



Tecnólogo en Análisis y Desarrollo de  
Software  
Ficha

**Funciones JS**

RUBEN FELIPE TOVAR AVILES

INSTRUCTOR: ANDRES MORENO COLLAZOS

SENA – CIES  
2900177  
2024



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

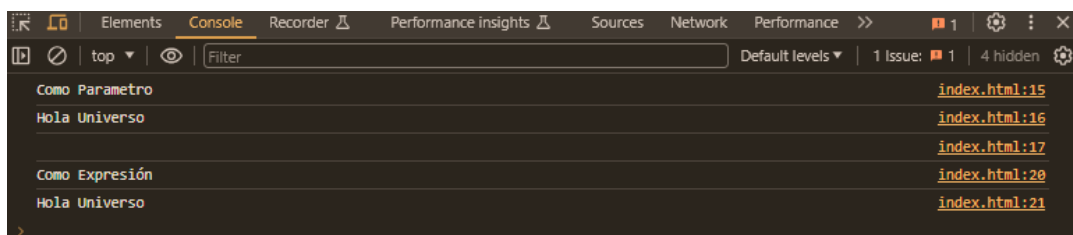
### Funciones JS

#### 01 - Saludo

Nombre de la función: saludo(pSaludo)		Versión: 2.0
Descripción: Función que saluda		
Variable	Tipo de variable	
saludar	string	
Código:		
<pre>function saludo(pSaludo){     let saludar = pSaludo;      return saludar; //Valor }</pre>		

Nombre de la función: const saludoExpresion = function(pSaludo)		Versión: 3.0
Descripción: Función que saluda		
Variable saludar	Tipo de variable string	
Código: <pre>const saludoExpresion = function(pSaludo){   let saludar = pSaludo;    return saludar; //Valor }</pre>		

#### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

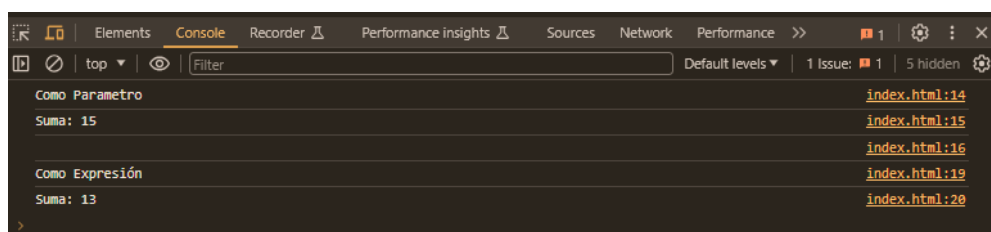
### Funciones JS

## 02 - Suma

Nombre de la función: suma(pNumUno, pNumDos)		Versión: 2.0
Descripción: Función que suma dos números		
Variable	Tipo de variable	
sumar	Int	
numUno	Int	
numDos	Int	
Código:		
<pre>function suma(pNumUno, pNumDos){     let numUno = pNumUno;     let numDos = pNumDos;     let sumar;     sumar = numUno + numDos      return sumar }</pre>		

Nombre de la función: sumaExpresion = function(pNumUno, pNumDos)		Versión: 3.0
Descripción: Función que suma dos números		
Variable	Tipo de variable	
sumar	Int	
numUno	Int	
numDos	Int	
Código:	<pre>const sumaExpresion = function(pNumUno, pNumDos){     let numUno = pNumUno     let numDos = pNumDos     let sumar     sumar = numUno + numDos      return sumar }</pre>	

## PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

## 03 - Operaciones

Nombre de la función: function suma(pNumUno, pNumDos)		Versión: 2.0
Descripción: Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable	Tipo de variable	
sumar	Int	
numUno	Int	
numDos	Int	
Código:		
<pre>//Como Parametro function suma(pNumUno, pNumDos){     let sumar;     numUno = pNumUno;     numDos = pNumDos;     sumar = numUno + numDos      return sumar }</pre>		

Nombre de la función: function resta(pNumUno, pNumDos)		Versión: 2.0
Descripción: Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable	Tipo de variable	
restar	Int	
numUno	Int	
numDos	Int	
Código:		
<pre>function resta(pNumUno, pNumDos){     let restar;     numUno = pNumUno;     numDos = pNumDos;     restar = numUno - numDos      return restar }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: function multiplicacion(pNumUno, pNumDos)		Versión: 2.0
Descripción: Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable multiplicar numUno numDos	Tipo de variable Int Int Int	
Código: <pre>function multiplicacion(pNumUno, pNumDos){     let multiplicar;     numUno = pNumUno;     numDos = pNumDos;     multiplicar = numUno * numDos      return multiplicar }</pre>		

Nombre de la función: function division(pNumUno, pNumDos)		Versión: 2.0
Descripción: Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable	Tipo de variable	
dividir	float	
numUno	Int	
numDos	Int	
Código:		
<pre>function division(pNumUno, pNumDos){     let dividir;     numUno = pNumUno;     numDos = pNumDos;     dividir = numUno / numDos      return dividir }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `function operaciones(pOperador, pNumUno, pNumDos)` Versión: 2.0

#### Descripción:

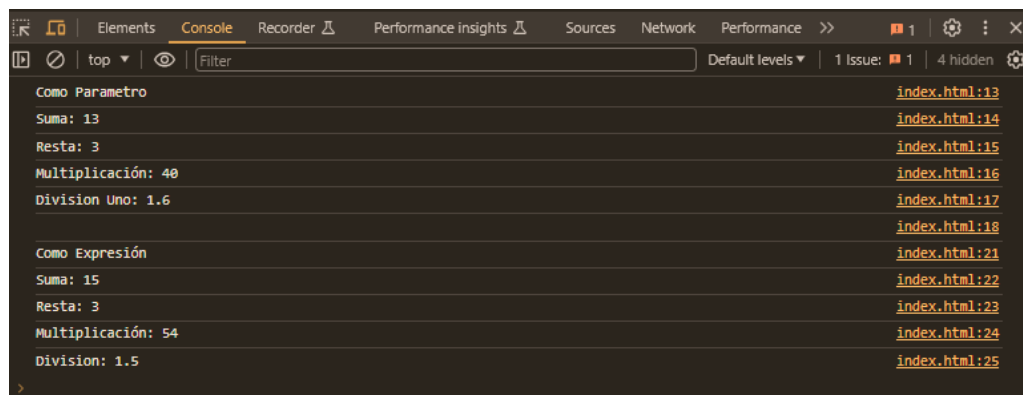
Función que hace operaciones básicas (sumar - restar - multiplicar - dividir)

Variable	Tipo de variable
operador	string
numUno	Int
numDos	Int

Código:

```
function operaciones(pOperador, pNumUno, pNumDos){  
    let operadaor = pOperador  
    numUno = pNumUno;  
    numDos = pNumDos;  
    if(operadaor == 'suma'){  
        return suma(numUno, numDos)  
    }else if(operadaor == 'resta'){  
        return resta(numUno, numDos)  
    }else if(operadaor == 'multiplicacion'){  
        return multiplicacion(numUno, numDos)  
    }else if(operadaor == 'division'){  
        return division(numUno, numDos)  
    }else{  
        return 'No se reconoce la operación'  
    }  
}
```

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: const sumaExpresion = function(pNumUno, pNumDos)		Versión: 3.0
Descripción: Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable	Tipo de variable	
sumar	Int	
numUno	Int	
numDos	Int	
Código: <pre>//Como Expresión const sumaExpresion = function(pNumUno, pNumDos){     let sumar;     numUno = pNumUno;     numDos = pNumDos;     sumar = numUno + numDos      return sumar }</pre>		

<b>Nombre de la función:</b> const restaExpresion = function(pNumUno, pNumDos)		<b>Versión:</b> 3.0
<b>Descripción:</b> Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable	Tipo de variable	
restar	Int	
numUno	Int	
numDos	Int	
Código: <pre>const restaExpresion = function(pNumUno, pNumDos){     let restar;     numUno = pNumUno;     numDos = pNumDos;     restar = numUno - numDos      return restar }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: const multiplicacionExpresio = function(pNumUno, pNumDos)		Versión: 3.0
Descripción: Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable	Tipo de variable	
multiplicar	Int	
numUno	Int	
numDos	Int	
Código:		
<pre>const multiplicacionExpresio = function(pNumUno, pNumDos){     let multiplicar;     numUno = pNumUno;     numDos = pNumDos;     multiplicar = numUno * numDos      return multiplicar }</pre>		

