Joanna Klukowska
joannakl@cs.nyu.edu

Project 2: San Francisco Movies

Due date: October 1, 11:55PM EST.

You may discuss any of the assignments with your classmates and tutors (or anyone else) but all work for all assignments must be entirely your own . Any sharing or copying of assignments will be considered cheating (this includes posting of partial or complete solutions on Piazza, GitHub or any other public forum). If you get significant help from anyone, you should acknowledge it in your submission (and your grade will be proportional to the part that you completed on your own). You are responsible for every line in your program: you need to know what it does and why. You should not use any data structures and features of Java that have not been covered in class (or the prerequisite class). If you have doubts whether or not you are allowed to use certain structures, just ask your instructor.

In this project you will provide a tool for searching through the data set of movies that were filmed on location in San Francisco.

Objectives

The goal of this programming project is for you to master (or at least get practice on) the following tasks:

- working with multi-file programs
- reading data from input files
- using and understanding command line arguments
- working with large data sets
- using the `ArrayList` class
- writing classes
- working with existing code
- extending existing classes (inheritance)
- parsing data

Most, if not all, of the skills that you need to complete this project are based on the material covered in cs101. But there might be certain topics for which you did not have to write a program or that you forgot. Make sure to ask questions during recitations, in class and on Piazza.

Start early! This project may not seem like much coding, but debugging always takes time.

For the purpose of grading, your project should be in the package called `project2`. This means that each of your submitted source code files should start with a line:

```
package project2;
```

Keep in mind that spelling and capitalization are important!

Dataset

In this project you will be working with open data. Wikipedia has a good description of open data: ”Open data is the idea that some data should be freely available to everyone to use and republish as they wish, without restrictions from copyright, patents or other mechanisms of control.”

The data set that you need can be found at DataSF portal. For your convenience a csv file is also located on the course website. It is a listing of filming locations of movies shot in San Francisco starting from 1924.[1] The data set is provided by San Francisco Film Commission.

The file that you download is a CSV (comma separated values) file - it is a simple text file and is processed like text file (but it can also be opened by most of the spreadsheet programs and displayed column-wise based on the locations of commas on each line). Since some of the columns contain longer text that might, optionally, contain commas as well, those entries are enclosed in a set of double quotes.

There are eleven columns in the dataset. Some of the columns may be empty.

[1] This data set is not complete and contains many errors. This does not affect the program that you will be writing, though.

Joanna Klukowska
joannakl@cs.nyu.edu

User Interface

Your program has to be a console based program (no graphical interface) - this means that the program should not open any windows or dialogs to prompt user for the input.

Program Usage

The program is started from the command line (or run within an IDE). It expects one command line argument.

This program should use command line arguments. When the user runs the program, they provide the name of the input file as a command line argument. (This way the user can specify a data set from another city or a different version of the data set).

The user may start the program from the command line or run it within an IDE like Eclipse - **from the point of view of your program this does not matter**.

If the name of the input file provided as a command line argument is incorrect or the file cannot be opened for any reason, the program should display an error message and terminate. It should not prompt the user for an alternative name of the file.

If the program is run without any arguments, the program should display an error message and terminate. It should not prompt the user for the name of the file.

The error messages should be specific and informative, for example:

```
Error: the file films.csv cannot be opened.
```

or

```
Usage Error: the program expects file name as an argument.
```

The above error messages generated by your code should be written to the `System.err` stream (not the `System.out` stream).[2]

Input and Output

The program should run in a loop that allows the user to issue different queries. The two types of queries are:

```
title KEYWORD
```

and

```
actor KEYWORD
```

In the first case, the program should display the list of all the movies in the data set whose title contains the given keyword (this search should be case insensitive). In the second case, the program should display the list of all the movies with actors whose names contain the given keyword (again, this search should be case insensitive). In both cases, the user may enter a multi-word KEYWORD.

On each iteration, the user should be prompted to enter a new query (for which the program computes the results) or the word 'quit' to indicate the termination of the program.

The user should not be prompted for any other response.

If the query entered by the user is invalid, the program should display an error message:

```
This is not a valid query. Try again.
```

and allow the user to provide an alternative query.

