VehicleController.java

```
package com.example.demo.controller;

import com.example.demo.model.Vehicle;
import com.example.demo.service.VehicleService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

import java.util.List;

@RestController
@RequestMapping("/api/vehicles")
public class VehicleController {

    @Autowired
    private VehicleService vehicleService;



    // Save a vehicle (helper endpoint to add test data)
    @PostMapping
    public Vehicle createVehicle(@RequestBody Vehicle vehicle) {
        return vehicleService.saveVehicle(vehicle);
    }

    // Q1: get all vehicles by year
    @GetMapping("/year/{year}")
    public List<Vehicle> getVehiclesByYear(@PathVariable int year) {
        return vehicleService.getVehiclesByServiceYear(year);
    }

    // Q2: get vehicle type by service id
    @GetMapping("/type/{serviceId}")
    public String getVehicleTypeByServiceId(@PathVariable Long serviceId) {
        return vehicleService.getVehicleTypeByServiceId(serviceId);
    }
    // Q3: delete all vehicles for a year
    @DeleteMapping("/year/{year}")
    public String deleteVehiclesByYear(@PathVariable int year) {
        vehicleService.deleteVehiclesByServiceYear(year);
        return "Deleted all vehicle service records for year " + year;
    }
}
```

Vehicle.java

```
package com.example.demo.model;

import jakarta.persistence.Entity;
import jakarta.persistence.GeneratedValue;
import jakarta.persistence.GenerationType;
import jakarta.persistence.Id;
import jakarta.persistence.Column;

@Entity
public class Vehicle {

    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private Long serviceId;

    @Column(nullable = false)
    private String vehicleType;

    private String vehicleNumber;
    private int serviceYear;

    // Constructors
    public Vehicle() {
    }

    public Vehicle(String vehicleType, String vehicleNumber, int serviceYear) {
        this.vehicleType = vehicleType;
        this.vehicleNumber = vehicleNumber;
        this.serviceYear = serviceYear;
    }

    // Getters and Setters
    public Long getServiceId() {
        return serviceId;
    }

    public void setServiceId(Long serviceId) {
        this.serviceId = serviceId;
    }

    public String getVehicleType() {
        return vehicleType;
    }
    public void setVehicleType(String vehicleType) {
        this.vehicleType = vehicleType;
    }

    public String getVehicleNumber() {
        return vehicleNumber;
    }

    public void setVehicleNumber(String vehicleNumber) {
        this.vehicleNumber = vehicleNumber;
    }

    public int getServiceYear() {
        return serviceYear;
    }

    public void setServiceYear(int serviceYear) {
        this.serviceYear = serviceYear;
    }
}
```

VehicleRepository.java

```
package com.example.demo.repository;

import com.example.demo.model.Vehicle;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.data.jpa.repository.Modifying;
import org.springframework.data.jpa.repository.Query;
import org.springframework.stereotype.Repository;
import org.springframework.transaction.annotation.Transactional;

import java.util.List;

@Repository
public interface VehicleRepository extends JpaRepository<Vehicle, Long> {

    // Q1: find all by service year
    List<Vehicle> findByServiceYear(int serviceYear);

    // Q2: get only vehicleType by serviceId
    @Query("SELECT v.vehicleType FROM Vehicle v WHERE v.serviceId = :serviceId")
    String findVehicleTypeByServiceId(Long serviceId);
    @Modifying
    @Transactional
    @Query("DELETE FROM Vehicle v WHERE v.serviceYear = :serviceYear")
    void deleteByServiceYear(int serviceYear);
}
```

VehicleServiceImpl.java

```
package com.example.demo.service.impl;

import com.example.demo.model.Vehicle;
import com.example.demo.repository.VehicleRepository;
import com.example.demo.service.VehicleService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;

import java.util.List;
```

```
@Service
public class VehicleServiceImpl implements VehicleService {

   @Autowired
   private VehicleRepository vehicleRepository;

   @Override
   public List<Vehicle> getVehiclesByServiceYear(int serviceYear) {
      return vehicleRepository.findByServiceYear(serviceYear);
   }

   @Override
   public String getVehicleTypeByServiceId(Long serviceId) {
      return vehicleRepository.findVehicleTypeByServiceId(serviceId);
   }

   @Override
   public void deleteVehiclesByServiceYear(int serviceYear) {
      vehicleRepository.deleteByServiceYear(serviceYear);
   }

   @Override
   public Vehicle saveVehicle(Vehicle vehicle) {
      return vehicleRepository.save(vehicle);
   }
}
```

