

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації та управління

Лабораторна робота №6
З дисципліни «Проектування інформаційних систем»
На тему: «Використання і створення API»

Виконав студент гр. ІС-91
Данилюк Ю.С.
Перевірив доц. каф. АСОІУ
Попенко В. Д.

Київ
2021

Вправа 6.1

Обрати зовнішній сервіс. і описати сценарій його використання. Створити в GitHub репозиторій з описом сценарію використання API зовнішнього застосування.

[API даних COVID-19](#)

Кожні 15 хвилин оновлюється статистика про коронавірус. Останні статистичні дані за країнами зібрані з кількох надійних джерел. Також можна переглянути загальну статистику та щоденні звіти. Маючи 6 серверів по всьому світу, [API даних COVID-19](#) також є найшвидшим API, пов'язаним з COVID-19 на ринку RapidAPI.

Я обрав цю [API](#), яка надає можливість отримати останні дані про COVID-19 для конкретної країни.

Дані надаються в наступному вигляді:

Body Schema Headers

```
[ 1 item
  0 : { 10 items
    "country" : "Italy"
    "code" : "IT"
    "confirmed" : 219070
    "recovered" : 105186
    "critical" : 1027
    "deaths" : 30560
    "latitude" : 41.87194
    "longitude" : 12.56738
    "lastChange" : "2020-05-10T18:12:02+02:00"
    "lastUpdate" : "2020-05-11T04:00:03+02:00"
  }
]
```

Вправа 6.2

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

Відображення даних отриманих з API:

COVID-19 data in Ukraine

Country	Confirmed	Recovered	Critical	Deaths	Date
Ukraine	1614707	1290158	177	31461	2021-03-26T16:45:03+01:00

Також я розмістив це на сторінці в GitHub:

<https://yuradanyliuk.github.io/SimpleUsingAPI/>

Таким чином, можна відіслати запит до API і подивитись відповідь у браузері. В HTML документі із зовнішнім кодом (у окремому файлі) JavaScript виклик виглядає так:

index.html:

```
<!DOCTYPE html>
<html lang="en">

  <head>
    <!-- Required meta tags -->
    <meta charset="UTF-8"/>
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <title>COVID-19</title>

    <title>COVID-19 data in Ukraine</title>
    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/
css/bootstrap.min.css"
      integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T" crossorigin="anon
ymous"/>

    <script src="functions.js"></script>
  </head>

  <body class="bg-info" onload="GetCountryData('Ukraine')">
    <h1 class="display-4 text-center">COVID-19 data in Ukraine</h1>
    <section class="container pt-3"></section>
  </body>

</html>
```

functions.js:

```
function GetCountryData(countryName)
{
  //Create request
```

```

let request = new XMLHttpRequest();
//Setting request
request.open("GET", "https://covid-19-
data.p.rapidapi.com/country?name="+countryName+"&format=json");
request.setRequestHeader("x-rapidapi-
key", "ee846d23cdmsh187974b70a52be6p160f83jsnf587f225711f");
request.setRequestHeader("x-rapidapi-host", "covid-19-data.p.rapidapi.com");
request.responseType = "json";
//Send request
request.send();
//On load output data
request.onload = function ()
{
    let colNames = ["Country", "Confirmed", "Recovered", "Critical", "Deaths", "Dat
e"];
    let response = request.response[0];
    let data = [response.country, response.confirmed, response.recovered, response.
critical, response.deaths, response.lastUpdate];
    OutputData(colNames,data);
};
}

function OutputData(colNames,data)
{
    //Get section element
    let section = document.querySelector("section");
    //Create table
    let table = document.createElement("table");
    table.classList.add("table", "table-bordered");
    //Create table heading
    let thead = document.createElement("thead");
    thead.classList.add("thead-dark");

    //Create table heading row and add elements to it
    let tr = document.createElement("tr");
    tr.classList.add("text-center");

    let t;
    for (var i = 0; i < colNames.length; i++)
    {
        t = document.createElement("th");
        t.textContent = colNames[i];
        t.scope = "col";
        tr.appendChild(t);
    }

    thead.appendChild(tr);

```

```

table.appendChild(thead);

//Add table rows from each json object
tr = document.createElement("tr");
tr.classList.add("table-secondary", "text-center");

//Create table heading row and add elements to it
for (var i = 0; i < data.length; i++)
{
    t = document.createElement("th");
    t.textContent = data[i];
    t.scope = "row";
    tr.appendChild(t);
}

table.appendChild(tr);
section.appendChild(table);
}

