Упражнение № 1

Напишите скрипт для получения погоды в Киеве, используя модуль http.

В это скрипте должны быть определены три функции:

getData - отправка запроса на OpenWeatherMap API

processData - обработка данных

saveToFile - запись данных в файл, используя fs.writeFile

var http = require('http');

var fs = require('fs');

var weather;

function getData(city, callback) {

http.get("http://api.openweathermap.org/data/2.5/weather?q="+String(city)+"&appid=c0c6030143a04eff2a48acd37bd81dbb", (res) => {

const { statusCode } = res;

const contentType = res.headers['content-type'];

let error;

if (statusCode !== 200) {

error = new Error('Request Failed.\n' +

`Status Code: ${statusCode}`);

} else if (!/^application\/json/.test(contentType)) {

error = new Error('Invalid content-type.\n' +

`Expected application/json but received ${contentType}`);

}

if (error) {

console.error(error.message);

// consume response data to free up memory

res.resume();

return;

}

res.setEncoding('utf8');

let rawData = '';

res.on('data', (chunk) => { rawData += chunk; });

res.on('end', () => {

try {

//console.log(parsedData);

return callback(rawData);

} catch (e) {

console.error(e.message);

}

});

}).on('error', (e) => {

console.error(`Got error: ${e.message}`);

});

}

function processData(data, callback) {

if (!data) {

console.log( "error message" + ': data not found');

} else {

//console.log(data);

var weather = JSON.parse(data);

weather = ("Country code: " + weather.sys.country + "\n"

+ "City: " + weather.name + "\n" +

"Temperature: " + (weather.main.temp - 273).toFixed(1) + " celsius\n" +

"Humidity: " + weather.main.humidity + "%\n\n");

return callback(weather);

}

}

function writeToFile(data) {

fs.writeFile('message.txt', data, (err) => {

if (err) throw err;

console.log('The file has been saved!');

});

}

getData("kiev", function(data) {

processData(data, function(data) {

writeToFile(data)

})

});

Упражение № 2

Напишите скрипт для получения погоды в Киеве, используя модуль request.

В это скрипте должны быть определены три функции:

getData - отправка запроса на OpenWeatherMap API

processData - обработка данных

saveToFile - запись данных в файл, используя fs.write

var fs = require('fs');

var request = require('request');

function getData(city,callback) {

request("http://api.openweathermap.org/data/2.5/weather?q="

+String(city)

+"&appid=c0c6030143a04eff2a48acd37bd81dbb", function (error, response, body) {

if (error) {

console.error(error.message);

// consume response data to free up memory

return;

}

if(response.statusCode !== 200) {

console.error(response && response.statusCode);

return;

}else {

console.log(response.statusCode);

return callback(body);

}

});

}

function processData(data, callback) {

if (!data) {

console.log( "error message" + ': data not found');

} else {

//console.log(data);

var weather = JSON.parse(data);

weather = ("Country code: " + weather.sys.country + "\n"

+ "City: " + weather.name + "\n" +

"Temperature: " + (weather.main.temp - 273).toFixed(1) + " celsius\n" +

"Humidity: " + weather.main.humidity + "%\n\n");

return callback(weather);

}

}

function saveToFile(data) {

fs.open('message2.txt', 'a', undefined, function(err, fd) {

if (err) {

console.error("open error!");

throw err;

}

fs.write(fd, data, 0, data.length, null, function(err, written) {

if (err) {

console.error("write error!");

throw err;

}

fs.close(fd, function(err) {

if (err) {

console.error("close error!");

throw err;

}

});

});

});

}

getData("kiev", function(data) {

processData(data, function(data) {

saveToFile(data)

})

});

Упражнение № 3

Напишите скрипт для получения погоды в Киеве, используя модуль request-promise.

В это скрипте должны быть определены три функции:

getData - отправка запроса на OpenWeatherMap API

processData - обработка данных

saveToFile - запись данных в файл, используя fs.createWriteStream

var fs = require('fs');

var rp = require('request-promise');

function getData(city, callback) {

rp("http://api.openweathermap.org/data/2.5/weather?q="

+String(city)

+"&appid=c0c6030143a04eff2a48acd37bd81dbb")

.then(function (response) {

return callback(response);

})

.catch(function (err) {

console.error(err);

return;

});

}

function processData(data, callback) {

if (!data) {

console.log( "error message" + ': data not found');

} else {

//console.log(data);

var weather = JSON.parse(data);

weather = ("Country code: " + weather.sys.country + "\n"

+ "City: " + weather.name + "\n" +

"Temperature: " + (weather.main.temp - 273).toFixed(1) + " celsius\n" +

"Humidity: " + weather.main.humidity + "%\n\n");

return callback(weather);

}

}

function saveToFile(data) {

var wstream = fs.createWriteStream('message3.txt');

wstream.write(data);

wstream.end();

}

getData("kiev", function(data) {

processData(data, function(data) {

saveToFile(data)

})

});

Упражнение № 4

Напишите скрипт, для получения погоды во всех трех городах последовательно и паралелльно.

Рефакторизуйте скрипт из упражнения #3 в модуль api.js, в котором экспортируется (module.exports) функция getWeather

var fs = require('fs');

var rp = require('request-promise');

function getData(city, callback) {

rp("http://api.openweathermap.org/data/2.5/weather?q="

+String(city)

+"&appid=c0c6030143a04eff2a48acd37bd81dbb")

.then(function (response) {

return callback(response);

})

.catch(function (err) {

console.error(err);

return;

});

}

function processData(data, callback) {

if (!data) {

console.log( "error message" + ': data not found');

} else {

//console.log(data);

var weather = JSON.parse(data);

weather = ("Country code: " + weather.sys.country + "\n"

+ "City: " + weather.name + "\n" +

"Temperature: " + (weather.main.temp - 273).toFixed(1) + " celsius\n" +

"Humidity: " + weather.main.humidity + "%\n\n");

return callback(weather);

}

}

var options = {

flags: 'a+'

};

function saveToFile(data, callback) {

var wstream = fs.createWriteStream('apiMessage.txt', options);

wstream.write(data);

wstream.end();

if (callback) callback();

}

var getWeather = function(city, callback) {

getData(city, function(data) {

processData(data, function(data) {

saveToFile(data, callback);

});

});

};

module.exports = {

getWeather: getWeather

};

Напишите скрипт для получения погоды для всех трех городов параллельно, используя async.each

var api = require('./api');

var each = require('async-each');

cities = ['kiev', 'london', 'new york'];

each(cities, api.getWeather, function(error) {

if (error) {

console.error(error);

return;

}else {

console.log('finish');

}

});

Напишите скрипт для получения погоды для всех трех городов последовательно, используя async.eachSeries

var api = require('./api');

var eachSeries = require('async-each-series');

cities = ['kiev', 'london', 'new york'];

eachSeries(cities, function (el, next) {

api.getWeather(el, next);

}, function (err) {

console.log('finished');

});