Session 9 Homework

Code For Everyone JavaScript

|  |  |
| --- | --- |
|  | **Study** |

1. Learn more about Promises, refer to these documents:
   1. [JavaScript Info: Promise basics](https://javascript.info/promise-basics) (Read till the end of [Consumers: then, catch, finally section](https://javascript.info/promise-basics#consumers-then-catch-finally))
   2. [MDN Promise](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise)

Your tasks:

1. Create a Promise that resolve: return ‘Promise is da best’ after 5 seconds. Take this Promise to test / consume in 2 ways
2. Use await inside an async function
3. Use then() function

1. Using fetch() to load data from an API, we can use 2 async/await to get the connection from fetch() and and the data from json() function, these 2 awaits can be replaced by 2 then() calls. Figure out how to do it, here is a hint:

const fetchPromise = fetch(<URL>);

**fetchPromise.then(connection => {**

**console.log(‘Connected’);**

**console.log(connection);**

**});**

After getting the connection you get can be accessed, get data by handling the Promise returned by json() function

const fetchPromise = fetch(<URL>);

fetchPromise.then(connection => {

console.log(‘Connected’);

console.log(connection);

**const jsonPromise = connection.json();**

**jsonPromise...**

});l

1. Learn how to create a quick, free API using [Google SpreadSheet](https://www.google.com/sheets/about/) and [SheetsDB](https://sheetdb.io/), register SheetsDB account and follow [this tutorial](https://sheetdb.io/documentation). After successfully creating your API and testing it with a Rest Client such as Insomnia Rest Client, submit the API URL

|  |  |
| --- | --- |
|  | **Review** |

1. Write a function with **no parameters / arguments, return a random number from 0 to 10**, then take it to the test as follows:

// <Your function here>

const x = <Your function name>();

if (x < 0) {

console.log(‘Failed: the number is smaller than 0’);

} else if (x > 10) {

console.log(‘Failed: the number is bigger than 10’);

} else {

console.log(‘Passed, bravo’);

}

1. Write a function with 2 parameters / arguments: a and b, return a random number from a to b, take it to the test as follows

// <Your function here>

const x = <Your function name>(4, 16);

if (x < 4) {

console.log(‘Failed: the number is smaller than 4’);

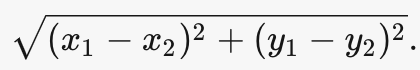
} else if (x > 16) {

console.log(‘Failed: the number is bigger than 16’);

} else {

console.log(‘Passed, bravo’);

}

1. Write a function to [calculate the distance between 2 points](https://www.wikihow.com/Find-the-Distance-Between-Two-Points) (x1, y1) and (x2, y2), the formula is 

// <Your function here>

const d = <Your function name>(3, 10, 0, 6);

if (x !== 5) {

console.log(‘Failed: the calculation is wrong’);

} else {

console.log(‘Passed, bravo’);

}

|  |  |
| --- | --- |
|  | **Serious exercices** |

1. Here is a free, public API, which, each day provide a different quote: [Quote of the Day API](http://quotes.rest/qod.json)

Consume it as [this video](https://youtu.be/RB6wjqdodq4).

You are recommended, but not required, to implement it by these steps:

* 1. Create UI structure in HTML file
  2. In JS file, fetch the API
  3. Destructure the returned data to get the quote content and quote author
  4. Show the quote content and author to users using DOM

1. Using [SheetsDb](https://sheetdb.io/), create a quick API for a Wish List. [Link video demo](https://youtu.be/gYSunpr8IhI). Write an app to
   1. Implement Add (POST) new wish list item. [Link demo](https://youtu.be/6Zld8q4mG4g).
   2. Implement Read (GET) all of wish list when the app starts. [Link Demo](https://youtu.be/Ftxx6ne4nrQ).
   3. Implement Remove (DELETE) a wish list item. [Link demo](https://youtu.be/0s1qr3WKtxs).
   4. (Optional) Implement Search (GET - SEARCH) wish list when users type in New item box. [Link demo](https://youtu.be/Z12XVjEYEb4).
2. (Optional) Here is a public API from Apple’s Itunes for affiliate developers to get the Itunes chart data: [Itunes RSS API](https://itunes.apple.com/us/rss/topsongs/all/limit=15/json). Implement an app to view the chart and play the audio preview. [Link demo](https://bit.ly/c4t-itunes).

Project Demo

1. **Educational - Quiz** <https://martindevox2811.github.io/C4EJS_Project/>
2. **Shop Mobile** [**https://luuthemanhgch17279.github.io/MobileShop.github.io/index.html**](https://luuthemanhgch17279.github.io/MobileShop.github.io/index.html)
3. **Music Search** <https://vck2004.github.io/blog-grid/>
4. **Gaminglaptop** <https://htuanmb.github.io/Project-c4e/index.html#>
5. **iTech** <https://tiangocode.github.io/KTL_iTech/>
6. Mobile Shop: <https://cachiengion314.github.io/mobileshop/index.html>
7. Kick-Lap: <https://hoanganhmdbg.github.io/kicklabpt/>
8. Tư vấn sức khỏe: <https://haibinh127.github.io/OfficialProject/>