


While - Ejercicio

Hacer un programa que imprima los números del `1 al 51` y debe hacer lo siguiente. Halla `múltiplos de 3 y múltiplos de 5`, y mostrar somos múltiplos de 3 y 5.

js

 Copiar código

```
let a = 1;


while(a <= 51){
  if(a % 3 == 0 && a % 5 == 0){
    console.log("Estos son los múltiplos de 3 y 5 " + a);
  }else if(a % 3 == 0){
    console.log("Estos son los múltiplos de 3: " + a);
  }else if(a % 5 == 0){
    console.log("Estos son los múltiplos de 5: " + a);
  }
  a++;
}
```

Function

Son **bloques de código** diseñados para realizar tareas específicas. Se pueden declarar con la palabra clave `function`, aceptar parámetros para recibir datos y devolver resultados usando `return`.


Ejemplos y formas crear funciones

js - Función forma 1

 Copiar código

```
// Función forma 1
function suma(a, b){
  let sumar = a + b;
  return sumar;
}
console.log(suma(5,6));
```


js - Función forma 2

 Copiar código

```
// Función forma 2
```

```
function sumar(){
  let a = 89;
  let b = 25;
  return a + b;
}
console.log(sumar());
```

js - Función forma 3


 Copiar código

```
// Función forma 3
function sumar1(){
  console.log("función sumar 1");
  console.log(123 - 97);
  console.log(123 * 97);
}
sumar1();
```

Función Anónima

Es una función que no tiene nombre y generalmente se define en el momento en que se necesita, sin asignarle un nombre específico. Se utilizan comúnmente como argumentos de otras funciones o para crear funciones autoejecutables.

js - Función Anónima


 Copiar código

```
// Función anónima:
const sumas = function(num1, num2){
  return num1 + num2;
}
console.log(sumas(6,9));
```

Función flecha


Es una forma corta y rápida de escribir funciones. No tiene su propio `this`, así que usa el `this` del lugar donde se encuentra. Son útiles para funciones pequeñas y simples, y para trabajar con el contexto que las rodea.

js - Función Flecha

 Copiar código


```
// Función flecha
let resultados = (a, b) => a + b;
console.log(resultados(12,12));
```

js - Función Flecha forma 2

 Copiar código


```
// Función flecha forma 2
const valores = () =>{
  let a = 15;
  let b = 17;
  return a * b;
}
console.log(valores());
```

js - Función Flecha forma 3

 Copiar código

```
// Función flecha forma 3
let mensaje = () => `Hola mundo`;
console.log(mensaje());
```

js - Función Flecha forma 4


 Copiar código

```
// Función Flecha forma 4
let sumados = function (n = 8, a ){
  return n * a;
}
console.log(sumados(undefined, 2));
```

Ejercicios con Arrays

Ejercicio: Has una función que **calcule la suma** de los número de un arreglo.

js


 Copiar código

```
function sumaArr(){
  let arreglo = [2, 5, 6, 8, 9, 12, 24, 56, 78];
  let sum = 0;
  for (let i = 0; i < arreglo.length; i++) {
    sum += arreglo[i];
  }
  console.log("La suma de mi arreglo es: " + sum);
}
sumaArr();
```

For

Ejercicio: Has una función que **recorra un array** usando **For**

js


 Copiar código

```
function recorner() {  
  let arreglos = ['Hola', 'Web', 'Php', 'Visual', '.net'];  
  for (let i = 0; i < arreglos.length; i++) {  
    console.log(arreglos[i]);  
  }  
}  
recorer();
```

Foreach

Ejercicio: Has una función que `recorra un array` usando `forEach`


js

 Copiar código

```
const array = [10,12,13,14, 'web', 'sql', '.net'];  
array.forEach(function(elementos){  
  console.log(elementos)  
});
```

Ejercicio: Has una función que `recorra un array` usando `forEach` y una `función flecha`

js

 Copiar código

```
const frutas = ['melon','piña', 'papaya','fresa'];  
frutas.forEach(element =>  
  console.log(element)  
);
```

Filter

Se utiliza para filtrar elementos de un arreglo según ciertos criterios y devolver un nuevo arreglo con los elementos que cumplen con dichos criterios

js

✓ Copiado!

```
const frutas = ['melon','piña', 'papaya','fresa', 'mandarina', 'naranja', 'uva' ];  
const result = frutas.filter(value => value.length < 7);  
console.log(result);
```

js - Filter con una función

✓ Copiado!

```
const numeros = [1,2,3,4,5,6,7,8,9,0,11,12,23,24,25,27,67,56,98,76];
const resp = numeros.filter(calcularN);

function calcularN (result){
  return result % 2 == 0;
}
console.log(resp);
```

Estudiar **Filter**, **Forich**