If the query entered by the user does not return any results, the program should print a message

```
No matches found. Try again.
```

and allow the user to provide an alternative query.

Matching reults display: If the query entered by the user matches one or more movies, the information about all the matching movies should be displayed in the following format:

[2] If you are not sure what the difference is, research it or ask questions.

Joanna Klukowska
joannakl@cs.nyu.edu

```
TITLE (YEAR)
-------------------------------------
director            : DIRECTOR
writer              : WRITER
starring            : ACTOR1, ACTOR2, ACTOR3,
filmed on location at:
    LOCATION1 (FUN_FACT1)
    LOCATION2 (FUN_FACT2)
    ...
    LOCATION_N (FUN_FACT_N)
```

All the words in uppercase letters are place-holders for the actual values from the data set. If any of the values is missing, it should be left blank and the corresponding commas and parenthesis should not be shown.

Sample interaction:

Sample user interaction is shown in Appendix 2.

Data Storage and Organization

You need to provide an implementation of several classes that store the data and compute the results when the program is executed.

In particular, your program must implement and use the following classes. You may implement additional classes and additional methods in the required classes, if you wish.

`Actor` Class

The `Actor` class is used to represent actors. It should store the name of an actor and it should provide a one parameter contructor. The constructor should throw an instance of `IllegalArgumentException` if it is called with a `null` prameter or an empty string parameter.

`Location` Class

The `Location` class is used to represent the filming locations and fun facts that may be associated with them. It should store the name of the location itself and the optional fun fact. The class should provide a constructor that takes as parameters two strings that represent the location and the fun fact, respectively. The constructor should throw an instance of `IllegalArgumentException` if it is called with the location string that is either `null` or empty. The fun fact is allowed to be `null` or empty (this will happen when there is no fun fact associated with the given location, i.e., the corresponding column in the data set is empty).

`Movie` Class

The `Movie` class is used to represent movies. It should store the following information:

- title
- year of release
- list of SF locations in which the movie was filmed (together with associated fun facts)
- director
- writer
- list of up to three actors

Note: most of the movies are represented on multiple rows of the data set.

This class should provide two constructors:

```
public Movie(String title, int year)
```

Joanna Klukowska
joannakl@cs.nyu.edu

```
public Movie(String title, int year, String director, String writer,
                                Actor actor1, Actor actor2, Actor actor3)
```

The following describes acceptable values for each of the constructor parameters. If a constructor is called with an invalid parameter, it should throw an instance of `IllegalArgumentException`.

- `title` - any non-empty string
- `year` - a number between 1900 and 2020 (including both end points)
- `director` - any non-empty string to indicate the name, `null` or an empty string when the data is not available (for example, when the column in the data set is empty)
- `writer` - any non-empty string to indicate the name, `null` or an empty string when the data is not available
- `actor1` - any non-emptyh string
- `actor2` - any non-empty string to indicate the name, `null` or an empty string when the data is not available
- `actor3` - any non-empty string to indicate the name, `null` or an empty string when the data is not available

There should be no default constructor.[3]

The class should provide a method

```
public void addLocation(Location loc)
```

that adds a given location to the list of filming locations for the current `Movie` object. The method should throw an instance of IllegalArgumentException if it is called with `null` parameter.

This class should implement `Comparable<Name>` interface. The comparison should be done by the `year` of release as the primary key (earlier years are considered to be smaller), and by the `title` as the secondary key (i.e., when two objects that have the same value of `year` are compared, the comparison should be performed by `title`) (the titles should be ordered according to the same rules as the String objects and the comparison should be case insensitive).

This class should override the `equals` methods. The two `Movie` objects should be considered equal, if the `tite` and `year` are identical (the title comparison should be case insensitive).

This class should override the `toString` method. It should return a string matching the following format:

```
TITLE (YEAR)
-----------------------------------
director            : DIRECTOR
writer              : WRITER
starring            : ACTOR1, ACTOR2, ACTOR3,
filmed on location at:
    LOCATION1 (FUN_FACT1)
    LOCATION2 (FUN_FACT2)
    ...
    LOCATION_N (FUN_FACT_N)
```

All the words in uppercase letters are place-holders for the actual values from the data set. If any of the values is missing, it should be left blank and the corresponding commas and parenthesis should not be shown. (This is the format used to display the results to the user.)

