# Project 1: Hangman Game

The goal of this project is to become more familiar with creating and using classes and objects in Java, to further introduce you to the concepts of abstraction and modularization, and to introduce you to collections of items. You will complete your Hangman so it contains everything we need to display and control your game.

# Instructions

1. You should have the interface for all of the pieces needed to complete the game, now your job is to make them work together.
   1. Coordinate the user input to appropriately change the alphabet panel, the person, and the guess phrase based on each letter pressed. You should have one key listener in this class and nowhere else (but you might be able to copy this code from AlphabetPanel’s key listener code). I recommend making your HangmanGame a JFrame, and put the code that normally goes in main for a JPanel into Hangman’s constructor. Then the HangmanGame main simply becomes: **new** HangmanGame();. See the other project descriptions for more details about handling user input.
   2. Ask the user if they want to play again. If they do, reset the guess phrase to a new phrase and reset the rest of the game.
   3. NOTES:
      1. Clicking in the window should not advance the person (so delete the mouse event handling that might be in person)
      2. Only incorrect letters should make a part of the person show up and **ONLY** the first time the incorrect letter is pressed (pressing the same incorrect letter over and over should not cause any more body parts to show).
2. **Style**: be sure to…
3. Comment each of your methods (including their parameters) & anything else that needs explanation.
4. Use proper indentation
5. Use appropriate variable/field names
6. Check your curly braces
7. EXTRA CREDIT: Add comments for correct/incorrect answers. NOTE: You can use two files for this and reuse the RandomString class. Display a corresponding comment from the person being hung, negative if the user/player guessed wrong, positive or encouraging if the user/player got it right.

# Hints / tips

* For the comments positive and negative you can reuse your RandomString class from Project 5
* When incorporating multiple panels in a single frame, you will need to set a layout scheme, such as BoxLayout.
* As part of project 1 you need to make the logic of the game work properly as well as ask the user if they would like to play again. You may find it useful to use the following logic:  
   **if** person or phrase is revealed  
   process the user's Y/N answer  
   **else**  
   process the input as usual for the game, updating the person, guess phrase panel,   
   & alphabet panel as appropriate  
   **if** person or phrase is revealed  
   ask the user if they would like to play again
* Your should create methods for processLetter (takes a letter as a parameter) and a reset method