

ЛАБОРАТОРНА РОБОТА № 2

Express, HBS. Проект “WeatherApp”

Завдання 1. Створення проекту.

- В середовищі розробки відкриваємо проект NodeJS.
- Створюємо папку поточного проекту NodeJS/WeatherApp, `npm init -y`.
- Встановлюємо фреймворк express для даного проекту.

Створюємо файл додатку *app.js*, в якому створюємо сервер на express.

Завдання 2. Nodemon.

Встановлюємо nodemon глобально:

```
npm install nodemon -g
```

Запускаємо застосунок, вказавши розширення файлів, які будуть змінюватись:

```
nodemon -e js,hbs,json app
```

Завдання 3. Обробка маршрутів.

Для прикладу встановлюємо обробник для маршрутів /weather:

```
app.get('/weather', (req, res) => {  
  res.render('weather', { cities: citiesWeatherData.cities });  
});
```

Завдання 4. Шаблонізація.

- Встановлюємо hbs: `npm install hbs`;
- Підключаємо модуль в коді: `const hbs = require("hbs");`
- Задаємо розширення hbs для html-рендеринга по замовчуванню:
`app.set('view engine', 'hbs');`

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.24.122.15.000 – Лр.2						
Змн.	Арк.	№ докум.	Підпис	Дата	Звіт з лабораторної роботи №2			Літ.	Арк.	Аркушів	
Розроб.		Невмержицький Т.В.									
Перевір.		Лисенко М.С.								1	4
Реценз.								ФІКТ, гр. КН-22-2			
Н. Контр.											
Зав.каф.											

Завдання 6. Розробити веб-застосунок для отримання даних про погоду.

- Дані про міста повинні отримуватись із json-файлу та передаватись в html-шаблон у вигляді масиву
- В шаблоні потрібно сформувати меню посилань з назвами міст
- 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/

Результат виконання програми:

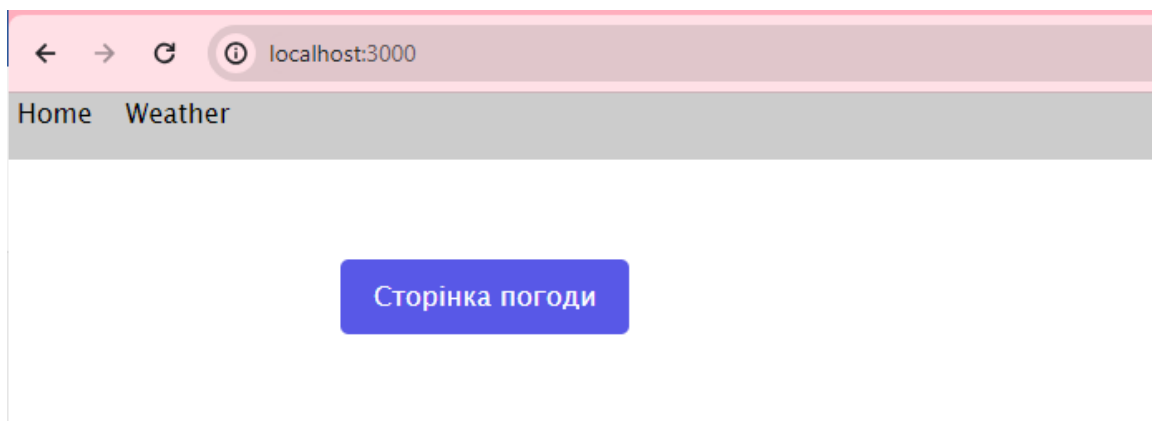


Рис. 2.1. Головна сторінка

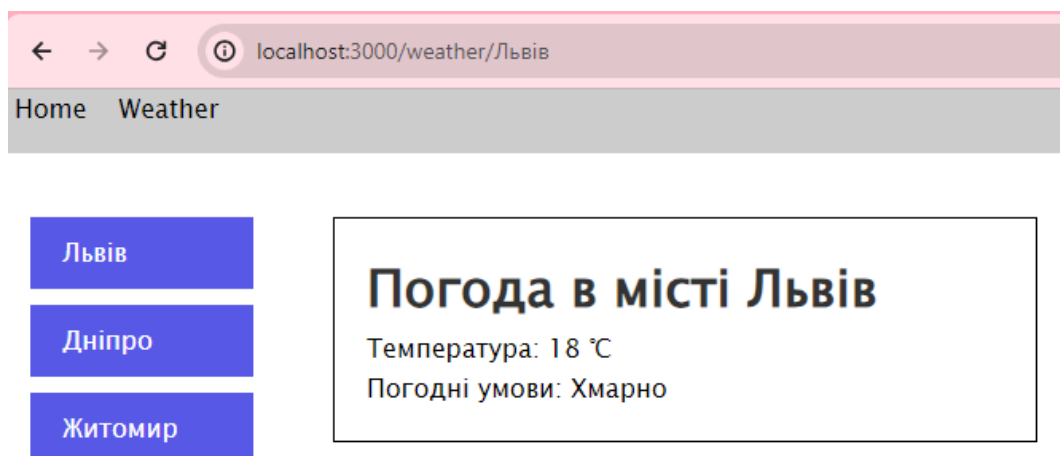


Рис. 2.2. Сторінка з погодою

Лістинг файлу app.js :

```
const hbs = require("hbs");
const fs = require("fs");
const express = require("express");
const app = express();

app.set('view engine', 'hbs');
hbs.registerPartials(__dirname + '/views/partials');

const citiesWeatherData = JSON.parse(fs.readFileSync('cities.json', 'utf8'));

app.get('/', (req, res) => {
  res.render('index.hbs');
});

app.get('/weather', (req, res) => {
  res.render('weather', { cities: citiesWeatherData.cities });
});

app.get('/weather/:city', (req, res) =>{
  const cityName = req.params.city;
  const cityData = citiesWeatherData.cities.find(city => city.name.toLowerCase()
=== cityName.toLowerCase());
  if (cityData !== null) {
    res.render('weather', { cityData: cityData, cities: citiesWeatherData.cities
});
  } else {
    res.send('City not found');
  }
});

app.listen(3000, () => {
  console.log("Example app listening on port 3000");
});
```

Лістинг weather.hbs:

```
{{> header}}
<div class="weather">
  <div class="city-buttons">
    {{#each cities}}
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.24.122.15.000 – Лр.2	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		3

```

<a href="/weather/{{name}}" class="city-button">{{name}}</a>
    {{/each}}
</div>
{{#if cityData}}
<div class="weather-info">
    <h1>Погода в місті {{ cityData.name }}</h1>

    <p>Температура: {{ cityData.temperature }} &#8451;</p>
    <p>Погодні умови: {{ cityData.condition }}</p>
</div>
</div>
{{/if}}
{{> footer}}

```

Лістинг файлу cities.json:

```

{
  "cities": [
    {
      "name": "Львів",
      "temperature": 18,
      "condition": "Хмарно"
    },
    {
      "name": "Дніпро",
      "temperature": 22,
      "condition": "Дощ"
    },
    {
      "name": "Житомир",
      "temperature": 20,
      "condition": "Сонячно"
    }
  ]
}

```

Посилання на репозиторій: https://github.com/tarasnevmer/Node.js_Lab_2

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.24.122.15.000 – Лр.2	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		4