



Tecnólogo en Análisis y Desarrollo de
Software
Ficha

Funciones/Arreglos JS

SENA

MANUAL FUNCIONES/ARREGLOS JAVASCRIPT

BRAYAN ESTIVEN CARVAJAL PADILLA

Análisis y Desarrollo de Software

Ficha: 2899747



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

1.

Nombre de la función: numeros()		Versión: 1.0
Descripción: Función que muestra los números de 1 al 10, los pares y los impares en arreglos		
	Tipo de variable: Numérico	
Código:		

```
function numeros() {  
  let numeros = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]  
  let pares = []  
  let impares = []  
  numeros.forEach(function(numero) {  
    if (numero % 2 === 0) {  
      pares.push(numero)  
    } else {  
      impares.push(numero)  
    }  
  })  
  return {  
    numeros: numeros,  
    pares: pares,  
    impares: impares  
  }  
}
```

Arreglo: ▾ Array(10) 1

0: 1
1: 2
2: 3
3: 4
4: 5
5: 6
6: 7
7: 8
8: 9
9: 10
length: 10
▸ [[Prototype]]: Array(0)

Pares: ▾ Array(5) 1

0: 2
1: 4
2: 6
3: 8
4: 10
length: 5
▸ [[Prototype]]: Array(0)

Impares: ▾ Array(5) 1

0: 1
1: 3
2: 5
3: 7
4: 9
length: 5
▸ [[Prototype]]: Array(0)



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

2.

Nombre de la función: tablas1()		Versión: 1.0
Descripción: Función que muestra la tabla del 5 y suma los pares e impares		
	Tipo de variable: Numérico	
Código:		

```
function tablas1(){
  let cont5=1
  let par = 0
  let impar = 0
  let tabla5=[]

  for (let i = 0; i < 5; i++){
    tabla5[i] = []
    for (let j = 0; j < 5; j++) {
      tabla5[i][j]=5*cont5
      if (tabla5[i][j] % 2 === 0) {
        par += tabla5[i][j]
      } else {
        impar += tabla5[i][j]
      }
      cont5++
    }
  }
  console.log(tabla5);
  console.log("Suma de los números pares:", par)
  console.log("Suma de los números impares:", impar)
  return ""
}
```

Tabla del 5

▼ Array(5) **1**

- ▶ 0: (5) [5, 10, 15, 20, 25]
- ▶ 1: (5) [30, 35, 40, 45, 50]
- ▶ 2: (5) [55, 60, 65, 70, 75]
- ▶ 3: (5) [80, 85, 90, 95, 100]
- ▶ 4: (5) [105, 110, 115, 120, 125]

length: 5

▶ [[Prototype]]: Array(0)

Suma de los números pares: 780

Suma de los números impares: 845

Tabla del 5

```
▼ Array(5) 1
  ▶ 0: (5) [5, 10, 15, 20, 25]
  ▶ 1: (5) [30, 35, 40, 45, 50]
  ▶ 2: (5) [55, 60, 65, 70, 75]
  ▶ 3: (5) [80, 85, 90, 95, 100]
  ▶ 4: (5) [105, 110, 115, 120, 125]
    length: 5
  ▶ [[Prototype]]: Array(0)
```

Suma de los números pares: 780

Suma de los números impares: 845



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

2.1

Nombre de la función: tablas2()		Versión: 1.0
Descripción: Función que muestra la tabla del 9 y suma los pares e impares		
	Tipo de variable: Numérico	
Código:		
<pre>function tablas2(){ let cont5=1 let par = 0 let impar = 0 let tabla9=[] for (let i = 0; i < 5; i++){ tabla9[i] = [] for (let j = 0; j < 5; j++){ tabla9[i][j]=9*cont5 if (tabla9[i][j] % 2 === 0) { par += tabla9[i][j] } else { impar += tabla9[i][j] } cont5++ } } console.log(tabla9); console.log("Suma de los números pares:", par) console.log("Suma de los números impares:", impar) return "" }</pre>		<div>Tabla del 9</div> <div>▼ Array(5) 1</div> <div>▶ 0: (5) [9, 18, 27, 36, 45]</div> <div>▶ 1: (5) [54, 63, 72, 81, 90]</div> <div>▶ 2: (5) [99, 108, 117, 126, 135]</div> <div>▶ 3: (5) [144, 153, 162, 171, 180]</div> <div>▶ 4: (5) [189, 198, 207, 216, 225]</div> <div>length: 5</div> <div>▶ [[Prototype]]: Array(0)</div> <div>Suma de los números pares: 1404</div> <div>Suma de los números impares: 1521</div>