VehicleService.java

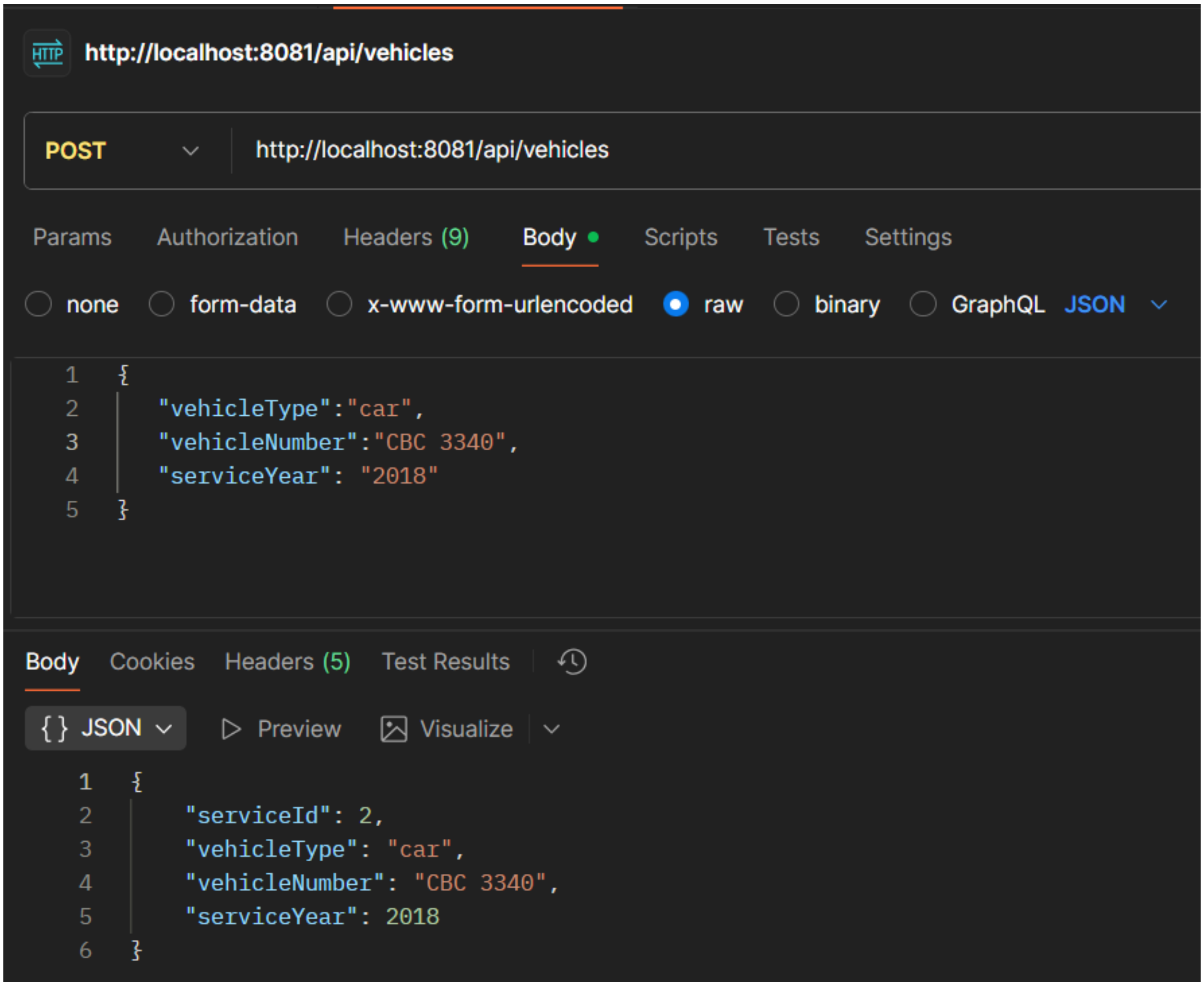
```
package com.example.demo.service;

import com.example.demo.model.Vehicle;
import java.util.List;

public interface VehicleService {
   List<Vehicle> getVehiclesByServiceYear(int serviceYear);
   String getVehicleTypeByServiceId(Long serviceId);
   void deleteVehiclesByServiceYear(int serviceYear);
   Vehicle saveVehicle(Vehicle vehicle); // helper to add sample data via API
}
```

