Міністерство освіти і науки україни

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Математичний факультет

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

Дисципліна «Автоматизоване тестування»

Лабораторна робота № 6 : «Тестування API»

Виконав студент гр. 8.1214

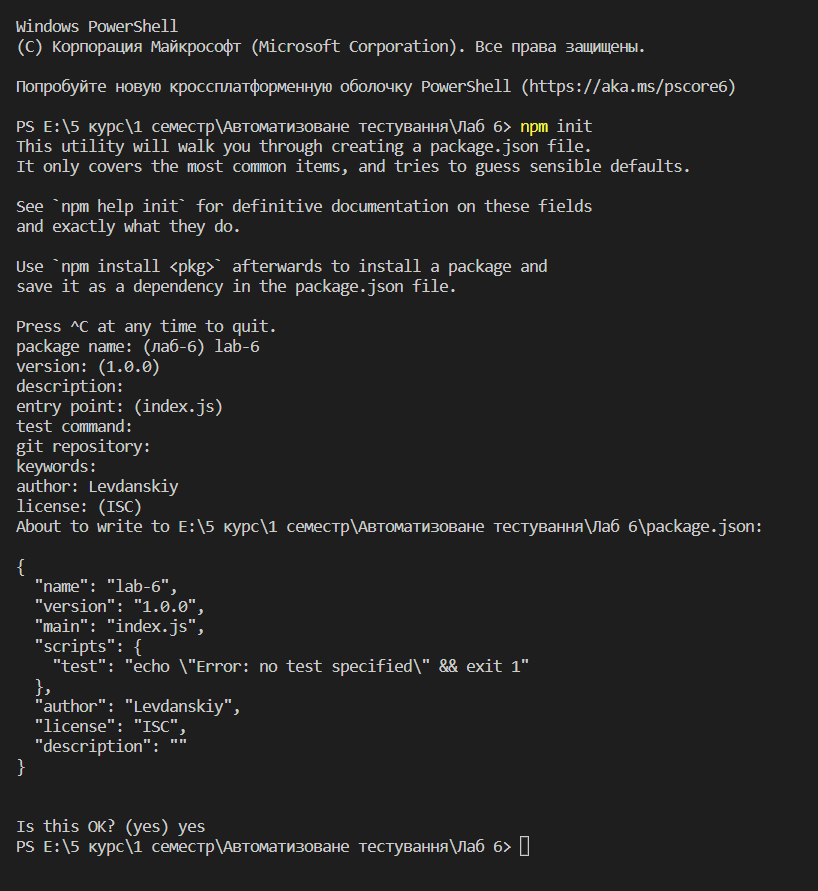
Левданський Михайло Владиславович

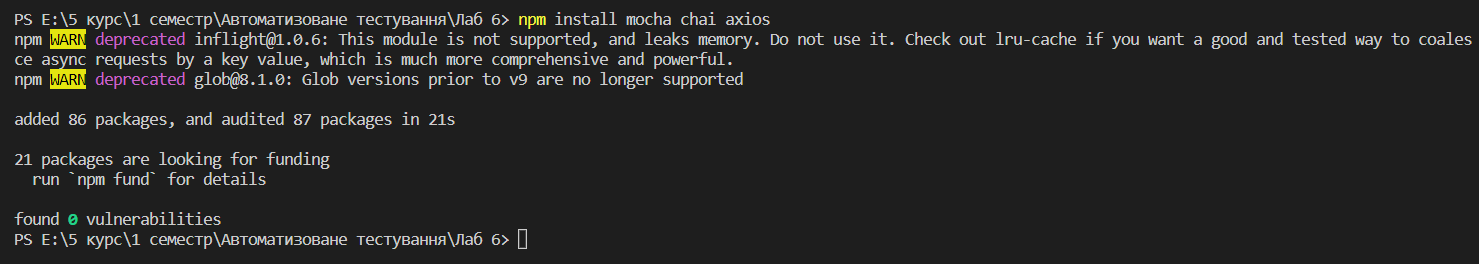
Перевірив: Викладач

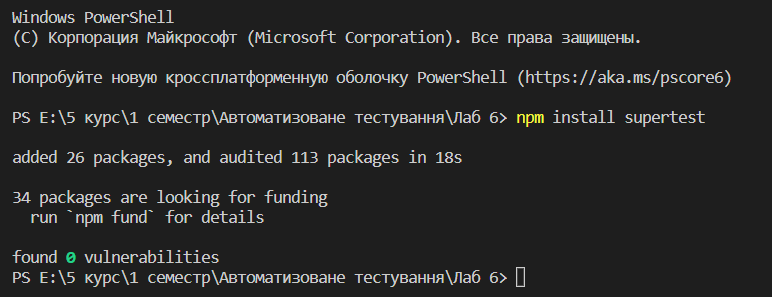
Кудін Олексій Володимирович

Запоріжжя

2024







const request = require('supertest');

let expect;

const API\_URL = 'https://gorest.co.in/public/v1';

const TOKEN = '24bfccebe2d69974f80ee2afe4bf706f24963fed5e71e99490f2f52c647f5c0c';

describe('GoRest API Testing', () => {

    before(async () => {

        expect = (await import('chai')).expect;

    });

    let userId;

    it('GET /users - Отримання списку користувачів', async () => {

        const res = await request(API\_URL)

            .get('/users')

            .set('Authorization', `Bearer ${TOKEN}`);

        expect(res.status).to.equal(200);

        expect(res.body).to.have.property('data');

    });

    it('POST /users - Створення нового користувача', async () => {

        const res = await request(API\_URL)

            .post('/users')

            .set('Authorization', `Bearer ${TOKEN}`)

            .send({

                name: "Тестовий Користувач",

                gender: "male",

                email: `testuser\_${Date.now()}@example.com`,

                status: "active"

            });

        expect(res.status).to.equal(201);

        expect(res.body).to.have.property('data');

        expect(res.body.data).to.have.property('id');

        console.log(res.body.data)

        userId = res.body.data.id;

    });

    it('PUT /users/:id - Оновлення даних користувача', async () => {

        const res = await request(API\_URL)

            .put(`/users/${userId}`)

            .set('Authorization', `Bearer ${TOKEN}`)

            .send({

                name: "Оновлений Користувач",

                status: "inactive"

            });

        expect(res.status).to.equal(200);

        expect(res.body.data).to.have.property('name', 'Оновлений Користувач');

        console.log(res.body.data)

    });

    it('DELETE /users/:id - Видалення користувача', async () => {

        const res = await request(API\_URL)

            .delete(`/users/${userId}`)

            .set('Authorization', `Bearer ${TOKEN}`);

        expect(res.status).to.equal(204);

    });

});