Tabla del 9

```
▼ Array(5) 1
  0: (5) [9, 18, 27, 36, 45]
  1: (5) [54, 63, 72, 81, 90]
  2: (5) [99, 108, 117, 126, 135]
  3: (5) [144, 153, 162, 171, 180]
  4: (5) [189, 198, 207, 216, 225]
  length: 5
  [[Prototype]]: Array(0)
```

Suma de los números pares: 1404

Suma de los números impares: 1521



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

3.

Nombre de la función: matrizX()		Versión: 1.0
Descripción:		
Función que muestra los números de la tabla del 5 que forman una X en el arreglo		
	Tipo de variable: Numérico	
Código:		
<pre>function matrizX() { let cont5 = 1; let tabla5 = [] for (let i = 0; i < 5; i++) { tabla5[i] = [] for (let j = 0; j < 5; j++) { tabla5[i][j] = 5 * cont5 cont5++ } } console.log(tabla5); for (let i = 0; i < 5; i++) { for (let j = 0; j < 5; j++) { if (i === j i + j === 4) { console.log(tabla5[i][j]) } } } }</pre>		



Tecnólogo en Análisis y Desarrollo de
Software
Ficha

Funciones/Arreglos JS

4.

Nombre del array: interno[]		Versión: 1.0
Descripción: Arreglo que contiene un bingo con números de la tabla del 5 y muestra las filas referente a las silabas B-I-N-G-O		
I, j, aumento	Tipo de variable: Numérico	



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

Código:

```
let interno = [];  
let i;  
let j;  
let aumento = 1;  
  
for (i = 0; i < 5; i++) {  
  interno[i] = [];  
  for (j = 0; j < 5; j++) {  
    interno[i][j] = 2 * aumento;  
    aumento++;  
  }  
}  
console.log(interno);  
  
//Letra B  
console.log("LETRA B \n");  
for (j = 0; j < 5; j++) {  
  console.log(interno[j][0] + " ");  
}  
  
//Letra I  
console.log("LETRA I \n");  
for (j = 0; j < 5; j++) {  
  console.log(interno[j][1] + " ");  
}  
  
//Letra N  
console.log("LETRA N \n");  
for (j = 0; j < 5; j++) {  
  console.log(interno[j][2] + " ");  
}  
  
//Letra G  
console.log("LETRA G \n");  
for (j = 0; j < 5; j++) {  
  console.log(interno[j][3] + " ");  
}  
  
//Letra O  
console.log("LETRA O \n");  
for (j = 0; j < 5; j++) {  
  console.log(interno[j][4] + " ");  
}
```

```
▼ (5) [Array(5), Array(5), Array(5), Array(5), Array(5)]  
  ▶ 0: (5) [2, 4, 6, 8, 10]  
  ▶ 1: (5) [12, 14, 16, 18, 20]  
  ▶ 2: (5) [22, 24, 26, 28, 30]  
  ▶ 3: (5) [32, 34, 36, 38, 40]  
  ▶ 4: (5) [42, 44, 46, 48, 50]  
    length: 5  
    ▶ [[Prototype]]: Array(0)  
  
LETRA B  
2  
12  
22  
32  
42  
  
LETRA I  
4  
14  
24  
34  
44  
  
LETRA N  
6  
16  
26  
36  
46  
  
LETRA G  
8  
18  
28  
38  
48  
  
LETRA O  
10  
20  
30  
40  
50
```

5.