DemoApplication

```
package com.example.demo;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class DemoApplication {
   public static void main(String[] args) {
      SpringApplication.run(DemoApplication.class, args);
   }
}
```

applicationProperties

```
spring.datasource.url=jdbc:mysql://localhost:3306/vehicle_db?createDatabaseIfNotExist=true&
useSSL=false&allowPublicKeyRetrieval=true&serverTimezone=UTC
spring.datasource.username=root
spring.datasource.password=1234

spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQLDialect

server.port=8081
```



http://localhost:8081/api/vehicles

POST http://localhost:8081/api/vehicles

Params Authorization Headers (9) Body Scripts Tests Settings

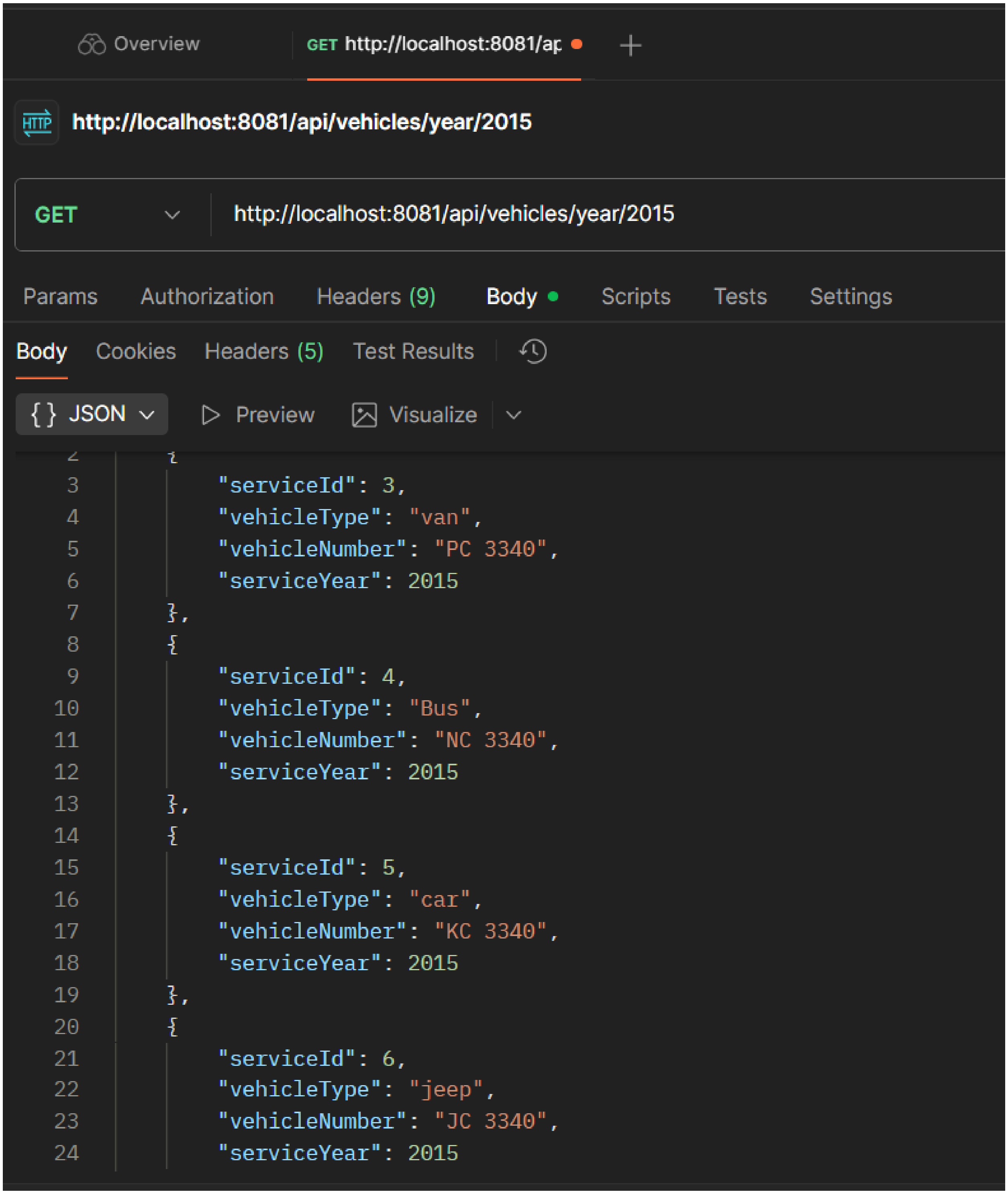
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "vehicleType":"car",
    "vehicleNumber":"CBC 3340",
    "serviceYear": "2018"
}
```

Body Cookies Headers (5) Test Results
{ } JSON Preview Visualize

