```java
import java.util.ArrayList;
import java.util.Collections;
import java.util.Iterator;
import java.util.List;
import java.util.Comparator;

// Java program to demonstrate working of Comparator
// interface more than one field
class ComparableStudent {

    // instance member variables
    private String name;
    private int age;

    // parameterized constructor
    public ComparableStudent(String name, Integer age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public Integer getAge() {
        return age;
    }

    public void setAge(Integer age) {
        this.age = age;
    }

    // overriding toString() method
    @Override
    public String toString() {
        return "Customer{" + "name=" + name + ",␣age=" + age + '}';
    }

    static class CustomerSortingComparator implements Comparator<ComparableStudent> {

        @Override
        public int compare(ComparableStudent student1, ComparableStudent student2) {

            // for comparison
            int nameCompare = student1.getName().compareTo(student2.getName());
            int ageCompare = student1.getAge().compareTo(student2.getAge());

            // 2-level comparison using if-else block
            if (nameCompare == 0) {
                return ((ageCompare == 0) ? nameCompare : ageCompare);
            } else {
                return nameCompare;
            }
        }
    }
```

```
    public static void main(String[] args) {

        // create arraylist of students for storing ComparableStudent objects
        List<ComparableStudent> students = new ArrayList<>();

        // create customer objects using constructor initialization
        ComparableStudent obj1 = new ComparableStudent("Ajay", 27);
        ComparableStudent obj2 = new ComparableStudent("Sneha", 23);
        ComparableStudent obj3 = new ComparableStudent("Simran", 37);
        ComparableStudent obj4 = new ComparableStudent("Ajay", 22);
        ComparableStudent obj5 = new ComparableStudent("Ajay", 29);
        ComparableStudent obj6 = new ComparableStudent("Sneha", 22);

        // add customer objects to arraylist
        students.add(obj1);
        students.add(obj2);
        students.add(obj3);
        students.add(obj4);
        students.add(obj5);
        students.add(obj6);

        // before Sorting arraylist: iterate using Iterator
        Iterator<ComparableStudent> studentIterator = students.iterator();

        System.out.println("Before␣Sorting:\n");
        while (studentIterator.hasNext()) {
            System.out.println(studentIterator.next());
        }

        // sorting using Collections.sort(arraylist, comparator);
        Collections.sort(students, new CustomerSortingComparator());

        // after Sorting arraylist: iterate using enhanced for-loop
        System.out.println("\n\nAfter␣Sorting:\n");
        for (ComparableStudent customer : students) {
            System.out.println(customer);
        }
    }
}
```