Nombre de la función: const divisionExpresion = function(pNumUno, pNumDos)		Versión: 3.0
Descripción: Función que hace operaciones básicas (sumar - restar – multiplicar – dividir)		
Variable	Tipo de variable	
dividir	Float	
numUno	Int	
numDos	Int	
Código: <pre>const divisionExpresion = function(pNumUno, pNumDos){     let dividir;     numUno = pNumUno;     numDos = pNumDos;     dividir = numUno / numDos      return dividir }</pre>		





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `const operacionesExpresion = function(pOperador, pNumUno, pNumDos)` Versión: 3.0

#### Descripción:

Función que hace operaciones básicas (sumar - restar - multiplicar - dividir)

Variable	Tipo de variable
operador	string
numUno	Int
numDos	Int

#### Código:

```
const operacionesExpresion = function(pOperador, pNumUno, pNumDos){  
    let operadaor = pOperador  
    numUno = pNumUno;  
    numDos = pNumDos;  
    if(operadaor == 'sumaExpresion'){  
        return sumaExpresion(numUno, numDos)  
    }else if(operadaor == 'restaExpresion'){  
        return restaExpresion(numUno, numDos)  
    }else if(operadaor == 'multiplicacionExpresion'){  
        return multiplicacionExpresion(numUno, numDos)  
    }else if(operadaor == 'divisionExpresion'){  
        return divisionExpresion(numUno, numDos)  
    }else{  
        return 'No se reconoce la operación'  
    }  
}
```

### PRUEBA DE ESCRITORIO

Elements	Console	Recorder	Performance insights	Sources	Network	Performance	>>	1	Settings	Close
Filter										
Default levels 1 Issue 1 4 hidden										
Como Parametro										
Suma: 13										<a href="#">index.html:13</a>
Resta: 3										<a href="#">index.html:14</a>
Multiplicación: 40										<a href="#">index.html:15</a>
Division Uno: 1.6										<a href="#">index.html:16</a>
										<a href="#">index.html:17</a>
										<a href="#">index.html:18</a>
Como Expresión										
Suma: 15										<a href="#">index.html:21</a>
Resta: 3										<a href="#">index.html:22</a>
Multiplicación: 54										<a href="#">index.html:23</a>
Division: 1.5										<a href="#">index.html:24</a>
										<a href="#">index.html:25</a>



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### 04 - Porcentaje

Nombre de la función: function porcentaje(pNumero)		Versión: 2.0
Descripción: Halla el porcentaje de un número		
Variable	Tipo de variable	
numero	Int	
porcentajeNumero	float	
Código:		
<pre>// Como Parametro function porcentaje(pNumero){     let numero = pNumero;     let porcentajeNumero     porcentajeNumero = numero / 100      return porcentajeNumero }</pre>		

Nombre de la función: const porcentajeExpresion = function(pNumero)		Versión: 3.0
<b>Descripción:</b> Halla el porcentaje de un número		
porcentaje	Tipo de variable: Alfanumérica	
<b>Código:</b> <pre>//Como Expresión const porcentajeExpresion = function(pNumero){     let numero = pNumero     let porcentajeNumero     porcentajeNumero = numero / 100      return porcentajeNumero }</pre>		

#### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

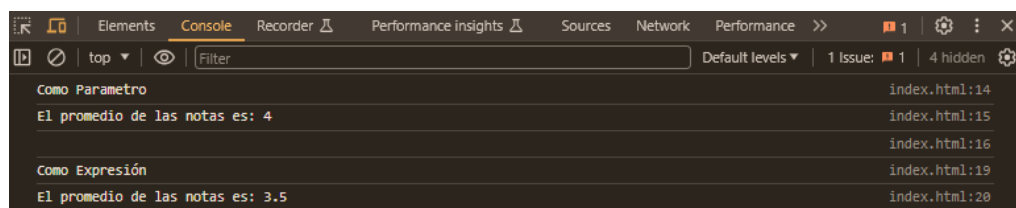
### Funciones JS

#### 05 – Promedio de Notas

Nombre de la función: function promedioNotas(pNota1, pNota2, pNota3)		Versión: 2.0
<b>Descripción:</b> Imprime el promedio de tres notas		
Variable	Tipo de variable	
nota1	Float	
nota2	Float	
nota3	Float	
promedio	Float	
Código:		
<pre>function promedioNotas(pNota1, pNota2, pNota3){     let nota1 = pNota1     let nota2 = pNota2     let nota3 = pNota3     let promedio     promedio = (nota1 + nota2 + nota3)/3;      return promedio }</pre>		

Nombre de la función: const promedioNotasExpresion = function(pNota1, pNota2, pNota3)		Versión: 3.0
Descripción: Imprime el promedio de tres notas		
Variable	Tipo de variable	
nota1	Float	
nota2	Float	
nota3	Float	
promedio	Float	
Código:		
<pre>const promedioNotasExpresion = function(pNota1, pNota2, pNota3){     let nota1 = pNota1     let nota2 = pNota2     let nota3 = pNota3     let promedio     promedio = (nota1 + nota2 + nota3)/3;      return promedio };</pre>		

#### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

## 06 – Promedio de Notas 2

Nombre de la función: function sumarPorcentaje(pNota, pPorcentaje)		Versión: 2.0
Descripción: Calcular el porcentaje ponderado de tres notas. Nota 1 = 30% Nota 2 = 30% Nota 3 = 40%		
Variable	Tipo de variable	
nota	Float	
porcentaje	Float	
resultadoPorcentaje	Float	
Código:		
<pre>function sumarPorcentaje(pNota, pPorcentaje){     let nota = pNota     let porcentaje = pPorcentaje     let resultadoPorcentaje      resultadoPorcentaje = nota * porcentaje      return resultadoPorcentaje }</pre>		

<b>Nombre de la función:</b> const sumarPorcentajeExpresion = function(pNota, pPorcentaje)		<b>Versión:</b> 3.0
<b>Descripción:</b> Calcular el porcentaje ponderado de tres notas. Nota 1 = 30% Nota 2 = 30% Nota 3 = 40%		
Variable	Tipo de variable	
nota	Float	
porcentaje	Float	
resultadoPorcentaje	Float	
Código:		
<pre>const sumarPorcentajeExpresion = function(pNota, pPorcentaje){     let nota = pNota     let porcentaje = pPorcentaje     let resultadoPorcentaje      resultadoPorcentaje = nota * porcentaje      return resultadoPorcentaje }</pre>		

## PRUEBA DE ESCRITORIO

Como Parametro	index.html:19
El porcentaje de la nota 1 es: 1.05	index.html:20
El porcentaje de la nota 2 es: 1.35	index.html:21
El porcentaje de la nota 3 es: 2	index.html:22
El la suma de los porcentajes es: 4.4	index.html:23
Como Expresion	index.html:24
El porcentaje de la nota 1 es: 1.3499999999999999	index.html:25
El porcentaje de la nota 2 es: 1.05	index.html:26
El porcentaje de la nota 3 es: 1.6	index.html:27
El la suma de los porcentajes es: 4	index.html:28



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### 07 – Área de Figuras Geométricas

Nombre de la función: function areaCuadrado(pLado)		Versión: 2.0
<b>Descripción:</b> Calcula el área de un cuadrado, rectángulo y triángulo		
Variable	Tipo de variable	
lado	Float	
area	Float	
Código:		
<pre>function areaCuadrado(pLado){     let lado = pLado     let area     area = lado * lado      return area }</pre>		

Nombre de la función: function areaRectangulo(pLado)		Versión: 2.0
Descripción: Calcula el área de un cuadrado, rectángulo y triángulo		
Variable	Tipo de variable	
base	Float	
altura	Float	
area	Float	
Código:		
<pre>function areaRectangulo(pBase, pAltura){     let base = pBase     let altura = pAltura     let area     area = base * altura      return area }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `function areaTriangulo(pLado)`

