

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Львівський національний університет імені Івана Франка
Факультет електроніки та комп'ютерних технологій

Звіт
про виконання лабораторної роботи №3
З курсу **“Веб програмування на стороні сервера”**
**“Практика програмування на JavaScript. Використання вбудованих та
сторонніх модулів”**

Виконав:
студент групи ФЕІ-25
МАЗУРИК Богуслав

Перевірив:
ЧМИХАЛО Олександр

Львів-2025

Мета:

- Використати навички програмування на JavaScript та роботи з JSON
- Опанувати використання вбудованих модулів Node.js
- Навчитися створювати package.json для проекту
- Навчитися встановлювати сторонні пакети npm

Хід роботи:

1. Ознайомився з методичкою до лабораторної роботи.
2. Створив новий репозиторій на гітхабі та файли package.json, main.js, .gitignore у папці лабораторної роботи. Також додав до цієї папки файл weather.json та встановив модуль commander для зручної обробки аргументів командного рядка.
3. Після підключення локального репозиторію до гітхабу та коміту проекту написав код який обробляє аргументи командного рядка, перевіряє коректність їх введення, витягує потрібну інформацію з файлу weather.json та формує відповідь на основі згаданих раніше аргументів.
4. Протестував програму та написав звіт.

Висновок:

В цілому, лабораторною роботою та своїм виконанням я задоволений. Вона не є складною і досить інтуїтивно зрозуміла.

Код та приклад роботи:

```
PS D:\Унік\2 курс\Web\лаб. 3> node main.js -i weather.json -d -r 8.2 -o result.txt
rainfall: 39.8, pressure: 1007
rainfall: 16.2, pressure: 1020.7
rainfall: 25.8, pressure: 1014.9
rainfall: 22.6, pressure: 1014.1
rainfall: 11, pressure: 996.8
rainfall: 17.4, pressure: 1009.1
rainfall: 14.4, pressure: 1013.4
rainfall: 18.8, pressure: 1006
rainfall: 12.2, pressure: 1004
rainfall: 9, pressure: 1007.8
rainfall: 16.2, pressure: 1005
rainfall: 11, pressure: 1011
rainfall: 10.4, pressure: 1014.5
rainfall: 19.8, pressure: 1008.4
rainfall: 16.8, pressure: 1019.6
rainfall: 19.2, pressure: 1010.3
rainfall: 9.8, pressure: 1006
rainfall: 17.4, pressure: 1017.8
rainfall: 13.2, pressure: 1016.7
Результат успішно записано в файл: result.txt
PS D:\Унік\2 курс\Web\лаб. 3> node main.js -d -r 9.1 -o result.txt -h
Please, specify input file
PS D:\Унік\2 курс\Web\лаб. 3> node main.js -i deletedfile.json -d -r 9.1 -o result.txt -h
Cannot find input file
PS D:\Унік\2 курс\Web\лаб. 3> node main.js -i weather.json -d -r 19 -o -h
rainfall: 39.8, pressure: 1007, humidity: 56
rainfall: 25.8, pressure: 1014.9, humidity: 79
rainfall: 22.6, pressure: 1014.1, humidity: 90
rainfall: 19.8, pressure: 1008.4, humidity: 50
rainfall: 19.2, pressure: 1010.3, humidity: 56
Помилка при збереженні файлу: TypeError [ERR_INVALID_ARG_TYPE]: The "path" argument must be of type string or an instance of Buffer or URL. Received type boolean (true)
    at Object.writeFileSync (node:fs:2422:14)
    at files:///D:/Users/ASUS/AppData/Local/Programs/Node.js/node_modules/esm/loader:647:26
    at ModuleJob.run (node:internal/modules/esm/module_job:343:25)
    at async onImport.tracePromise.__proto__ (node:internal/modules/esm/loader:647:26)
    at async asyncRunEntryPointWithESMLoader (node:internal/modules/run_main:117:5) {
  code: 'ERR_INVALID_ARG_TYPE'
}
PS D:\Унік\2 курс\Web\лаб. 3>
```

```
result.txt
1  rainfall: 39.8, pressure: 1007
2  rainfall: 16.2, pressure: 1020.7
3  rainfall: 25.8, pressure: 1014.9
4  rainfall: 22.6, pressure: 1014.1
5  rainfall: 11, pressure: 996.8
6  rainfall: 17.4, pressure: 1009.1
7  rainfall: 14.4, pressure: 1013.4
8  rainfall: 18.8, pressure: 1006
9  rainfall: 12.2, pressure: 1004
10 rainfall: 9, pressure: 1007.8
11 rainfall: 16.2, pressure: 1005
12 rainfall: 11, pressure: 1011
13 rainfall: 10.4, pressure: 1014.5
14 rainfall: 19.8, pressure: 1008.4
15 rainfall: 16.8, pressure: 1019.6
16 rainfall: 19.2, pressure: 1010.3
17 rainfall: 9.8, pressure: 1006
18 rainfall: 17.4, pressure: 1017.8
19 rainfall: 13.2, pressure: 1016.7
```

```
import fs from 'fs';

import { program } from 'commander';

program.option('-i, --input <path>', 'шлях до вхідного json файлу');
program.option('-o, --output [path]', 'шлях для збереження результату');
program.option('-d, --display', 'виведення результату в консоль');
program.option('-h, --humidity', 'вивести вологість');
program.option('-r, --rainfall <number>', 'фільтрувати за кількістю опадів');

program.parse(process.argv);

const options = program.opts();

if (!options.input) {
  console.error("Please, specify input file");
  process.exit(1);
}

let data;

try {
  const fileContent = fs.readFileSync(options.input, 'utf8');
  data = JSON.parse(fileContent);
} catch (error) {
  console.error("Cannot find input file");
  process.exit(1);
}

let processedData = data;

if (options.rainfall) {
  const rainLevel = parseFloat(options.rainfall);
```

```
processedData = processedData.filter(item => {  
    return item.Rainfall > rainLevel;  
});  
}  
  
const resultLines = processedData.map(item => {  
    let line = `rainfall: ${item.Rainfall}, pressure: ${item.Pressure3pm}`;  
    if (options.humidity) {  
        line += `, humidity: ${item.Humidity3pm}`;  
    }  
    return line;  
})  
  
const outputString = resultLines.join("\n");  
  
if (options.display) {  
    console.log(outputString);  
}  
  
if (options.output) {  
    try {  
        fs.writeFileSync(options.output, outputString);  
        console.log('Результат успішно записано в файл:', options.output);  
    } catch (error) {  
        console.error('Помилка при збереженні файлу:', error);  
    }  
}
```