

## Checklist for the Baby Names Project:

I don't usually create such detailed guidelines for a project, but realizing that you may not have recalled all of them after break, I did my best to assemble a checklist to help you include all details in your projects. Make sure to read the project description too. I hope this helps.

### External Perfection:

- Make sure you print out the line of data and line of meaning to console, not just create the graphics. Output should look exactly like this (without bolding):  

```
This program graphs the popularity of a name
in statistics recorded since the year 1900.

Type a name: lisa
Lisa 0 0 0 0 464 38 1 6 31 113 298
LISA f English, German, Swedish Short form of ELIZABETH or ELISABETH.
```
- If the name is not found, no graphics nor the meaning should be generated. Simply:  

```
Type a name: zOIDberG
"zOIDberG" not found.
```
- Make sure your program work properly with any number of decades of data (at least 1), test both "names.txt" and "names2.txt" file.
- Account for changes to the Class Constants:
  - Introduction's year should reflect the starting year Class Constant (may not always be 1900).
  - Make sure the width of the bars change with the class constant for width of each decade. i.e. for the width to 50, each red decade bar is 50px apart and 25px (half) thick.
- Things NOT to worry about (no points taken off for):
  - You do NOT need to scale the width of the overall graphics panel with the decade width.
  - Don't worry if the definition prints out beyond the width of the graphics panel.
- Follow all specific guidelines in the project description for creating the Graphics output. Make sure to account for 0's and scale bar heights correctly.

### Internal Greatness:

- Include two Class Constants (please place them in this order):
  - Starting year of the input data, as an integer (default of 1900)
  - The width of each decade on the DrawingPanel, as an integer (default of 60)
- Each method should perform a single clear and coherent task; Separate out functionality from input/output:
  - Search for name method should not read input nor print out.
  - The method that asks the user for a name should not also read lines of input from a file.
  - Split the displaying of graphical data into at least two methods.** For example, you could have one method to draw "fixed" graphics (yellow bars, etc.) and another for graphics that come from the file (red bars, ranks).
- Avoid redundancy: Make your Search for name in a file method reusable for both files.
- Don't scan files more than once – return key data and store values in variables.
- The main method should contain any key variables of data and pass them clearly so the reader can understand how information flows in the program.
- The main method should not draw on a DrawingPanel, nor read lines of input from a file (nextLine).
- Use overall good style: include header block, method comments, good variable & method names, lines of 100 character width, etc. Please add comments for any tricky bits.