 МІНІСТЕРСТВО ОСВІТИ I НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ   
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

**Лабораторний практикум №8**

з дисципліни «*Веб-технології та веб-дизайн - 1*»

# на тему: *«*Reactive Extentions*»*

**Виконав:**

студент гр. БС-81

Збаровський Д. Д.

**Перевірив:**

ас. каф. БМК Матвійчук О. В.

Зараховано від \_\_\_.\_\_\_.\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис викладача)

Київ 2020

**Завдання:**

Створити 1 сторінковий сайт який має містити наступну інформацію:

Час з моменту останнього оновлення даних. (Бажано в header, та в секундах з точністю до 0.1)

Дані по позиціях на біржі для 8 найменувань (на ваш вибір)

**Пункт 1:**

Доступ до наних біржі відбувається за допомогою API описаного тут: <https://iexcloud.io/docs/api/>

Ключ доступу: pk\_ba9bb61a36a24ec3a5ae70362941f41c

Частота оновлень: 20 сек

p.s: даний ключ має задавитись конфігураційно та не повинен фігурувати на gitHub

**Пункт 2:**

Таблиця з результатами повинна містити наступні поля:

Назву компанії (Приклад: "Apple Inc.")

Назву позиції (Приклад: "AAPL")

Останню ціну позиції (Приклад: "158.73")

Різницю з моменту останнього оновлення (Приклад: "-1.67")

**При виконанні роботи обов'язково використовувати бібліотеку RXjs**

PS1: Потрібно чітко відділяти потоки роботи з даними

PS2: Статичні поля  необхідно запитувати лише 1 раз

Різницю в часі з моменту останньої угоди (Приклад: "2.9с") // постійно оновлювати

**Лістинг сторінки:**

**index.html**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Лабораторна №8</title>

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

</head>

<body>

<h3 id="update">Завантаження...</h3>

<table id="data\_table">

<tr>

<th>Назва компанії</th>

<th>Назва позиції</th>

<th>Остання ціна позиції</th>

<th>Різниця з моменту останнього оновлення</th>

</tr>

</table>

<script src="https://unpkg.com/rxjs/bundles/rxjs.umd.min.js"></script>

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

</body>

</html>

**Style.css**

table {

font-size: 14px;

background: white;

max-width: 50%;

width: 50%;

border-collapse: collapse;

text-align: left;

margin: 0 auto;

}

th {

font-weight: normal;

color: #039;

border-bottom: 2px solid #6678b1;

padding: 10px 8px;

}

td {

color: #669;

padding: 9px 8px;

transition: .3s linear;

}

tr:hover td {color: #6699ff;}

#update {

text-align: center;

margin: 50px auto;

color: #039;

}

**Js.js**

const { timer } = rxjs;

const { fromFetch } = rxjs.fetch;

const { filter, map, switchMap } = rxjs.operators;

const token = prompt("Введіть ключ");

const positions = ['aapl', 'amzn', 'msft', 'tsla', 'googl','dis','pep','nflx'];

const api = 'https://cloud.iexapis.com/stable';

function quote(position, field)

{

return fromFetch(`${api}/stock/${position}/quote/${field}?token=${token}`).pipe(switchMap(res => res.json()));

}

function myFunction()

{

const table = document.getElementById('data\_table');

const time = timer(250, 100);

const update$ = time.pipe(filter(x => x % 200 == 0));

for (const position of positions)

{

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

const td\_company\_name = document.createElement('td');

const td\_position = document.createElement('td');

const td\_price = document.createElement('td');

const td\_change = document.createElement('td');

tr.appendChild(td\_company\_name);

tr.appendChild(td\_position);

tr.appendChild(td\_price);

tr.appendChild(td\_change);

table.appendChild(tr);

quote(position, 'companyName').subscribe(text => td\_company\_name.innerHTML = text);

quote(position, 'symbol').subscribe(text => td\_position.innerHTML = text);

update$.subscribe(() =>

{

quote(position, 'latestPrice').subscribe(text => td\_price.innerHTML = text);

quote(position, 'change').subscribe(text => td\_change.innerHTML = text);

});

}

const update\_element = document.getElementById('update');

let update = 0;

update$.subscribe(() =>

{

update = 0.0;

});

time.subscribe(x =>

{

update += 0.1;

update\_element.innerHTML = `Останнє оновлення: ${update.toFixed(1)}с тому`;

});

}

myFunction();