Q1.

```
import java.util.*;

class Driver {
    int id;
    String name;
    double rating;
    int tripsCompleted;


    public Driver(int id, String name, double rating, int tripsCompleted) {
        this.id = id;
        this.name = name;
        this.rating = rating;
        this.tripsCompleted = tripsCompleted;
    }


    @Override
    public String toString() {
        return "ID: " + id + ", Name: " + name + ", Rating: " + rating + ", Trips: " + tripsCompleted;
    }
}

public class SmartCabFleet {
    public static void main(String[] args) {


        List<Driver> drivers = new ArrayList<>();
        drivers.add(new Driver(101, "Amit", 4.9, 250));
        drivers.add(new Driver(102, "Riya", 4.8, 310));
        drivers.add(new Driver(103, "Karan", 4.9, 300));
        drivers.add(new Driver(104, "Neha", 4.7, 280));
        drivers.add(new Driver(105, "Rahul", 4.9, 200));


        Collections.sort(drivers, (d1, d2) -> {
            if (Double.compare(d2.rating, d1.rating) == 0) { // same rating
                return Integer.compare(d2.tripsCompleted, d1.tripsCompleted); // sort by trips desc
            } else {
                return Double.compare(d2.rating, d1.rating); // sort by rating desc
            }
        });


        Map<Integer, Driver> driverMap = new LinkedHashMap<>();
        for (Driver d : drivers) {
            driverMap.put(d.id, d);
        }


        System.out.println("Top 3 Drivers:");
        int count = 0;
        for (Map.Entry<Integer, Driver> entry : driverMap.entrySet()) {
            System.out.println(entry.getValue());
            count++;
            if (count == 3) break;
        }
    }
}
```

Q2.

```
import java.util.*;


public class LibraryInventorySystem {



  private HashMap<String, TreeSet<String>> library = new HashMap<>();


    public LibraryInventorySystem() {
    library.put("Science", new TreeSet<>(Arrays.asList("Physics", "Chemistry", "Biology")));
    library.put("Technology", new TreeSet<>(Arrays.asList("AI", "Networking", "Cloud Computing")));
    library.put("Fiction", new TreeSet<>(Arrays.asList("Harry Potter", "Sherlock Holmes", "Alice in Wonderland")));
  }


    public void displayAllBooks() {
    System.out.println("Library Inventory:");
    Iterator<Map.Entry<String, TreeSet<String>>> categoryIterator = library.entrySet().iterator();


    while (categoryIterator.hasNext()) {
      Map.Entry<String, TreeSet<String>> entry = categoryIterator.next();
      String category = entry.getKey();
      TreeSet<String> books = entry.getValue();


      System.out.println("\nCategory: " + category);
      Iterator<String> bookIterator = books.iterator();
      while (bookIterator.hasNext()) {
        System.out.println(" - " + bookIterator.next());
      }
```

```
        }
    }



    public void removeBooksStartingWith(char letter) {
        System.out.println("\nRemoving all books starting with '" + letter + "'...");
        for (Map.Entry<String, TreeSet<String>> entry : library.entrySet()) {
            Iterator<String> bookIterator = entry.getValue().iterator();
            while (bookIterator.hasNext()) {
                String book = bookIterator.next();
                if (book.toLowerCase().startsWith(String.valueOf(letter).toLowerCase())) {
                    bookIterator.remove();  // Safe removal using Iterator
                }
            }
        }
    }

    public static void main(String[] args) {
        LibraryInventorySystem lib = new LibraryInventorySystem();

        System.out.println("Before Removal:");
        lib.displayAllBooks();

        // Remove all books starting with 'A'
        lib.removeBooksStartingWith('A');

        System.out.println("\nAfter Removal:");
        lib.displayAllBooks();
    }
}
```