The class should provide getters and setters for any data fields that your program may need access to from the outside of the class.

`MovieList` Class

The `MovieList` class should be used to store all the `Movie` objects. This class should inherit from the `ArrayList<Movie>` class.

The class needs to provide a default constructor that creates an empty `MovieList` object.

The class should implement the following two methods:

```
public MovieList getMatchingTitles ( String keyword )
```

[3] A default constructor is one that can be used without passing any arguments.

Joanna Klukowska
joannakl@cs.nyu.edu

```
public MovieList getMatchingActor ( String keyword )
```

The first method should return a list of `Movie` objects whose titles contain the `keyword` as a substring. The second method should return a list of `Movie` object whose actors' names contain the `keyword` as a substring. The cantainment should be determined in case insesitive way in both cases. The returned lists should be sorted according to the natural order[4] of their elements. These methods should return `null` if the functions are called with a keyword that is either equal to `null` or an empty string. They should also return `null` if there are no matches for the keyword.

The class should override the `toString` method. This method should return a `String` containing a semi-colon and space separated list of the titles of the `Movie` objects stored in the list. For example, if the three Movie objects stored in the list arre *On the Road*, *Budding Prospects, Pilot* and *The House on Telegraph Hill*, than the returned `String` should be

`"On the Road; Budding Prospects, Pilot; The House on Telegraph Hill"`.

You may implement other methods, if you wish.

`SFMovieData` Class

The `SFMovieData` class is the actual program. This is the class that should contain the `main` method. It is responsible for opening and reading the data files, obtaining user input, performing some data validation and handling all errors that may occur (in particular, it should handle any exceptions thrown by your other classes and terminate gracefully, if need be, with a friendly error message presented to the user).

You may implement other methods in this class to modularize the design.

Given Code

To simplify parsing of the data from the input file, you can use the function `splitCSVLine`. Given a row (a single line) from the input data set, it returns an `ArrayList<String>` object containing all the individual entries from that row. This function produces empty strings to represent entries that were blank within the input line before the last comma. If the last columns in the row are missing, the returned `ArrayList<String>` object has fewer entries.

For example: given a string containing the line:

```
"Alcatraz",2012,"Francisco St from Larkin to Polk",,"Bonanza Productions Inc.","Warner Bros. Television","J.J. Abrams
","Steven Lilien","Sarah Jones","Jorge Garcia",
```

it returns an `ArrayList<String>` object with ten strings:

```
["Alcatraz", "2012","Francisco St from Larkin to Polk","","Bonanza Productions Inc.","Warner Bros. Television","J
.J. Abrams","Steven Lilien","Sarah Jones","Jorge Garcia"]
```

The source code for `splitCSVLine` is given in Appendix 1. You can use this method as is or modify it if you wish.

Programming Rules

You should follow the rules outlined in the document *Code conventions* posted on the course website at https://cs.nyu.edu/~joannakl/cs102_f18/notes/CodeConventions.pdf.

The data file should be read only once! Your program needs to store the data in memory resident data structures.

You may not use any of the collection classes that were not covered in cs101 (for this assignment, do not use `LinkedList`, `Stack`, `Queue`, `PriorityQueue`, or any classes implementing `Map` interface).

You may use any exception-related classes.

You may use any classes to handle the file I/O, but probably the simplest ones are `File` and `Scanner` classes. You are responsible for knowing how to use the classes that you select.

[4] The natural order of elements is determined by the `compareTo` method defined in the class of the elements.

Joanna Klukowska
joannakl@cs.nyu.edu

Working on This Assignment

You should start right away!

You should modularize your design so that you can test it regularly. Make sure that at all times you have a working program. You can implement methods that perform one task at a time. This way, if you run out of time, at least parts of your program will be functioning properly.

You should make sure that you are testing the program on much smaller data set for which you can determine the correct output manually. You can create a test input file that contains only a few rows.

You should make sure that your program's results are consistent with what is described in this specification by running the program on carefully designed test inputs and examining the outputs produced to make sure they are correct. The goal in doing this is to try to find the mistakes you have most likely made in your code.