Nombre del array: interno[]		Versión: 1.0
Descripción: Arreglo que muestra los números de la tabla del 5 que forman tres X en diferentes posiciones		
l, j, aumento	Tipo de variable: Numérico	



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

Código:

```
let interno = [];  
let i;  
let j;  
let aumento = 1;  
  
for (i = 0; i < 5; i++) {  
  interno[i] = [];  
  for (j = 0; j < 5; j++) {  
    interno[i][j] = 2 * aumento;  
    aumento++;  
  }  
}  
console.log(interno);  
  
//Muestra los numeros ubicados en X  
for (i = 0; i < 3; i++) {  
  for (j = 0; j < 3; j++) {  
    if (i === j || i + j === 2) {  
      console.log("x1:" + interno[i][j])  
    }  
  }  
}  
console.log("")  
  
for (i = 2; i < 5; i++) {  
  for (j = 0; j < 3; j++) {  
    if (i + j === (j + 1) * 2 || i + j === 4) {  
      console.log("x2:" + interno[i][j])  
    }  
  }  
}  
console.log("")  
  
for (i = 0; i < 3; i++) {  
  for (j = 2; j < 5; j++) {  
    if (i + j === (j - 1) * 2 || i + j === 4) {  
      console.log("x3:" + interno[i][j])  
    }  
  }  
}  
console.log("")
```

▼ Array(5) 1

- ▶ 0: (5) [2, 4, 6, 8, 10]
- ▶ 1: (5) [12, 14, 16, 18, 20]
- ▶ 2: (5) [22, 24, 26, 28, 30]
- ▶ 3: (5) [32, 34, 36, 38, 40]
- ▶ 4: (5) [42, 44, 46, 48, 50]

length: 5

▶ [[Prototype]]: Array(0)

x1: 2
x1: 6
x1: 14
x1: 22
x1: 26

x2: 22
x2: 26
x2: 34
x2: 42
x2: 46

x3: 6
x3: 10
x3: 18
x3: 26
x3: 30

6.

Nombre de la función: agregarRegistro(...productos)		Versión: 1.0
Descripción: Función que agrega productos en un array de una tienda		
...productos	Tipo de variable: Cadena de caracteres	



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

Código:

```
// Agrega registros
function agregarRegistro(...productos) {
  ...tienda.push(...productos);
  ...numeroRegistros = tienda.length;
}

Tienda: ▼ Array(6) 1
  ▶ 0: {producto: 'Arroz', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 777, precio: 2450}
  ▶ 1: {producto: 'Papa', tipoProducto: 'carnes', tipoUnidad: 'kilogramos', cantidad: 111, precio: 1000}
  ▶ 2: {producto: 'Mora', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 500, precio: 1500}
  ▶ 3: {producto: 'Res', tipoProducto: 'carnes', tipoUnidad: 'gramos', cantidad: 500, precio: 7500}
  ▶ 4: {producto: 'Banano', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 100, precio: 1500}
  ▶ 5: {producto: 'Chocolate', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 300, precio: 8300}
  length: 6
  __proto__: Array(0)
```

6.1

Nombre de la función: modificarRegistro(productoActual, cambios)		Versión: 1.0
Descripción: Función que modifica productos en un array de una tienda		
productoActual, cambios	Tipo de variable: Cadena de caracteres	
Código:		
<pre>// Modifica registros function modificarRegistro(productoActual, cambios) { tienda.forEach((producto, index) => { if (producto.producto === productoActual) { Object.assign(tienda[index], cambios); } }); }</pre>		
<div>Tienda:  Array(6) </div> <div><div>▶ 0: {producto: 'Arroz', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 777, precio: 2450}</div><div>▶ 1: {producto: 'Papa', tipoProducto: 'carnes', tipoUnidad: 'kilogramos', cantidad: 111, precio: 1000}</div><div>▶ 2: {producto: 'Mora', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 500, precio: 1500}</div><div>▶ 3: {producto: 'Res', tipoProducto: 'carnes', tipoUnidad: 'gramos', cantidad: 500, precio: 7500}</div><div>▶ 4: {producto: 'Banano', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 100, precio: 1500}</div><div>▶ 5: {producto: 'Chocolate', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 300, precio: 8300}</div><div>length: 6</div><div>__proto__: Array(0)</div></div>		

