

# Rajalakshmi Engineering College

Name: SANJAY V  
Email: 241801247@rajalakshmi.edu.in  
Roll no: 241801247  
Phone: 7397492247  
Branch: REC  
Department: AI & DS - Section 3  
Batch: 2028  
Degree: B.E - AI & DS

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**

```
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;  
}
```

```
@Override  
public boolean equals(Object obj) {  
    if (this == obj) return true;  
    if (obj == null || getClass() != obj.getClass()) return false;  
    Vehicle vehicle = (Vehicle) obj;  
    return regNumber.equals(vehicle.regNumber);  
}
```

```
@Override  
public int hashCode() {  
    return Objects.hash(regNumber);  
}
```

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

```
public class Main {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        int N = scanner.nextInt();  
        HashSet<Vehicle> vehicles = new HashSet<>();  
  
        for (int i = 0; i < N; i++) {  
            String regNumber = scanner.next();  
            String ownerName = scanner.next();  
            String vehicleType = scanner.next();  
            Vehicle v = new Vehicle(regNumber, ownerName, vehicleType);  
            vehicles.add(v);  
        }  
  
        for (Vehicle v : vehicles) {  
            System.out.println(v);  
        }  
  
        scanner.close();  
    }  
}
```

}  
}  
**Status : Correct**

**Marks : 10/10**