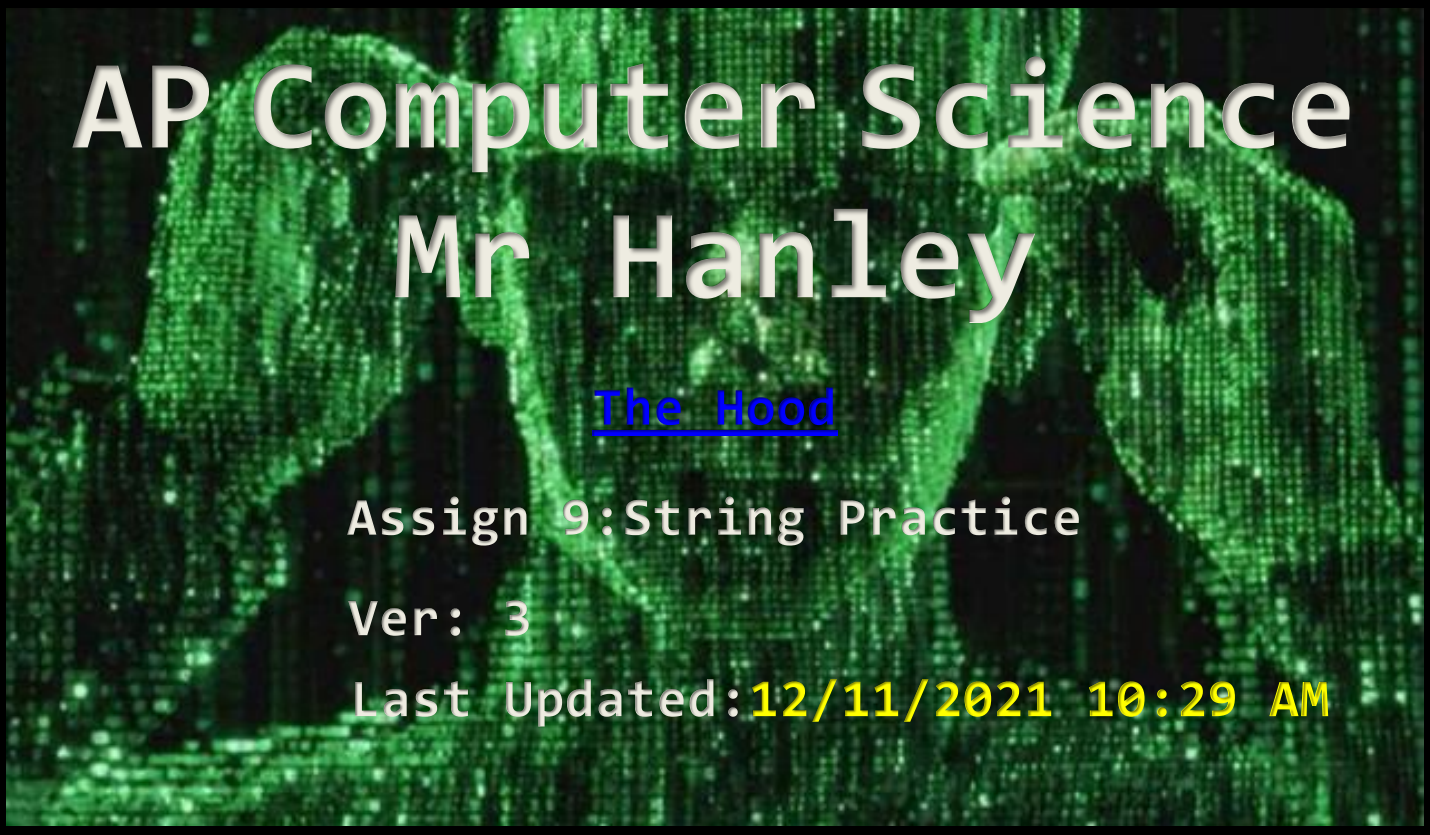

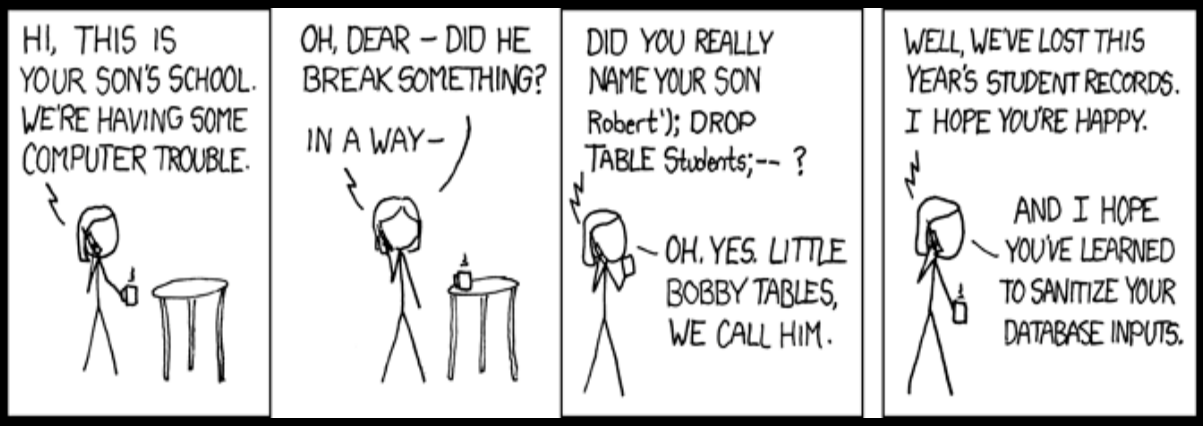

Assignment 9: String Practice

Binary **Ones Comp** **Twos Comp**

Write a class that features a series of String methods

There is a file called StringPractSkeleton.zip on the website

Download the StringPractice Skeleton from website

Expand the Skeleton in your account
Look for the file called StringMethods.java

Each of the methods is **"stub coded",** which means the name and inputs and return values are set up but the **details are missing**.

You must finish them one by one.
You can test them using the Graphical User Interface. By clicking the buttons, you will see if your result is correct!!

To finish the compareTo, you will need to add some code to the Frame

1. Have the user enter a song title. Count the number of words in the song title

2. Build a String of Pascal's Triangle according to the number of rows input
   - Examples;
     Input 4:

```
          1
        1   1
      1   2   1
    1   3   3   1
```

Input 5:

```
          1
        1   1
      1   2   1
    1   3   3   1
  1   4   6   4   1
```

For #'s 3 and 4, you will be connecting to the internet to download the html text of a web site. A method inside StringMethods.java called openURL will handle this for you!!!

3. Given a typical html file, write a method that returns the title of the page. Most pages have a title
   (If no title is found, please return "No Title" so the GUI can display the appropriate message)

4. Given a typical html file, write a method that returns an array of strings, each array element contains one of the hyperlinks on the page
   NOTE: You are only responsible for <a href= links, many pages have other links as well, like href=, I ignore these but you are welcome to include them

   public String[] findLinks(String page)
   NOTE: You will have to count the links first to set the array size or return an ArrayList to properly handle the exact number of links(requires change to Frame if you want to return an ArrayList)

For the page
https://sites.google.com/a/shenschools.org/augustbiology/home
as of 3/9/2021 You should get 24 links
