any updates/corrections on this assignment. If you have any questions, please check this site before emailing karim@cmu.edu*

http://www.andrew.cmu.edu/user/karim/mssm2018/

Good luck!