**Frontend Assignment**

**JavaScript**

**MODULE: 2 (Data Types and Objects)**

**Write the code, one line for each action:**

1. **Create an empty object user.**

**ANS.**  let user = {}

1. **Add the property name with the value John.**

**ANS.**  let user = {name: “John”}

1. **Add the property surname with the value Smith.**

**ANS.**  let user = {name: “John”, surname:”Smith”}

1. **Change the value of the name to Pete.**

**ANS.**  let user = {name: “John”, surname: “Smith”}

user.name = “Pete”;

1. **Remove the property name from the object.**

**ANS.**  let user = {name: “John”, surname: “Smith”}

deleteuser.name = “Pete”;

**Is array copied?**

let fruits = ["Apples", "Pear", "Orange"]; // push a new value into the "copy" let shoppingCart = fruits;

shoppingCart.push("Banana"); // what's in fruits?

alert(fruits.length ); // length is 4.

**Map to names.**

let john = {name: "John", age: 25};

let pete = {name: "Pete", age: 30};

let mary = {name: "Mary", age: 28};

let users = [john, pete, mary ];

let names = /\* users.map(item=>item.name);\*/

alert(names); // John, Pete, Mary

**Map to objects.**

let john = { name: "John", surname: "Smith", id: 1 }; let pete = { name: "Pete", surname: "Hunt", id: 2 }; let mary = { name: "Mary", surname: "Key", id: 3 }; let users = [ john, pete, mary ];

let usersMapped = /\* ... your code ...

users.map(user =>(fullname:`{$user.name} {$user.surname}`, id: user.id));

\*/

usersMapped = [ {fullName: "John Smith", id: 1}, { fullName: "Pete Hunt", id: 2 }, { fullName: "Mary Key", id: 3 } ]

usersMapped = [ { fullName: "John Smith", id: 1 }, { fullName: "Pete Hunt", id: 2 }, { fullName: "Mary Key", id: 3 } ]

**Sum the properties There is a salaries object with arbitrary number of salaries. Write the function sumSalaries(salaries) that returns the sum of all salaries using Object.values and the for..of loop. If salaries is empty, then the result must be 0.**

let salaries = {"John": 100, "Pete": 300, "Mary": 250 };

function sumSalaries(salaries)

{

let sum = 0;

for(let salary of Object.values(salaries))

{

sum +=salary;

}

}

alert(sumSalaries(salaries) ); // 650

**Destructuring assignment We have an object: Write the Destructuring assignment that reads:**

a) Name property into the variable name.

b) Year’s property into the variable age.

c) isAdmin property into the variable isAdmin (false, if no such property)

d) let user = {name: "John", years: 30};

**ANS.**

let user = {

name: "John",

years: 30

};

let {name, years: age, isAdmin = false} = user;

alert( name ); // John

alert( age ); // 30

alert( isAdmin ); // false

**Turn the object into JSON and back Turn the user into JSON and then read it back into another variable.**

user = {name: "John Smith", age: 35}

**ANS.**

let user2 = JSON.parse(JSON.stringify(user));

alert(user2);

**MODULE: 3 (Document, Event and Controls)**

**Create a program to hide/show the password**

**ANS.**

Password:<input type="password" value="" id="myInput">

  <input type="checkbox" onclick="showpsd()">Show password

<script>

    function showpsd(){

      let psd = document.getElementById('myInput');

      if(psd.type === 'password')

      {

        psd.type = 'text';

      }

      else{

        psd.type = 'password';

      }

    }

  </script>

**Create a program that will select all the classes and loop over and whenever i click the button the alert should show**

**ANS.**

**Create a responsive header using proper JavaScript**

**ANS.** Click below the link to show an example of header:

<https://github.com/Chigs18/Assignment/blob/main/header.html>

**Create a form and validate using JavaScript**

**ANS.** Click below the link to show an example of Form with Validate:

<https://github.com/Chigs18/Assignment/blob/main/validate_form_with_js.html>

**Create a modal box using css and Js with three buttons**

**ANS.** Click below the link to show an example of Model box:

<https://github.com/Chigs18/Assignment/blob/main/model.html>

**Use external js library to show slider**

**ANS.** Click below the link to show an example of Slider:

<https://github.com/Chigs18/Assignment/tree/main/slider>

**Prevent the browser when i click the form submit button**

**ANS.** In JavaScript, we are provided with an in−built preventDefault() method that is used to stop the submission of the form and also to prevent the browser from refreshing itself while performing some task.

There are two ways in which we can use the preventDefault() method to stop the form submission –

<form onsubmit="event.preventDefault();">

//content of the form tag

</form>

**MODULE: 4 (New Request)**

**What is JSON ?**

**ANS.** The JavaScript JSON is an acronym of JavaScript Object Notation. It provides a format for storing and transporting data. It is a lightweight human readable collection of data that can be accessed in a logical manner.

* It generates and stores the data from user input.
* It can transport the data from the server to client, client to server, and server to server.
* It can also build and verifying the data.

**What is promises ?**

**ANS.** A Promise is a proxy for a value not necessarily known when the promise is created. It allows you to associate handlers with an asynchronous action's eventual success value or failure reason. This lets asynchronous methods return values like synchronous methods: instead of immediately returning the final value, the asynchronous method returns a promise to supply the value at some point in the future.

A Promise is in one of these states:

* pending: initial state, neither fulfilled nor rejected.
* fulfilled: meaning that the operation was completed successfully.
* rejected: meaning that the operation failed.

**Write a program of promises and handle that promises also**

**ANS.** Program link is below:

<https://github.com/Chigs18/Assignment/blob/main/promises.html>

**Use fetch method for calling an api** [**https://fakestoreapi.com/products**](https://fakestoreapi.com/products)

**ANS.** Program link is below:

<https://github.com/Chigs18/Assignment/blob/main/fetchApi.html>

**Display all the product from the api in your HTML page**