Deliverables

Your project files should be submitted to Web-CAT by the due date and time specified. Note that there is also an optional Skeleton Code assignment which will indicate level of coverage your tests have achieved (there is no late penalty since the skeleton code assignment is ungraded for this project). The files you submit to skeleton code assignment may be incomplete in the sense that method bodies have at least a return statement if applicable or they may be essentially completed files. In order to avoid a late penalty for the project, you must submit your completed code files to Web-CAT no later than 11:59 PM on the due date for the completed code assignment. If you are unable to submit via Web-CAT, you should e-mail your project Java files in a zip file to your TA before the deadline. The grades for the Completed Code submission will be determined by the tests that you pass or fail in your test files and by the level of coverage attained in your source files as well as usual correctness tests in Web-CAT.

From Project 9 originally, but should be copied from Project 10

- SoftballPlayer.java
- Outfielder.java, OutfielderTest.java
- Infielder.java, InfielderTest.java
- Pitcher.java, PitcherTest.java
- ReliefPitcher.java, ReliefPitcherTest.java

From Project 10

- NameComparator.java, NameComparatorTest.java
- RatingComparator.java, RatingComparatorTest.java
- SoftballTeam.java, SoftballTeamTest.java [must be modified as described below]

New in Project 11

- InvalidCategoryException.java
- SoftballPlayersPart3.java, SoftballPlayersPart3Test.java

Recommendations

You should create a new folder for Part 3 and copy your relevant Part 2 source and test files to it. You should create a jGRASP project and add the class and test files as they are created.

Specifications – Use arrays in this project; ArrayLists are not allowed!

Overview: This project is Part 3 of three that involves the rating and reporting for softball players. In Part 1 (Project 9), you developed Java classes including an abstract SoftballPlayer class and subclasses of it that represent categories of softball players: outfielders, infielders, pitchers, and relief pitchers. In Part 2 (Project 10), you implemented three additional classes: (1) NameComparator that implements the Comparator interface for SoftallPlayer, (2) RatingComparator that implements the Comparator interface for SoftballPlayer, and (3) SoftballTeam that represents a team of softball players and includes several specialized methods. In Part 3 (Project 11), you are to add exception handling. You will need to do the following: (1) create a new class named InvalidCategoryException which extends the Exception class, (2) add try-catch statements to catch IOException in the main method of the SoftballPlayersPart3 class, and (3) modify the readPlayerFile in the SoftballTeam class

to catch/handle InvalidCategoryException and NumberFormatException in the event that either type exception is thrown while reading the input file.

Note that the main method in SoftballPlayersPart3 should create a SoftballTeam object and then invoke the readPlayerFile method on the SoftballTeam object to read data from a file and add softball players to the team. You can use SoftballPlayersPart3 in conjunction with interactions by running the program in the canvas (or debugger with a breakpoint) and single stepping until the variables of interest are created. You can then enter interactions in the usual way. In addition to the source files, you will create a JUnit test file for the indicated source files and write one or more test methods to ensure the classes and methods meet the specifications. You should create a jGRASP project upfront and then add the source files as they are created. All of your files should be in a single folder.

- **SoftballPlayer, Outfielder, Infielder, Pitcher, ReliefPitcher, NameComparator, RatingComparator**

  Requirements and Design: No changes from the specifications in Projects 9 and 10. .

- **InvalidCategoryException.java**

  Requirements and Design: InvalidCategoryException is a user defined exception created by extending the Exception class. This exception is to be thrown and caught in the readPlayerFile method in the SoftballTeam class when a line of input data contains an invalid player category. The constructor for InvalidCategoryException takes a single String parameter representing *category* and invokes the super constructor with the following String:

  ```
  "For category: " + "\"" + category + "\""
  ```

  This string will be the toString() value of an InvalidCategoryException when it occurs. For a similar constructor, see InvalidLengthException.java in 11_Exceptions\Examples\Polygons from this week's class notes.

- **SoftballTeam.java**

  Requirements and Design: The SoftballTeam class provides methods for reading in the data file and generating reports.

  Design: In addition to the specifications in Project 10, the existing readPlayerFile method must be modified to catch following: InvalidCategoryException and NumberFormatException.

