# Selection Sort

1. **SelectionSortSimple.java**:  Create a method `selectionSort` that uses a selection sort to arrange an array of double values in ascending order. Create a `main` method in the class to test the method.

2. If a selection sort were to be used to sort the data shown below in alphabetical order, show the data after each pass of the sort.

   Renee　　　　Brien　　　　Vincent　　　　Doris　　　　Scarlett

3. **SelectionSortString.java**: Write a program that asks user to input 10 random strings then sorts the list using selection sort and outputs the sorted list.

4. **SelectionSortNoSwapping.java**:  In our version of selection sort, if the largest item is already at right location, the algorithm still swaps that value with itself, even though it is not necessary.
   a)  How could the algorithm be changed to avoid this unnecessary swapping?
   b)  Why might it better to leave the algorithm as it is?

5. **SelectionSortPartialList.java**:  Sometimes we are only interested in knowing the values that would occupy one end of the list if the list were sorted.  As an example, we may want to know the scores of only the top ten competitors in a contest.  Write a method that implements the selection sort so that instead of sorting the entire array, it puts the *k* largest values in order in the last *k* positions in the array.  The value of *k* should be a parameter of the method. Create a main method to test accordingly.