Project 3

CMPT220L
Due on Nov 13, 2020 by 11:59 PM
Points: 100

Baby Name Popularity Ranking

Problem Description

The popularity ranking of baby names from years 2001 to 2010 is downloaded from `www.ssa.gov/oact/babynames` and stored in files named `babynameranking2001.txt`, `babynameranking2002.txt`, ..., `babynameranking2010.txt`. Each file contains one thousand lines. Each line contains a ranking, a boy's name, number for the boy's name, a girl's name, and number for the girl's name. For example, the first two lines in the file `babynameranking2010.txt` are as follows:

```
1  Jacob 21,875  Isabella  22,731
2  Ethan 17,866  Sophia  20,477
```

So, the boy's name Jacob and girl's name Isabella are ranked #1 and the boy's name Ethan and girl's name Sophia are ranked #2. 21,875 boys are named Jacob and 22,731 girls are named Isabella.

Write a program that prompts the user to enter the year, gender, and followed by a name, and displays the ranking of the name for the year. Here is a sample run:

```
Enter the year: 2010
Enter the gender: M
Enter the name: Javier
Javier is ranked #190 in year 2010
```

```
Enter the year: 2010
Enter the gender: F
Enter the name: ABC
The name ABC is not ranked in year 2010
```

Deliverables

1. Create a UML diagram for a class called `BabyName`. The class should hold all the information related to a baby name. **10 Points**
2. Implement the `BabyName` class. **20 Points**
3. Create a method with the following header and use it to load the names from the file: **20 Points**

   ```
   private ArrayList<BabyName> loadNames(String fileName)
   ```

4. Create a method with the following header to implement the search of names: **20 Points**

   ```
   private static BabyName findName(String name, int year, ArrayList<BabyName> names)
   ```

5. Create Java program called `Project3`. **20 Points**
6. Comment your code. **10 Points**

Submission

Submit the following items:

1. Create a PDF file with the UML diagram and submit to GitHub.
2. Compile, test, and submit your Java program to GitHub (you must submit the program regardless whether it's complete or incomplete, correct or incorrect)

Place your `.java` file under the corresponding folder in your local copy of the GitHub repository, commit and push it to the remote repository. Make sure that the professor has access to the repository (`jfac65-marist`).

```
cmpt220lastname\
   prj\
      3\
         Project3.pdf
         Project3.java
```