```

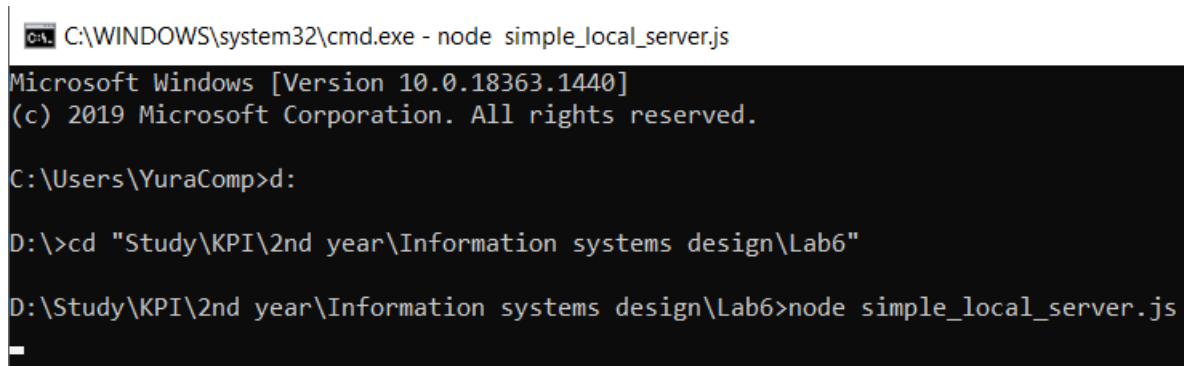
Посилання на репозиторій на GitHub де розміщені всі скрипти:

<https://github.com/yuradanyliuk/SimpleUsingAPI>

Вправа 6.3

Реалізувати на будь-якій мові програмування http-сервер, організувати звертання до нього і отримати відповідь у браузері. Звертання має містити Ваш логін у Moodle. Відповідь має містити Ваші особисті дані (прізвище, ім'я, курс, група).

Створимо сервер локально на комп'ютері і доступ до нього здійснюємо через порт 3000. Скрипт `simple_local_server.js` запустимо з консолі Windows.



```

C:\WINDOWS\system32\cmd.exe - node simple_local_server.js
Microsoft Windows [Version 10.0.18363.1440]
(c) 2019 Microsoft Corporation. All rights reserved.

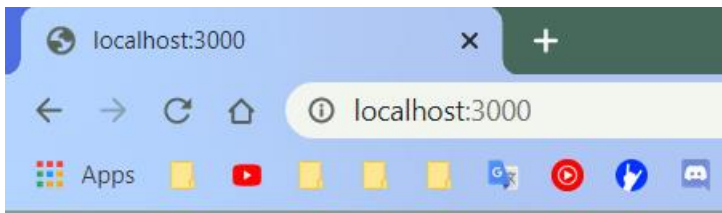
C:\Users\YuraComp>d:

D:\>cd "Study\KPI\2nd year\Information systems design\Lab6"

D:\Study\KPI\2nd year\Information systems design\Lab6>node simple_local_server.js

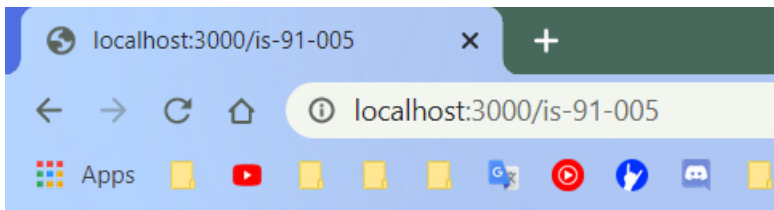
```

У браузері по адресі <http://localhost:3000/> бачимо відформатовану згідно таблиці стилів відповідь сервера.



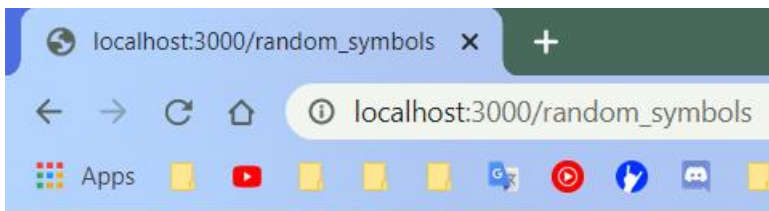
Home

Реалізовуємо звертання яке містити мій логін у Moodle. Відповідь містить мій особисті дані (прізвище, ім'я, курс, група).



Yuriy Danyliuk IS-91 2nd year

Реалізовуємо звертання яке не обробляється.



Not found

Код на JavaScript для побудови http-сервера засобами Node.js:

```
const http = require('http')

http.createServer(function(request, response)
{
    response.setHeader("Content-Type", "text/html; charset=utf-8;");
    if(request.url === "/home" || request.url === "/")
        response.write('<h2>Home</h2>');
    else if(request.url === "/is-91-005")
        response.write('<h2>Yuriy Danyliuk IS-91 2nd year</h2>');
    else
        response.write('<h2>Not found</h2>');

    response.end();
}).listen(3000);
```