

Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут ім. Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки

Звіт до комп'ютерного практикуму №4
З дисципліни «Основи Back-end технологій»

ВИКОНАЛА:

Студентка групи ІМ-21

Рабійчук Дар'я Олександрівна

№ у списку(варіант) - 4

ПЕРЕВІРИВ:

доц. Голубєв Л.П.

Київ 2025

Тема: NodeJs. Створення серверу за допомогою express. Обробка маршрутів. Шаблонізація.

Завдання.

Розробити веб-застосунок для отримання даних про погоду

- В шаблоні потрібно сформувати меню посилань з назвами міст (одним із пунктів меню має бути пункт із зазначенням місцезнаходження автора)
- Формат рядка запиту для отримання даних про погоду: `/weather/{city}`, де `city` - назва вибраного міста
- Дані про погоду можна отримати відправкою запиту на OpenWeatherMap
- Advanced. Отримати дані про погоду в місцезнаходженні користувача за таким URI: `/weather/`

App.js

```
const express = require("express");

const hbs = require("hbs");
const path = require("path");
const axios = require("axios");
const fetch = require('node-fetch');

const app = express();
const port = 3000;

app.set("view engine", "hbs");
app.set("views", path.join(__dirname, "views"));
hbs.registerPartials(path.join(__dirname, "/views/partials"));

const cityMap = {
  Lviv: "Lviv",
  Ternopil: "Ternopil",
  Odesa: "Odesa",
  Kyiv: "Kyiv",
  Cherkasy: "Cherkasy",
  Obukhiv: "Obukhiv"
};

app.get("/", (req, res) => {
  res.render("weather", { cities: Object.keys(cityMap) });
});

app.get("/weather/:city", async (req, res) => {
  const urlCity = req.params.city;
  const city = cityMap[urlCity];
```

```
    if (!city) {
      return res.render("weather", { error: "Місто не знайдено", cities:
Object.keys(cityMap) });
    }

    const apiKey = "7f4a351bddd5db7ab96a4851a6ba05aa";
    try {
      const response = await
axios.get(`https://api.openweathermap.org/data/2.5/weather`, {
        params: {
          q: city,
          appid: apiKey,
          units: "metric",
          lang: "ua"
        }
      });
      const weather = response.data;
      res.render("weather", { weather, cities: Object.keys(cityMap), selectedCity:
city });
    } catch (error) {
      res.render("weather", { error: "Місто не знайдено", cities:
Object.keys(cityMap) });
    }
  });

app.listen(port, () => {
  console.log(`Example app listening on port ${port}`);
});
```

Запуск програми:

```
PS C:\Users\HP\Desktop\универ\backend\weatherApp> nodemon app.js
[nodemon] 3.1.9
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node app.js`
Example app listening on port 3000
█
```

Результат виконання:

Weather App KPI

Select a city to view weather:

Lviv

Ternopil

Odesa

Kyiv

Cherkasy

Obukhiv

Погода в Kyiv

Температура: 4.29°C

Опис: уривчасті хмари

Weather App KPI

Select a city to view weather:

Lviv

Ternopil

Odesa

Kyiv

Cherkasy

Obukhiv

Погода в Cherkasy

Температура: 3.22°C

Опис: уривчасті хмари

Weather App KPI

Select a city to view weather:

Lviv

Ternopil

Odesa

Kyiv

Cherkasy

Obukhiv

Погода в Lviv

Температура: 5.06°C

Опис: хмарно

Висновок

У результаті виконання лабораторної роботи було створено веб-застосунок на платформі Node.js з використанням фреймворку Express. Основні досягнення:

- 1) Реалізовано обробку маршрутів за допомогою `express.Router`, включаючи динамічний маршрут `/weather/:city`, який дозволяє отримувати прогноз погоди для обраного міста.
- 2) Використано шаблонізатор Handlebars (hbs) для динамічного формування HTML-сторінки.
- 3) Побудовано інтерфейс із меню кнопок міст, включаючи пункт із зазначенням місцезнаходження користувача.

4) Здійснено інтеграцію з OpenWeatherMap API, через який у реальному часі отримуються дані про погоду.

5) Реалізовано вивід помилки у випадку, якщо вказане місто не знайдено або виникла помилка під час запиту.

Даний застосунок демонструє базові принципи побудови серверного додатку на Node.js, включаючи маршрутизацію, роботу з API, шаблонізацію та стилізацію інтерфейсу.