NOTE: I go to the next > which are why mine are so long, should probably just go to the next "/" and then they would be shorter

"https://sites.google.com/a/shenschools.org/augustbiology/" id="sites-chrome-userheader-logo"
"https://sites.google.com/a/shenschools.org/augustbiology/" dir="ltr" id="sites-chrome-userheader-title"
"/a/shenschools.org/augustbiology/home/ap-biology" jotId="wuid:gx:4c33d97c7387afb8" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/ap-biology/course-information" jotId="wuid:gx:52f7b0a22cfcd683" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/ap-biology/ap-biology-homework" jotId="wuid:gx:7c53d220cbc0b865" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/ap-biology/resources" jotId="wuid:gx:d6611b161753f7d" class="sites-navigation-link"

"/a/shenschools.org/augustbiology/home/ap-biology/resources/class-notes"
jotId="wuid:gx:353a05b289e432b9" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment"
jotId="wuid:gx:795586925338595a" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment/course-information"
jotId="wuid:gx:2455e156bcf84054" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment/homework"
jotId="wuid:gx:2e21d84598922722" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment/regents-biology-resources" jotId="wuid:gx:6dedbbdd943585ec" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment/regents-biology-resources/class-notes" jotId="wuid:gx:3698265d10556b34" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment/regents-biology-resources/labs" jotId="wuid:gx:4205162af3e9a965" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment/regents-biology-resources/review" jotId="wuid:gx:243a3f00439ddd26" class="sites-navigation-link"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment/regents-biology-resources/worksheets" jotId="wuid:gx:e7536ee576e1ea7" class="sites-navigation-link"
"mailto:augugabr@shenet.org"
"http://www.shenet.org" target="_blank" rel="nofollow"
"/a/shenschools.org/augustbiology/home/ap-biology" dir="ltr"
"/a/shenschools.org/augustbiology/home/regents-biology-living-environment" dir="ltr"
"mailto:augugabr@shenet.org" target="_blank"
"http://sites.google.com/site"