6.2

Nombre de la función: eliminarRegistros(...productos)	Versión: 1.0
Descripción: Función que elimina productos en un array de una tienda	



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

...productos Tipo de variable: Cadena de caracteres

Código:

```
// Elimina registros
function eliminarRegistros(...productos) {
  ... productos.forEach(producto => {
    ... tienda = tienda.filter(item => item.producto !== producto);
    ... });
}
```

Tienda: **Array(6)**

- ▶ 0: {producto: 'Arroz', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 777, precio: 2450}
- ▶ 1: {producto: 'Papa', tipoProducto: 'carnes', tipoUnidad: 'kilogramos', cantidad: 111, precio: 1000}
- ▶ 2: {producto: 'Mora', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 500, precio: 1500}
- ▶ 3: {producto: 'Res', tipoProducto: 'carnes', tipoUnidad: 'gramos', cantidad: 500, precio: 7500}
- ▶ 4: {producto: 'Banano', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 100, precio: 1500}
- ▶ 5: {producto: 'Chocolate', tipoProducto: 'fruver', tipoUnidad: 'gramos', cantidad: 300, precio: 8300}

length: 6
[[Prototype]]: Array(0)

7.

Nombre del array: registrosNomina[]

Versión: 1.0

Descripción:

Arreglo que muestra los registros ingresados

Tipo de variable: Numérico, Cadena de caracteres

Código:

```
// Arreglo [Nombre:Valor]
registrosNomina = [
  { id: 1076501813, nombres: 'Brayan Estiven', apellidos: 'Carvajal Padilla', cargo: 'Gerente', valorDia: 200000, diasTrab: 30 },
  { id: 1075679234, nombres: 'Aylin Celeste', apellidos: 'Carvajal Padilla', cargo: 'Operario', valorDia: 80000, diasTrab: 30 },
  { id: 1073209845, nombres: 'Lindora', apellidos: 'Padilla Vanegas', cargo: 'Supervisor', valorDia: 300000, diasTrabajados: 30 },
  { id: 1072903176, nombres: 'Joaquín Emilio', apellidos: 'Carvajal Quintero', cargo: 'Supervisor', valorDia: 300000, diasTrabajados: 30 },
  { id: 1056798009, nombres: 'Daniela Andrea', apellidos: 'Carvajal Padilla', cargo: 'Operario', valorDia: 85000, diasTrabajados: 30 },
  { id: 1089567890, nombres: 'Jonathan Emilio', apellidos: 'Carvajal Padilla', cargo: 'Operario', valorDia: 85000, diasTrabajados: 30 },
  { id: 1023456789, nombres: 'Nicole Fernanda', apellidos: 'Criollo Alvares', cargo: 'Analista', valorDia: 570000, diasTrabajados: 30 },
  { id: 1098234567, nombres: 'Luis Alejandro', apellidos: 'Macana Espinosa', cargo: 'Técnico', valorDia: 120000, diasTrabajados: 30 },
  { id: 1077384965, nombres: 'Anibal', apellidos: 'Alvarado Andrade', cargo: 'Técnico', valorDia: 120000, diasTrabajados: 30 },
  { id: 1078924056, nombres: 'Diego Fernando', apellidos: 'Cuellar Hernandez', cargo: 'Técnico', valorDia: 120000, diasTrabajados: 30 }
];
console.log(registrosNomina);
```

7.1

Nombre de la función: salario(pdias, pVdia)

Versión: 1.0



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

Descripción:

Función que calcula el salario de un empleado

pdias, pVdia

Tipo de variable: Numérico

Código:

```
function salario(pdias, pVdia) {  
  let dias = pdias;  
  let Vdia = pVdia;  
  let totalSalario = dias * Vdia;  
  return totalSalario;  
}  
  
Array(10)   
▶ 0: (11) [1076501813, 'Brayan Estiven', 'Carvajal Padilla', 'Gerente', 44000000, 0, 5280000, 7040000, 2288000, 3520000, 25872000]  
▶ 1: (11) [1075679234, 'Aylin Celeste', 'Carvajal Padilla', 'Operario', 2080000, 120000, 249600, 332800, 108160, 0, 1509440]  
▶ 2: (11) [1073209845, 'Lindora', 'Padilla Vanegas', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]  
▶ 3: (11) [1072903176, 'Joaquin Emilio', 'Carvajal Quintero', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]  
▶ 4: (11) [1056798009, 'Daniela Andrea', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]  
▶ 5: (11) [1089567890, 'Jonathan Emilio', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]  
▶ 6: (11) [1023456789, 'Nicole Fernanda', 'Criollo Alvares', 'Analista', 12540000, 0, 1504800, 2006400, 652080, 250800, 8125920]  
▶ 7: (11) [1098234567, 'Luis Alejandro', 'Macana Espinosa', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
▶ 8: (11) [1077384965, 'Anibal', 'Alvarado Andrade', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
▶ 9: (11) [1078924056, 'Diego Fernando', 'Cuellar Hernandez', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
length: 10
```

7.2

Nombre de la función: salud(pdias, pVdia)

Versión: 1.0

Descripción:

Función que calcula el costo salud de un empleado

pdias, pVdia

Tipo de variable: Numérico

Código:

```
function salud(pdias, pVdia) {  
  let pagoSalud  
  pagoSalud = salario(pdias, pVdia) * 0.12  
  return pagoSalud  
}  
  
Array(10)   
▶ 0: (11) [1076501813, 'Brayan Estiven', 'Carvajal Padilla', 'Gerente', 44000000, 0, 5280000, 7040000, 2288000, 3520000, 25872000]  
▶ 1: (11) [1075679234, 'Aylin Celeste', 'Carvajal Padilla', 'Operario', 2080000, 120000, 249600, 332800, 108160, 0, 1509440]  
▶ 2: (11) [1073209845, 'Lindora', 'Padilla Vanegas', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]  
▶ 3: (11) [1072903176, 'Joaquin Emilio', 'Carvajal Quintero', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]  
▶ 4: (11) [1056798009, 'Daniela Andrea', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]  
▶ 5: (11) [1089567890, 'Jonathan Emilio', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]  
▶ 6: (11) [1023456789, 'Nicole Fernanda', 'Criollo Alvares', 'Analista', 12540000, 0, 1504800, 2006400, 652080, 250800, 8125920]  
▶ 7: (11) [1098234567, 'Luis Alejandro', 'Macana Espinosa', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
▶ 8: (11) [1077384965, 'Anibal', 'Alvarado Andrade', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
▶ 9: (11) [1078924056, 'Diego Fernando', 'Cuellar Hernandez', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
length: 10
```

7.3



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

Nombre de la función: SubTrans(pdias, pVdia)

Versión: 1.0

Descripción:

Función que calcula el subsidio de transporte de un empleado

pdias, pVdia

Tipo de variable: Numérico

Código:

```
function SubTrans(pdias, pVdia) {  
  let SalarioMinimo = 1600000  
  let SalarioTrans = salario(pdias, pVdia)  
  let SubTransporte  
  if (SalarioTrans <= 2 * SalarioMinimo) {  
    SubTransporte = 120000  
  }  
  else {  
    SubTransporte = 0  
  }  
  return SubTransporte  
}
```

```
0: (11) [1076501813, 'Brayan Estiven', 'Carvajal Padilla', 'Gerente', 4400000, 0, 5280000, 7040000, 2288000, 3520000, 25872000]  
1: (11) [1075679234, 'Aylin Celeste', 'Carvajal Padilla', 'Operario', 2080000, 120000, 249600, 332800, 108160, 0, 1509440]  
2: (11) [1073209845, 'Lindora', 'Padilla Vanegas', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]  
3: (11) [1072903176, 'Joaquin Emilio', 'Carvajal Quintero', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]  
4: (11) [1056798009, 'Daniela Andrea', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]  
5: (11) [1089567890, 'Jonathan Emilio', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]  
6: (11) [1023456789, 'Nicole Fernanda', 'Criollo Alvares', 'Analista', 12540000, 0, 1504800, 2006400, 652080, 250800, 8125920]  
7: (11) [1098234567, 'Luis Alejandro', 'Macana Espinosa', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
8: (11) [1077384965, 'Anibal', 'Alvarado Andrade', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
9: (11) [1078924056, 'Diego Fernando', 'Cuellar Hernandez', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]  
length: 10
```

