**LAB # 05**

**Sorting on Linear Array**

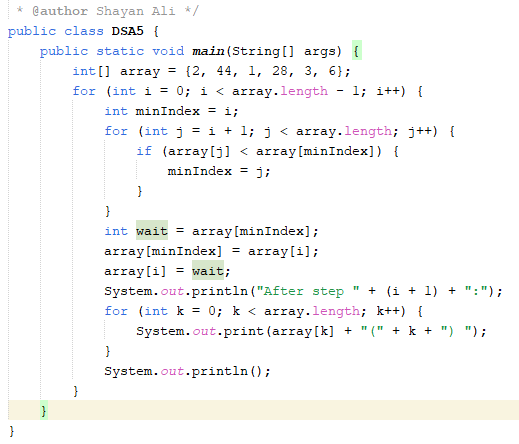
**OBJECTIVE:** To sort a linear array using Selection Sort, Bubble Sort and Merge Sort.

**Lab Task**

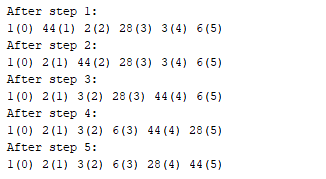
1. Write a program for Selection sort that sorts an array containing numbers, prints all

the sort values of array each followed by its location.

**Code:**

****

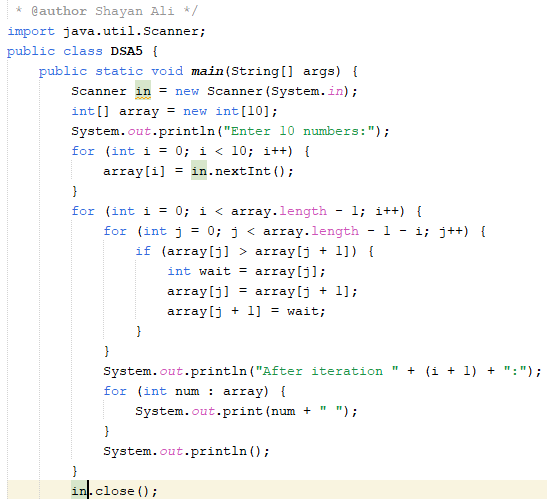
**Output:**

****

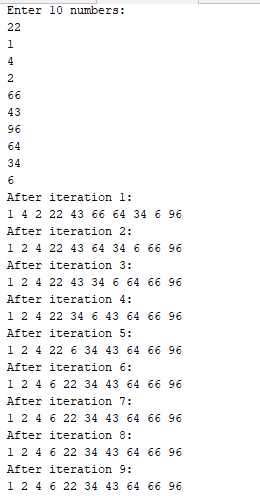
1. Write a program that takes 10 numbers as input in an array. Sort the elements of array

by using Bubble sort. Print each iteration of the sorting process.