Versión: 2.0

#### Descripción:

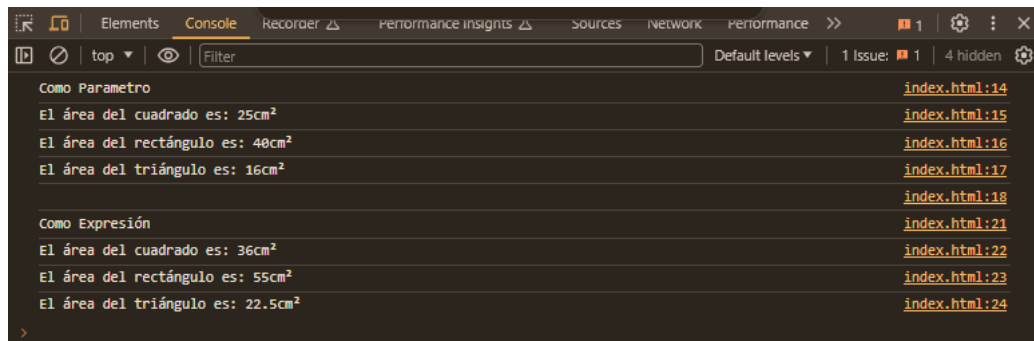
Calcula el área de un cuadrado, rectángulo y triángulo

Variable	Tipo de variable
base	Float
altura	Float
area	Float

Código:

```
function areaTriangulo(pBase, pAltura){  
    let base = pBase  
    let altura = pAltura  
    let area  
    area = (base * altura)/2  
  
    return area  
}
```

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: const <u>areaCuadradoExpresion</u> = function(pLado)		Versión: 3.0
Descripción: Calcula el área de un cuadrado, rectángulo y triángulo		
Variable	Tipo de variable	
lado	Float	
area	Float	
Código: <pre>const areaCuadradoExpresion = function(pLado){     let lado = pLado     let area     area = lado * lado      return area }</pre>		

Nombre de la función: <code>const areaRectanguloExpresion = function(pBase, pLado)</code>		Versión: 3.0
Descripción: Calcula el área de un cuadrado, rectángulo y triángulo		
Variable	Tipo de variable	
base	Float	
altura	Float	
area	Float	
Código:		
<pre>const areaRectanguloExpresion = function(pBase, pAltura){     let base = pBase     let altura = pAltura     let area     area = base * altura      return area }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `const areaTrianguloExpresion = function(pBase, pLado)`

Versión: 3.0

#### Descripción:

Calcula el área de un cuadrado, rectángulo y triángulo

#### Variable

base  
altura  
area

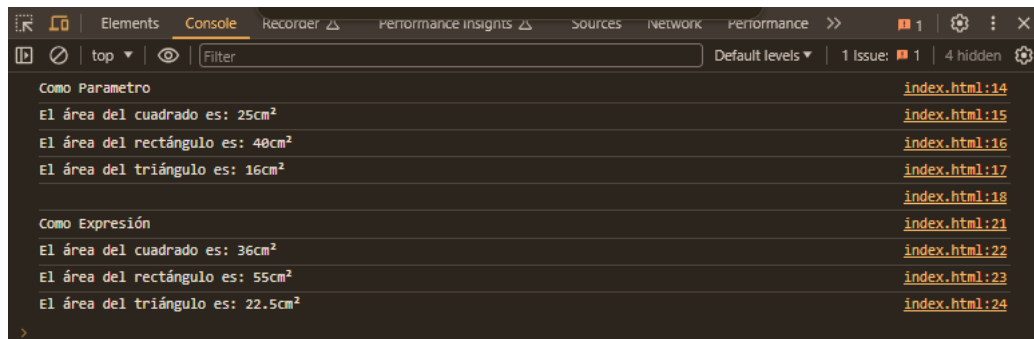
Float  
Float  
Float

#### Tipo de variable

#### Código:

```
const areaTrianguloExpresion = function(pBase, pAltura){  
  let base = pBase  
  let altura = pAltura  
  let area  
  area = (base * altura)/2  
  
  return area  
}
```

## PRUEBA DE ESCRITORIO







# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### 08 – Sueldo de Persona

Nombre de la función: function calcularSueldo(pValorD, pDiasT)		Versión: 2.0
<b>Descripción:</b> Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)		
Variable	Tipo de variable	
valorD	Float	
diasT	Int	
sueldo	Float	
Código:		
<pre>function calcularSueldo(pValorD, pDiasT){     let valorD = pValorD;     let diasT = pDiasT;     let sueldo     sueldo = valorD * diasT;     return sueldo; }</pre>		

Nombre de la función: function calcularPension(pSueldo)		Versión: 2.0
<b>Descripción:</b> Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)		
Variable	Tipo de variable	
pen	Float	
Código:		
<pre>function calcularPension(pValorD, pDiasT){     let pen;     pen = calcularSueldo(pValorD, pDiasT) * 0.16;     return pen; }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

<b>Nombre de la función:</b> function calcularSalud(pSueldo)	<b>Versión:</b> 2.0
--	---------------------

**Descripción:**

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable	Tipo de variable
salud	Float

Código:

```
function calcularSalud(pValorD, pDiasT){  
    let salud;  
    salud = calcularSueldo(pValorD, pDiasT) * 0.12;  
    return salud;  
}
```

<b>Nombre de la función:</b> function calcularARL(pSueldo)	<b>Versión:</b> 2.0
--	---------------------

**Descripción:**

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable	Tipo de variable
ARL	Float

Código:

```
function calcularARL(pValorD, pDiasT){  
    let arl;  
    arl = calcularSueldo(pValorD, pDiasT) * 0.16;  
    return arl;  
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `function calcularSueldoTotal(pSueldo)` Versión: 2.0

#### Descripción:

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable	Tipo de variable
sueldoCalculado	Float
pen	Float
salud	Float
arl	Float
desc	Float
sueldoTotal	Float

#### Código:

```
function calcularSueldoTotal(pValorD, pDiasT) {  
    let sueldoCalculado = calcularSueldoExpresion(pValorD, pDiasT);  
    let pen = calcularPension(pValorD, pDiasT);  
    let salud = calcularSalud(pValorD, pDiasT);  
    let arl = calcularARL(pValorD, pDiasT);  
    let desc = pen + salud + arl;  
    let sueldoTotal = sueldoCalculado - desc;  
    return sueldoTotal;  
}
```

## PRUEBA DE ESCRITORIO

```
Como Parametro                                index.html:17  
Al valor del siguiente sueldo 3960000 se le descuenta:  index.html:18  
Pensión: 633600                                index.html:19  
Salud: 475200                                  index.html:20  
ARL: 633600                                    index.html:21  
El sueldo total del trabajador es: 2217600        index.html:22  
                                                index.html:23  
Como Expresión                                index.html:26  
Al valor del siguiente sueldo 3960000 se le descuenta:  index.html:27  
Pensión: 633600                                index.html:28  
Salud: 475200                                  index.html:29  
ARL: 205920                                    index.html:30  
El sueldo total del trabajador es: 2645280        index.html:31  
Live reload enabled.                            index.html:63
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** const calcularSueldoExpresion = function(pValorD, pDiasT)

**Versión:** 2.0

**Descripción:**

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable	Tipo de variable
valorD	Float
diasT	Int
sueldo	Float

Código:

```
const calcularSueldoExpresion = function(pValorD, pDiasT){  
    let valorD = pValorD;  
    let diasT = pDiasT;  
    let sueldo;  
    sueldo = valorD * diasT;  
    return sueldo;  
}
```

**Nombre de la función:** const calcularPensionExpresion = function(pSueldo)

**Versión:** 2.0

**Descripción:**

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable	Tipo de variable
pen	Float

Código:

```
const calcularPensionExpresion = function(pValorD, pDiasT){  
    let pen;  
    pen = calcularSueldo(pValorD, pDiasT) * 0.16;  
    return pen;  
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** const calcularSaludExpresion = function(pSueldo)

**Versión:** 2.0

**Descripción:**

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable  
salud

Float

Tipo de variable

Código:

```
const calcularSaludExpresion = function(pValorD, pDiasT){  
  let salud;  
  salud = calcularSueldo(pValorD,pDiasT) * 0.12;  
  return salud;  
}
```

**Nombre de la función:** const calcularARLExpresion = function(pSueldo)

**Versión:** 2.0

**Descripción:**

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable  
ARL

Float

Tipo de variable

Código:

```
const calcularARLExpresion = function(pValorD, pDiasT){  
  let arl;  
  arl = calcularSueldo(pValorD,pDiasT) * 0.16;  
  return arl;  
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `const calcularSueldoTotalExpresion = function()`

Versión: 2.0

#### Descripción:

Calcular el sueldo de un trabajador con deducciones de pensión, salud y ARL. (pensión = 0.16, salud = 0.12, ARL = 0.052)

Variable	Tipo de variable
sueldoCalculado	Float
pen	Float
salud	Float
arl	Float
desc	Float
sueldoTotal	Float