  - `readPlayerFile` has no return value and accepts the data file name as a String. Remember to include the `throws IOException` clause in the method declaration. This method creates a Scanner object to read in the file and then reads it in line by line. The first line contains the team name and each of the remaining lines contains the data for a player. After reading in the team name, the "player" lines should be processed as follows. A player line is read in, a second scanner is created on the line, and the individual values for the player are read in. After the values on the line have been read in, an "appropriate" SoftballPlayer object created. If there is room on the roster, the player is added to the roster array and the

player count is incremented. Any player lines/records read from the file after the limit of *MAX_PLAYERS* players has been reached should be added to the excluded array with appropriate prefix message (`Maximum player count of ___ exceeded for:` where the blank is `MAX_PLAYERS`) and its count should be incremented. If excluded array is full, the line/record should just be skipped, and the ignored count should be incremented. The data file is a "comma separated values" file; i.e., if a line contains multiple values, the values are delimited by commas. So when you set up the scanner for the player lines, you need to set the delimiter to use a "," by calling the `useDelimiter(",")` method on the Scanner object. Each player line in the file begins with a category for the softball player (O, I, P, and R are valid categories for softball players indicating **O**utfielder, **I**nfielder, **P**itcher, and **R**eliefPitcher respectively. The second field in the record is the player's number, followed by the data for the name, position, specialization factor, and batting average. The last items correspond to the data needed for the particular category (or subclass) of SoftballPlayer. For each *incorrect* line scanned (i.e., a line of data contains an invalid category or invalid numeric data), your method will need to handle the invalid items properly. If the line of data begins with an invalid category, your program should throw an InvalidCategoryException (see description above). If a line of data has a valid category, but includes invalid numeric data (e.g., the value for *battingAvg* contains an alphabetic character), a NumberFormatException (see notes on last page) will be thrown automatically by the Java Runtime Environment (JRE). The code that checks for player category should be in a try statement and the code that adds a record with an invalid player category to the excluded records array should now be placed in the catch clause that follows the try statement. That is, your readPlayerFile method should catch and handle InvalidCategoryException and NumberFormatException as follows. In each catch clause, a String object should be created consisting of

```
e + " in: " + line
```

where *e* is the exception and *line* is the line with the invalid data. The String object should be added to the excludedRecords array.

The file *softball_player_data3a.csv* is available for download from the course web site. Below are example data records (the first line/record containing the team name is followed by player lines/records). Note that the last two lines will each cause an exception to be thrown. L is an invalid category and .480a in not a double value.

```
Auburn Stars
O,32,Pat Jones,RF,1.0,.375,.950
I,23,Jackie Smith,3B,1.25,.275,.850
P,43,Jo Williams,RHP,2.0,.125,22,4,2.85
R,34,Sammi James,LHP,2.0,.125,5,4,3.85,17
L,23,Gayle Adams,2B,1.25,.225,.875
O,09,Pat Williams,RF,1.0,.480a,.950
```

