AP CS A Celebrity Lab Checklist

1. ☐ Work through Lab exercises on paper for parts 1-3
2. ☐ Write Celebrity.java (2)
3. Write the PSVM in CelebrityRunner
   a. ☐ Create a CelebrityGame instance
4. Finish CelebrityGame.java (3)
   a. ☐ Data members
      i. ☐ `Celebrity` instance `gameCelebrity`
      ii. ☐ `ArrayList<Celebrity>` instance `celebGameList`
      iii. ☐ `CelebrityFrame` instance `gameWindow`
   b. Constructor
      i. ☐ Initialize data members
      ii. ☐ Call `prepareGame` helper
   c. Methods
      i. ☐ Complete `prepareGame`
      ii. ☐ Complete game size method
      iii. Complete Celebrity related methods
         1. ☐ `sendClue`
         2. ☐ `sendAnswer`
      iv. Complete validation methods
         1. ☐ `validateCelebrity`
         2. ☐ `validateClue`
      v. ☐ Complete `addCelebrity`
      vi. Complete `play`
         1. ☐ Check the size of the `ArrayList<Celebrity>`
         2. ☐ Set the current celebrity
         3. ☐ Call the `replaceScreen` method properly
      vii. Complete `processGuess`
         1. ☐ Use the `String` methods `equalsIgnoreCase(text)` and `trim()` on the parameter
      viii. ☐ Complete sendClue()
5. Complete the `CelebrityFrame` class (GUI Appendix)
   a. Constructor
      i. ☐ Leave the `super()` call where it is
      ii. ☐ Initialize the controller data member from the parameter
      iii. Initialize the three panels
         1. ☐ `panelCards` with a parameter of a new `CardLayout()`
         2. ☐ `gamePanel` with a reference to the controller as a parameter
         3. ☐ `startPanel` with a reference to the controller as a parameter
         4. ☐ Call the `setupFrame` helper

b. `setupFrame` helper method
   i. ☐ Add the `gamePanel` and `startPanel` to the `panelCards` member with the Strings GAME and START respectively
   ii. ☐ Set the size of the frame to 800 by 800
   iii. ☐ Set the title to be Celebrity Game
   iv. ☐ Add the panel cards to the frame
   v. ☐ Set the default close operation
   vi. ☐ Set the window to NOT be resizable
   vii. ☐ Call `replaceScreen` with a parameter of START
   viii. ☐ Set the window to be visible!!
c. `replaceScreen` method
   i. ☐ Check the if the parameter equals GAME
      1. ☐ If it does, call `gamePanel.addClue` with a parameter of `controller.sendClue()`
   ii. ☐ Tell the `panelCards` layout manager to show the correct screen
6. Complete the StartPanel class
   a. The setupPanel helper method
      i. ☐ Set the layout manager
      ii. Add all the GUI components
         1. ☐ clueLabel
         2. ☐ celebrityRadio
         3. ☐ literatureRadio
         4. ☐ answerField
         5. ☐ startButton
         6. ☐ celebrityCountLabel
         7. ☐ addCelebrityButton
      iii. ☐ Set the default type of Celebrity
      iv. ☐ Enable the startButton
      v. ☐ Add both RadioButtons to the typeGroup