***Practica Calificada 1***

Pregunta 1

Código:

//Practica pregunta 1

//LET

const nombre\_persona="Javier";

console.log(nombre\_persona);

//CONST

let nombre\_persona2="Raul";

console.log(nombre\_persona2);

//Spread Operator(Clonar)

let empleado\_ejemplo=[

    {id:100,nombre:"Guille",apellidos:"Garcia Porras",categoria:"Jefe de Proyectos",tiempoServicio:1000,estado:0,sueldo:4000},

    {id:102,nombre:"Sebastian",apellidos:"Ramirez Lenci",categoria:"Medico",tiempoServicio:1000,estado:0,sueldo:2500},

    {id:101,nombre:"Ursula",apellidos:"Dominguez Arroyo",categoria:"Analista",tiempoServicio:1000,estado:0,sueldo:2500},

];

let empleado\_practica={...empleado\_ejemplo};

console.table(empleado\_practica);

//Destructuring

let {apellidos,categoria}=empleado\_practica[1];

console.log(apellidos,categoria);

//Template Literals

console.log(`El trabajador ${apellidos} ocupa el cargo de ${categoria}`);

Captura de pantalla con la imagen de una pantalla

Descripción generada automáticamente con confianza media

Pregunta 2

//Practica pregunta 2

//Funcion que permita evaluar si un numero es impar o par

let par\_impar=(nume)=>{

    if (nume%2==0){

        console.log("par");

    }

    else console.log("impar");

}

par\_impar(15);

par\_impar(8);

//Funcion que calcule el promedio

let suma\_a=(arreglo)=>{

    let suma\_v=0

    for(let i=0;i<arreglo.length;i++) suma\_v+=arreglo[i];

    return suma\_v/arreglo.length;

};

console.log(suma\_a([10,16,19,39,85,17]));

//Funcion que evalue multiplicidad de 5

let mul5=(nume)=>{

    if (nume%5==0){

        console.log("Es multiplo de 5");

    }

    else console.log("No es multiplo de 5");

}

mul5(15);

mul5(865743);

//Funcion que devuelva un arreglo de numeros pares

let pares=(arreglo)=>{

    let arreglo\_par=arreglo.map((n)=>{

         if (n%2==0) return n;

         else return "a";

    });

    return arreglo\_par.filter((n)=>n!="a");

};

console.log(pares([20,50,26,3201]));

// Generar mensaje

let Generate\_message=(destinatario,emisor,mensaje)=>{

    return `${emisor} quiere enviarle un mensaje a ${destinatario} el cual contiene lo siguiente: ${mensaje}`;

};

console.log(Generate\_message("Carlos","Javier","Tengo algo urgente que decirte"));

Pantalla de computadora con fondo negro

Descripción generada automáticamente con confianza media

Pregunta 3

//Practica pregunta 3

const arreglo\_original=[5,86,19,103,215];

let arreglo\_copia={...arreglo\_original};

let arreglo\_agregado4=[...arreglo\_original,16,59,49,78];

console.log("arreglo original",arreglo\_original);

console.log("arreglo copia",arreglo\_copia);

console.log("arreglo agregado",arreglo\_agregado4);

Un reloj digital

Descripción generada automáticamente con confianza media

Pregunta 4

//Practica pregunta 4

let calculadora=(num1,num2,operador)=>{

    if(operador==="+") return num1+num2;

    else if(operador==="-") return num1-num2;

    else if (operador==="x") return num1\*num2;

    else if(operador==="/") return num1/num2;

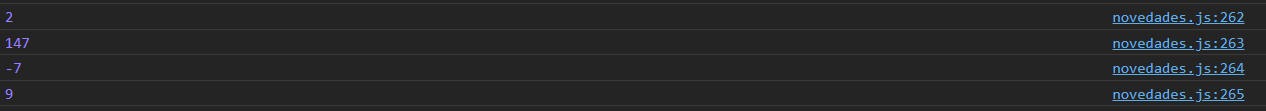
}

console.log(calculadora(46,23,"/"));

console.log(calculadora(49,3,"x"));

console.log(calculadora(16,23,"-"));

console.log(calculadora(6,3,"+"));



Pregunta 5

// Practica pregunta 5

import { platos } from './platos.js';

console.table(platos.filter((n)=>(n.region==="Costa")));

console.table(platos.filter((n)=>n.likes===Math.max(...platos.map((n)=>n.likes))));

console.table(platos.filter((n)=>n.precio===Math.max(...platos.map((n)=>n.precio))));

console.table(platos.filter((n)=>n.precio===Math.min(...platos.map((n)=>n.precio))));

export {platos};

const platos=[

     {nombre:"Ceviche",region:"Costa",likes:1000,precio:20},

     {nombre:"Malaya",region:"Costa",likes:300,precio:10},

     {nombre:"Juane",region:"Selva",likes:1000,precio:15},

     {nombre:"Cuy",region:"Sierra",likes:650,precio:15}

];

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica

Descripción generada automáticamente