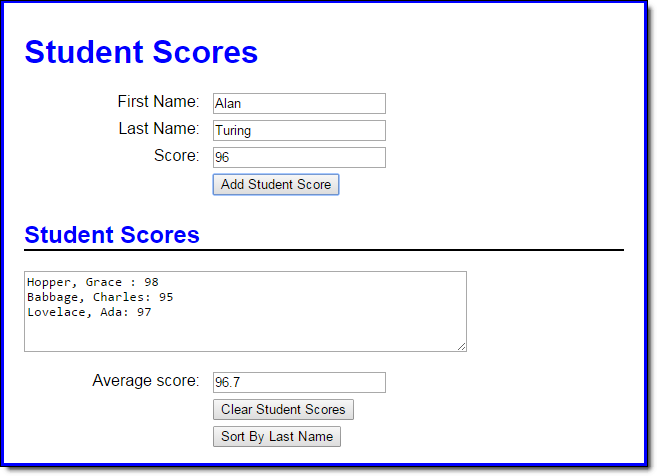
# Lab 8-1 Develop the Student Scores application

In this exercise, you’ll develop an application that tracks student’s scores, tallies the average of the entered scores, and sorts the entered students by last name. The interface looks like this:



1. Open the HTML and JavaScript files in this folder.
2. In the JavaScript file, note that six functions are supplied. The $ function. The start of a displayScores function. The start of an addScore function that ends by clearing the add form and setting the focus on its first field. The start of a clearScores function that ends by clearing the display area and setting the focus on the first name field. The start of a sortScores function. And an onload event handler that attaches the addScore, clearScores, and sortScores functions to the click events of the appropriate buttons and sets the focus on the first name field.
3. To start, code two global arrays, one to hold the score values and the other to hold the strings that display the students’ names and scores.
4. In the displayScores function, add the code that calculates the average score of all the scores in the first array, and stores it in the text box below the text area. Then, add the code that gets the students’ names and scores in the second array and displays it in the text area.
5. In the addScore function, use the push method to save the score in the first array and to save the name and score string (as shown in the text box) in the second array. Then, call the displayScores function to redisplay the updated data.
6. In the clearScores function, add code that clears both global arrays.
7. In the sortScores function, add code that sorts the students by last name and then re-displays the score information.

# Lab 8-2 Develop the Student Scores application with objects

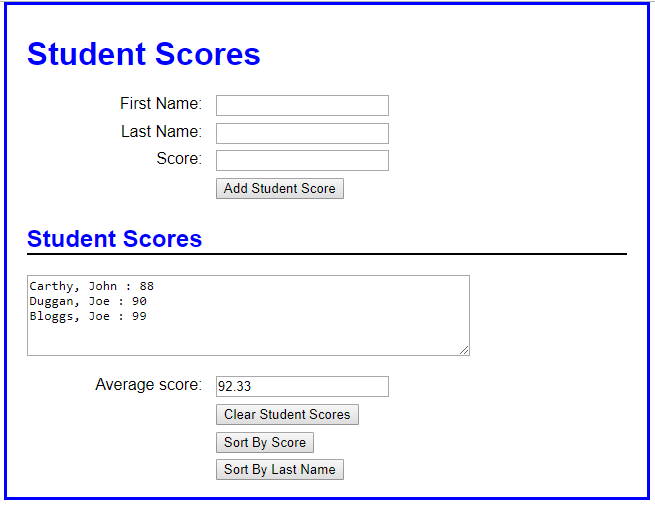
In this exercise, you’ll develop the same application as before but this time using objects to store each students’ details. E.g.

{ firstName:”Joe”, lastName:”Bloggs”, score: 99 }

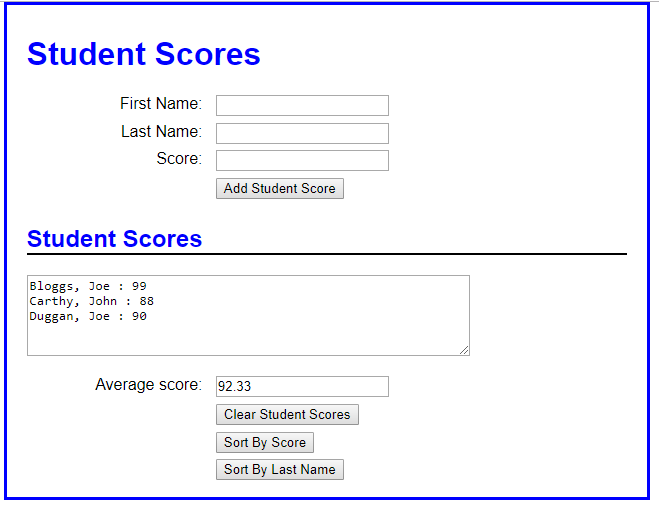
There’s no need to write methods for this object, just a constructor function. All the functions should work the same as the previous application but with the following additions:

1. The student data should be stored in localStorage so that it persists between browser sessions.
2. You should add an extra button to sort by score.

Sorted by Score:



Sorted by lastName:



Notes:

* You’ll use JSON.stringify() and JSON.parse() to serialise/deserialise the array of student objects for localStorage.
* You should try out some of the array methods we studied e.g.
  + forEach() could be used to sum the scores when getting the average, and when displaying the names. Try using an arrow function.
  + You’ll need to use sort() with a comparison function. See the notes for details.