Міністерство освіти та науки України

Національний технічний університет України “КПІ ім. Ігоря Сікорського”

Факультет інформатики та обчислювальної техніки

Кафедра автоматизованих систем обробки інформації та управління

**ЗВІТ**

про виконання

комп’ютерного практикуму №7

З дисципліни «Проектування інформаційних систем»

на тему:

**«**Лабораторна робота 7. Доступ до

зовнішнього API**»**

**Мощицька Ольга**

**ІС-83**

Київ  2020

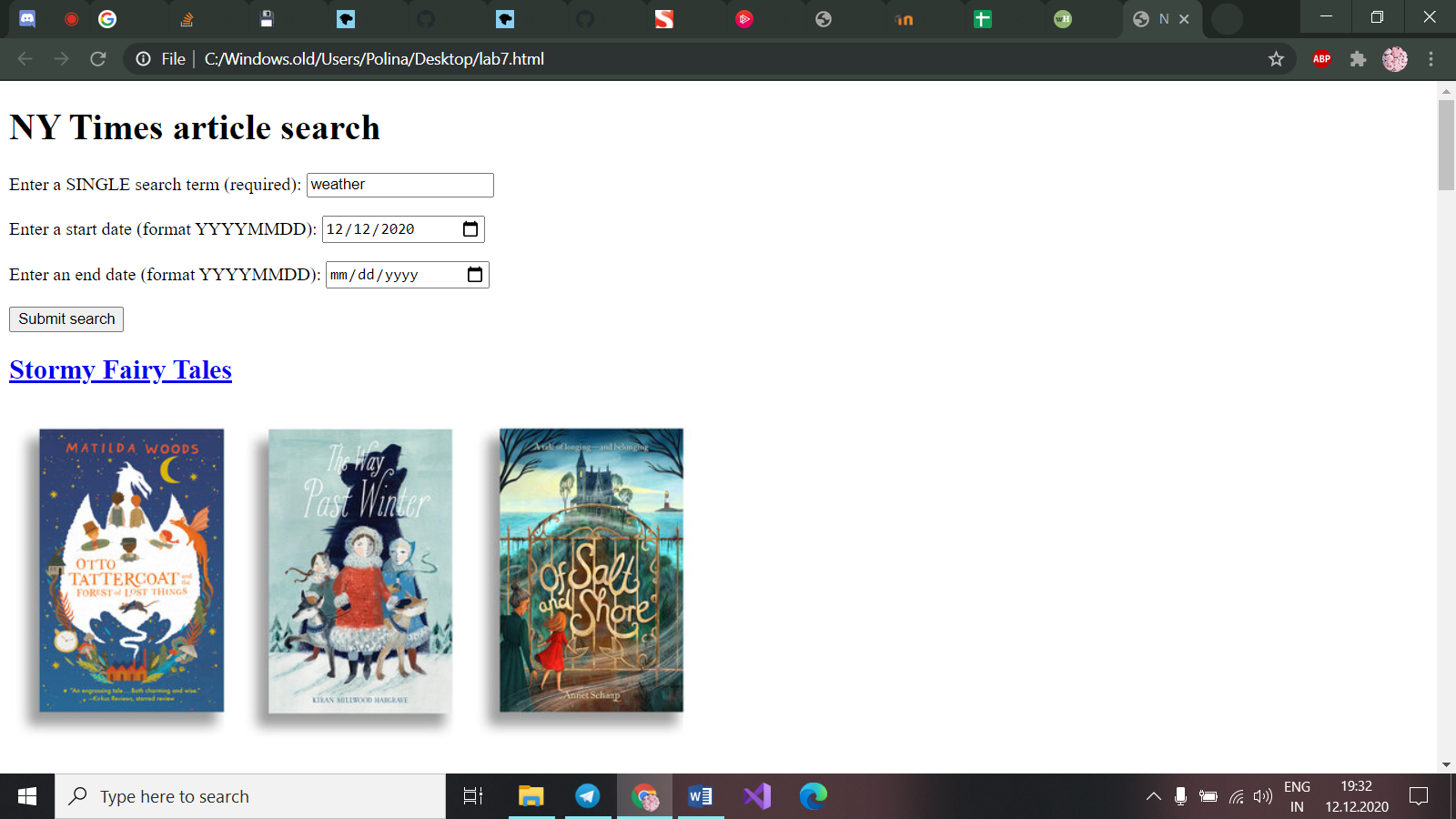
**Лабораторна робота 7. Доступ до**

**зовнішнього API**

Завдання: Підключити зовнішній сервіс до свого рішення. Сервіс можна використовувати існуючий або створити власний. Приклади зовнішніх сервісів наведені в Додатку.

Вправа 7.1. Обрати зовнішній сервіс. і описати сценарій його використання. Додати до існуючого в GitHub репозіторію з ЛР6 опис сценарію використання API зовнішнього застосування.

Вправа 7.2. Реалізувати на будь-якій мові програмування виклик API зовнішнього застосування і візуалізацію відповіді. Додати в GitHub код і опис прикладу застосування API.



