

Міністерство освіти і науки України

Національний технічний університет України „КПІ імені Ігоря Сікорського ”

Факультет інформатики та обчислювальної техніки

Кафедра інформаційних систем та технологій

**Звіт до комп’ютерного практикуму №5**

**З дисципліни «Основи Back-end технологій»**

Прийняв: Виконав:

Викладач Зубко Роман Студент 3 курсу Бондарчук Анастасія, гр. ІП-12

«\_29\_»\_\_травня\_\_2024 р.

**2024 р.**

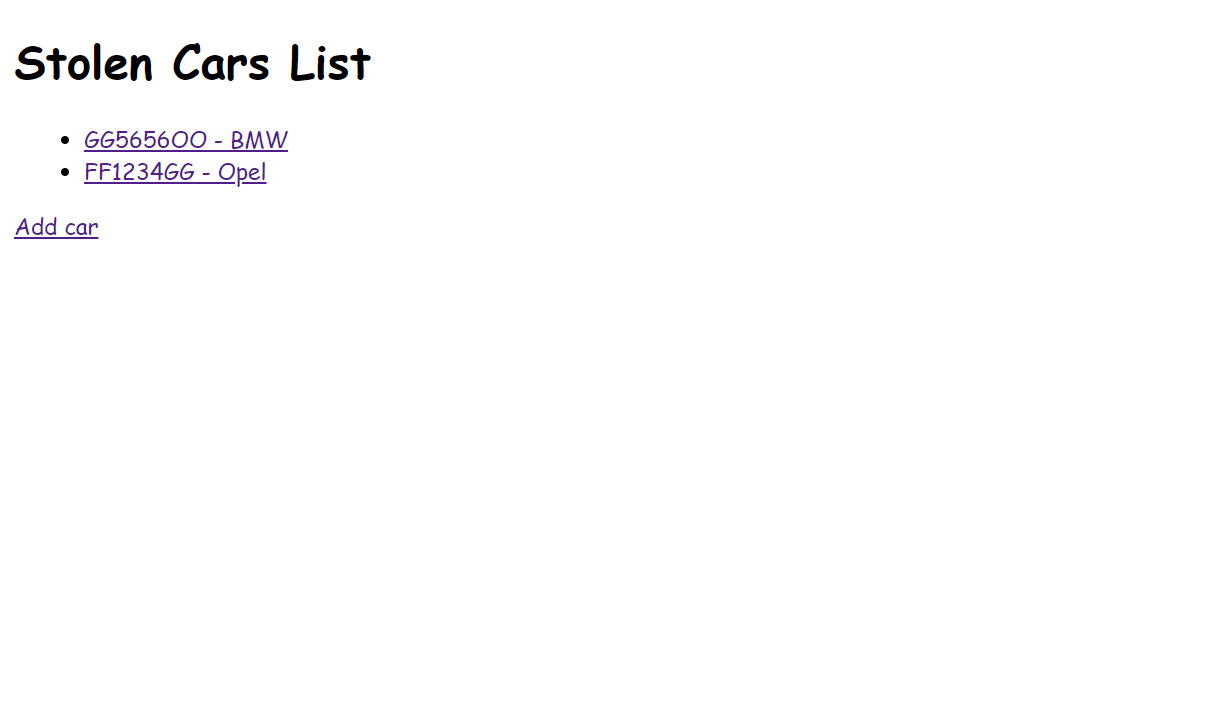
