# Assignment 6: Sorting Words

## Objectives

* Work with if statements.
* Work with Strings
* Compare to see if two strings are equal.
* Compare multiple strings in order to sort them.

## Motivation

This is a variation on problem number 7 on page 186 in the textbook, except that all names should be unique.

## Instructions

Name your project FirstnameLastnameAssignment6

Have your program do the following.

1. Greet the user.
   * "Welcome to Dr. Church's Name Sorter."
   * "All names must be unique."
2. You should prompt the user for three names.
   * Each prompt should be clearly labeled. The grader must count off if you say "Enter three names". You should prompt for each individual name.
   * A name can be any sequence of characters and will include spaces, which means the Scanner.next method is insufficient for this assignment.
3. You should then make sure that no two names are the same.
   * In other words, "Alice", "Rebecca", and "Alice" uses a duplicate name. Your program should report that the first and third names are identical. It should not print anything more in the application.
4. (Provided that all three names are unique.) The program should then print the names alphabetically.

Notes:

* Your instructor doesn't like nested if statements, so don't use them.
* Your instructor does like boolean flags, so you are encouraged to use them on this assignment.
* Your instructor did use logical operators.
* This assignment will make use of String.equals and String.compareTo.

Your source code must include the following documentation:

* Your name
* The class (CS 2070) and the section number (on ground is 08, online is W1).
* The date on which you turned in the assignment.
* A short description of the software. Usually a sentence or two is sufficient.

## Example Run

Welcome to Dr. Church's Name Sorter.  
All names must be unique.  
Enter the first name: Eve  
Enter the second name: Alice  
Enter the third name: Bob  
  
Here are the sorted names.  
Alice  
Bob  
Eve

## Example Run

Welcome to Dr. Church's Name Sorter.  
All names must be unique.  
Enter the first name: Alice  
Enter the second name: Bob  
Enter the third name: Eve  
  
Here are the sorted names.  
Alice  
Bob  
Eve

## Example Run

Welcome to Dr. Church's Name Sorter.  
All names must be unique.  
Enter the first name: Alice  
Enter the second name: Rebecca  
Enter the third name: Alice  
  
Names one and three are identical.

## Example Run

Welcome to Dr. Church's Name Sorter.  
All names must be unique.  
Enter the first name: Alice  
Enter the second name: Bob  
Enter the third name: Bob  
Names two and three are identical.

## Turning it in.

To turn in your application, find the folder containing your entire project (not the folder with the "java" file), zip it up, and turn it in.