Each class that you submit will be tested by itself without the context of other classes that you are implementing for this assignment. This means that you need to make sure that your methods can perform their tasks correctly even if they are executed in situations that would not arise in the context of this specific program.

You should backup your code after each time you spend some time working on it. Save it to a flash drive, email it to yourself, upload it to your Google drive, do anything that gives you a second (or maybe third copy). Computers tend to break just a few days or even a few hours before the due dates - make sure that you have working code if that happens.

Grading

If your program does not compile or if it crashes (almost) every time it is run, you will get a zero on the assignment. Make sure that you are submitting functioning code, even if it is not a complete implementation.

If the program does not adhere to the specification, the grade will be low and will depend on how easy it is to figure out what the program is doing.

40 points class correctness: correct behavior of methods of the required classes and correct behavior of the program

35 points design and the implementation of the required classes and any additional classes

25 points proper documentation, program style and format of submission

How and What to Submit

For the purpose of grading, your project should be in the package called `project2`. This means that each of your submitted source code files should start with a line:

```
package project2;
```

Your should submit all your source code files (the ones with .java extensions only) in a single **zip** file to Gradescope.

You can produce a zip file directly from Eclipse:

- right click on the name of the package (inside the `src` folder) and select Export...
- under General pick Archive File and click Next
- in the window that opens select appropriate files and settings:
  - in the right pane pick ONLY the files that are actually part of the project, but make sure that you select all files that are needed
  - in the left pane, make sure that no other directories are selected
  - click Browse and navigate to a location that you can easily find on your system (Desktop or folder with the course materials or ...)
  - in Options select "Save in zip format", "Compress the contents of the file" and "Create only selected directories"
- click Finish

Joanna Klukowska
joannakl@cs.nyu.edu

Appendix 1: Parsing CSV file

```
/**
 * Splits the given line of a CSV file according to commas and double quotes
 * (double quotes are used to surround multi-word entries so that they may contain commas)
 * @author Joanna Klukowska
 * @param textLine  a line of text to be passed
 * @return an Arraylist object containing all individual entries found on that line
 */
public static ArrayList<String> splitCSVLine(String textLine){

if (textLine == null ) return null;

  ArrayList<String> entries = new ArrayList<String>();
  int lineLength = textLine.length();
  StringBuffer nextWord = new StringBuffer();
  char nextChar;
  boolean insideQuotes = false;
  boolean insideEntry= false;

  // iterate over all characters in the textLine
  for (int i = 0; i < lineLength; i++) {
    nextChar = textLine.charAt(i);

    // handle smart quotes as well as regular quotes
    if (nextChar == '"' || nextChar == '\u201C' || nextChar =='\u201D') {

      // change insideQuotes flag when nextChar is a quote
      if (insideQuotes) {
        insideQuotes = false;
        insideEntry = false;
      }else {
        insideQuotes = true;
        insideEntry = true;
      }
    } else if (Character.isWhitespace(nextChar)) {
      if ( insideQuotes || insideEntry ) {
      // add it to the current entry
        nextWord.append( nextChar );
      }else { // skip all spaces between entries
        continue;
      }
    } else if ( nextChar == ',') {
      if (insideQuotes){ // comma inside an entry
        nextWord.append(nextChar);
      } else { // end of entry found
        insideEntry = false;
        entries.add(nextWord.toString());
        nextWord = new StringBuffer();
      }
    } else {
      // add all other characters to the nextWord
      nextWord.append(nextChar);
      insideEntry = true;
    }

  }
  // add the last word ( assuming not empty )
  // trim the white space before adding to the list
  if (!nextWord.toString().equals("")) {
    entries.add(nextWord.toString().trim());
  }

  return entries;
}