Скріншот із сайту

Я використовувала API NYTimes. Документація API New York Times знаходиться на https: / /developer.nytimes.com/. Більшість API вимагають використання якогось ключа розробника з міркувань безпеки та підзвітності. Щоб зареєструватися для ключа NYTimes API, дотримуючись інструкцій на <https://developer.nytimes.com/get-started>.

У підсумку програма дозволить мені ввести в пошуку термін та необов’язково дати початку та закінчення, які потім буде використовувати для запиту API пошуку статей та відображення результатів пошуку. На скріншоті є приклад пошуку статей за темою погода.

Посилання, яке я використовувала

*const baseURL = 'https://api.nytimes.com/svc/search/v2/articlesearch.json';*

*const key = '3Cm3bHxG1I3ROE2N94Y8vw7347XEAaQk';*

Код наведений нжче, також посилання на гітхаб із файлом

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>NY Times API example</title>

<link href="nytimes.css" rel="stylesheet">

</head>

<body>

<h1>NY Times article search</h1>

<div class="wrapper">

<div class="controls">

<form>

<p>

<label for="search">Enter a SINGLE search term (required): </label>

<input type="text" id="search" class="search" required>

</p>

<p>

<label for="start-date">Enter a start date (format YYYYMMDD): </label>

<input type="date" id="start-date" class="start-date" pattern="[0-9]{8}">

</p>

<p>

<label for="end-date">Enter an end date (format YYYYMMDD): </label>

<input type="date" id="end-date" class="end-date" pattern="[0-9]{8}">

</p>

<p>

<button class="submit">Submit search</button>

</p>

</form>

</div>

<div class="results">

<nav>

<button class="prev">Previous 10</button>

<button class="next">Next 10</button>

</nav>

<section>

</section>

</div>

</div>

<script>

// Defining a baseURL and key to as part of the request URL

const baseURL = 'https://api.nytimes.com/svc/search/v2/articlesearch.json';

const key = '3Cm3bHxG1I3ROE2N94Y8vw7347XEAaQk';

let url;

// Grab references to all the DOM elements you'll need to manipulate

const searchTerm = document.querySelector('.search');

const startDate = document.querySelector('.start-date');

const endDate = document.querySelector('.end-date');

const searchForm = document.querySelector('form');

// This is never used

// const submitBtn = document.querySelector('.submit');

const nextBtn = document.querySelector('.next');

const previousBtn = document.querySelector('.prev');

const section = document.querySelector('section');

const nav = document.querySelector('nav');

// Hide the "Previous"/"Next" navigation to begin with, as we don't need it immediately

nav.style.display = 'none';

// define the initial page number and status of the navigation being displayed

let pageNumber = 0;

// This is never used

// let displayNav = false;

// Event listeners to control the functionality

// Event listeners to control the functionality

searchForm.addEventListener('submit', submitSearch);

nextBtn.addEventListener('click', nextPage);

previousBtn.addEventListener('click', previousPage);

function submitSearch(e){

pageNumber = 0;

fetchResults(e);

}

function fetchResults(e) {

// Use preventDefault() to stop the form submitting

e.preventDefault();

// Assemble the full URL

url = baseURL + '?api-key=' + key + '&page=' + pageNumber + '&q=' + searchTerm.value + '&fq=document\_type:("article")';

if(startDate.value !== '') {

url += '&begin\_date=' + startDate.value;

};

if(endDate.value !== '') {

url += '&end\_date=' + endDate.value;

};

// Use fetch() to make the request to the API

fetch(url).then(function(result) {

return result.json();

}).then(function(json) {

displayResults(json);

});

}

function displayResults(json) {

while (section.firstChild) {

section.removeChild(section.firstChild);

}

const articles = json.response.docs;

if(articles.length === 10) {

nav.style.display = 'block';

} else {

nav.style.display = 'none';

}

if(articles.length === 0) {

const para = document.createElement('p');

para.textContent = 'No results returned.'

section.appendChild(para);

} else {

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

const article = document.createElement('article');

const heading = document.createElement('h2');

const link = document.createElement('a');

const img = document.createElement('img');

const para1 = document.createElement('p');

const para2 = document.createElement('p');

const clearfix = document.createElement('div');

const current = articles[i];

console.log(current);

link.href = current.web\_url;

link.textContent = current.headline.main;

para1.textContent = current.snippet;

para2.textContent = 'Keywords: ';

for(let j = 0; j < current.keywords.length; j++) {

const span = document.createElement('span');

span.textContent = current.keywords[j].value + ' ';

para2.appendChild(span);

}

if(current.multimedia.length > 0) {

img.src = 'http://www.nytimes.com/' + current.multimedia[0].url;

img.alt = current.headline.main;

}

clearfix.setAttribute('class','clearfix');

article.appendChild(heading);

heading.appendChild(link);

article.appendChild(img);

article.appendChild(para1);

article.appendChild(para2);

article.appendChild(clearfix);

section.appendChild(article);

}

}

};

function nextPage(e) {

pageNumber++;

fetchResults(e);

};

function previousPage(e) {

if(pageNumber > 0) {

pageNumber--;

} else {

return;

}

fetchResults(e);

};

</script>

</body>

</html>