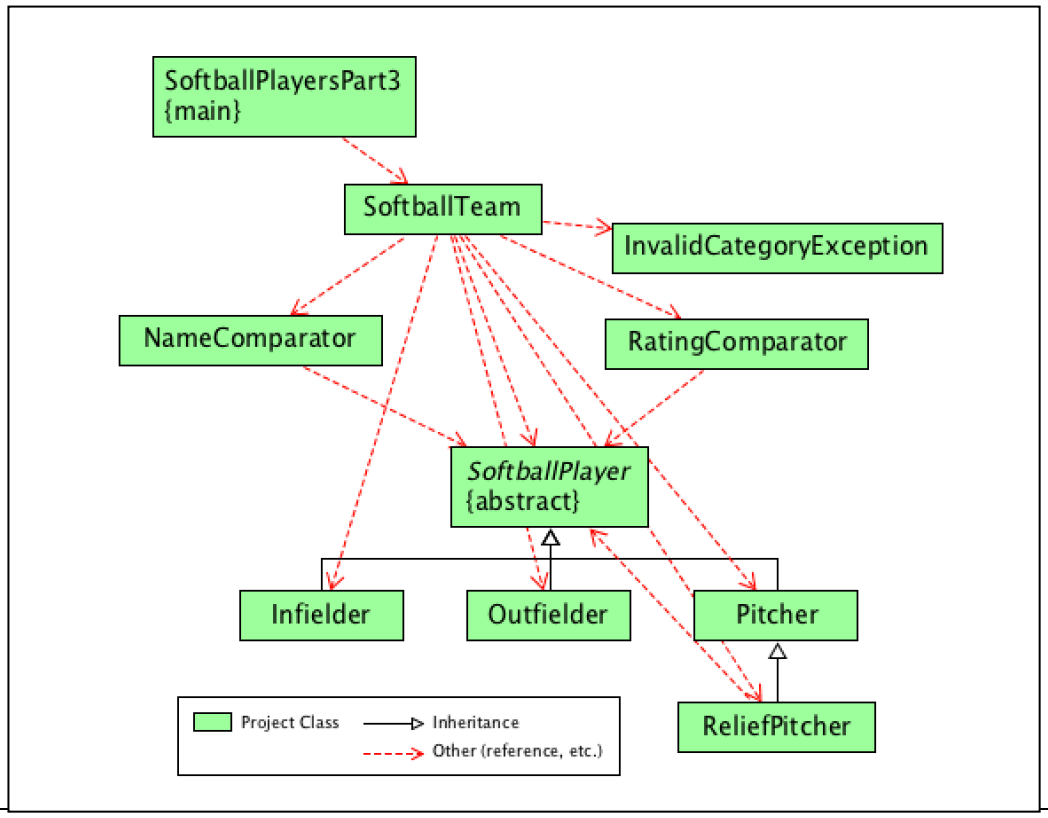
- **SoftballPlayersPart3.java**

  Requirements and Design: The SoftballPlayersPart3 class has only a main method as described below. In addition to the specifications in Project 10, the main method should be modified to catch and handle an IOException if one is thrown in the readPlayerFile method.

  - As before, `main` reads in the file name as the first argument, args[0], of the command line arguments, creates an instance of SoftballTeam, and then calls the readPlayerFile method in the SoftballTeam class to read in the data file and generate the five reports as shown in the output examples beginning on the next page. The main method should not include the *throws IOException* in the declaration. Instead, the main method should include a try-catch statement to catch IOException when/if it is thrown in the readPlayerFile method in the SoftballPlayer class. This exception is most likely to occur when an incorrect file name is passed to the readPlayerFile method. After this exception is caught and the appropriate message is printed in main, your program should end. See the second example output on the following page.

    Example data files can be downloaded from the Lab web page, and the program output for *softball_player_data3a.csv*, *softball_player_data3b.csv, and softball_player_data3c.csv* begins on the next page.

UML Class Diagram



Example Output when file name is missing as command line argument

```
 ----jGRASP exec: java SoftballPlayersPart3
File name expected as command line argument.
Program ending.

 ----jGRASP: operation complete.
```

Example Output when attempting to read a file that is not found

```
 ----jGRASP exec: java SoftballPlayersPart3 not_a_real_file.csv
Attempted to read file: not_a_real_file.csv (No such file or directory)
Program ending.

 ----jGRASP: operation complete.
```