Código:

```
const calcularSueldoTotalExpresion = function(pValorD, pDiasT) {  
    let sueldoCalculado = calcularSueldoExpresion(pValorD, pDiasT);  
    let pen = calcularPensionExpresion(pValorD, pDiasT);  
    let salud = calcularSaludExpresion(pValorD, pDiasT);  
    let arl = calcularARLExpresion(pValorD, pDiasT);  
    let desc = pen + salud + arl;  
    let sueldoTotal = sueldoCalculado - desc;  
    return sueldoTotal;  
}
```

### PRUEBA DE ESCRITORIO

The screenshot shows a web browser console with the following output:

```
Como Parametro                                index.html:17  
Al valor del siguiente sueldo 3960000 se le descuenta:  index.html:18  
Pensión: 633600                                index.html:19  
Salud: 475200                                  index.html:20  
ARL: 633600                                    index.html:21  
El sueldo total del trabajador es: 2217600        index.html:22  
  
Como Expresión                                index.html:26  
Al valor del siguiente sueldo 3960000 se le descuenta:  index.html:27  
Pensión: 633600                                index.html:28  
Salud: 475200                                  index.html:29  
ARL: 205920                                    index.html:30  
El sueldo total del trabajador es: 2645280        index.html:31  
Live reload enabled.                             index.html:63
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

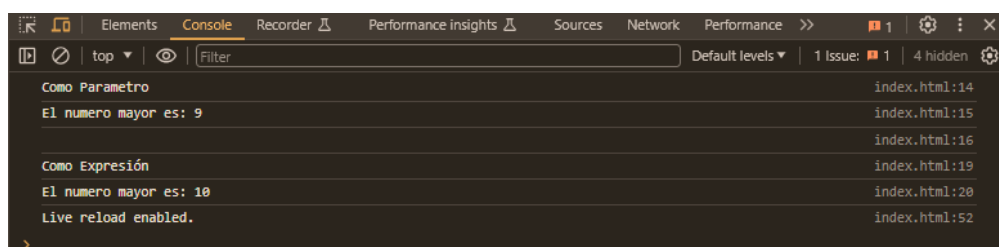
## CONDICIONALES

### 09 – Numero Mayor

Nombre de la función: function numeroMayor(pNum1, pNum2)		Versión: 2.0
Descripción: Imprimir el mayor de dos números		
Variable	Tipo de variable	
num1	Int	
num2	Int	
Código:		
<pre>function numeroMayor(pNum1, pNum2){     let num1 = pNum1;     let num2 = pNum2;      if(num1 &gt; num2){         return num1;     }else{         return num2;     } }</pre>		

Nombre de la función: const numeroMayorExpresion = function(pNum1, pNum2)		Versión: 3.0
Descripción: Imprimir el mayor de dos números		
Variable	Tipo de variable: Alfanumérica	
num1	Int	
num2	Int	
Código:	<pre>const numeroMayorExpresion = function(pNum1, pNum2){   let num1 = pNum1;   let num2 = pNum2;    if(num1 &gt; num2){     return num1;   }else{     return num2;   } }</pre>	

## PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

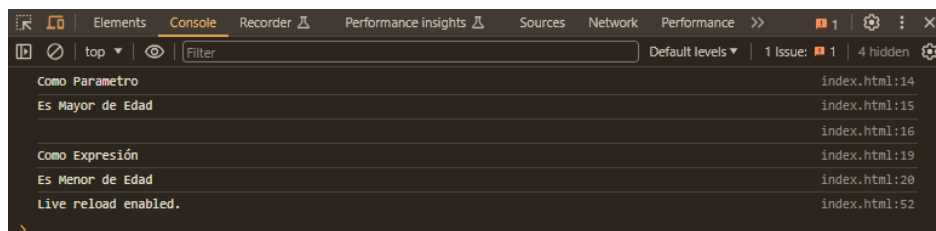
### Funciones JS

## 10 – Calcular Edad

Nombre de la función: function mayorEdad(pFechaN, pYearActual)		Versión: 2.0
Descripción: Calcular la edad de una persona e imprimir si es mayor de edad		
Variable	Tipo de variable	
fechaN	Int	
yearActual	Int	
edad	Int	
Código:		
<pre>function mayorEdad(pFechaN, pYearActual){   let fechaN = pFechaN;   let yearActual = pYearActual   let edad   edad = yearActual - fechaN    if(edad &gt; 17){     return "Es Mayor de Edad"   }else{     return "Es Menor de Edad"   } }</pre>		

Nombre de la función: const mayorEdadExpresion = function(pFechaN, pYearActual)		Versión: 3.0
Descripción: Calcular la edad de una persona e imprimir si es mayor de edad		
Variable	Tipo de variable	
fechaN	Int	
yearActual	Int	
edad	Int	
Código:		
<pre>const mayorEdadExpresion = function(pFechaN, pYearActual){   let fechaN = pFechaN;   let yearActual = pYearActual   let edad   edad = yearActual - fechaN    if(edad &gt; 17){     return "Es Mayor de Edad"   }else{     return "Es Menor de Edad"   } }</pre>		

## PRUEBA DE ESCRITORIO







# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

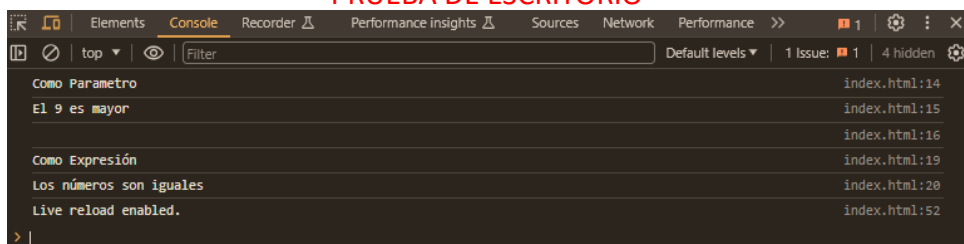
### Funciones JS

## 11 – Mayor de Dos Números

Nombre de la función: function numeroMayor(pNum1, pNum2)		Versión: 2.0
Descripción: Imprime el mayor de dos números validando que no sean iguales		
Variables	Tipo de variable	
num1	Int	
num2	Int	
Código:		
<pre>function numeroMayor(pNum1, pNum2){     let num1 = pNum1;     let num2 = pNum2;      if(num1 == num2){         return "Los números son iguales ";     }else if(num1 &gt; num2){         return "El " + num1 + " es mayor";     }else{         return "El " + num2 + " es mayor";     } }</pre>		

Nombre de la función: const numeroMayorExpresion = function(pNum1, pNum2)		Versión: 3.0
Descripción: Imprime el mayor de dos números validando que no sean iguales		
Variable	Tipo de variable	
num1	Int	
num2	Int	
Código:		
<pre>const numeroMayorExpresion = function(pNum1, pNum2){   let num1 = pNum1;   let num2 = pNum2;    if(num1 == num2){     return "Los números son iguales ";   }else if(num1 &gt; num2){     return "El " + num1 + " es mayor";   }else{     return "El " + num2 + " es mayor";   } }</pre>		

## PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

## 12- Mayor de Tres Números

Nombre de la función: function numeroMayor(pNum1, pNum2, pNum3){		Versión: 2.0
Descripción: Imprime el mayor de tres números con la misma validación		
Variables	Tipo de variable	
num1	Int	
num2	Int	
num3	Int	
Código:		
<pre>function numeroMayor(pNum1, pNum2, pNum3){     let num1 = pNum1;     let num2 = pNum2;     let num3 = pNum3;      if(num1 == num2    num1 == num2    num2 == num3){         return "Los números son iguales";     }     else if(num1 &gt; num2 &amp;&amp; num1 &gt; num3){         return "El primer número es mayor: " + num1;     }     else if(num2 &gt; num1 &amp;&amp; num2 &gt; num3){         return "El segundo número mayor es: " + num2;     }     else{         return "El tercer número mayor es: " + num3;     } }</pre>		

## PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función:

Versión: 3.0

**Descripción:**

Imprimir el mayor de tres números con la misma validación

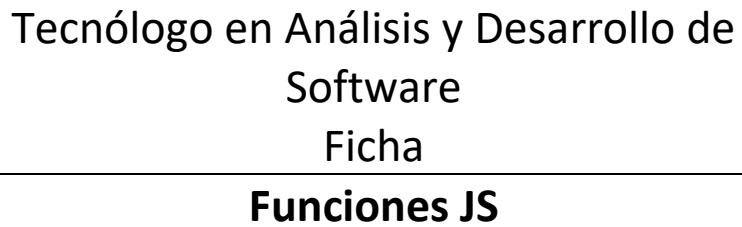
Variables	Tipo de variable
num1	Int
num2	Int
num3	Int

Código:

```
const numeroMayorExpresion = function(pNum1, pNum2, pNum3){  
    let num1 = pNum1;  
    let num2 = pNum2;  
    let num3 = pNum3;  
  
    if(num1 == null || num2 == null || num3 == null){  
        return "No hay números"  
    }else{  
        if(num1 == num2 || num1 == num2 || num2 == num3){  
            return "Los números son iguales";  
        }  
        else if(num1 > num2 && num1 > num3){  
            return "El primer número es mayor: " + num1;  
        }  
        else if(num2 > num1 && num2 > num3){  
            return "El segundo número mayor es: " + num2;  
        }  
        else{  
            return "El tercer número mayor es: " + num3;  
        }  
    }  
}
```

### PRUEBA DE ESCRITORIO





<b>Nombre de la función:</b> function areaCuadrados(plado1, plado2, plado3)		<b>Versión:</b> 2.0
<b>Descripción:</b> Calcular el área 3 de cuadrados e imprimir el mayor		
Variables	Tipo de variable	
l1	Float	
l2	Float	
l3	Float	
cu1	Float	
cu2	Float	
cu3	Float	

**Código:**

```
function areaCuadrados(plado1, plado2, plado3){
  let l1 = plado1;
  let l2 = plado2;
  let l3 = plado3;

  let cu1 = l1 * l1;
  let cu2 = l2 * l2;
  let cu3 = l3 * l3;

  if(l1 == null || l2 == null || l3 == null){
    return "No hay valores"
  }else{
    if(cu1 == cu2 || cu1 == cu2 || cu2 == cu3){
      return "El área de los cuadrados son iguales";
    }
    else if(cu1 > cu2 && cu1 > cu3){
      return "El área mayor es la del primer cuadrado: " + cu1 + " cm\u00b2";
    }
    else if(cu2 > cu1 && cu2 > cu3){
      return "El área mayor es la del segundo cuadrado: " + cu2 + " cm\u00b2";
    }
    else{
      return "El área mayor es la del tercer cuadrado: " + cu3 + " cm\u00b2";
    }
  }
}
```



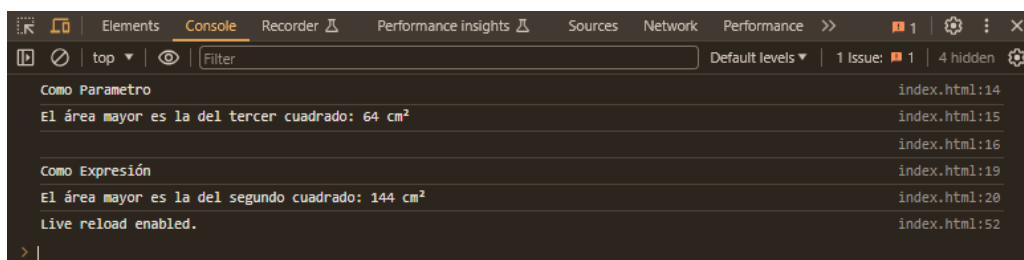
# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: const areaCuadradosExpresion = function(pLado1, pLado2, pLado3)		Versión: 3.0
Descripción: Calcular el área 3 de cuadrados e imprimir el mayor		
Variables	Tipo de variable	
l1	Float	
l2	Float	
l3	Float	
cu1	Float	
cu2	Float	
cu3	Float	
Código:		
<pre>const areaCuadradosExpresion = function(pLado1, pLado2, pLado3){     let l1 = pLado1;     let l2 = pLado2;     let l3 = pLado3;      let cu1 = l1 * l1;     let cu2 = l2 * l2;     let cu3 = l3 * l3;      if(l1 == null    l2 == null    l3 == null){         return "No hay valores"     }else{         if(cu1 == cu2    cu1 == cu2    cu2 == cu3){             return "El área de los cuadrados son iguales";         }         else if(cu1 &gt; cu2 &amp;&amp; cu1 &gt; cu3){             return "El área mayor es la del primer cuadrado: " + cu1 + " cm\u00b2";         }         else if(cu2 &gt; cu1 &amp;&amp; cu2 &gt; cu3){             return "El área mayor es la del segundo cuadrado: " + cu2 + " cm\u00b2";         }         else{             return "El área mayor es la del tercer cuadrado: " + cu3 + " cm\u00b2";         }     } }</pre>		

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### 14 – Mayor de Tres Edades

Nombre de la función: function mayorEdadUno(pFechaN1, pYearActual1)		Versión: 2.0
Descripción: Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades		
Variable	Tipo de variable	
fechaN1	Int	
yearActual	Int	
edad1	Int	
mensaje	String	
Código:		
<pre>function mayorEdadUno(pFechaN1, pYearActual1){     let fechaN1 = pFechaN1;     let yearActual1 = pYearActual1;     let edad1 = yearActual1 - fechaN1;     let mensaje = (edad1 &gt; 17) ? "Es Mayor de edad" : "Es Menor de edad";      return { texto: mensaje, edad: edad1 } }</pre>		

<b>Nombre de la función:</b> function mayorEdadDos(pFechaN2, pYearActual2)		<b>Versión:</b> 2.0
<b>Descripción:</b> Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades		
Variable	Tipo de variable	
fechaN2	Int	
yearActual	Int	
edad2	Int	
mensaje	String	
Código:		
<pre>function mayorEdadDos(pFechaN2, pYearActual2){     let fechaN2 = pFechaN2;     let yearActual2 = pYearActual2;     let edad2 = yearActual2 - fechaN2;     let mensaje = (edad2 &gt; 17) ? "Es Mayor de edad" : "Es Menor de edad";      return { texto: mensaje, edad: edad2 } }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: function mayorEdadTres(pFechaN3, pYearActual3)		Versión: 2.0
Descripción: Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades		
Variable	Tipo de variable	
fechaN3	Int	
yearActual	Int	
edad3	Int	
mensaje	String	
-Código:		
<pre>function mayorEdadTres(pFechaN3, pYearActual3){     let fechaN3 = pFechaN3;     let yearActual3 = pYearActual3;     let edad3 = yearActual3 - fechaN3;     let mensaje = (edad3 &gt; 17) ? "Es Mayor de edad" : "Es Menor de edad";      return { texto: mensaje, edad: edad3 } }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `function promedioEdades(pFechaN1, pYearActual1, pFechaN2, pYearActual2, pFechaN3, pYearActual3)`

Versión: 2.0

**Descripción:** Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades

Variable	Tipo de variable
edad1	Int
edad2	Int
edad3	Int
prom	Float

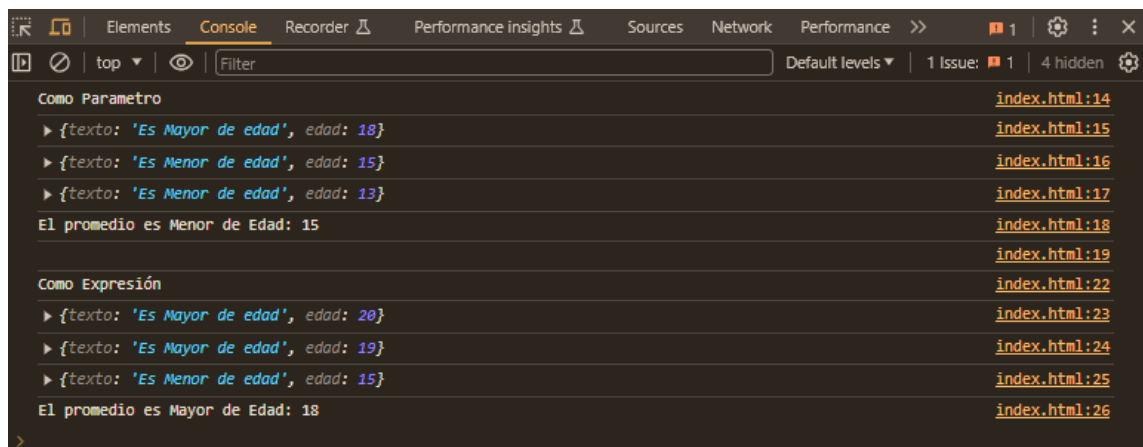
Código:

```
function promedioEdades(pFechaN1, pYearActual1, pFechaN2, pYearActual2, pFechaN3, pYearActual3){
  let edad1 = mayorEdadUno(pFechaN1, pYearActual1).edad;
  let edad2 = mayorEdadDos(pFechaN2, pYearActual2).edad;
  let edad3 = mayorEdadTres(pFechaN3, pYearActual3).edad;

  let prom = (edad1 + edad2 + edad3) / 3;

  if(prom > 17){
    return "El promedio es Mayor de Edad: " + prom.toFixed(0)
  }else{
    return "El promedio es Menor de Edad: " + prom.toFixed(0)
  }
}
```

## PRUEBA DE ESCRITORIO







# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** `const mayorUnoEdadExpresion = function(pFechaN1, pYearActual1)` **Versión: 3.0**

**Descripción:** Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades

Variable	Tipo de variable
fechaN1	Int
yearActual	Int
edad1	Int
mensaje	String

Código:

```
const mayorEdadUnoExpresion = function(pFechaN1, pYearActual1){  
    let fechaN1 = pFechaN1;  
    let yearActual1 = pYearActual1;  
    let edad1 = yearActual1 - fechaN1;  
    let mensaje = (edad1 > 17) ? "Es Mayor de edad" : "Es Menor de edad";  
    return { texto: mensaje, edad: edad1 }  
}
```

**Nombre de la función:** `const mayorEdadDosExpresion = function(pFechaN2, pYearActual2)` **Versión: 3.0**

**Descripción:** Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades

Variable	Tipo de variable
fechaN2	Int
yearActual	Int
edad2	Int
mensaje	String

Código:

```
const mayorEdadDosExpresion = function(pFechaN2, pYearActual2){  
    let fechaN2 = pFechaN2;  
    let yearActual2 = pYearActual2;  
    let edad2 = yearActual2 - fechaN2;  
    let mensaje = (edad2 > 17) ? "Es Mayor de edad" : "Es Menor de edad";  
    return { texto: mensaje, edad: edad2 }  
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** `const mayorUnoEdadTresExpresion = function(pFechaN1, pYearActual1)`

**Versión:** 3.0

**Descripción:** Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades

Variable

Tipo de variable

fechaN3

Int

yearActual

Int

edad3

Int

mensaje

String

Código:

```
const mayorEdadTresExpresion = function(pFechaN3, pYearActual3){  
    let fechaN3 = pFechaN3;  
    let yearActual3 = pYearActual3;  
    let edad3 = yearActual3 - fechaN3;  
    let mensaje = (edad3 > 17) ? "Es Mayor de edad" : "Es Menor de edad";  
    return { texto: mensaje, edad: edad3 }  
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** `const promedioEdades = function(pFechaN1, pYearActual1, pFechaN2, pYearActual2, pFechaN3, pYearActual3)`

**Versión:** 3.0

**Descripción:** Calcular la edad de una persona e imprimir si es mayor o menor de edad y calcular el promedio de edades

Variable

Tipo de variable

edad1	Int
edad2	Int
edad3	Int
prom	Float

Código:

```
const promedioEdadesExpresion = function(pFechaN1, pYearActual1, pFechaN2, pYearActual2, pFechaN3, pYearActual3){  
    let edad1 = mayorEdadUnoExpresion(pFechaN1, pYearActual1).edad;  
    let edad2 = mayorEdadDosExpresion(pFechaN2, pYearActual2).edad;  
    let edad3 = mayorEdadTresExpresion(pFechaN3, pYearActual3).edad;  
  
    let prom = (edad1 + edad2 + edad3) / 3;  
  
    if(prom > 17){  
        return "El promedio es Mayor de Edad: " + prom.toFixed(0)  
    }else{  
        return "El promedio es Menor de Edad: " + prom.toFixed(0)  
    }  
}
```

## PRUEBA DE ESCRITORIO

Como Parametro	
▶ {texto: 'Es Mayor de edad', edad: 18}	<a href="#">index.html:14</a>
▶ {texto: 'Es Mayor de edad', edad: 15}	<a href="#">index.html:15</a>
▶ {texto: 'Es Menor de edad', edad: 16}	<a href="#">index.html:16</a>
▶ {texto: 'Es Menor de edad', edad: 13}	<a href="#">index.html:17</a>
El promedio es Menor de Edad: 15	<a href="#">index.html:18</a>
	<a href="#">index.html:19</a>
Como Expresión	
▶ {texto: 'Es Mayor de edad', edad: 20}	<a href="#">index.html:22</a>
▶ {texto: 'Es Mayor de edad', edad: 19}	<a href="#">index.html:23</a>
▶ {texto: 'Es Menor de edad', edad: 15}	<a href="#">index.html:24</a>
El promedio es Mayor de Edad: 18	<a href="#">index.html:25</a>
	<a href="#">index.html:26</a>



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### 15 – Sueldo Persona 2

Nombre de la función: function <u>calcularSueldo</u> (pValorD, pDiasT)		Versión: 2.0
Descripción: Calcular el sueldo de una persona todo incluido		
Variable	Tipo Variable	
valorD	Float	
diasT	Int	
salario	Float	
Codigo: <div><pre>function calcularSueldo(pValorD, pDiasT){     let valorD = pValorD;     let diasT = pDiasT;     let sueldo     sueldo = valorD * diasT;     return sueldo; }</pre></div>		

Nombre de la función: calcularSubsidioTransporte(pSalarioM)		Versión: 2.0
Descripción: Calcular el sueldo de una persona todo incluido		
Variable	Tipo Variable	
salarioM	Float	
subsidioT	Float	
Codigo: <pre>function calcularSubsidioTransporte(pValorD, pDiasT, pSalarioM){     let salarioM = pSalarioM     let sueldoCalculado = calcularSueldo(pValorD, pDiasT)     let subsidioT     // let subsidioT = calcularSueldo(43333, 30) &lt; salarioM * 2 ? 114000 : 0;     if(sueldoCalculado &lt; salarioM * 2){         subsidioT = 114000;     }else{         subsidioT = 0;     }     return subsidioT }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** `function calcularPensionI(pSueldo)`

**Versión:** 2.0

**Descripción:**

Calcular el sueldo de una persona todo incluido

Variable	Tipo Variable
pen	Float

Código:

```
function calcularPension(pValorD, pDiasT){  
    let pen;  
    pen = calcularSueldo(pValorD, pDiasT) * 0.16;  
    return pen;  
}
```

**Nombre de la función:** `function calcularSalud(pSueldo)`

**Versión:** 2.0

**Descripción:**

Calcular el sueldo de una persona todo incluido

Variable	Tipo Variable
salud	Float

Código:

```
function calcularSalud(pValorD, pDiasT){  
    let salud;  
    salud = calcularSueldo(pValorD, pDiasT) * 0.12;  
    return salud;  
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: function <u>calcularARL</u> (pSueldo)		Versión: 2.0
<b>Descripción:</b> Calcular el sueldo de una persona todo incluido		
Variable	Tipo Variable	
arl	Float	
Codigo:		
<pre>function calcularARL(pValorD, pDiasT){     let arl;     arl = calcularSueldo(pValorD, pDiasT) * 0.052;     return arl; }</pre>		

Nombre de la función: calcularRetencion(pSalarioM)		Versión: 2.0
Descripción: Calcular el sueldo de una persona todo incluido		
Variable	Tipo Variable	
salarioM	Float	
retencion	Float	
Codigo:		
<pre>function calcularRetencion(pValorD, pDiasT, pSalarioM){     let salarioM = pSalarioM     let sueldoCalculado = calcularSueldo(pValorD, pDiasT)     let retencion     // let retencion = calcularSueldo(43333, 30) &lt; salarioM * 4 ? 0 : sueldoP * 0.04     if(sueldoCalculado &lt; salarioM * 4){         retencion = 0;     }else{         retencion = sueldoCalculado * 0.04;     }     return retencion }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `calcularSueldoTotal()`

Versión: 1.0

#### Descripción:

Calcular el sueldo de una persona todo incluido

Variable	Tipo Variable
sueldoCalculado	Float
subsidioT	Float
pen	Float
salud	Float
arl	Float
retención	Float
desc	Float
sueldoTotal	Float

Código:

```
function calcularSueldoTotal(pValorD, pDiasT, pSalarioM) {  
    let sueldoCalculado = calcularSueldo(pValorD, pDiasT);  
    let subsidioT = calcularSubsidioTransporte(pValorD, pDiasT, pSalarioM);  
    let pen = calcularPension(pValorD, pDiasT);  
    let salud = calcularSalud(pValorD, pDiasT);  
    let arl = calcularARL(pValorD, pDiasT);  
    let retencion = calcularRetencion(pValorD, pDiasT, pSalarioM);  
    let desc = pen + salud + arl;  
    let sueldoTotal = ((sueldoCalculado + subsidioT) - (retencion + desc)).toFixed(0);  
    return sueldoTotal;  
}
```

