**Node.js Workshop 3: Processing JSON and using API’s**

After completing this workshop the student is knows how to:

* Process JSON files from web
* Parse JSON and show the output on browser
* Deploy the code in Heroku

*Create a new folder called WS3 for these assignments. Place all your code there.*

**Read and process JSON files locally**

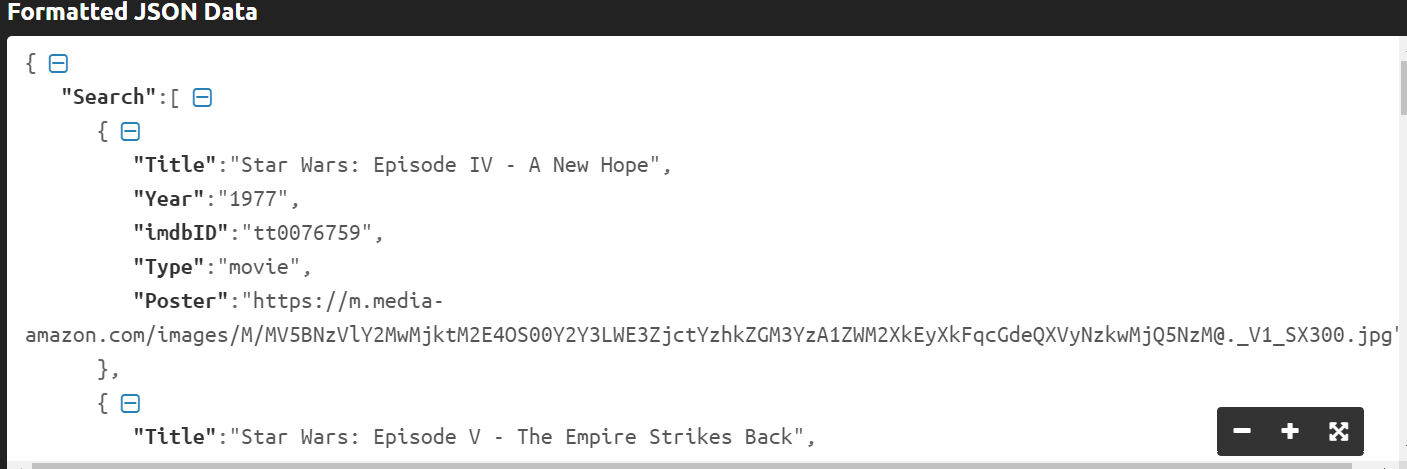
Use lecture notes as a guide. There is plenty of [tutorials available in the Internet](https://www.codementor.io/nodejs/tutorial/how-to-use-json-files-in-node-js).

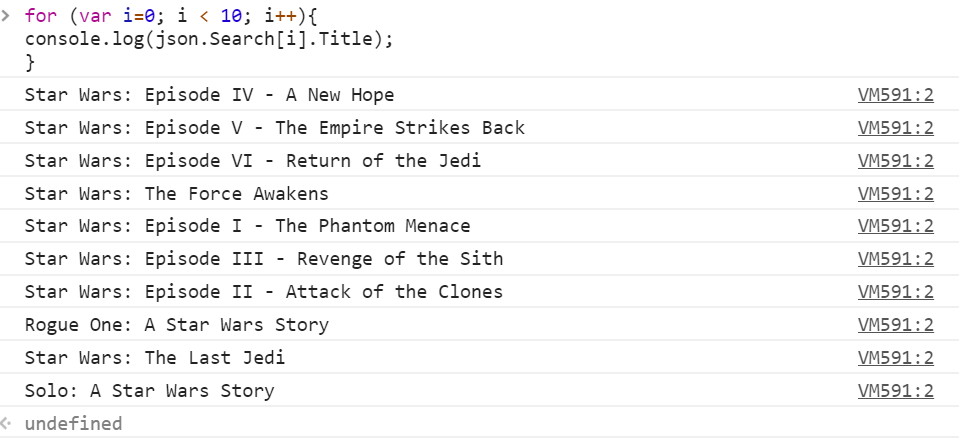
**Read and process JSON files online through API’s**

**Exercise 1: movies.js**

1. Create a new program “movies.js”, which will call JSON API from the web, parse it and output on the console and to the browser. This time we will use the Open Movie Database API, see the docs here: <http://www.omdbapi.com/>.

You can see a sample set of results here: <http://www.omdbapi.com/?s=star+wars&apikey=cbbc6750>

Often times Online JSON formatters make reading API responses much easier. Try pasting the response here: <https://jsonformatter.curiousconcept.com/>  
  


1. Pasting the JSON response in JS console as variable (“var json”) helps to understand it:  
     
   
2. Then we can loop through it:  
     
   
3. Register here in order to get your own API key: <http://www.omdbapi.com/apikey.aspx>

You can use the API by adding parameters after the URL.  
http://www.omdbapi.com/?apikey=[yourkey]&

1. Create a web server and query for your favourite movies/series and parse the output as HTML formatted table to the browser.
2. Notice that the resultset has an image link, make sure your HTML code renders it nicely. You can see examples on howto use the API here: <http://www.omdbapi.com/#usage>

**Exercise 3: Deploy your code to Heroku web service**.

Push (upload) all the exercises completed here to you GitHub-account. Then deply the site to Heroku.

Heroku is a web service for deploying web applications online. Heroku can get your code from GitHub, so you need to authenticate Heroku to read your repository.

Some considerations though.

1. You need to use variable called PORT instead of a fixed port in your code
2. You need to have a file called package.json in your app directory, with the following structure

index.js:

const PORT = process.env.PORT || 5000;

var http = require("http");

//create a server object:

http

.createServer(function(request, response) {

response.writeHead(200, { "Content-Type": "text/plain" });

response.write("Hello World!\n"); //write a response to the client

response.end("This is the end"); //end the response

})

.listen(PORT); //the server object listens on port 8080

package.json:

{

"name": "node workshop 3",

"version": "0.1.0",

"description": "A sample Node.js app in Heroku",

"engines": {

"node": "12.x"

},

"main": "index.js",

"scripts": {

"start": "node index.js",

"test": "node test.js"

}

}