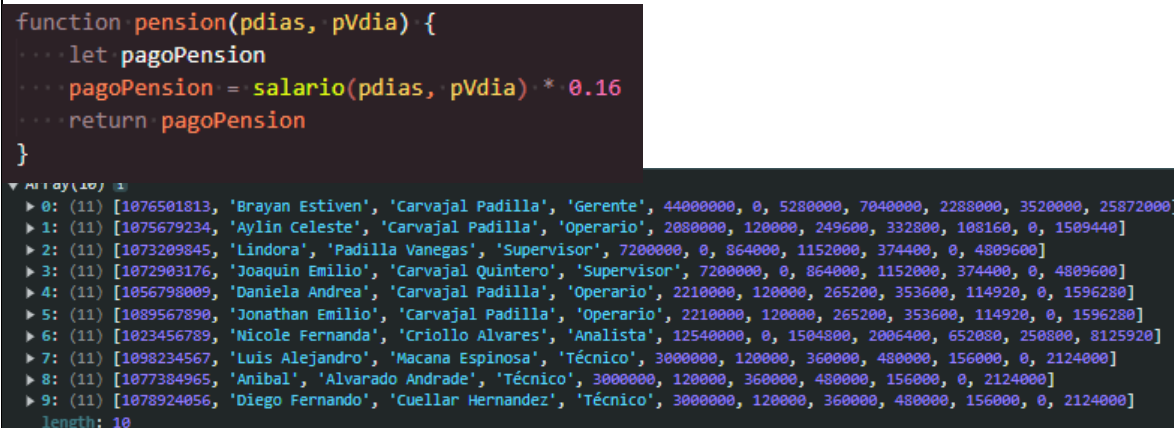


Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

7.4

Nombre de la función: pension(pdias, pVdia)		Versión: 1.0
Descripción: Función que calcula la pensión de un empleado		
pdias, pVdia		Tipo de variable: Numérico
Código: <pre>function pension(pdias, pVdia) { let pagoPension pagoPension = salario(pdias, pVdia) * 0.16 return pagoPension }</pre> 		

7.5

Nombre de la función: arl(pdias, pVdia)		Versión: 1.0
Descripción: Función que calcula el arl de un empleado		
pdias, pVdia	Tipo de variable: Numérico	
Código: <pre>function arl(pdias, pVdia) { ...let pagoArl ...pagoArl = salario(pdias, pVdia) * 0.052 ...return pagoArl }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

```
▼ Array(10) 1
  ▶ 0: (11) [1076501813, 'Brayan Estiven', 'Carvajal Padilla', 'Gerente', 44000000, 0, 5280000, 7040000, 2288000, 3520000, 25872000]
  ▶ 1: (11) [1075679234, 'Aylin Celeste', 'Carvajal Padilla', 'Operario', 2080000, 120000, 249600, 332800, 108160, 0, 1509440]
  ▶ 2: (11) [1073209845, 'Lindora', 'Padilla Vanegas', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]
  ▶ 3: (11) [1072903176, 'Joaquín Emilio', 'Carvajal Quintero', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]
  ▶ 4: (11) [1056798009, 'Daniela Andrea', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]
  ▶ 5: (11) [1089567890, 'Jonathan Emilio', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]
  ▶ 6: (11) [1023456789, 'Nicole Fernanda', 'Criollo Alvares', 'Analista', 12540000, 0, 1504800, 2006400, 652080, 250800, 8125920]
  ▶ 7: (11) [1098234567, 'Luis Alejandro', 'Macana Espinosa', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]
  ▶ 8: (11) [1077384965, 'Anibal', 'Alvarado Andrade', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]
  ▶ 9: (11) [1078924056, 'Diego Fernando', 'Cuellar Hernandez', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]
  length: 10
```

