Do the following in the language of your choice.

1. Write a file with “EmployeeId, Salary” as the two fields. Randomly generate the salary and use the iteration number for the employee id.
2. Create files with the following number of rows = 11,101, 1001,10001,100001,1000001
3. Read the files one at a time and store the salary in an array or dynamic array .
4. Use pre-existing sort function in the coding language library. And find the total time needed for sorting salaries of each file. (You can also see the Sorting class in GitHub repo for your reference, or use it if you are writing in Java)

Use the GitHub code as a reference.

The runtime of bubble sort and merge sort on the salaries are as below:

Data (n)	11	101	1001	10001	100001	1000001
bubblesort runtime in ms	1.187876	1.987627	13.632842	398.986879	74469.9278	
mergesort runtime in ms	0.504049	1.588339	4.400743	11.825108	58.805142	497.074712

This readings are for Java language on Mac OS.

You will get comparable but different runtimes.