

CS 377: Database Systems

Project #3

Wednesday, March 23, 2016 at 11:59 pm

SUBMISSION: Please submit the project electronically via Blackboard before 11:59 pm. Any submissions afterwards will be marked late and deducted accordingly unless you create a README file that specifies that you are using one of your 2 late assignment exceptions. You should use one file for each problem and follow these guidelines:

- Each file should be named <FirstnameLname>Prob<Number>.java. Use camel case to differentiate your first name and last name. For example, the instructor's submission for problem number 1 would be: JoyceHoProb1.java.
- Each command-line Java program should print an introductory message that includes your name and the problem number and the following line:

```
/* THIS CODE IS MY OWN WORK.  
IT WAS WRITTEN WITHOUT CONSULTING CODE WRITTEN BY OTHER STUDENTS.  
_Your_Name_Here_ */
```

1. **Matt Damon Recent Movies** (30 points): Write a command line Java program that does the following:
 - prints an introductory message to the screen that includes your name, the problem number, and the honor code (from above)
 - prints the most recent 10 movies that Matt Damon starred in
 - each movie should be on a separate line with the role and the year separated by a space, semicolon, and a space (e.g., Titan A.E ; Cale Tucker ; 2000)
2. **IMDB Actor Movie Finder** (70 points): Write a command line Java program that finds the movies where a list of actors were all in together. The program should allow for any number of actors, specified by the first and last name. The application should:
 - expect at least two command-line arguments - if there are no command line arguments complain and exit
 - each pair of command line arguments will be of the form first name, last name (e.g., java JoyceHoProb2 Matt Damon would search for movies with Matt Damon, java JoyceHoProb2 Matt Damon Brad Pitt would search for movies with Matt Damon AND Brad Pitt, etc.)
 - checks to see that the actor/actress is in the database - if not found, the program should complain which actor/actress was not found and exit
 - prints an introductory message to the screen that includes your name, the problem number, and the honor code (from above)
 - prints that no result was found if there is no movie where all the actors were in together
 - print the total number of movies (count) where all the actors were in the same movie as a newline
 - prints each movie on a separate line with the year separated by a space, semicolon, and a space (e.g., java JoyceHoProb2 Matt Damon would have a line with Titan A.E ; 2000)