```
{
    "serviceId": 2,
    "vehicleType": "car",
    "vehicleNumber": "CBC 3340",
    "serviceYear": 2018
}
```

1. How can we retrieve a list of all vehicles serviced in a specific year?

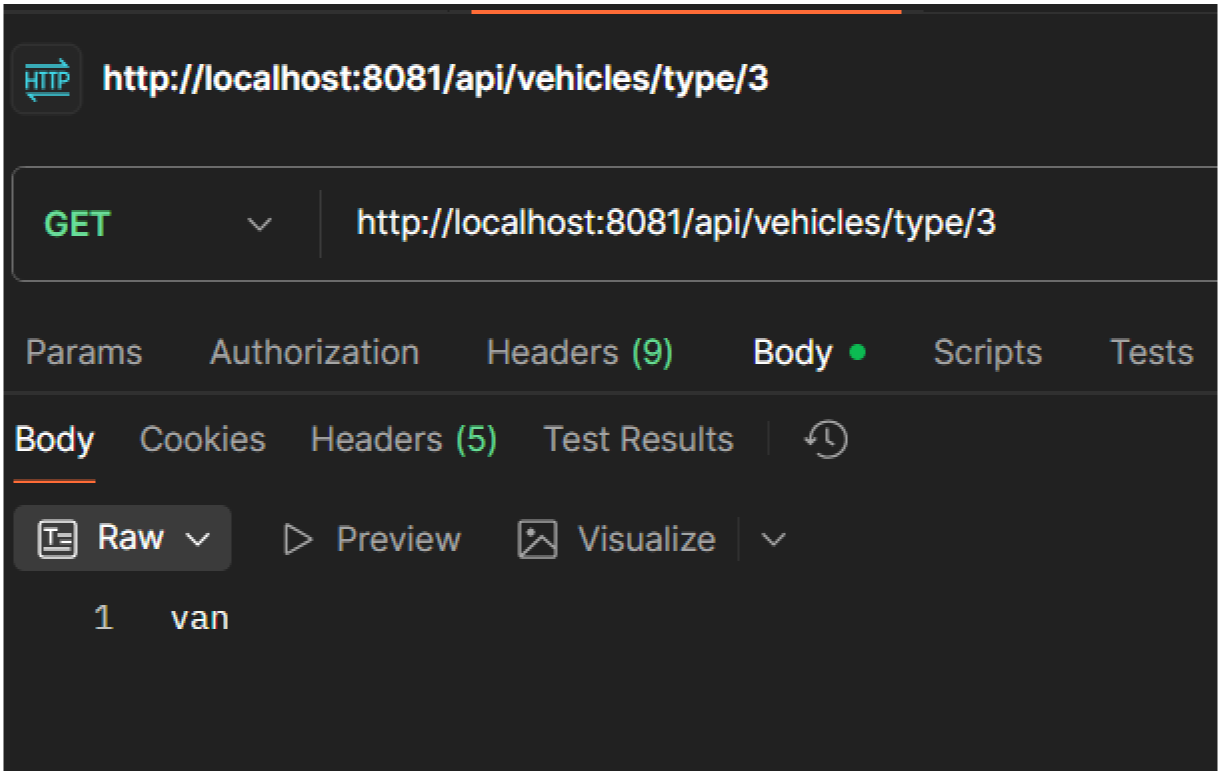


http://localhost:8081/api/vehicles/year/2015

GET http://localhost:8081/api/vehicles/year/2015

