 МІНІСТЕРСТВО ОСВІТИ I НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ   
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

**Комп’ютерний практикум №8**

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

**Виконав:**

студент гр. БС-83

Розмариця О.А.

**Перевірив:**

ас. каф. БМК Матвійчук О. В.

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

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

(підпис викладача)

Київ-2020

**Завдання:**

//const key = '';

const filter1 = 'https://cloud.iexapis.com/v1/stock/market/batch?symbols=TSLA,SCTY,PYPL,AAPL,MSFT,COKE,PEP,GOOG&types=quote&filter=symbol,companyName,latestPrice,latestUpdate&token=' + key;

const filter2 = 'https://cloud.iexapis.com/v1/stock/market/batch?symbols=TSLA,SCTY,PYPL,AAPL,MSFT,COKE,PEP,GOOG&types=quote&filter=latestPrice,latestUpdate&token=' + key;

let timeUpdate = 0;

function buildTable(data){

let tbody = document.createElement('tbody');

let nextTimeUpdate = document.createElement('span');

nextTimeUpdate.id = 'updateTime';

document.getElementById('tableCaption').appendChild(nextTimeUpdate);

for (i in data){

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

tr.id = i;

for (j in data[i].quote){

let td = document.createElement('td');

td.id = j;

td.textContent = data[i].quote[j];

tr.appendChild(td);

}

tbody.appendChild(tr);

}

document.getElementById('tableBase').appendChild(tbody);

}

let req = new XMLHttpRequest();

req.open('GET', filter1);

req.onreadystatechange = function(){

if (req.readyState === 4 && req.status === 200){

buildTable(calculateLatestUpdate(JSON.parse(req.responseText)));

timeUpdate = Date.now();

}

}

req.send('');

function updateTable(data){

for (i in data){

for (j in data[i].quote){

document.getElementById('tableBase').querySelector('#'+i+'> #'+j).textContent = data[i].quote[j];

}

}

}

function calculateLatestUpdate(data) {

let temp = Date.now();

for (i in data) {

data[i].quote.latestUpdate = Math.round((data[i].quote.latestUpdate - temp)/10)/100;

}

return data;

}

let interval$ = rxjs.interval(20000)

.subscribe( () => {

req.open('GET', filter2);

req.onreadystatechange = function() {

if (req.readyState === 4 && req.status === 200) {

updateTable(calculateLatestUpdate(JSON.parse(req.responseText)));

timeUpdate = Date.now();

}

}

req.send('');

});

let updateTime$ = rxjs.interval(100)

.subscribe( () => {

if (timeUpdate != 0) {

document.getElementById('updateTime').textContent = Math.round((Date.now() - timeUpdate) / 100 ) / 10;

}

});