Example Output for `softball_player_data_part3a.csv`

```
 ----jGRASP exec: java SoftballPlayersPart3 softball_player_data_part3a.csv

---------------------------------------
Team Report for Auburn Stars
---------------------------------------

32 Pat Jones (RF) .375
Specialization Factor: 1.0 (class Outfielder) Rating: 3.562

23 Jackie Smith (3B) .275
Specialization Factor: 1.25 (class Infielder) Rating: 2.922

43 Jo Williams (RHP) 22 wins, 4 losses, 2.85 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: 3.740

34 Sammi James (LHP) 5 wins, 4 losses, 17 saves, 3.85 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 2.474


---------------------------------------
Team Report for Auburn Stars (by Number)
---------------------------------------
23 Jackie Smith 3B .275
32 Pat Jones RF .375
34 Sammi James LHP 5 wins, 4 losses, 17 saves, 3.85 ERA
43 Jo Williams RHP 22 wins, 4 losses, 2.85 ERA

---------------------------------------
Team Report for Auburn Stars (by Name)
---------------------------------------
34 Sammi James LHP 5 wins, 4 losses, 17 saves, 3.85 ERA
32 Pat Jones RF .375
23 Jackie Smith 3B .275
43 Jo Williams RHP 22 wins, 4 losses, 2.85 ERA

---------------------------------------
Team Report for Auburn Stars (by Rating)
---------------------------------------
```

```
3.74 43 Jo Williams RHP 22 wins, 4 losses, 2.85 ERA
3.56 32 Pat Jones RF .375
2.92 23 Jackie Smith 3B .275
2.47 34 Sammi James LHP 5 wins, 4 losses, 17 saves, 3.85 ERA

---------------------------------------
Excluded Records Report
---------------------------------------
InvalidCategoryException: For category: "L" in: L,23,Gayle Adams,2B,1.25,.225,.875
java.lang.NumberFormatException: For input string: ".480a" in: O,09,Pat Williams,RF,1.0,.480a,.950
Number of ignored records from file: 0

 ----jGRASP: operation complete.
```

