**Day 1 Lab Assignments**

1. Create a text json file containing an array of products as the following (or use the given json file):

[{ "name": "p1", "color": "red", "price": 100 },

    { "name": "p2", "color": "blue", "price": 200 },

    { "name": "p3", "color": "black", "price": 300 }

]

* 1. Use file system module to read the given products file into variable: productsDB.
  2. Use JSON.parse(), to convert the data to JSON.

1. Create a http server (on port 4000) using Nodejs as a simple restfull API:
   1. Handle the following endpoints:
      * + “/home” prints “welcome to our APIs” in bold text.
        + “/products” returns the array of products that you read from the file (use JSON.strigify() to send the data).
        + “/products/1” that returns specific product with the given id (for id=1, return product in index 0 of the array products, and so on).
        + “/” default route, redirects to “/home” endpoint.
        + For any other route, return 404 status error.
   2. Try your server from browser or from Postman.
   3. Create HTML page containing a text input for product id and button that when clicked, makes ajax request to the server, and return the product with the given id.
      * + You will need to set response headers to allow CORS.

const headers = {

        'Access-Control-Allow-Origin': '\*',

        'Access-Control-Allow-Methods': 'OPTIONS, GET, POST',

*// 'Access-Control-Allow-Headers': '\*',*

*/\*\* add other headers as per requirement \*/*

    };

    response.**writeHead**(200, headers);

1. Run npm init command to initialize your (package.json) file for your app.
2. Search for any node module in NPM (Validator module for example) and install it and make a demo on it using some of its functions.
   1. What’s npm? And how to use it to install packages locally or globally?
   2. Use “--save” when you install the package, WHY?!

**Bonus:**

1. Add new API endpoints for other CRUD operations (insert, update, delete).
2. Read the following about http:
   1. <https://www.slideshare.net/skgopu/web-http-request-to-response-life-cycle>
   2. <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>
   3. <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status>
   4. <https://searchapparchitecture.techtarget.com/definition/RESTful-API>
   5. <https://rapidapi.com/blog/api-glossary/endpoint/>
   6. <https://dev.to/dangolant/things-i-brushed-up-on-this-week-the-http-request-lifecycle->
3. Make HTTP server that when receive request (/home), send him home.html file that contains any data.
4. Create TCP server using NodeJS, and make simple chat application.
5. Install express module and try it.
6. Instal MongoDB, and use NodeJS to connect to MongoDB, and display data from it.

**<Script>document.write(“Thank YOU”) </Script>**