```
    {
        "serviceId": 3,
        "vehicleType": "van",
        "vehicleNumber": "PC 3340",
        "serviceYear": 2015
    },
    {
        "serviceId": 4,
        "vehicleType": "Bus",
        "vehicleNumber": "NC 3340",
        "serviceYear": 2015
    },
    {
        "serviceId": 5,
        "vehicleType": "car",
        "vehicleNumber": "KC 3340",
        "serviceYear": 2015
    },
    {
        "serviceId": 6,
        "vehicleType": "jeep",
        "vehicleNumber": "JC 3340",
        "serviceYear": 2015
```

2. How can we retrieve only the vehicle type given the service ID?

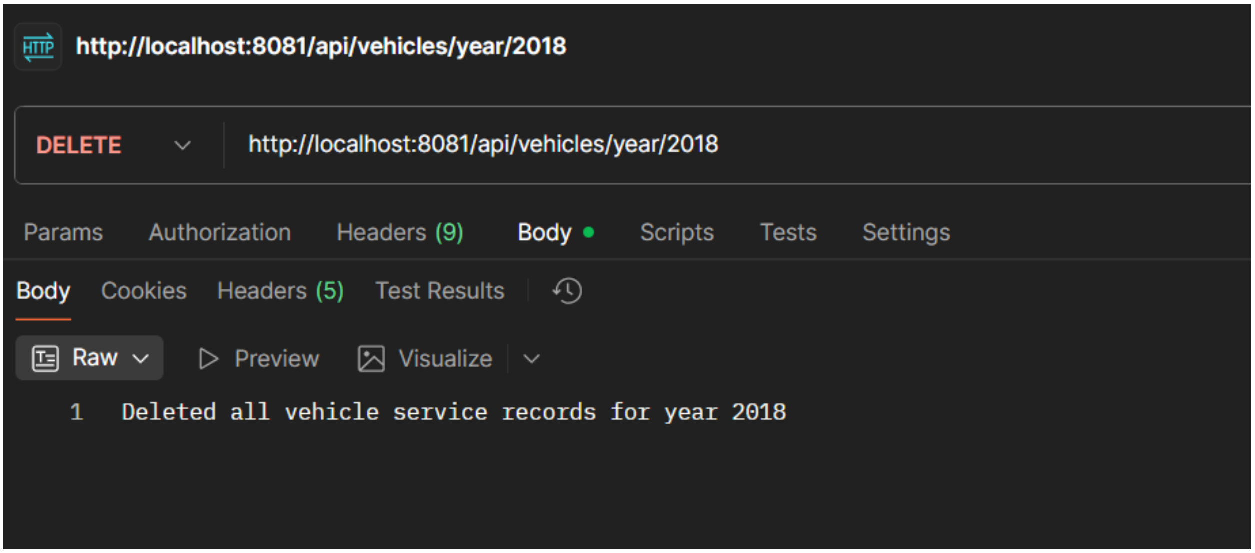


http://localhost:8081/api/vehicles/type/3

GET http://localhost:8081/api/vehicles/type/3

