CITP 190 – Introduction to Java
Project 10

This project requires you to create one project. The project will work with students. Please be sure to download and unzip the project start files. You will need to add this file to your Project in the same manner that you added the start file to Project 9.

To receive full credit for this project you must submit the following:

- A detailed design diagram for `main()` and for each method in the Student class.
- A class diagram for the Student class.
- The source code for each source code file (the .java file) following the coding standards.
- The bytecode for each source code file (the.class file)
- Proof of the correctness of your output using the test data provided. Please note that you must provide proof for all test data. You may provide additional test data.
- A capture of the output of the program. The output must show all data from your proof.

Submit all files as one (1) ZIP file to the Project 10 Drop Box in the course site.

Grading:

Program design	5 points
Design diagrams	5 points
Following course standards	5 points
Proof	3 points
Screen captures	2 points

Important notes:

1. Output that is not presented as shown (including spaces and spelling) will result in a 0 grade for this project.
2. Code that does not follow the standards will result in a 0 grade for this project.
3. Uploading more than one zip file will result in a 0 grade for this project.
4. Not using all the test data provided will result in a 0 grade for this project.

Project 10: Display a sorted list of student scores

Console

```
Welcome to the Student Scores Application.

Enter number of students to enter: 4

Student 1 last name: Steelman
Student 1 first name: Andrea
Student 1 score: 95

Student 2 last name: Murach
Student 2 first name: Joel
Student 2 score: 92

Student 3 last name: Lowe
Student 3 first name: Doug
Student 3 score: 82

Student 4 last name: Murach
Student 4 first name: Mike
Student 4 score: 93

Lowe, Doug: 82
Murach, Joel: 92
Murach, Mike: 93
Steelman, Andrea: 95
```

Operation

- This application asks the user for the number of students that will be entered.
- The application then accepts the last name, first name, and score for the number of students specified by the user and stores the student information in an array.
  Important Note: although the test data has 6 students, the program must work for any number of students including zero.
- Once all students have been entered, the application prints the students and their scores in alphabetical order by last name. Note the format of the display - the student's last name, a comma, a space, the student's first name, a colon, a space, the student's score.

Specifications

- The program should implement a class named Student that:
  - stores the last name, first name, and score for each student.
  - has appropriate constructors, setters and getters.
  - implements the IComparable interface so the students can be sorted by name. If two students have the same last name, the first name should be used to determine the final sort order.
- The program should use an array to store the Student objects. Then, it should sort the array prior to printing the student list.
- The program should use a for loop for the entry of the student data and an enhanced for loop to display the student information.
- Use the SortedStudentsApp class as a start for your project. It contains a start for the the `main()` method and the following methods:
  - `getStudentCount` - This method should be used to get the number of students the user will enter.
  - `getString` - This method should be used to get all string data from the user.
  - `getScore` - This method should be used to get the score from the user.

For an extra challenge (but no extra points) modify the program so the user can sort by name or by score. If two students have the same score, the last name should be used to determine the final sort order. If two students have the same score and the same last name, the first name should be used to determine the final sort order.

Test Data

- Number of students to enter: 6

First Name	Last Name	Score
Kris	Kringle	85
Mary	Jones	96
Sam	Spade	68
Terry	Programmer	84
Sky	Masterson	75
Indiana	Jones	48

Proof

For this program the proof is what the program should display after all the students have been entered.