**MODULE: 1 (Introduction and Code Quality)**

1. Write a program to Show an alert.

<script>

// Show an alert

alert("Hello, How Are You !");

</script>

1. What will be the result for these expressions?

<script>

console.log(5 > 4); // true

console.log("apple" > "pineapple"); // false

console.log("2" > "12"); // true

console.log(undefined == null); // true

console.log(undefined === null); // false

console.log(null == "\n0\n"); // false

console.log(null === +"\n0\n"); // false

</script>

1. Will alert be shown? if ("0") { alert( 'Hello'); }

* Yes

1. What is the code below going to output? alert( null || 2 || undefined );

* 2

1. The following function returns true if the parameter age is greater than 18. Otherwise it asks for a confirmation and returns its result:

script>

function checkAge(age) {

if (age > 18) {

return true;

} else {

return confirm("Did parents allow you?");

}

}

let age = prompt("How old are you?", 18);

if (checkAge(age)) {

alert("Access granted");

} else {

alert("Access denied");

}

</script>

1. Replace Function Expressions with arrow functions in the code below:

Function :

ask(question, yes, no)

{ if (confirm(question))yes();

Else

no();

}

ask("Do you agree?", function()

{ alert("You agreed."); },

{ alert("You agreed."); },

function() {

alert("You canceled the execution."); }

}

<script>

      function ask(question, yes, no) {

        if (confirm(question)) yes();

        else no();

      }

      ask(

        "Do you agree?",

        () => alert("You agreed."),

        () => alert("You canceled the execution.")

      );

    </script>

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

1. Write the code, one line for each action: a) Create an empty object user. b) Add the property name with the value John. c) Add the property surname with the value Smith. d) Change the value of the name to Pete. e) Remove the property name from the object.

<script>

      let user = {};

      user.name = "John";

      user.surname = "Smith";

      user.name = "Pete";

      delete user.name;

</script>

1. 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 ); // ?

<script>

      let fruits = ["Apples", "Pear", "Orange"];

      let shoppingCart = fruits;

      shoppingCart.push("Banana"); // Push a new value into the array via shoppingCart

      alert(fruits.length); // Display the length of the array via fruits

    </script>

O/P : 4

1. 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 = /\* ... your code \*/ alert( names ); // John, Pete, Mary

<script>

      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((user) => user.name);

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

    </script>

1. 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 ... \*/ /\* usersMapped = [ { fullName: "John Smith", id: 1 }, { fullName: "Pete Hunt", id: 2 }, { fullName: "Mary Key", id: 3 } ] \*/ alert( usersMapped[0].id ) // 1 alert( usersMapped[0].fullName ) // John Smithv

<script>

      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 = users.map((user) => ({

        fullName: `${user.name} ${user.surname}`,

        id: user.id,

      }));

      alert(usersMapped[0].id); // 1

      alert(usersMapped[0].fullName); // John Smith

    </script>

1. 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 }; alert( sumSalaries(salaries) ); // 650

<script>

        function sumSalaries(salaries) {

            let sum = 0;

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

                sum += salary;

            }

            return sum;

        }

        let salaries = {

            "John": 100,

            "Pete": 300,

            "Mary": 250

        };

        console.log(sumSalaries(salaries)); // 650

    </script>

1. 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};

<script>

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

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

      console.log(name); // "John"

      console.log(age); // 30

      console.log(isAdmin); // false

</script>

1. 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};

<script>

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

      // Convert the object to JSON

      let userJSON = JSON.stringify(user);

      console.log(userJSON); // '{"name":"John Smith","age":35}'

      // Parse the JSON back into an object

      let newUser = JSON.parse(userJSON);

      console.log(newUser); // { name: "John Smith", age: 35 }

</script>