**Code:**

****

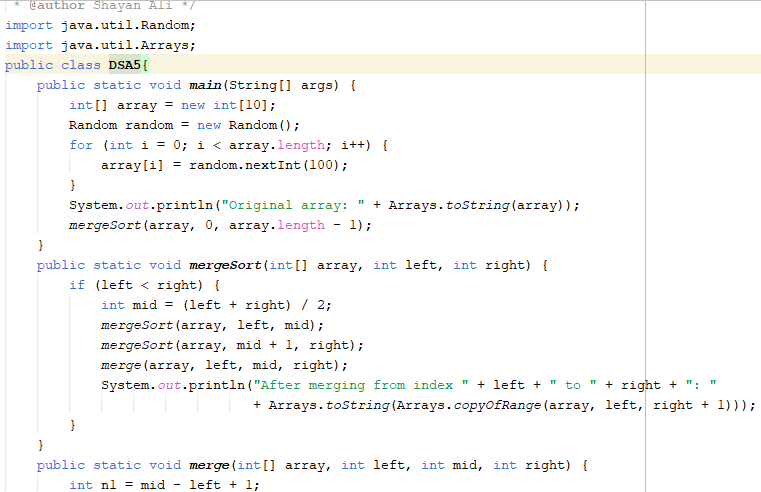
**Output:**

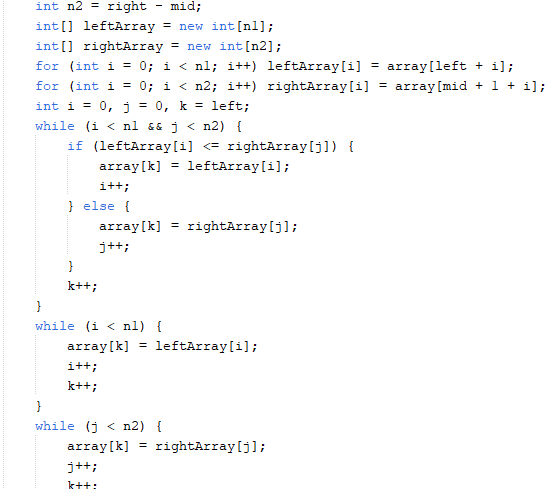
****

1. Write a program that takes 10 random numbers in an array. Sort the elements of array

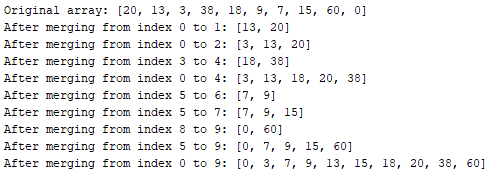
by using Merge sort applying recursive technique. Print each iteration of the sorting process.

**Code:**

****

****

**Output:**

****

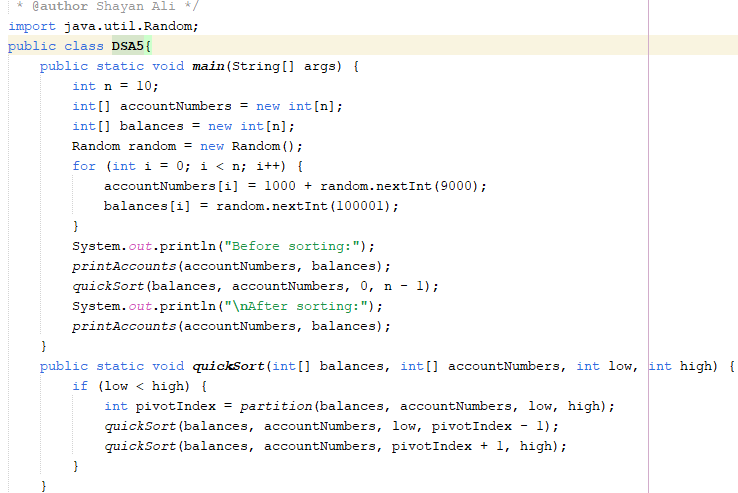
**Home Task**

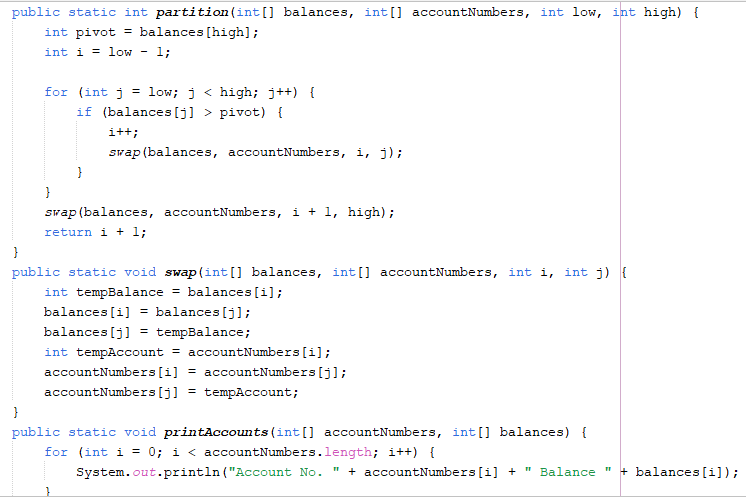
1. Declare an array of size n to store account balances. Initialize with values 0 to 100000 and sort Account No’s according to highest balance values by using Quick sort, For e.g.:

Account No. 3547 Balance 28000

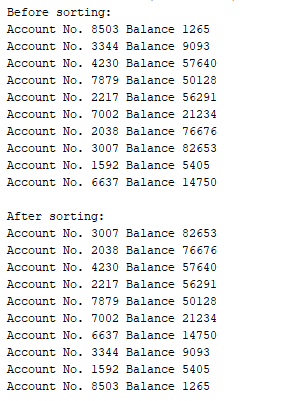
Account No. 1245 Balance 12000

**Code:**

****

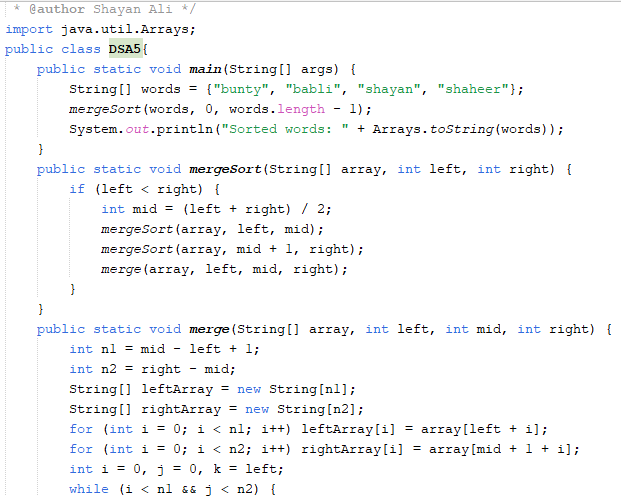
****

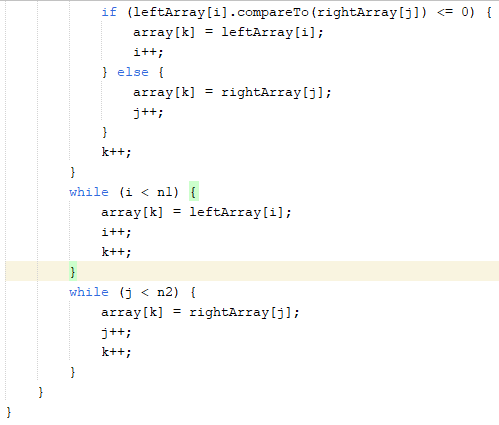
**Output:**

****

1. Write a program which takes an unordered list of integers (or any other objects e.g. String), you have to rearrange the list in their natural order using merge sort.