```

Joanna Klukowska
joannakl@cs.nyu.edu

Appendix 2: Sample Interaction

Note that some of the text is wider than the shown display. Your program does not need to worry about line-wrapping.

```
Search the database by matching keywords to titles or actor names.
  To search for matching titles, enter
    title KEYWORKD
  To search for matching actor names, enter
    actor KEYWORD
  To finish the program, enter
    quit



Enter your search query:

title kill

A View to a Kill (1985)
-----------------------------------
director            : John Glen
writer              : Richard Maibaum
starring            : Roger Moore,Christopher Walken,Tanya Roberts,
filmed on location at:
    City Hall (The dome of SF's City Hall is almost a foot taller than that of the US Capitol Building. In 1954, Joe DiMa
    Port of San Francisco
    Lefty O' Doul Drawbridge/ 3rd Street Bridge (3rd Street, China Basin) (This is SF's only drawbridge, and was named af
    Golden Gate Bridge (With 23 miles of ladders and 300,000 rivets in each tower, the Golden Gate Bridge was the world's
    Embarcadero Freeway (Embarcadero Freeway, which was featured in the film, was demolished in 1989 because of structura
    Potrero Hill (The most crooked street in San Francisco is actually Potrero Hill's Vermont Street between 20th St & 22
    City Hall (The dome of SF's City Hall is almost a foot taller than that of the US Capitol Building. In 1954, Joe DiMa
    Chinatown (First established in the mid-19th Century, SF's Chinatown is the oldest and largest Chinatown in the US.)
    Burger Island (901 3rd Street, China Basin)
    Van Ness Avenue
    Taylor and Jefferson Streets (Fisherman's Wharf)



Enter your search query:

actor Humphrey Bogart

The Maltese Falcon (1941)
-----------------------------------
director            : John Huston
writer              : John Huston
starring            : Humphrey Bogart,Mary Astor,Gladys George,
filmed on location at:
    Golden Gate Bridge (With 23 miles of ladders and 300,000 rivets in each tower, the Golden Gate Bridge was the world's
    Ferry Building (Every hour and half-hour, the clock bell atop the Ferry Building chimes portions of the Westminster Q
    Burritt Alley (Off Bush Street, between Powell and Stockton Streets)

Dark Passage (1947)
-----------------------------------
director            : Delmer Daves
writer              : Delmer Daves & David Goodis
starring            : Humphrey Bogart,Lauren Bacal,Bruce Bennett,
filmed on location at:
    Golden Gate Bridge (With 23 miles of ladders and 300,000 rivets in each tower, the Golden Gate Bridge was the world's
    The Malloch Apartment Building (1360 Montgomery Street)
    Filbert Steps, Filbert Street

The Caine Mutiny (1954)
-----------------------------------
director            : Edward Dmytryk
writer              : Stanley Roberts
starring            : Humphrey Bogart,Fred MacMurray,Jose Ferrer,
```

Joanna Klukowska
joannakl@cs.nyu.edu

```
filmed on location at:
    The Embarcadero
    Golden Gate Bridge (With 23 miles of ladders and 300,000 rivets in each tower, the Golden Gate Bridge was the world's



Enter your search query:

actor man

The Graduate (1967)
-------------------------------------
director            : Mike Nichols
writer              : Calder Willingham
starring            : Anne Bancroft,Dustin Hoffman,Katharine Ross,
filmed on location at:
    San Francisco Zoo (2701 Sloat Blvd.)
    Bay Bridge (Before opening in 1936, the bridge was blessed by Cardinal Secretary of State Eugenio Pacelli, who later

The Conversation (1974)
-------------------------------------
director            : Francis Ford Coppola
writer              : Francis Ford Coppola
starring            : Gene Hackman,John Cazale,Allen Garfield,
filmed on location at:
    American Roofing Co. Building (297 Kansas Street, Potrero Hill)
    Alta Plaza Park (Steiner Street) (The park was originally a rock quarry and served as a campground for many survivors
    Alcoa Building (1 Maritime Plaza) (A partially-above ground parking structure near the building made it necessary for
    Alamo Square (Hayes Valley)
    Westin St. Francis Hotel (335 Powell Street, Union Square) (The hotel was originally supposed to be named the Crocker
    Union Square (During the Civil War, pro-Union rallies were held in the Square, and thus the area was called Union Squ
    Saks Fifth Avenue (384 Post Street, Union Square)
    One Embarcadero Center (Financial District)
    Neiman Marcus (150 Stockton Street, Union Square) (In the film the City of Paris Department Store is featured. That e
    Financial District
    Cathedral Hill Hotel (1101 Van Ness Avenue, Civic Center)

The Towering Inferno (1974)
-------------------------------------
director            : John Guillermin
writer              : Stirling Silliphant
starring            : Steve McQueen,Paul Newman,William Holden,
filmed on location at:
    2898 Vallejo Street
    Bank of America Building (555 California Street, Financial District) (The Bank of America Building was the tallest bu
    San Francisco Fire Station 38 (2150 California Street, Pacific Heights)
    Hyatt Regency Hotel (5 Embarcadero Center, Financial District)
    Grace Cathedral Episcopal Church (1100 California Street) (Grace Cathedral Episcopal Church is the West Coast's large
    Firestation #38 (California & Laguna)
    Fairmont Hotel (950 Mason Street, Nob Hill) (In 1945 the Fairmont hosted the United Nations Conference on Internatio

High Anxiety (1977)
-------------------------------------
director            : Mel Brooks
writer              : Mel Brooks
starring            : Mel Brooks,Madeline Kahn,Cloris Leachman,
filmed on location at:
    Hyatt Regency Hotel (5 Embarcadero Center)
    Fort Point (Presidio, Golden Gate National Recreation Area) (Built in 1853, Fort Point is the only West Coast fort bu
    Conservatory of Flowers, Golden Gate Park (The Conservatory, unveiled in 1879, is the oldest public conservatory in t

Superman (1978)
-------------------------------------
director            : Richard Donner
writer              : Jerry Siegel
starring            : Christopher Reeve,Gene Hackman,Marlon Brando,
filmed on location at:
    Golden Gate Bridge (With 23 miles of ladders and 300,000 rivets in each tower, the Golden Gate Bridge was the world's

Maxie (1985)
```