7.6

Nombre de la función: retencion(pdias, pVdia)		Versión: 1.0
Descripción: Función que calcula la retención de un empleado		
pdias, pVdia	Tipo de variable: Numérico	
Código:		
<pre>function retencion(pdias, pVdia) { let SalarioMin = 1600000; let SalarioRet = salario(pdias, pVdia); let Retencion; if (SalarioRet > 6 * SalarioMin && SalarioRet < 8 * SalarioMin) { Retencion = SalarioRet * 0.02; } else if (SalarioRet > 8 * SalarioMin && SalarioRet < 12 * SalarioMin) { Retencion = SalarioRet * 0.04; } else if (SalarioRet > 12 * SalarioMin) { Retencion = SalarioRet * 0.08; } else { Retencion = 0; } return Retencion; }</pre> <pre>▶ 0: (11) [1076501813, 'Brayan Estiven', 'Carvajal Padilla', 'Gerente', 44000000, 0, 5280000, 7040000, 2288000, 3520000, 25872000] ▶ 1: (11) [1075679234, 'Aylin Celeste', 'Carvajal Padilla', 'Operario', 2080000, 120000, 249600, 332800, 108160, 0, 1509440] ▶ 2: (11) [1073209845, 'Lindora', 'Padilla Vanegas', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600] ▶ 3: (11) [1072903176, 'Joaquin Emilio', 'Carvajal Quintero', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600] ▶ 4: (11) [1056798009, 'Daniela Andrea', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280] ▶ 5: (11) [1089567890, 'Jonathan Emilio', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280] ▶ 6: (11) [1023456789, 'Nicole Fernanda', 'Criollo Alvares', 'Analista', 12540000, 0, 1504800, 2006400, 652080, 250800, 8125920] ▶ 7: (11) [1098234567, 'Luis Alejandro', 'Macana Espinosa', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000] ▶ 8: (11) [1077384965, 'Anibal', 'Alvarado Andrade', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000] ▶ 9: (11) [1078924056, 'Diego Fernando', 'Cuellar Hernandez', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000] length: 10</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones/Arreglos JS

7.7

Nombre de la función: pagoTotal(pdias, pVdia)		Versión: 1.0
Descripción: Función que calcula el pago total de un empleado		
pdias, pVdia	Tipo de variable: Numérico	
Código:		
<pre>function pagoTotal(pdias, pVdia) { --let totalSalario = salario(pdias, pVdia); --let totalDeducciones = salud(pdias, pVdia) + pension(pdias, pVdia) + arl(pdias, pVdia) + retencion(pdias, pVdia) --return totalSalario + SubTrans(pdias, pVdia) - totalDeducciones; }</pre> <p>▼ Array(10) ⓘ</p> <ul style="list-style-type: none">▶ 0: (11) [1076501813, 'Brayan Estiven', 'Carvajal Padilla', 'Gerente', 4400000, 0, 5280000, 7040000, 2288000, 3520000, 25872000]▶ 1: (11) [1075679234, 'Aylin Celeste', 'Carvajal Padilla', 'Operario', 2080000, 120000, 249600, 332800, 108160, 0, 1509440]▶ 2: (11) [1073209845, 'Lindora', 'Padilla Vanegas', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]▶ 3: (11) [1072903176, 'Joaquin Emilio', 'Carvajal Quintero', 'Supervisor', 7200000, 0, 864000, 1152000, 374400, 0, 4809600]▶ 4: (11) [1056798009, 'Daniela Andrea', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]▶ 5: (11) [1089567890, 'Jonathan Emilio', 'Carvajal Padilla', 'Operario', 2210000, 120000, 265200, 353600, 114920, 0, 1596280]▶ 6: (11) [1023456789, 'Nicole Fernanda', 'Criollo Alvares', 'Analista', 12540000, 0, 1504800, 2006400, 652080, 250800, 8125920]▶ 7: (11) [1098234567, 'Luis Alejandro', 'Macana Espinosa', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]▶ 8: (11) [1077384965, 'Anibal', 'Alvarado Andrade', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000]▶ 9: (11) [1078924056, 'Diego Fernando', 'Cuellar Hernandez', 'Técnico', 3000000, 120000, 360000, 480000, 156000, 0, 2124000] <p>length: 10</p>		