**Лабораторна робота**

**З дисципліни:**

**Програмування WEB додатків**

**На тему: Node js**

Виконав:

Студент 3 курсу

Групи ПЗС-2044

Бойко Денис

**Лабораторна робота**

**Тема:** Node js

**Мета:** Навчитись використувати Node js.

Хід роботи

1. Через консоль підключив компоненти необхідні для роботи з Node js, та створив файл index.js в корені папки з майбутнім проектом

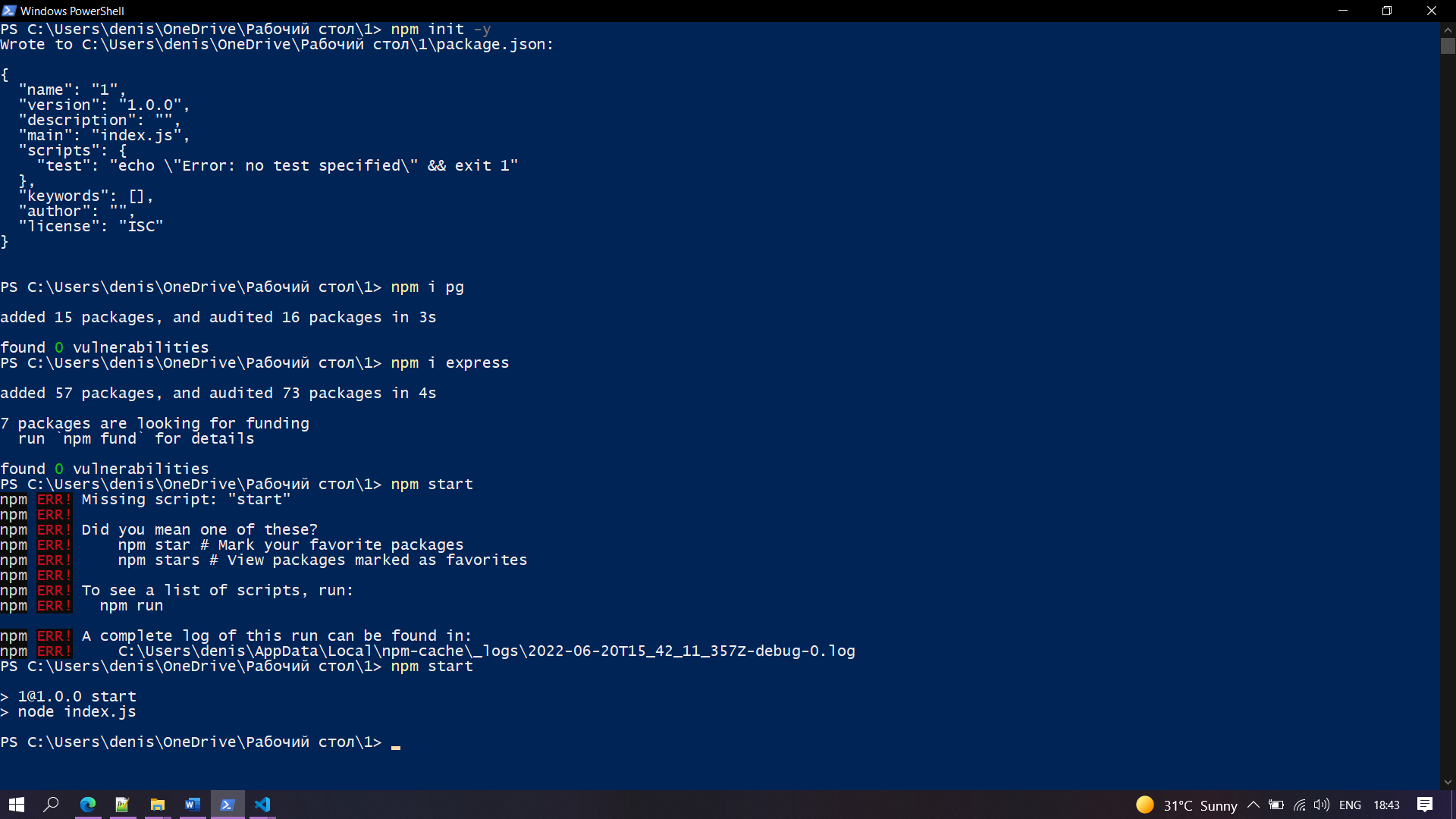


Рисунок 1 – створений конфігураційний файл для Node js

1. Загрузив за допомогою консолі необхідні модулі Node js

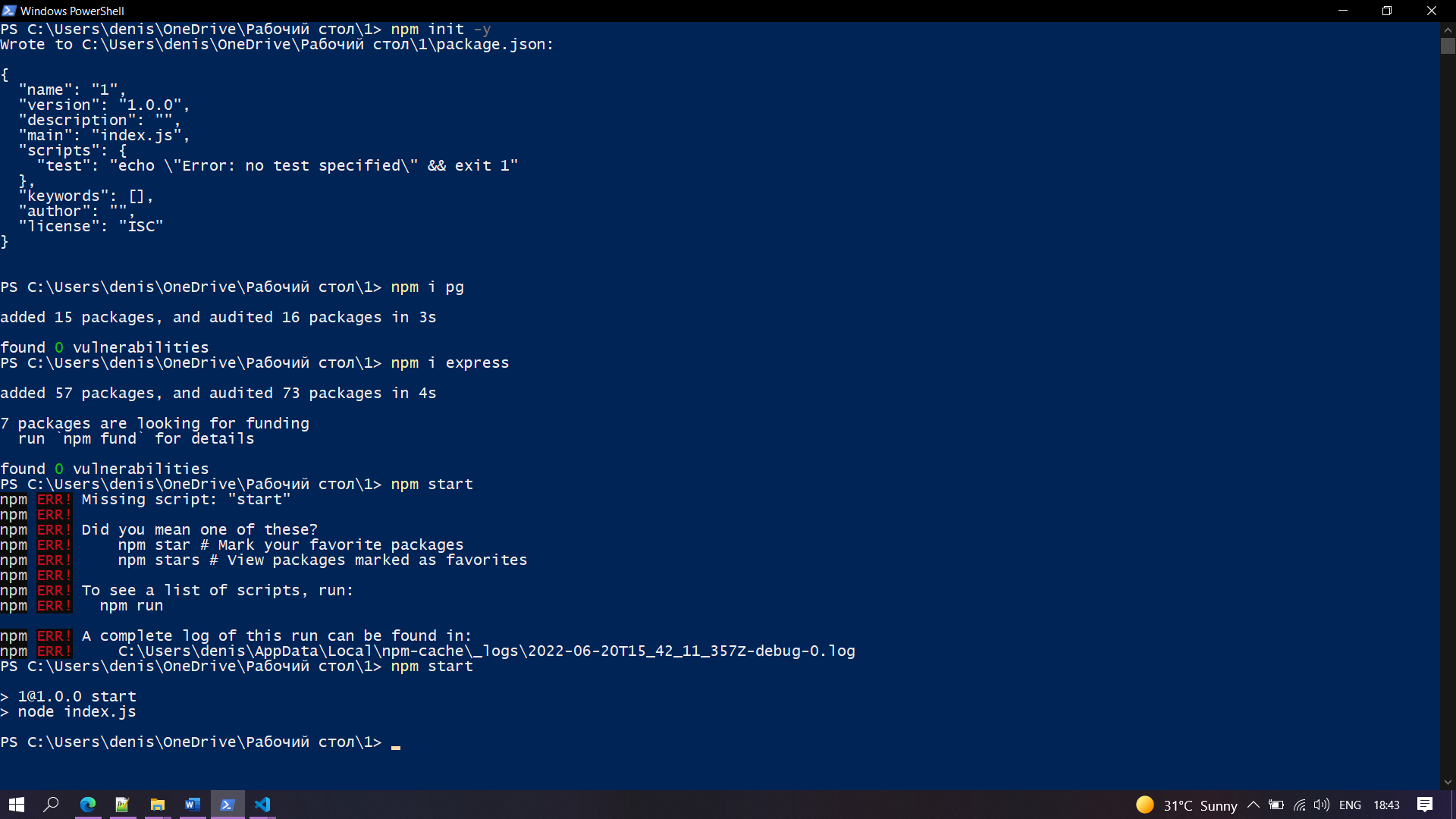


Рисунок 2 - Загружені за допомогою консолі необхідні модулі Node js

1. Створив сервер і підключив Бд до Node js

Лістинг програми:

const http = require("http");

const fs = require("fs");

const { Client } = require('pg')

const client = new Client({

user: 'postgres',

host: 'localhost',

database: 'mob\_phone',

password: '1234566',

port: 5432,

})

client.connect(function (err) {

if (err) throw err;

console.log("Connected!");

});

http.createServer(function (request, response) {

if (request.url === "/" && request.method === "GET") {

const filePath = "index.html";

fs.access(filePath, fs.constants.R\_OK, err => {

if (err) {

response.statusCode = 404;

response.end("Resourse not found!");

}

else {

fs.createReadStream(filePath).pipe(response);

}

});

}

}).listen(8081);

console.log('Server running at <http://127.0.0.1:8081/>');

1. Робота з Бд.