For the page
https://racketlounge.com/history-of-tennis/
as of 3/9/2021 You should get 26 links

"https://racketlounge.com"
"https://racketlounge.com/tennis/" class="menu-link elementor-item"
"https://racketlounge.com/pickleball/" class="menu-link elementor-item"
"https://racketlounge.com/squash/" class="menu-link elementor-item"
"https://racketlounge.com/racquetball/" class="menu-link elementor-item"
"https://racketlounge.com/table-tennis/" class="menu-link elementor-item"
"https://racketlounge.com/badminton/" class="menu-link elementor-item"
"https://racketlounge.com/tennis/" class="menu-link elementor-item"
"https://racketlounge.com/pickleball/" class="menu-link elementor-item"
"https://racketlounge.com/squash/" class="menu-link elementor-item"
"https://racketlounge.com/racquetball/" class="menu-link elementor-item"
"https://racketlounge.com/table-tennis/" class="menu-link elementor-item"
"https://racketlounge.com/badminton/" class="menu-link elementor-item"
"https://racketlounge.com/category/tennis/" rel="category tag"
"https://www.topcourt.com/n/ROUBEN-2XKFYA6WNW" target="_blank" rel="nofollow"
class="elementor-button-link elementor-button elementor-size-md" role="button"
"https://www.olympic.org/news/a-brief-history-of-tennis" rel=""
"https://www.topcourt.com/n/ROUBEN-2XKFYA6WNW" target="_blank" rel="noopener"
"https://www.topcourt.com/n/ROUBEN-2XKFYA6WNW" target="_blank" rel="nofollow"
"https://www.topcourt.com/n/ROUBEN-2XKFYA6WNW" target="_blank" rel="nofollow"
class="elementor-button-link elementor-button elementor-size-md" role="button"
"https://hi.switchy.io/TB" target="_blank"
"https://racketlounge.com/best-tennis-ball-machines/" rel="prev"

"https://racketlounge.com/best-ping-pong-paddles/" rel="next"
"https://racketlounge.b-cdn.net/about-us/"
"https://racketlounge.b-cdn.net/contact/"
"https://swiy.io/topcourt" target="_blank" class="elementor-button-link elementor-button elementor-size-md" role="button"
https://malcare.com?src=BA42AD&utm_source=mcbadge&utm_medium=usersite&utm_campaign=badge target="_blank"

5. Allow a user to enter two strings, print out the result of the first compareTo'd the second(you will need to edit the GUI to get it to work since there is really no need to deal with the StringMethods.java on this one.)

6. Get an password from the passwordField.
   You will get in the String from the GUI and a Level...

   A. Level 1: Then decide if the password is at least 6 characters. If the password is less than 6 characters, give the user an error message. If the password is greater than or equal to 6 characters and has at least one non alpha, return true and a happy face will appear in the GUI

      Also confirm that at least one of the characters is non alphabetical (symbol or number) HINT:
      Character.isLetter(text.charAt(i)) is helpful here!!!!

      If the character is not sufficient length or doesn't contain at least one non-alphabetical character, then sad face (return false)

   B. Second Level Password: Level 2
      Second level password requirements:
      Length: 8, At least one upper, At least one digit, At least one symbol(non letter and non digit)

Project Name	AP Student String SolutionNB
Class 1 Name	StringApp
Class 2 Name	StringMethods(this is the primary file you are working with)
Class 3 Name	StringFrame

RUBRIC	
Song Title	20
Pascal's Triangle	35
Get the Title	20
Get the Links	40
Compare To	10
Password Checker	10 – first, 15 second
Comments (finish any unfinished pre or post conditions) and add comments to your String Methods	10
TOTAL	160

Artificial_Intelligence
Dynamic_Memory
Waterfall_Model
Nested_Loops
Stacks
Short_Circuit_Evaluation
Insertion_Sort
Binary_Search
Inheritance
Files
Linear_nlogn
Extreme_Programming
Client_Server
Hong_Kong_Fooey
Video_Games
String
Recursion
Search
File_Processing
Arrays
Selection_Sort
Big_O_Notation
2D_Arrays
Hangman
ArrayList