Joanna Klukowska
joannakl@cs.nyu.edu

```
-------------------------------------
director            : Paul Aaron
writer              : Patricia Resnick
starring            : Glenn Close,Mandy Patinkin,Ruth Gordon,
filmed on location at:
    Asian Art Museum (200 Larkin Street, Civic Center) (The Dalai Lama opened an exhibition on Wisdom and Compassion at t
    Grace Cathedral Episcopal Church (1100 California Street) (Grace Cathedral Episcopal Church is the West Coast's large
    722 Steiner Street (Postcard Row, Alamo Square, Hayes Valley) (The 6 Victorian homes across from Alamo Square Park ar

Dawn of the Planet of the Apes (2014)
-------------------------------------
director            : Matt Reeves
writer              : Rick Jaffa
starring            : Gary Oldman,Keri Russell,Andy Serkis,
filmed on location at:
    University Club
    Filbert St. from Hyde to Leavenworth
    California St from Mason to Kearny
    City Hall
    Alioto Park
    California & Powell
    Columbus & Pacific Ave

GirlBoss (2017)
-------------------------------------
director            : Jamie Babbit, Amanda Brotchie, Steven K. Tsuchida, Christian Ditter, John Riggi
writer              : Kay Cannon
starring            : Britt Robertson,Ellie Reed,Amanda Rea,
filmed on location at:
    Mission Dolores Park, 19th St. & Dolores St. (After the devastating earthquake of 1906, the park served as a refugee
    Green and Divisadero
    1375 Haight St.
    Mission Thrift, 2330 Mission St.
    The Castro Theater, 429 Castro St.
    Clarion Alley between Valencia and Mission (The murals that make up this famous alley were first painted in 1992 and
    Macchiarini Steps, Kearny between Vallejo and Broadway (The namesake for the steps was an Italian-American Californi
    Fillmore and Waller
    La Taqueria, 2889 Mission St. (The restaurant was given the honor of America's Best Burrito in 2014 by ESPN's FiveThi
    Various Haight St. Vintage Shops
    Clayton Post Office, 554 Clayton St.
    Taylor and Union
    Filbert between Leavenworth and Hyde
    Filbert between Grant and Kearny
    Filbert and Kearny Steps
    Boulevard Restaurant, 1 Mission St.
    Grant between California and Washington
    Musee Mecanique, Fisherman's Wharf (An interactive museum consisting of 20th century penny arcade games and artifacts
    Mission between 19th and 25th St.
    Underground SF, 424 Haight St.
    Fillmore between Haight and Waller


Enter your search query:

xxx

This is not a valid query. Try again.


Enter your search query:

title xxx

No matches found. Try again.


Enter your search query:

quit
```