Example Output for `softball_player_data_part3b.csv`

```
 ----jGRASP exec: java SoftballPlayersPart3 softball_player_data_part3b.csv

---------------------------------------
Team Report for My Bigger Team
---------------------------------------

21 Jodi Doe (RF) .305
Specialization Factor: 1.0 (class Outfielder) Rating: 2.989

11 Tina Dobbs (RF) .350
Specialization Factor: 1.0 (class Outfielder) Rating: 3.395

13 Nina Dobbs (LF) .478
Specialization Factor: 1.0 (class Outfielder) Rating: 4.541

12 Poppi Ledet (LF) .325
Specialization Factor: 1.0 (class Outfielder) Rating: 3.120

14 Sruthi Yalamanchili (CF) .285
Specialization Factor: 1.0 (class Outfielder) Rating: 2.679

29 Sandy Chintapalli (1B) .265
Specialization Factor: 1.25 (class Infielder) Rating: 2.915

18 Buddy Bell (2B) .325
Specialization Factor: 1.25 (class Infielder) Rating: 3.494

19 Gigi de la Hoya (2B) .278
Specialization Factor: 1.25 (class Infielder) Rating: 3.301

10 Mikie Mahtook (3B) .298
Specialization Factor: 1.25 (class Infielder) Rating: 3.464

22 Matty Ott (SS) .298
Specialization Factor: 1.25 (class Infielder) Rating: 3.278

23 Leah Coleman (SS) .350
Specialization Factor: 1.25 (class Infielder) Rating: 4.244

25 Erin Noland (RHP) 5 wins, 11 losses, 4.3 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: -.906

26 Jackie Malkovic (RHP) 6 wins, 10 losses, 5.4 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: -.500

27 Lois Gibson (RHP) 8 wins, 7 losses, 3.5 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: .178

28 Gina Malika (LHP) 7 wins, 8 losses, 1.6 ERA
```

```
Specialization Factor: 2.0 (class Pitcher) Rating: -.308

16 Tika Brando (LHP) 9 wins, 7 losses, 1.7 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: .593

30 Belinda Striker (LHP) 10 wins, 6 losses, 10 saves, 1.8 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.333

31 Lilly Dean (RHP) 11 wins, 5 losses, 3 saves, 1.9 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 2.069

32 Briana Wilson (RHP) 4 wins, 4 losses, 14 saves, 2.0 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.111

33 Janine Mason (RHP) 5 wins, 3 losses, 12 saves, 2.1 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.011

34 Green Lantern (LHP) 14 wins, 2 losses, 3 saves, 2.2 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.125

35 Bruce Wayne (LHP) 15 wins, 1 losses, 4 saves, 2.3 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.636

36 Billie Gates (LHP) 16 wins, 0 losses, 2 saves, 2.4 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.529


---------------------------------------
Team Report for My Bigger Team (by Number)
---------------------------------------
10 Mikie Mahtook 3B .298
11 Tina Dobbs RF .350
12 Poppi Ledet LF .325
13 Nina Dobbs LF .478
14 Sruthi Yalamanchili CF .285
16 Tika Brando LHP 9 wins, 7 losses, 1.7 ERA
18 Buddy Bell 2B .325
19 Gigi de la Hoya 2B .278
21 Jodi Doe RF .305
22 Matty Ott SS .298
23 Leah Coleman SS .350
25 Erin Noland RHP 5 wins, 11 losses, 4.3 ERA
26 Jackie Malkovic RHP 6 wins, 10 losses, 5.4 ERA
27 Lois Gibson RHP 8 wins, 7 losses, 3.5 ERA
28 Gina Malika LHP 7 wins, 8 losses, 1.6 ERA
29 Sandy Chintapalli 1B .265
30 Belinda Striker LHP 10 wins, 6 losses, 10 saves, 1.8 ERA
31 Lilly Dean RHP 11 wins, 5 losses, 3 saves, 1.9 ERA
32 Briana Wilson RHP 4 wins, 4 losses, 14 saves, 2.0 ERA
33 Janine Mason RHP 5 wins, 3 losses, 12 saves, 2.1 ERA
34 Green Lantern LHP 14 wins, 2 losses, 3 saves, 2.2 ERA
35 Bruce Wayne LHP 15 wins, 1 losses, 4 saves, 2.3 ERA
36 Billie Gates LHP 16 wins, 0 losses, 2 saves, 2.4 ERA

---------------------------------------
Team Report for My Bigger Team (by Name)
---------------------------------------
18 Buddy Bell 2B .325
16 Tika Brando LHP 9 wins, 7 losses, 1.7 ERA
29 Sandy Chintapalli 1B .265
23 Leah Coleman SS .350
19 Gigi de la Hoya 2B .278
31 Lilly Dean RHP 11 wins, 5 losses, 3 saves, 1.9 ERA
13 Nina Dobbs LF .478
11 Tina Dobbs RF .350
21 Jodi Doe RF .305
36 Billie Gates LHP 16 wins, 0 losses, 2 saves, 2.4 ERA
27 Lois Gibson RHP 8 wins, 7 losses, 3.5 ERA
34 Green Lantern LHP 14 wins, 2 losses, 3 saves, 2.2 ERA
```

```
12 Poppi Ledet LF .325
10 Mikie Mahtook 3B .298
28 Gina Malika LHP 7 wins, 8 losses, 1.6 ERA
26 Jackie Malkovic RHP 6 wins, 10 losses, 5.4 ERA
33 Janine Mason RHP 5 wins, 3 losses, 12 saves, 2.1 ERA
25 Erin Noland RHP 5 wins, 11 losses, 4.3 ERA
22 Matty Ott SS .298
30 Belinda Striker LHP 10 wins, 6 losses, 10 saves, 1.8 ERA
35 Bruce Wayne LHP 15 wins, 1 losses, 4 saves, 2.3 ERA
32 Briana Wilson RHP 4 wins, 4 losses, 14 saves, 2.0 ERA
14 Sruthi Yalamanchili CF .285

---------------------------------------
Team Report for My Bigger Team (by Rating)
---------------------------------------
4.54 13 Nina Dobbs LF .478
4.24 23 Leah Coleman SS .350
3.64 35 Bruce Wayne LHP 15 wins, 1 losses, 4 saves, 2.3 ERA
3.53 36 Billie Gates LHP 16 wins, 0 losses, 2 saves, 2.4 ERA
3.49 18 Buddy Bell 2B .325
3.46 10 Mikie Mahtook 3B .298
3.39 11 Tina Dobbs RF .350
3.33 30 Belinda Striker LHP 10 wins, 6 losses, 10 saves, 1.8 ERA
3.30 19 Gigi de la Hoya 2B .278
3.28 22 Matty Ott SS .298
3.12 34 Green Lantern LHP 14 wins, 2 losses, 3 saves, 2.2 ERA
3.12 12 Poppi Ledet LF .325
3.11 32 Briana Wilson RHP 4 wins, 4 losses, 14 saves, 2.0 ERA
3.01 33 Janine Mason RHP 5 wins, 3 losses, 12 saves, 2.1 ERA
2.99 21 Jodi Doe RF .305
2.92 29 Sandy Chintapalli 1B .265
2.68 14 Sruthi Yalamanchili CF .285
2.07 31 Lilly Dean RHP 11 wins, 5 losses, 3 saves, 1.9 ERA
0.59 16 Tika Brando LHP 9 wins, 7 losses, 1.7 ERA
0.18 27 Lois Gibson RHP 8 wins, 7 losses, 3.5 ERA
-0.31 28 Gina Malika LHP 7 wins, 8 losses, 1.6 ERA
-0.50 26 Jackie Malkovic RHP 6 wins, 10 losses, 5.4 ERA
-0.91 25 Erin Noland RHP 5 wins, 11 losses, 4.3 ERA

---------------------------------------
Excluded Records Report
---------------------------------------
java.lang.NumberFormatException: For input string: "1.0a" in: O,15,Kavya Krishnappa,CF,1.0a,0.298,0.93
java.lang.NumberFormatException: For input string: "0.350b" in: I,17,Janie Doe,1B,1.25,0.350b,0.97
java.lang.NumberFormatException: For input string: "0.94c" in: I,20,Daisy Doalot,3B,1.25,0.285,0.94c
InvalidCategoryException: For category: "H" in: H,24,Nola Austin,LHP,2.0,0.225,4,12,1.2
Number of ignored records from file: 0

 ----jGRASP: operation complete.
```