## PRUEBA DE ESCRITORIO

The screenshot shows a web browser console with two test cases. The first test case, labeled 'Como Parametro', shows the calculation of the total salary for a person with a base salary of \$129,999.00, a transport subsidy of \$114,000, a pension of \$207,998.4, health insurance of \$155,998.8, and ARL of \$67,599.48. The total salary calculated is \$982,393. The second test case, labeled 'Como Expresión', shows the calculation of the total salary for a person with a base salary of \$540,000.00, a transport subsidy of \$0, a pension of \$86,400, health insurance of \$64,800, and ARL of \$28,800. The total salary calculated is \$339,120.00.

```
Como Parametro index.html:16  
El sueldo de la persona antes de procesos era de: $1299990 index.html:17  
Al subsidio de Transporte se le dio un valor de: $114000 index.html:18  
Pensión: $207998.4 index.html:19  
Salud: $155998.8 index.html:20  
ARL: $67599.48 index.html:21  
Se le hizo una retencion de: $0 index.html:22  
El sueldo total de la persona es: $982393 index.html:23  
  
Como Expresión index.html:24  
Como Parametro index.html:30  
El sueldo de la persona antes de procesos era de: $540000 index.html:31  
Al subsidio de Transporte se le dio un valor de: $0 index.html:32  
Pensión: $86400 index.html:33  
Salud: $64800 index.html:34  
ARL: $28800 index.html:35  
Se le hizo una retencion de: $216000 index.html:36  
El sueldo total de la persona es: $3391200 index.html:37  
index.html:38  
index.html:39
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

<b>Nombre de la función:</b> const calcularSueldoExpresion = function(pValorD, pDiasT)		<b>Versión:</b> 3.0
<b>Descripción:</b> Calcular el sueldo de una persona todo incluido		
Variable	Tipo de Variables	
valorD	Float	
diasT	Int	
Salario	Float	
<b>Código:</b> <pre>const calcularSueldoExpresion = function(pValorD, pDiasT){     let valorD = pValorD;     let diasT = pDiasT;     let sueldo     sueldo = valorD * diasT;     return sueldo; }</pre>		

<b>Nombre de la función:</b> const calcularSubsidioTransporteExpresion = function(pValorD, pDiasT)		<b>Versión:</b> 3.0
<b>Descripción:</b> Calcular el sueldo de una persona todo incluido		
Variable	Tipo de Variable	
salarioM	Float	
subsidioT	Float	
<b>Código:</b> <pre>const calcularSubsidioTransporteExpresion = function(pValorD, pDiasT, pSalarioM){     let salarioM = pSalarioM     let sueldoCalculado = calcularSueldo(pValorD, pDiasT)     let subsidioT     // let subsidioT = calcularSueldoExpresion(180000, 30) &lt; salarioM * 2 ? 114000 : 0;     if(sueldoCalculado &lt; salarioM * 2){         subsidioT = 114000;     }else{         subsidioT = 0;     }     return subsidioT }</pre>		





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: const calcularPensionExpresion = function(pSueldo)		Versión: 3.0
Descripción: Calcular el sueldo de una persona todo incluido		
Variable	Tipo de Variables	
pen	Float	
Código:		
<pre>const calcularPensionExpresion = function(pValorD, pDiasT){     let pen;     pen = calcularSueldo(pValorD, pDiasT) * 0.16;     return pen; }</pre>		

Nombre de la función: const calcularSaludExpresion = function(pSueldo)		Versión: 3.0
Descripción: Calcular el sueldo de una persona todo incluido		
Variable	Tipo de Variables	
salud	Float	
Código:		
<pre>const calcularSaludExpresion = function(pValorD, pDiasT){   let salud;   salud = calcularSueldo(pValorD, pDiasT) * 0.12;   return salud; }</pre>		



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** const calcularARLEspression = function(pSuelto)

**Versión:** 3.0

**Descripción:**

Calcular el sueldo de una persona todo incluido

Variable	Tipo de Variables
arl	Float

Código:

```
const calcularARLEspression = function(pValorD, pDiasT){  
    let arl;  
    arl = calcularSuelto(pValorD, pDiasT) * 0.052;  
    return arl;  
}
```

**Nombre de la función:** const calcularRetencionEspression = function(pSalarioM)

**Versión:** 3.0

**Descripción:**

Calcular el sueldo de una persona todo incluido

Variable	Tipo de Variables
salarioM	Float
retencion	Float

Código:

```
const calcularRetencionEspression = function(pValorD, pDiasT, pSalarioM){  
    let salarioM = pSalarioM  
    let sueldoCalculado = calcularSuelto(pValorD, pDiasT)  
    let retencion  
    // let retencion = calcularSueltoEspression(180000, 30) < salarioM * 4 ? 0 : calcular  
    if(sueldoCalculado < salarioM * 4){  
        retencion = 0;  
    }else{  
        retencion = sueldoCalculado * 0.04;  
    }  
    return retencion  
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: const calcularSueldoTotalExpresion = function()		Versión: 1.0
Descripción:		
Calcular el sueldo de una persona todo incluido		
Variable	Tipo de Variables	
sueldoCalculado	Float	
subsidioT	Float	
pen	Float	
salud	Float	
arl	Float	
retención	Float	
desc	Float	
sueldoTotal	Float	
Código:		
<pre>const calcularSueldoTotalExpresion = function(pValorD, pDiasT, pSalarioM) {   let sueldoCalculado = calcularSueldoExpresion(pValorD, pDiasT);   let subsidioT = calcularSubsidioTransporteExpresion(pValorD, pDiasT, pSalarioM);   let pen = calcularPensionExpresion(pValorD, pDiasT);   let salud = calcularSaludExpresion(pValorD, pDiasT);   let arl = calcularARLExpresion(pValorD, pDiasT);   let retencion = calcularRetencionExpresion(pValorD, pDiasT, pSalarioM);   let desc = pen + salud + arl;   let sueldoTotal = ((sueldoCalculado + subsidioT) - (retencion + desc)).toFixed(0);   return sueldoTotal; }</pre>		

## PRUEBA DE ESCRITORIO

Elements	Console	Recorder	Performance insights	Sources	Network	Performance
top	Filter					
Default levels   1 Issue: 1   4 hidden						
Como Parametro						
El sueldo de la persona antes de procesos era de: \$1299990						<a href="#">index.html:16</a>
Al subsidio de Transporte se le dio un valor de: \$114000						<a href="#">index.html:17</a>
Pensión: \$207998.4						<a href="#">index.html:18</a>
Salud: \$155998.8						<a href="#">index.html:19</a>
ARL: \$67599.48						<a href="#">index.html:20</a>
Se le hizo una retencion de: \$0						<a href="#">index.html:21</a>
El sueldo total de la persona es: \$982393						<a href="#">index.html:22</a>
						<a href="#">index.html:23</a>
						<a href="#">index.html:24</a>
Como Expresión						
Como Parametro						<a href="#">index.html:30</a>
El sueldo de la persona antes de procesos era de: \$5400000						<a href="#">index.html:31</a>
Al subsidio de Transporte se le dio un valor de: \$0						<a href="#">index.html:32</a>
Pensión: \$864000						<a href="#">index.html:33</a>
Salud: \$648000						<a href="#">index.html:34</a>
ARL: \$280800						<a href="#">index.html:35</a>
Se le hizo una retencion de: \$216000						<a href="#">index.html:36</a>
El sueldo total de la persona es: \$3391200						<a href="#">index.html:37</a>
						<a href="#">index.html:38</a>
						<a href="#">index.html:39</a>



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### 16 – Promedio de Notas 3

Nombre de la función: function calcularSumaPorcentaje(pNota1, pNota2, pNota3,)		Versión: 2.0
Descripción: Calcular el promedio de notas y demás		
Variable	Tipo de variable	
nota1	Float	
nota2	Float	
nota3	Float	
porc1	Float	
porc2	Float	
porc3	Float	
sumPorc	Float	
Codigo: <pre>function calcularSumaPorcentaje(pNota1, pNota2, pNota3,){     let nota1 = pNota1     let nota2 = pNota2     let nota3 = pNota3      let porce1 = nota1 * 0.21     let porce2 = nota2 * 0.35     let porce3 = nota3 * 0.45      let sumPorc = porce1 + porce2 + porce3     return sumPorc }</pre>		



