**Assignment 5**

1. **Create a Mongoose schema for cars which contains following properties**
2. **Manufacturer, make, model, year of manufacture – validate year of manufacture should be between 2020 and 2024.**
3. **Ensure that same model car should not be present for any of the manufacturers.**
4. **Validate the enumerators for car such as [hatchback, sedan, suv].**
5. **Validate engine number to accept numbers which is of 10 digits.**
6. **Validate chassis number which is alphanumerical and should be between 10-15 digits.**

**config.mongoose.js**

var mongoose = require('mongoose');

var uri = 'mongodb://127.0.0.1:27017/car';

module.exports = function() {

var db = mongoose.connect(uri);

console.log('DB Connection');

require('../students/models/models.student');

// require('../students/models/models.courses');

require('../students/models/models.users');

return db;

};

**controllers.student.js**

var app = require('../../config/config.express');

var Car = require('../models/models.student');

// var Course = require('../models/models.courses');

app.post('/', function(req, res, next) {

var car = new Car(req.body);

car.save(function(err) {

if (err) {

return next(err);

} else {

res.json(car);

}

});

});

**model.student.js**

var mongoose = require('mongoose');

var CarSchema = new mongoose.Schema({

\_id:{type:Number},

manufacturer: { type: String, trim: true, required: true },

make:{type:String,trim: true},

model: {type: String,unique: true, trim: true},

type: { type: String, enum: ['hatchback', 'sedan','suv']},

engine\_num:{type:String, trim: true,unique: true,

validate: [

function(v) {

return /^[0-9]{10}$/.test(v);

}

]

},

//min:10,max:15,

chasis\_num:{type:String, trim: true,unique: true,

validate: [

function(v) {

return /^[A-Za-z0-9]{10,15}$/.test(v); }

]},

manufacture\_year: { type: Date, min: '2020-01-01',max: '2024-01-01' },

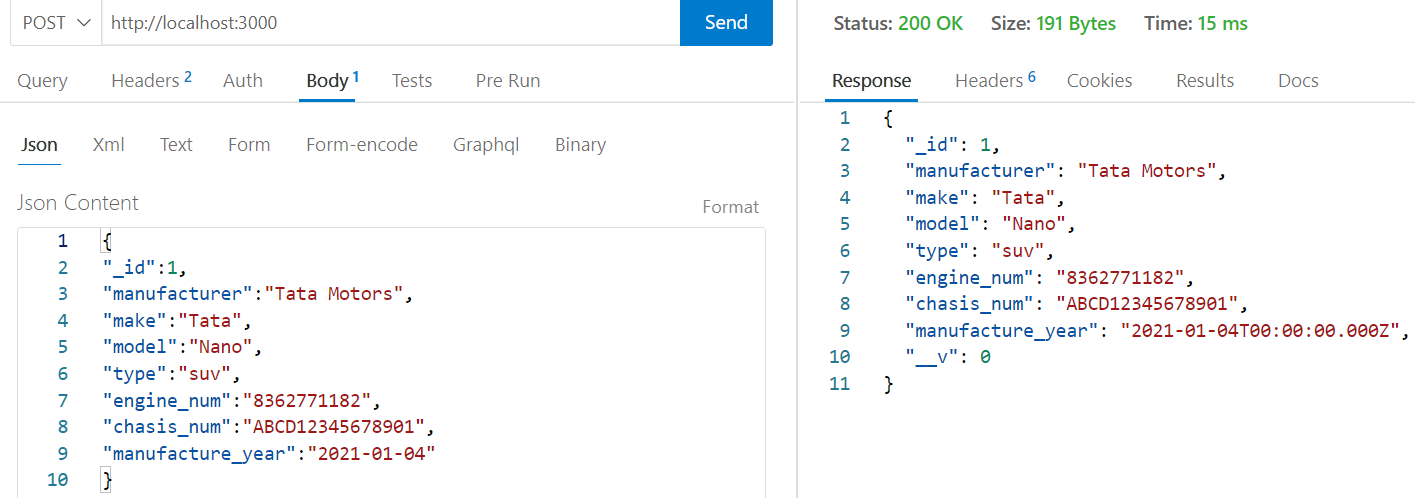
});

var Car = mongoose.model('Car', CarSchema);

module.exports = Car;