Example Output for `softball_player_data3c.csv`

```
 ----jGRASP exec: java SoftballPlayersPart3 softball_player_data_part3c.csv

---------------------------------------
Team Report for My Biggest Team File
---------------------------------------

21 Jodi Doe (RF) .305
Specialization Factor: 1.0 (class Outfielder) Rating: 2.989

11 Tina Dobbs (RF) .350
Specialization Factor: 1.0 (class Outfielder) Rating: 3.395
```

```
13 Nina Dobbs (LF) .478
Specialization Factor: 1.0 (class Outfielder) Rating: 4.541

12 Poppi Ledet (LF) .325
Specialization Factor: 1.0 (class Outfielder) Rating: 3.120

14 Sruthi Yalamanchili (CF) .285
Specialization Factor: 1.0 (class Outfielder) Rating: 2.679

29 Sandy Chintapalli (1B) .265
Specialization Factor: 1.25 (class Infielder) Rating: 2.915

18 Codi Bell (2B) .325
Specialization Factor: 1.25 (class Infielder) Rating: 3.494

19 Gigi de la Hoya (2B) .278
Specialization Factor: 1.25 (class Infielder) Rating: 3.301

10 Mikie Mahtook (3B) .298
Specialization Factor: 1.25 (class Infielder) Rating: 3.464

22 Matty Ott (SS) .298
Specialization Factor: 1.25 (class Infielder) Rating: 3.278

23 Leah Coleman (SS) .350
Specialization Factor: 1.25 (class Infielder) Rating: 4.244

25 Erin Noland (RHP) 5 wins, 11 losses, 4.3 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: -.906

26 Jackie Malkovic (RHP) 6 wins, 10 losses, 5.4 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: -.500

27 Lois Gibson (RHP) 8 wins, 7 losses, 3.5 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: .178

28 Gina Malika (LHP) 7 wins, 8 losses, 1.6 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: -.308

16 Tika Brando (LHP) 9 wins, 7 losses, 1.7 ERA
Specialization Factor: 2.0 (class Pitcher) Rating: .593

30 Belinda Striker (LHP) 10 wins, 6 losses, 10 saves, 1.8 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.333

31 Lilly Dean (RHP) 11 wins, 5 losses, 3 saves, 1.9 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 2.069

32 Briana Wilson (RHP) 4 wins, 4 losses, 14 saves, 2.0 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.111

33 Janine Mason (RHP) 5 wins, 3 losses, 12 saves, 2.1 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.011

34 Green Lantern (LHP) 14 wins, 2 losses, 3 saves, 2.2 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.125

35 Brice Wayne (LHP) 15 wins, 1 losses, 4 saves, 2.3 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.636

36 Billie Gates (LHP) 16 wins, 0 losses, 2 saves, 2.4 ERA
Specialization Factor: 2.0 (class ReliefPitcher) Rating: 3.529

37 Anita Jones (RF) .375
Specialization Factor: 1.0 (class Outfielder) Rating: 3.562


---------------------------------------
Team Report for My Biggest Team File (by Number)
```

```
---------------------------------------
10 Mikie Mahtook 3B .298
11 Tina Dobbs RF .350
12 Poppi Ledet LF .325
13 Nina Dobbs LF .478
14 Sruthi Yalamanchili CF .285
16 Tika Brando LHP 9 wins, 7 losses, 1.7 ERA
18 Codi Bell 2B .325
19 Gigi de la Hoya 2B .278
21 Jodi Doe RF .305
22 Matty Ott SS .298
23 Leah Coleman SS .350
25 Erin Noland RHP 5 wins, 11 losses, 4.3 ERA
26 Jackie Malkovic RHP 6 wins, 10 losses, 5.4 ERA
27 Lois Gibson RHP 8 wins, 7 losses, 3.5 ERA
28 Gina Malika LHP 7 wins, 8 losses, 1.6 ERA
29 Sandy Chintapalli 1B .265
30 Belinda Striker LHP 10 wins, 6 losses, 10 saves, 1.8 ERA
31 Lilly Dean RHP 11 wins, 5 losses, 3 saves, 1.9 ERA
32 Briana Wilson RHP 4 wins, 4 losses, 14 saves, 2.0 ERA
33 Janine Mason RHP 5 wins, 3 losses, 12 saves, 2.1 ERA
34 Green Lantern LHP 14 wins, 2 losses, 3 saves, 2.2 ERA
35 Brice Wayne LHP 15 wins, 1 losses, 4 saves, 2.3 ERA
36 Billie Gates LHP 16 wins, 0 losses, 2 saves, 2.4 ERA
37 Anita Jones RF .375

---------------------------------------
Team Report for My Biggest Team File (by Name)
---------------------------------------
18 Codi Bell 2B .325
16 Tika Brando LHP 9 wins, 7 losses, 1.7 ERA
29 Sandy Chintapalli 1B .265
23 Leah Coleman SS .350
19 Gigi de la Hoya 2B .278
31 Lilly Dean RHP 11 wins, 5 losses, 3 saves, 1.9 ERA
13 Nina Dobbs LF .478
11 Tina Dobbs RF .350