**Code:**

****

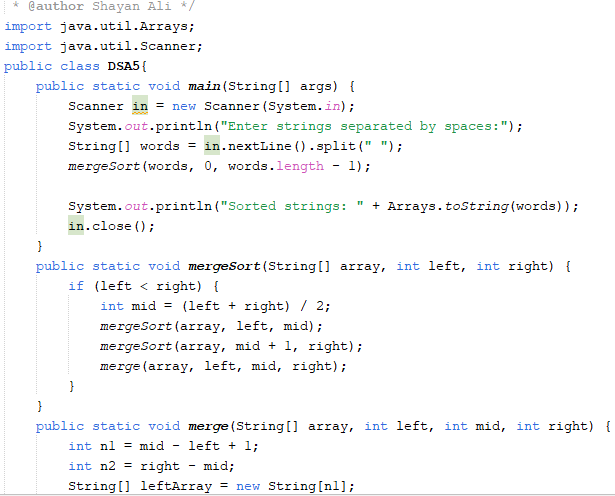
****

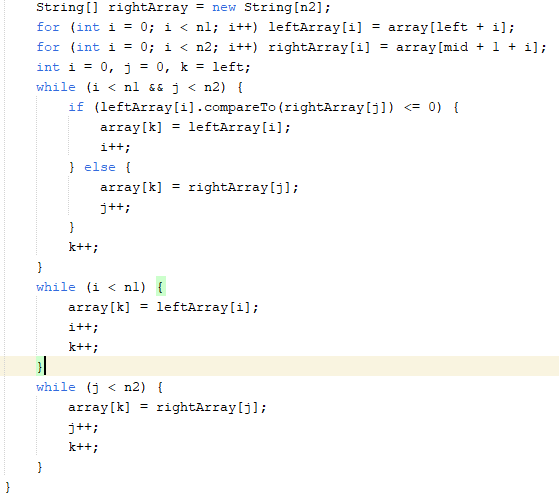
**Output:**

****

1. You are given an unordered list of integers or strings. Write a program to Take this list as input. Sort it in **natural order** using Merge Sort. For integers, this means ascending order. For strings, this means alphabetical order. Print the sorted list.

**Code:**

****

****

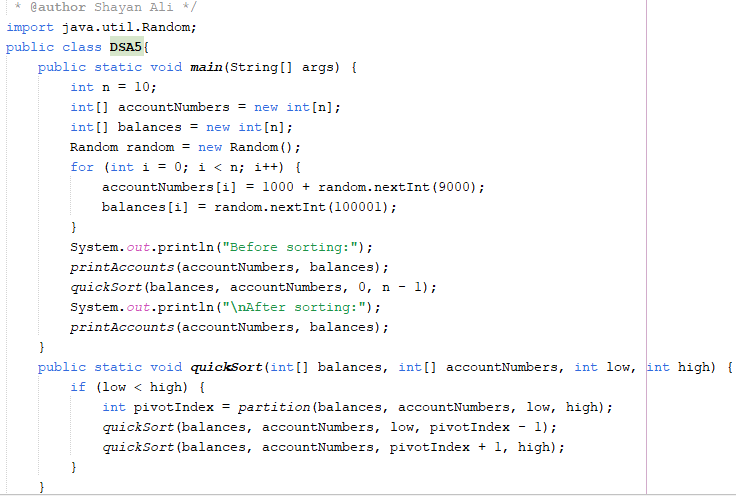
**Output:**

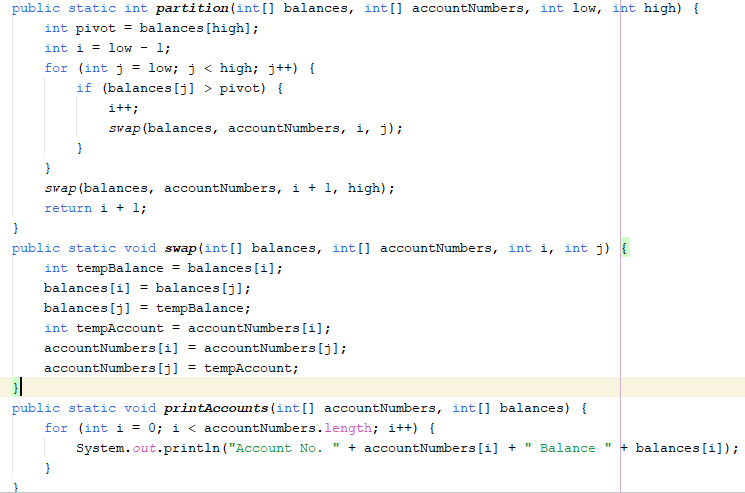


1. You are given a set of bank accounts, each with a unique account number and a balance. Write a Java program to Declare an array of size n to store account balances. Initialize each balance randomly with values between 0 and 100,000. Sort the accounts in **descending order** of their balances using Quick Sort. Print the sorted list in the format

.

**Code:**

****

****

**Output:**