Connecting Thunder Client to MongoDB compass



{

"\_id":1,

"manufacturer":"Tata Motors",

"make":"Tata",

"model":"Nano",

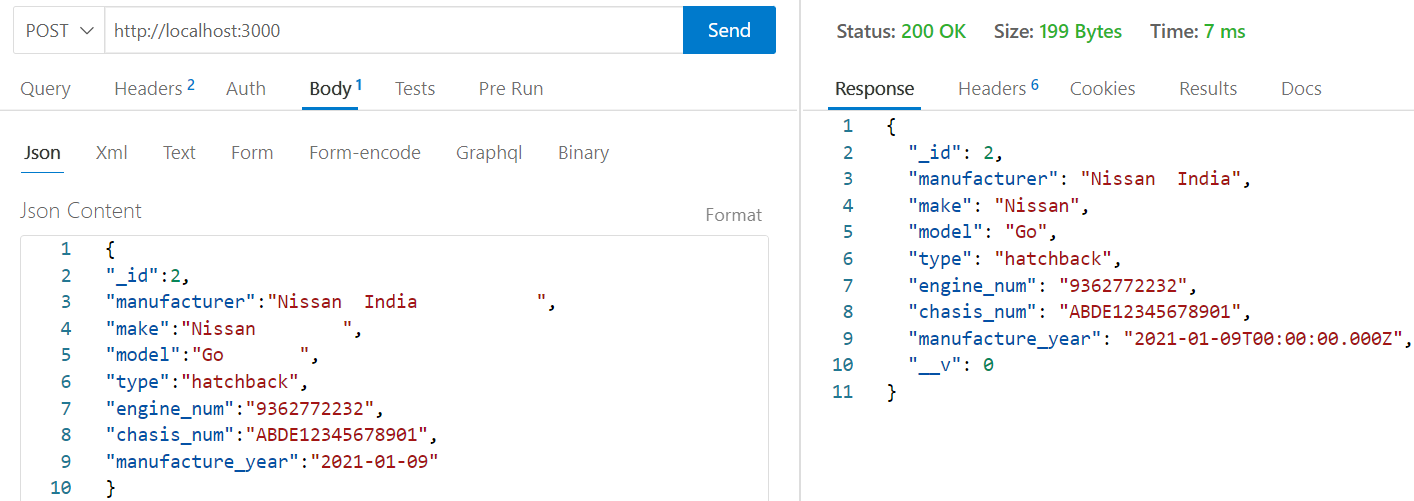
"type":"suv",

"engine\_num":"8362771182",

"chasis\_num":"ABCD12345678901",

"manufacture\_year":"2021-01-04"

}



{

"\_id":2,

"manufacturer":"Nissan India ",

"make":"Nissan ",

"model":"Go ",

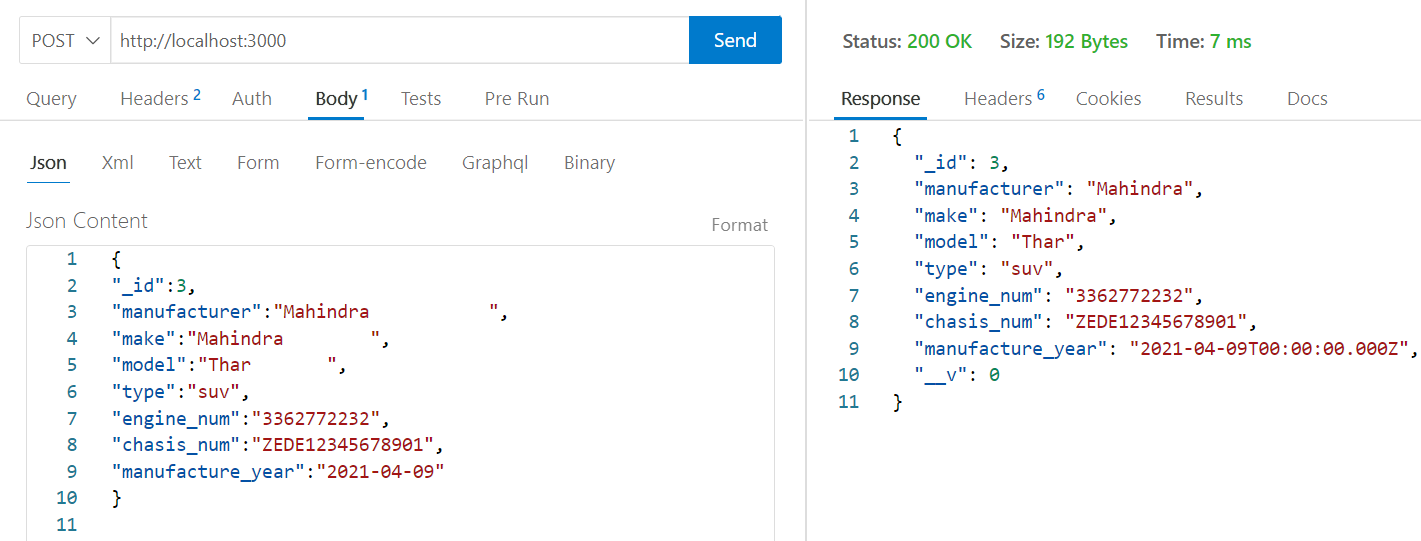
"type":"hatchback",

"engine\_num":"9362772232",

"chasis\_num":"ABDE12345678901",

"manufacture\_year":"2021-01-09"