**Лабораторна робота №5.**

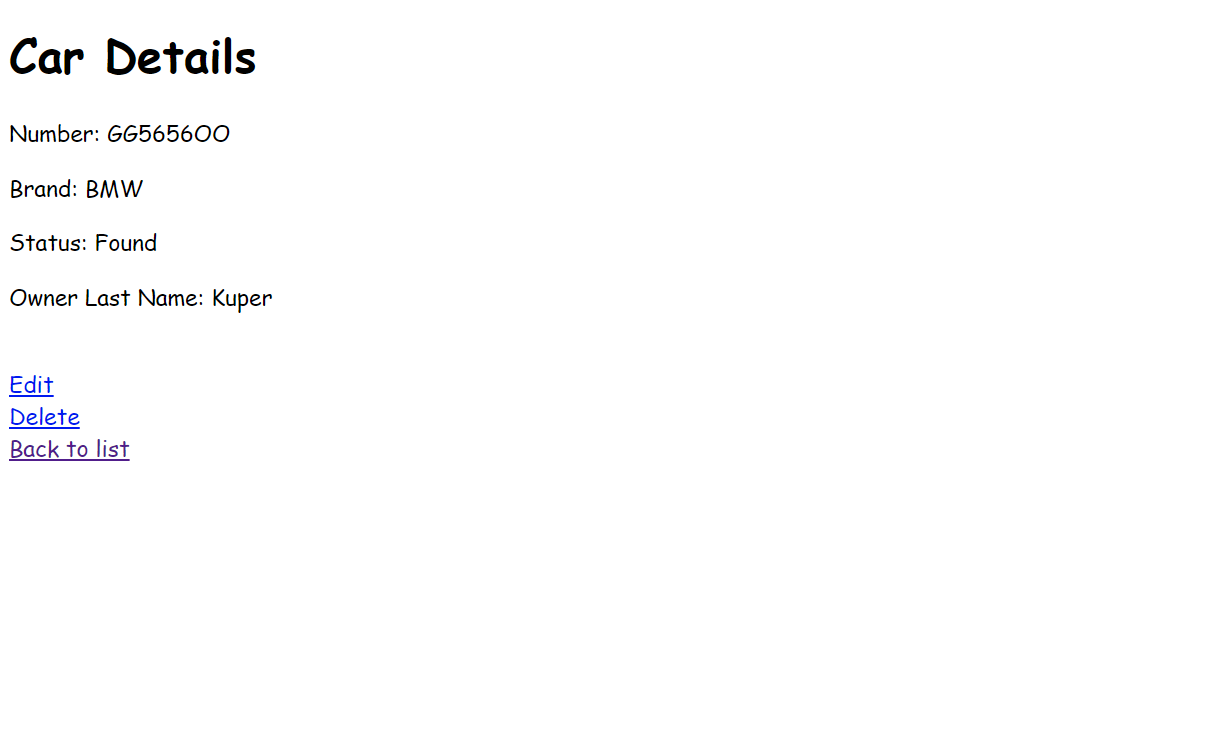
**NodeJS. Робота з БД MongoDB. Додаток що реалізує CRUD операції в БД.**

**Завдання.**

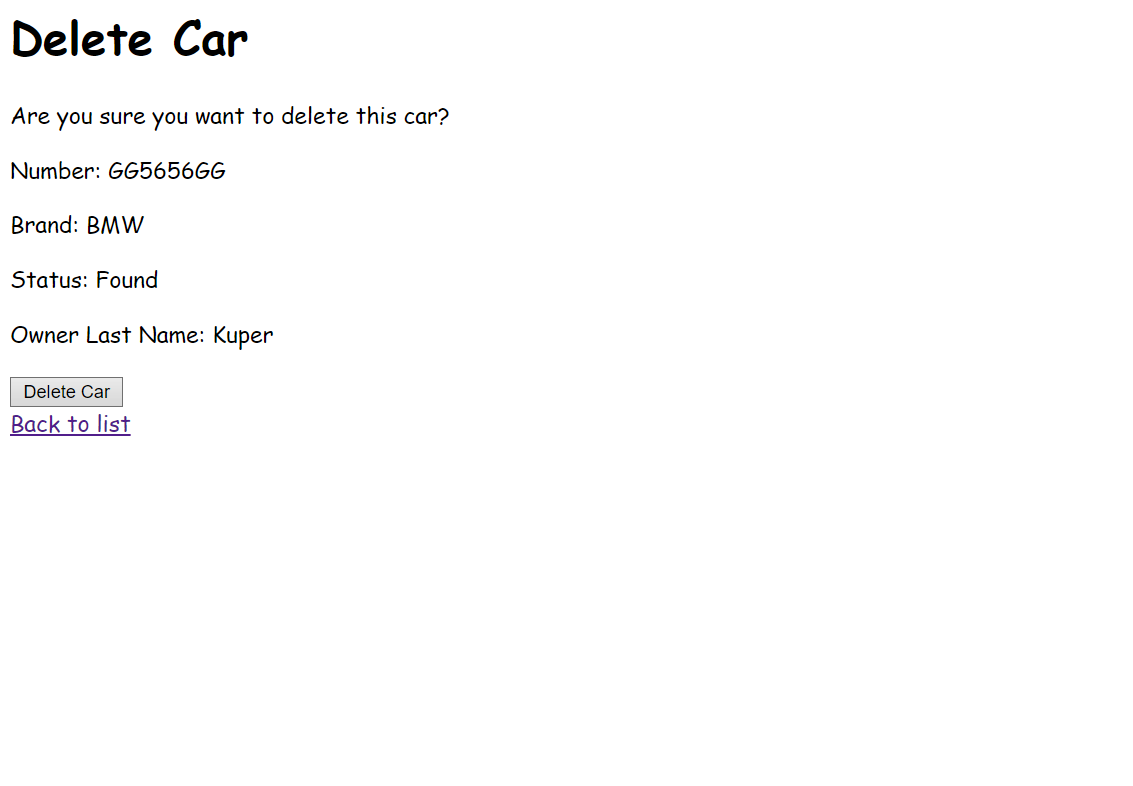
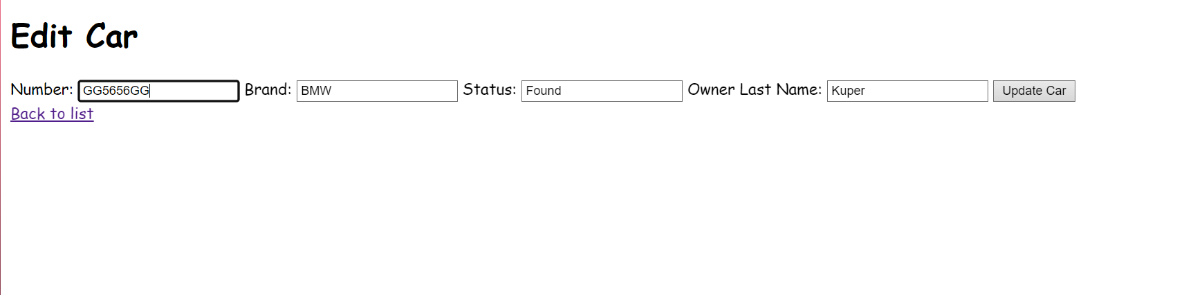
1. Створити додаток, що реалізує CRUD операції з БД додавання, читання, редагування та видалення записів БД.
2. Забезпечити роутінг запитів та виведення результатів запитів на WEB- сторінку.
3. Додати новий роут для виведення інформації у вигляді json-файлу.

