



Certified Tech Developer

The Ultimate Degree

Programación Imperativa

*/*Loop de pares*

Deberás crear una función llamada loopDePares que reciba como parámetro un número y haga un loop de 0 a 100 mostrando en la consola cada número del loop. En caso de que el número de la iteración sumado con el número pasado por parámetro sea par, mostrará en la consola: "El número X es par"./*

```
function loopDePares(num){  
  for(let i = 0; i <= 100; i++){  
  
    if( (num + i) % 2 == 0){  
      console.log(`El número ${num+i} es par`);  
    } else {  
      console.log(i);  
    }  
  }  
}
```

//LoopDePares(6);



*/*Loop de impares con palabra*

*Deberás crear una función llamada LoopDeImpares que reciba como parámetro un número y una palabra, y haga un loop de 0 a 100 mostrando en la consola cada número del loop. Luego, modificar el código para que, en caso de que ese número sumado con el número pasado por parámetro sea impar, muestre en la consola la palabra pasada por parámetro. */*

```
function loopDeImpares(num, palabra){  
  for(let i = 0; i <= 100; i++){  
  
    if( (num+i) % 2 !== 0){  
      console.log(palabra);  
    } else {  
      console.log(i);  
    }  
  }  
}
```

```
//LoopDeImpares(4, 'hola');
```

// Deberás crear una función llamada sumatoria que reciba un número como parámetro y que devuelva la sumatoria de todos sus números anteriores, incluso ese mismo. Ejemplo:

// sumatoria(3) debe retornar 6 porque hace (1+2+3)



```
// sumatoria(8) debe retornar 36
```

```
function sumatoria(num){  
  
    let contador = 0;  
  
    for(let i = 1; i <= num; i++){  
  
        contador = contador + i;  
  
        //contador+= i;  
  
    }  
  
    return contador;  
}
```

```
//console.log(sumatoria(8));
```

//Deberás crear una función llamada nuevoArreglo que reciba un número como parámetro y que devuelva un nuevo arreglo con tantos elementos como el número que le hayas pasado. Ejemplo:

```
//nuevoArreglo(5) debe retornar [1,2,3,4,5]
```

```
function nuevoArreglo(num){  
  
    let arr = [];  
  
    for(let i = 1; i <= num; i++){
```



```
        arr.push(i);

    }

    return arr;
}

console.log(nuevoArreglo(15));

//Deberás crear una función llamada split que reciba un string y simule el
comportamiento de la función original. Si no estás seguro de cómo funciona,
Google puede ayudarte. Importante: no podés usar el String.split(). Ejemplo:

// split("hola") debe retornar ["h","o","l","a"]
// split("chau") debe retornar ["c","h","a","u"]

function split(string){

    let arr= [];

    for(let i = 0; i < string.length; i++){

        arr.push(string[i]);

    }

    return arr;
}
```



```
//console.log(split('hola'));

// Deberás crear una función llamada caracterDelMedio que reciba un string
por parámetro y devuelva sus caracteres del medio. Ejemplo:

// caracterDelMedio("Digital House") debe retornar "l"
// caracterDelMedio("hola") debe retornar "ol"
// caracterDelMedio("cosas") debe retornar "s"

let caracterDelMedio = (palabra) => {

    // let palabra2 = palabra.replace(/ /g, "")

    // console.log(palabra)

    let posicionDelMedio = Math.ceil(palabra.length / 2) - 1;

    if (palabra.length % 2 === 0) {

        return palabra[posicionDelMedio] + palabra[posicionDelMedio + 1];

    } else {

        return palabra[posicionDelMedio];

    }

};

//console.log(caracterDelMedio("digita"));
```



//Deberás crear una función llamada arrayHandler que reciba dos arreglos de igual largo como parámetros y muestre en la consola "Soy {elemento de array 1} y yo soy {elemento de array 2}". Ejemplo:

// arrayHandler([1,2,3,4], ["h","o","l","a"]) debe mostrar:

// Soy 1 y yo soy h

// Soy 2 y yo soy o

// Soy 3 y yo soy l

// Soy 4 y yo soy a

```
let arrayHandler = (array1, array2) => {  
  if(array1.length === array2.length){  
    for (let i = 0; i < array1.length; i++) {  
      console.log(`Soy ${array1[i]}, y yo soy ${array2[i]}`);  
    }  
  }else{  
    console.log("los arrays no tienen la misma longitud")  
  }  
};
```

//arrayHandler([1, 2, 3, 4], ["h", "o", "l", "a"]);

// Deberás crear una función que convierta de snake_case o kebab-case a camelCase. La primera letra del resultado debe estar en mayúsculas solo si en el string original estaba en mayúscula. Ejemplo:

// "el-guerrero-silencioso" debería convertirse en "elGuerreroSilencioso"



```
// "El_guerrero_silencioso" debería convertirse en "ElGuerreroSilencioso"

function conversor(str) {

  for (let i = 0; i < str.length; i++) {

    if (str[i] === "_" || str[i] === "-") {

      str = str.replace(str[i + 1], str[i + 1].toUpperCase());

      str = str.replace(str[i], "");

    }

  }

  return str;

}

// console.log(conversor("Hola_mina_gonzalez"));
// console.log(conversor("hola-mina-gonzalez"));


// Deberás crear una función llamada palindromo que indique si una palabra es
palíndroma o no. Debe retornar "true" en caso de que lo sea, y "false" en
caso de que no. Ejemplo:

// palindromo("anilina") debe retornar true
// palindromo("Ana") debe retornar true
// palindromo("Enrique") debe retornar false


let palindromo = (string)=>{

  let inverso = ""
```



```
    for (let i = string.length - 1; i >= 0; i--) {  
        inverso += string[i]  
    }  
  
    return string === inverso ? true : false  
}  
  
//console.log(palindromo("menem"))
```