}



{

"\_id":3,

"manufacturer":"Mahindra ",

"make":"Mahindra ",

"model":"Thar ",

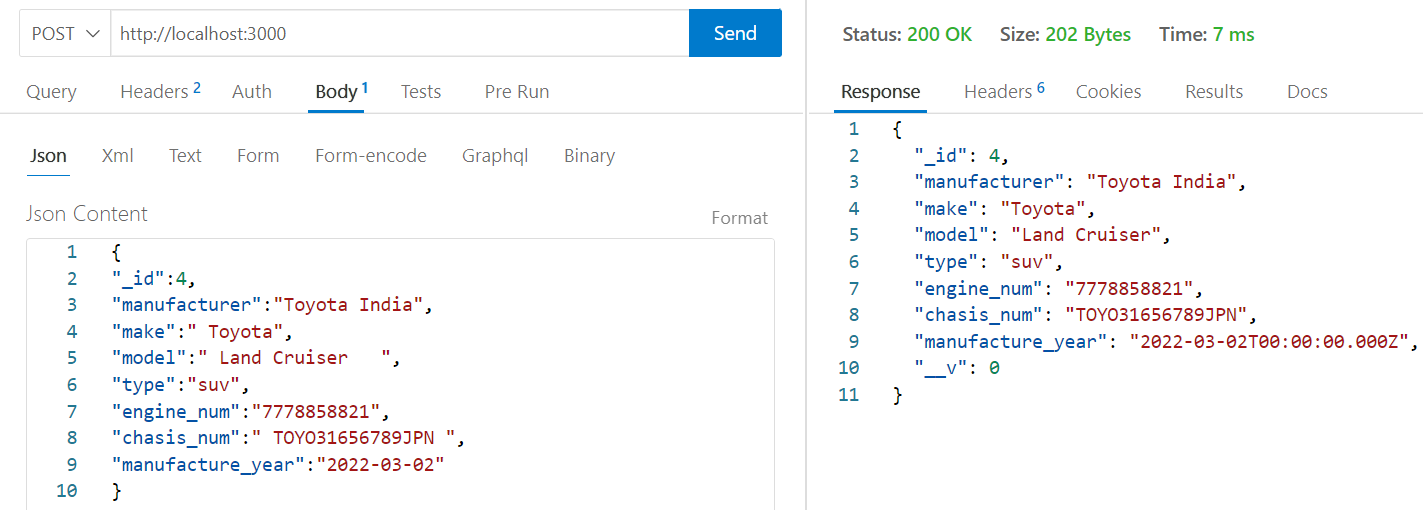
"type":"suv",

"engine\_num":"3362772232",

"chasis\_num":"ZEDE12345678901",

"manufacture\_year":"2021-04-09"

}



{

"\_id":4,

"manufacturer":"Toyota India",

"make":" Toyota",

"model":" Land Cruiser ",

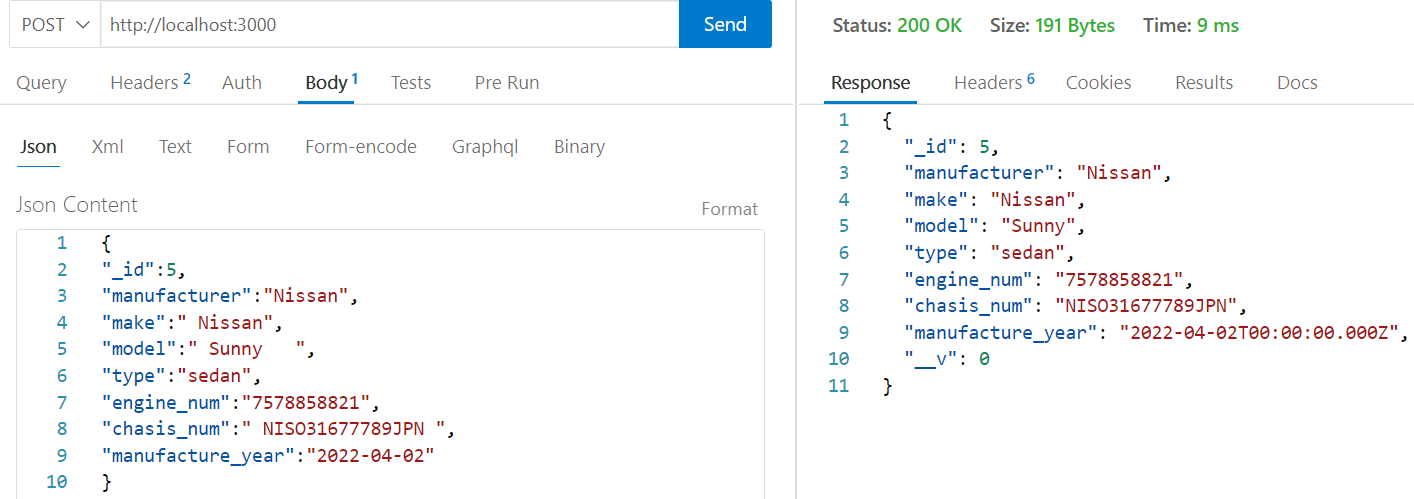
"type":"suv",

"engine\_num":"7778858821",

"chasis\_num":" TOYO31656789JPN ",

"manufacture\_year":"2022-03-02"

}

****

{

"\_id":5,

"manufacturer":"Nissan",

"make":" Nissan",

"model":" Sunny ",

"type":"sedan",

"engine\_num":"7578858821",

"chasis\_num":" NISO31677789JPN ",

"manufacture\_year":"2022-04-02"

}

**Output in MomgoDB Compass**

