

# Rajalakshmi Engineering College

Name: Harish V.H  
Email: 240701175@rajalakshmi.edu.in  
Roll no: 240701175  
Phone: 9080255347  
Branch: REC  
Department: CSE - Section 2  
Batch: 2028  
Degree: B.E - CSE

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 10\_Q1

Attempt : 1  
Total Mark : 10  
Marks Obtained : 10

#### Section 1 : COD

##### 1. Problem Statement

A city traffic management system needs to track vehicles entering a toll booth. Each vehicle is uniquely identified by its registration number. The system should allow adding vehicles to a record, ensuring that no duplicate registration numbers exist. The vehicles should be stored in a HashSet, which does not guarantee any specific order.

Your task is to implement a program using a HashSet that allows adding vehicle details and displaying the records.

##### ***Input Format***

The first line of input contains an integer N - the number of vehicles.

The next N lines contain details of each vehicle in the format: "RegNumber

OwnerName VehicleType"

1. RegNumber (String) - A unique registration number (Alphanumeric).
2. OwnerName (String) - The name of the vehicle owner.
3. VehicleType (String, Car, Bike, or Truck) - The type of vehicle.

If a vehicle with the same registration number is already present, ignore the duplicate entry.

### **Output Format**

The output prints the unique vehicle records in any order (since HashSet does not maintain order).

Output format: "RegNumber OwnerName VehicleType"

Refer to the sample output for formatting specifications.

### **Sample Test Case**

Input: 5

KA01AB1234 John Car

MH02CD5678 Alice Bike

DL03EF9012 Bob Truck

TN04GH3456 Mike Car

KA01AB1234 John Car

Output: TN04GH3456 Mike Car

KA01AB1234 John Car

MH02CD5678 Alice Bike

DL03EF9012 Bob Truck

### **Answer**

// You are using Java

import java.util.\*;

class Vehicle {

String regNumber;

String ownerName;

String vehicleType;

public Vehicle(String regNumber, String ownerName, String vehicleType) {

```

        this.regNumber = regNumber;
        this.ownerName = ownerName;
        this.vehicleType = vehicleType;
    }

    // To avoid duplicates based on Registration Number
    @Override
    public int hashCode() {
        return regNumber.hashCode();
    }

    @Override
    public boolean equals(Object obj) {
        Vehicle v = (Vehicle) obj;
        return this.regNumber.equals(v.regNumber);
    }

    @Override
    public String toString() {
        return regNumber + " " + ownerName + " " + vehicleType;
    }
}

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        int n = Integer.parseInt(sc.nextLine());
        HashSet<Vehicle> set = new HashSet<>();

        for (int i = 0; i < n; i++) {
            String line = sc.nextLine();
            String[] parts = line.split(" ");

            String reg = parts[0];
            String owner = parts[1];
            String type = parts[2];

            Vehicle v = new Vehicle(reg, owner, type);
            set.add(v); // duplicate regNumber will be ignored
        }
    }
}

```

```
// Print unique vehicle records
for (Vehicle v : set) {
    System.out.println(v);
}

sc.close();
}
```

**Status :** Correct

**Marks :** 10/10