**Хід роботи**

1. Створено додаток, що реалізує CRUD операції з БД додавання, читання, редагування та видалення записів БД.

Мал.1 – Головна сторінка

Мал.2 – Сторінка читання даних

Мал.3 – Сторінка редагування даних

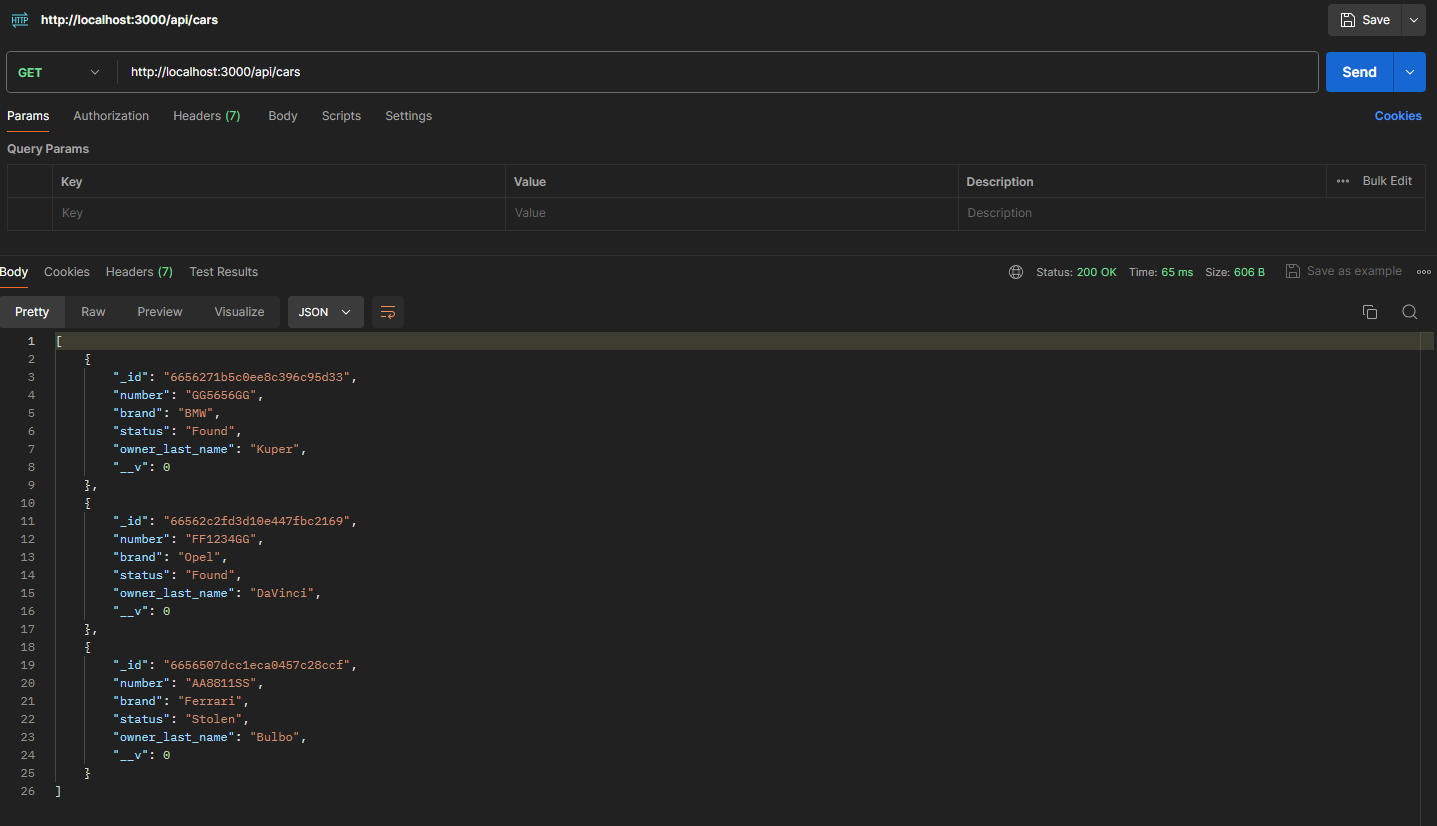
Мал.4 – Сторінка видалення даних

Мал.5 – Сторінка додавання даних

1. Було забезпечено роутінг запитів та виведення результатів запитів на WEB- сторінку.

const express = ***require***('express');  
const ***router*** = express.Router();  
const ***StolenCar*** = ***require***('../models/StolenCar');  
  
// Create  
***router***.post('/', async (req, res) => {  
 try {  
 const newCar = new ***StolenCar***(req.body);  
 const savedCar = await newCar.save();  
 res.status(201).json(savedCar);  
 } catch (error) {  
 res.status(400).json({ message: error.message });  
 }  
});  
  
// Read all  
router.get('/', async (req, res) => {  
 try {  
 const cars = await StolenCar.find();  
 res.status(200).json(cars);  
 } catch (error) {  
 res.status(500).json({ message: error.message });  
 }  
});  
  
// Read one  
router.get('/:id', async (req, res) => {  
 try {  
 const car = await StolenCar.findById(req.params.id);  
 if (car == null) {  
 return res.status(404).json({ message: 'Car not found' });  
 }  
 res.status(200).json(car);  
 } catch (error) {  
 res.status(500).json({ message: error.message });  
 }  
});  
  
// Update  
router.put('/:id', async (req, res) => {  
 try {  