21 Jodi Doe RF .305
36 Billie Gates LHP 16 wins, 0 losses, 2 saves, 2.4 ERA
27 Lois Gibson RHP 8 wins, 7 losses, 3.5 ERA
37 Anita Jones RF .375
34 Green Lantern LHP 14 wins, 2 losses, 3 saves, 2.2 ERA
12 Poppi Ledet LF .325
10 Mikie Mahtook 3B .298
28 Gina Malika LHP 7 wins, 8 losses, 1.6 ERA
26 Jackie Malkovic RHP 6 wins, 10 losses, 5.4 ERA
33 Janine Mason RHP 5 wins, 3 losses, 12 saves, 2.1 ERA
25 Erin Noland RHP 5 wins, 11 losses, 4.3 ERA
22 Matty Ott SS .298
30 Belinda Striker LHP 10 wins, 6 losses, 10 saves, 1.8 ERA
35 Brice Wayne LHP 15 wins, 1 losses, 4 saves, 2.3 ERA
32 Briana Wilson RHP 4 wins, 4 losses, 14 saves, 2.0 ERA
14 Sruthi Yalamanchili CF .285

---------------------------------------
Team Report for My Biggest Team File (by Rating)
---------------------------------------
4.54 13 Nina Dobbs LF .478
4.24 23 Leah Coleman SS .350
3.64 35 Brice Wayne LHP 15 wins, 1 losses, 4 saves, 2.3 ERA
3.56 37 Anita Jones RF .375
3.53 36 Billie Gates LHP 16 wins, 0 losses, 2 saves, 2.4 ERA
3.49 18 Codi Bell 2B .325
3.46 10 Mikie Mahtook 3B .298
3.39 11 Tina Dobbs RF .350
3.33 30 Belinda Striker LHP 10 wins, 6 losses, 10 saves, 1.8 ERA
3.30 19 Gigi de la Hoya 2B .278
3.28 22 Matty Ott SS .298
3.12 34 Green Lantern LHP 14 wins, 2 losses, 3 saves, 2.2 ERA
```

```
3.12 12 Poppi Ledet LF .325
3.11 32 Briana Wilson RHP 4 wins, 4 losses, 14 saves, 2.0 ERA
3.01 33 Janine Mason RHP 5 wins, 3 losses, 12 saves, 2.1 ERA
2.99 21 Jodi Doe RF .305
2.92 29 Sandy Chintapalli 1B .265
2.68 14 Sruthi Yalamanchili CF .285
2.07 31 Lilly Dean RHP 11 wins, 5 losses, 3 saves, 1.9 ERA
0.59 16 Tika Brando LHP 9 wins, 7 losses, 1.7 ERA
0.18 27 Lois Gibson RHP 8 wins, 7 losses, 3.5 ERA
-0.31 28 Gina Malika LHP 7 wins, 8 losses, 1.6 ERA
-0.50 26 Jackie Malkovic RHP 6 wins, 10 losses, 5.4 ERA
-0.91 25 Erin Noland RHP 5 wins, 11 losses, 4.3 ERA

---------------------------------------
Excluded Records Report
---------------------------------------
java.lang.NumberFormatException: For input string: "1.0a" in: O,15,Kavya Krishnappa,CF,1.0a,0.298,0.93
java.lang.NumberFormatException: For input string: "0.350b" in: I,17,Janie Doe,1B,1.25,0.350b,0.97
java.lang.NumberFormatException: For input string: "0.94c" in: I,20,Daisy Doalot,3B,1.25,0.285,0.94c
InvalidCategoryException: For category: "H" in: H,24,Nola Austin,LHP,2.0,0.225,4,12,1.2
Maximum player count of 24 exceeded for: O,38,Betty Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,39,Cate Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,40,Dee Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,41,Edie Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,42,Fay Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,43,Gigit Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,44,Hattie Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,45,Isabel Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,46,Jane Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,47,Kathy Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,48,Lola Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,49,Mary Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,50,Nina Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,51,Olivia Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,52,Pat Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,53,Quie Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,54,Reta Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,55,Siena Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,56,Tina Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,57,Ubi Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,58,Victoria Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,59,Willow Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,60,Xena Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,61,Yani Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,62,Zeta Jones,RF,1.0,.375,.950
Maximum player count of 24 exceeded for: O,63,Zaata Jones,RF,1.0,.375,.950
Number of ignored records from file: 5

 ----jGRASP: operation complete.
```