```
van
```

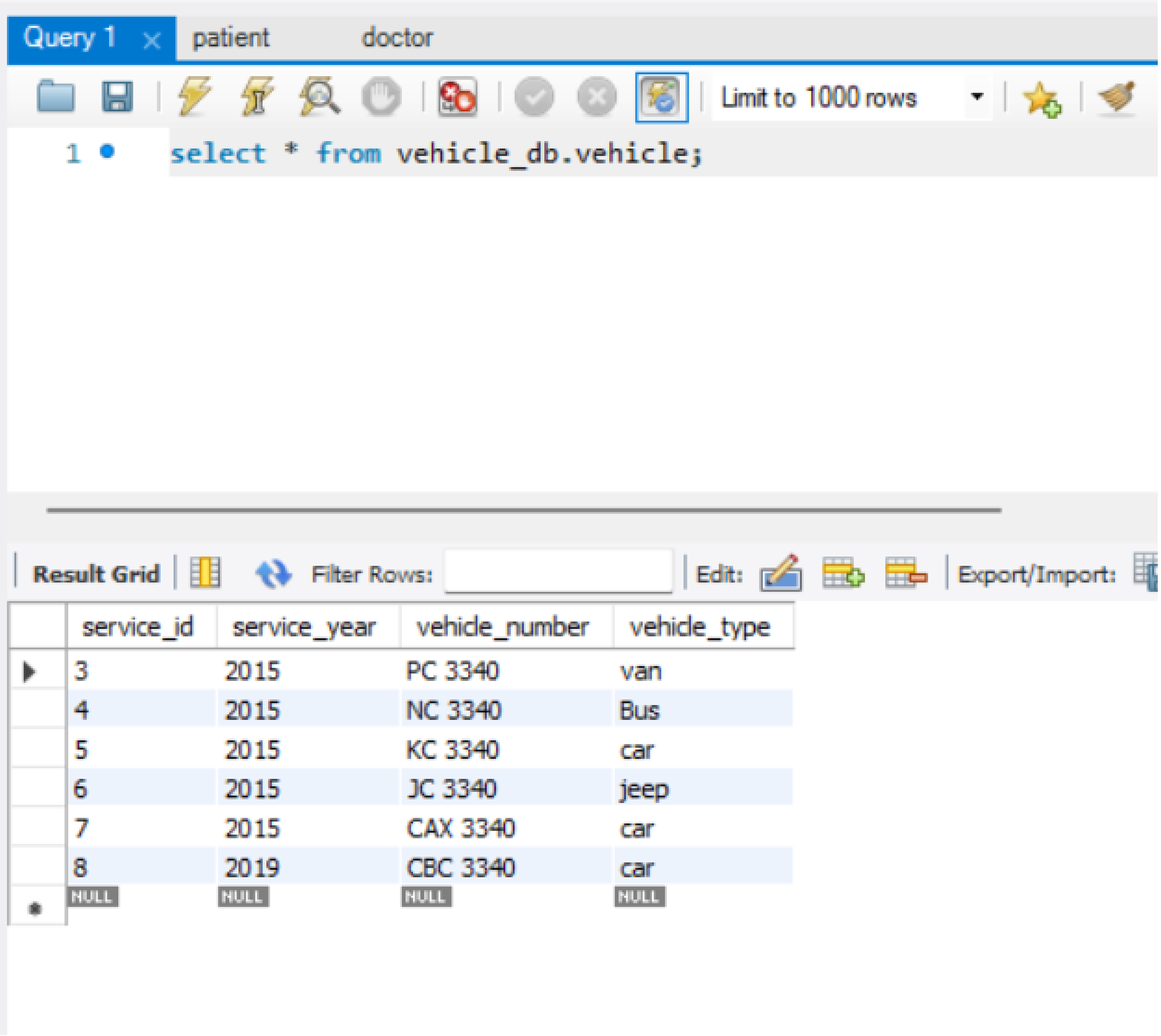
3. How can we remove all vehicle service records for a given year?



http://localhost:8081/api/vehicles/year/2018

DELETE http://localhost:8081/api/vehicles/year/2018

```
Deleted all vehicle service records for year 2018
```



```
select * from vehicle_db.vehicle;
```

service_id	service_year	vehicle_number	vehicle_type
3	2015	PC 3340	van
4	2015	NC 3340	Bus
5	2015	KC 3340	car
6	2015	JC 3340	jeep
7	2015	CAX 3340	car
8	2019	CBC 3340	car
NULL	NULL	NULL	NULL