 const updatedCar = await StolenCar.findByIdAndUpdate(req.params.id, req.body, { new: true });  
 if (updatedCar == null) {  
 return res.status(404).json({ message: 'Car not found' });  
 }  
 res.status(200).json(updatedCar);  
 } catch (error) {  
 res.status(400).json({ message: error.message });  
 }  
});  
  
// Delete  
***router***.delete('/:id', async (req, res) => {  
 try {  
 const car = await ***StolenCar***.findByIdAndDelete(req.params.id);  
 if (car == null) {  
 return res.status(404).json({ message: 'Car not found' });  
 }  
 res.status(204).json({ message: 'Deleted' });  
 } catch (error) {  
 res.status(500).json({ message: error.message });  
 }  
});  
  
module.exports = router;

1. Додати новий роут для виведення інформації у вигляді json-файлу.

const express = ***require***('express');  
const ***router*** = express.Router();  
const ***StolenCar*** = ***require***('../models/StolenCar');  
  
// Виведення всіх записів у вигляді JSON  
***router***.get('/cars', async (req, res) => {  
 try {  
 const cars = await ***StolenCar***.find();  
 res.json(cars);  
 } catch (error) {  
 res.status(500).json({ message: error.message });  
 }  
});  
  
***module***.exports = ***router***;

Мал.6 – Виведення результату у json-форматі

**Висновок:** Під час виконання лабораторної роботи було створено сайт для інтерактивного виведення результатів роботи CRUD операцій над базою даних за допомогою nodejs та mongoose технологій.

**Код програми додається:** <https://github.com/NastasaBondarchuck/BackEnd_Lab5-6>