Вивів таблицю *teleph* з підключеної Бд

Лістинг запиту

else if (request.url === "/teleph" && request.method === "GET") {

response.writeHead(200, { "Content-Type": "application/json" });

let query = `

SELECT \*

FROM teleph

`;

let mas = [];

client.query(query, (err, res) => {

if (err) {

console.error(err);

return;

}

let row

for (row of res.rows) {

mas.push(row);

}

response.end(JSON.stringify(mas));

response.end();

});

}

Прийом на сервері

function GetTeleph(){

fetch('http://127.0.0.1:8081/teleph')

.then((response) => {

return response.json();

})

.then((data) => {

let itemList = data;

let items\_table = document.getElementById('table');

items\_table.innerHTML = '<th scope="col">Код</th> <th scope="col">Назва моделі</th> <th scope="col">код бренду</th> <th scope="col">код процесору</th><th scope="col">код типу екрану</th>';

for(let i = 0; i < itemList.length; i++)

{

items\_table.innerHTML+= '<tr><td>' + itemList[i].id\_teleph + '</td><td>' + itemList[i].model\_teleph + '</td><td>' + itemList[i].id\_brend + '</td><td>' + itemList[i].id\_cpu + '</td><td>' + itemList[i].id\_screen + '</td></tr>';

}

});

}

Результат виконання:

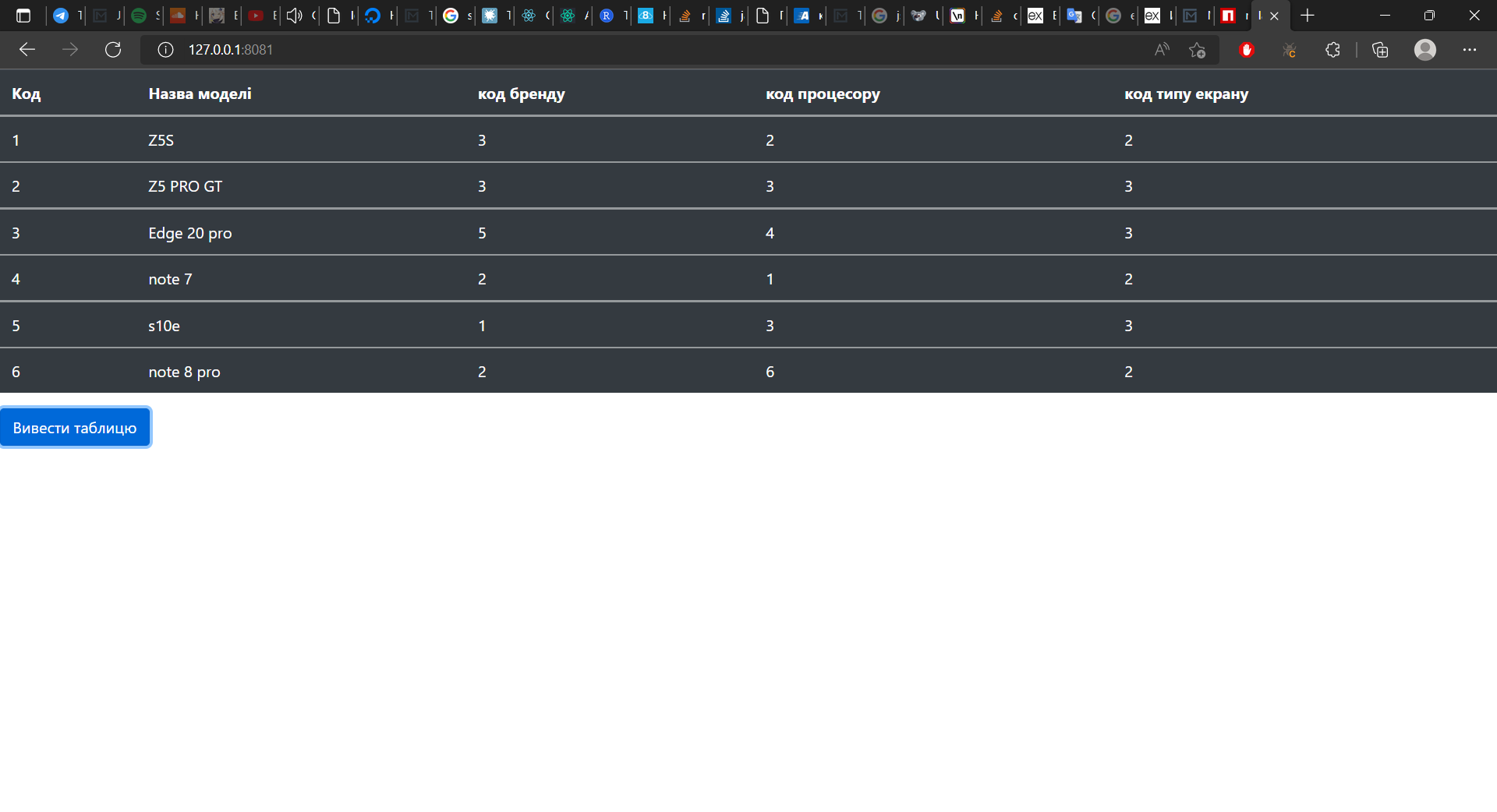


Рисунок 3 – результат підключення сервера Node js і виводу інформації з сервера Posgresql pg admin

Висновок: Отже, я навчився використувати Node js для створення серверів бекенду