Notes:

1. This project assumes that you are reading each double value as a String using next() and then parsing it into a double with Double.parseDouble(...) as shown in the following example.

   ```
   . . . Double.parseDouble(myInput.next());
   ```

   This form of input will throw a java.lang.NumberFormatException if the value is not a double.

   If you are reading in each double value as a double using nextDouble(), for example

   ```
   . . . myInput.nextDouble();
   ```

   then a java.util.InputMismatchException will be thrown if the value read in is not a double.

For this assignment, you should change your input to use Double.parseDouble(...) rather than nextDouble(), since Web-CAT is looking for NumberFormatException rather than java.util.InputMismatchException.

2. If you are using the JUnit Assert.assertArrayEquals method to check two Cardholder arrays for equality, then the `equals` and `hashCode` methods must be implemented in your Cardholder class; that is, Assert.assertArrayEquals calls equals(Object obj) on each object in the array, so Cardholder must have an equals method that overrides the one inherited from the Object class. If Cardholder does not override equals(Object obj), then the JUnit Assert.assertArrayEquals method will use the inherited equals(Object obj) method which means two Cardholder arrays will be equal only if they are aliases.

   Below is a simplified `equals` method and `hashCode` method you are free to use.

```
public boolean equals(Object obj) {
   if (!(obj instanceof SoftballPlayer)) {
      return false;
   }
   else {
      Cardholder c = (SoftballPlayer) obj;
      return (name.equalsIgnoreCase(c.getName()));
   }
}

public int hashCode() {
   return 0;
}
```