import java.util.ArrayList;

import java.util.Collections;

import java.util.Comparator;

import java.util.List;

public class Trip implements Comparable<Trip> {

// ... Your class members and constructor ...

@Override

public int compareTo(Trip trip) {

// Compare based on departure time

if (this.getDepartureTime() < trip.getDepartureTime()) {

return -1;

} else if (this.getDepartureTime() > trip.getDepartureTime()) {

return 1;

} else {

// If departure times are equal, compare by arrival time in descending order

return Double.compare(trip.getArrivalTime(), this.getArrivalTime());

}

}

// Define a separate compareTo method for sorting by latest arrival time

public int compareToByLatestArrival(Trip trip) {

// Compare based on arrival time in descending order

return Double.compare(trip.getArrivalTime(), this.getArrivalTime());

}

@Override

public String toString() {

return "Trip{" +

"idOfTrip=" + idOfTrip +

", departureTime=" + departureTime +

", arrivalTime=" + arrivalTime +

'}';

}

public static void main(String[] args) {

Trip l1 = new Trip(0, 0, 1, 10.25, 11.50, 1);

Trip l2 = new Trip(1, 1, 0, 13.00, 13.50, 1);

Trip l3 = new Trip(2, 2, 3, 9.50, 10.75, 1);

Trip l4 = new Trip(3, 0, 1, 10.25, 11.50, 1);

Trip l5 = new Trip(4, 1, 0, 13.00, 13.25, 1);

List<Trip> trips = new ArrayList<>(); // Initialize a list of type Trip

trips.add(l1);

trips.add(l2);

trips.add(l3);

trips.add(l4);

trips.add(l5);

// Sort by earliest departure time

Collections.sort(trips);

System.out.println("Sorted by Earliest Departure Time:");

for (Trip trip : trips) {

System.out.println(trip);

}

// Sort by latest arrival time

Collections.sort(trips, new Comparator<Trip>() {

@Override

public int compare(Trip trip1, Trip trip2) {

return trip1.compareToByLatestArrival(trip2);

}

});

System.out.println("\nSorted by Latest Arrival Time:");

for (Trip trip : trips) {

System.out.println(trip);

}

}

}