HIMMAT AHIR

Advance JavaScript

* Write the code, one line for each action:

**a) Create an empty object user.**

**Ans :- var details = {**

**}**

**b) Add the property name with the value John.**

**Ans :- var details = {**

**}**

**details.name = "john"**

**console.log(details)**

**c) Add the property surname with the value Smith.**

**Ans :- var details = {**

**}**

**details.surname = "Smith"**

**console.log(details)/**

**d) Change the value of the name to Pete.**

**Ans :- var details = {**

**}**

**details.surname = "Smith"**

**details.surname = "Pete"**

**console.log(details)**

**e) Remove the property name from the object.**

**Ans :- var details = {**

**}**

**details.surname = "Smith"**

**details.surname = "Pete"**

**delete details.name**

**console.log(details)**

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 ); // ?

<!DOCTYPE html>

<script>

    "use strict";

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

    let shoppingCart = fruits;

    shoppingCart.push("Banana");

    alert(fruits.length); // 4

</script>

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

<!DOCTYPE html>

<script>

"use strict";

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

</script>

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 Smith**

**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**

**}));**

**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 Smith**

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

Function: sumsalaries(salaries);

    {

        let sum = 0;

        For

            (let,salary, object .values(salaries))

        {

            sum += salary;

        }

        return sum; // 650

    };

    let salaries =

    {

        "John": 100,

        "Pete": 300,

        "Mary": 250

    };

    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};

**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};

**let user = {**

**name: "John Smith",**

**age: 35**

**};**

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