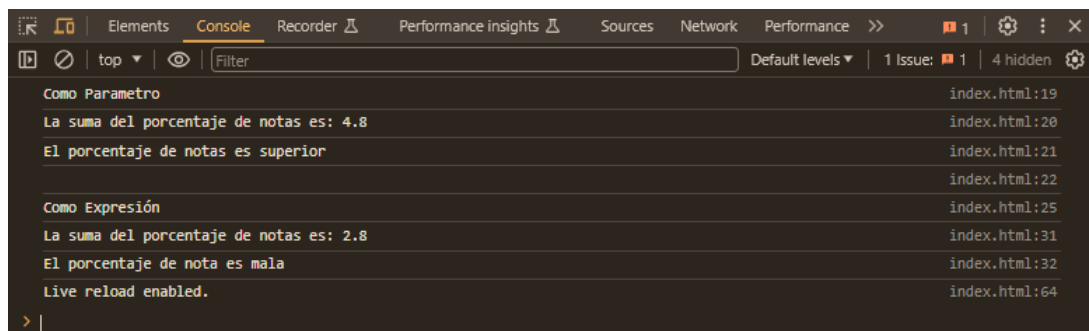
# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: function validarNota(pNota1, pNota2, pNota3)		Versión: 2.0
Descripción: Calcular el promedio de notas y demás		
Variable	Tipo de variable	
sumPorc	Float	
Codigo: <pre>function validarNota(pNota1, pNota2, pNota3){     let sumPorc = calcularSumaPorcentaje(pNota1, pNota2, pNota3)     if(sumPorc &gt; 4.5){         return "El porcentaje de notas es superior"     }else if (sumPorc &lt;= 4.5 &amp;&amp; sumPorc &gt; 3.5){         return "El porcentaje de nota es buena"     }else if (sumPorc &lt;= 3.5 &amp;&amp; sumPorc &gt;= 3.0){         return "El porcentaje de nota es media"     }else{         return "El porcentaje de nota es mala"     } }</pre>		

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `const calcularSumaPorcentajeExpresion = function(pNota1, pNota2, pNota3)`

Versión: 3.0

#### Descripción:

Calcular el promedio de notas y demás

Variable	Tipo de variable
nota1	Float
nota2	Float
nota3	Float
porc1	Float
porc2	Float
porc3	Float
sumPorc	Float

Código:

```
const calcularSumaPorcentajeExpresion = function(pNota1, pNota2, pNota3,){  
    let nota1 = pNota1  
    let nota2 = pNota2  
    let nota3 = pNota3  
  
    let porce1 = nota1 * 0.21  
    let porce2 = nota2 * 0.35  
    let porce3 = nota3 * 0.45  
  
    let sumPorc = porce1 + porce2 + porce3  
    return sumPorc  
}
```



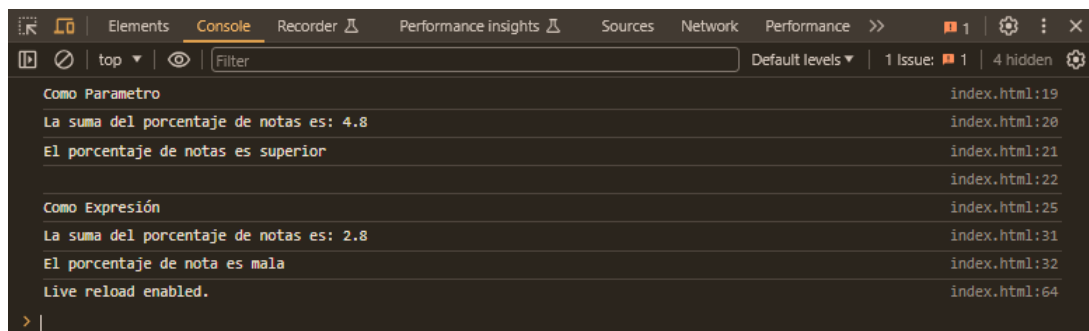
# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: const validarNotaExpresion = function(pNota1, pNota2, pNota3)		Versión: 3.0
Descripción: Calcular el promedio de notas y demás		
Variable	Tipo de variable	
sumPorc	Float	
Código:		
<pre>const validarNotaExpresion = function(pNota1, pNota2, pNota3){     let sumPorc = calcularSumaPorcentajeExpresion(pNota1, pNota2, pNota3)     if(sumPorc &gt; 4.5){         return "El porcentaje de notas es superior"     }else if (sumPorc &lt;= 4.5 &amp;&amp; sumPorc &gt; 3.5){         return "El porcentaje de nota es buena"     }else if (sumPorc &lt;= 3.5 &amp;&amp; sumPorc &gt;= 3.0){         return "El porcentaje de nota es media"     }else{         return "El porcentaje de nota es mala"     } }</pre>		

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

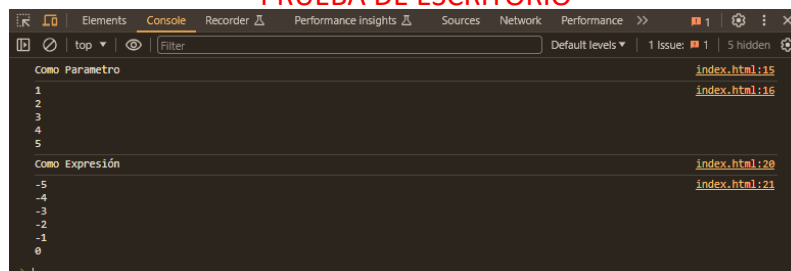
## CICLO WHILE

### 17 – Contar al 5

Nombre de la función: function contador(pNumero)		Versión: 2.0
Descripción: Imprime los números del 1 al 5		
Variable	Tipo de variable	
numero	Int	
resultado	String	
contar	Int	
Código:	<pre>function contador(pNumero){   let numero = pNumero   let contar = 0;   let resultado = ""   while(contar &lt; numero){     contar = contar + 1;     resultado += `\${contar}\n`   }   return resultado }</pre>	

Nombre de la función: const contadorExpresion = function(pNumero){		Versión: 3.0
Descripción: Imprimir los números del 1 al 5		
Variable	Tipo de variable	
numero	Int	
resultado	String	
contar	Int	
Código:		
<pre>const contadorExpresion = function(pNumero){     let numero = pNumero     let contar = -6;     let resultado = ""     while(contar &lt; numero){         contar = contar + 1;         resultado += `\${contar}\n`     }     return resultado }</pre>		

## PRUEBA DE ESCRITORIO







# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

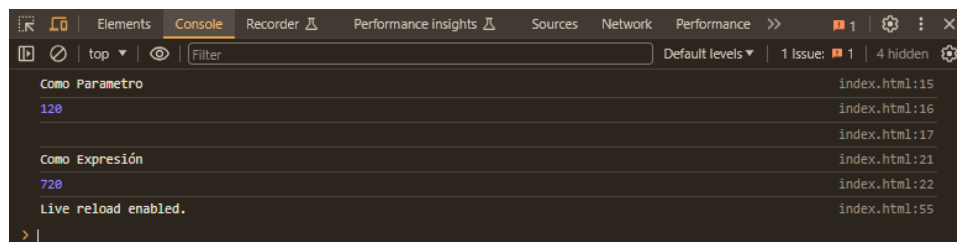
### Funciones JS

#### 18 – Factorial de 5

Nombre de la función: function factorial(pNumero)		Versión: 2.0
<b>Descripción:</b> Imprimir la factorial de 5		
Variable	Tipo de variable	
numero	Int	
fac	Int	
resultado	String	
contar	Int	
<b>Código:</b>		
<pre>function factorial(pNumero){     let numero = pNumero     let fac = 1     let contar = 0;     while(contar &lt; numero){         contar = contar + 1;         fac = fac * contar     }     return fac }</pre>		

Nombre de la función: const factorialExpresion = function(pNumero)		Versión: 3.0
Descripción: Imprimir la factorial de 5		
Variable	Tipo de variable	
numero	Int	
fac	Int	
resultado	String	
contar	Int	
Código:		
<pre>const factorialExpresion = function(pNumero){     let numero = pNumero     let fac = 1     let contar = 0;     while(contar &lt; numero){         contar = contar + 1;         fac = fac * contar     }     return fac }</pre>		

#### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

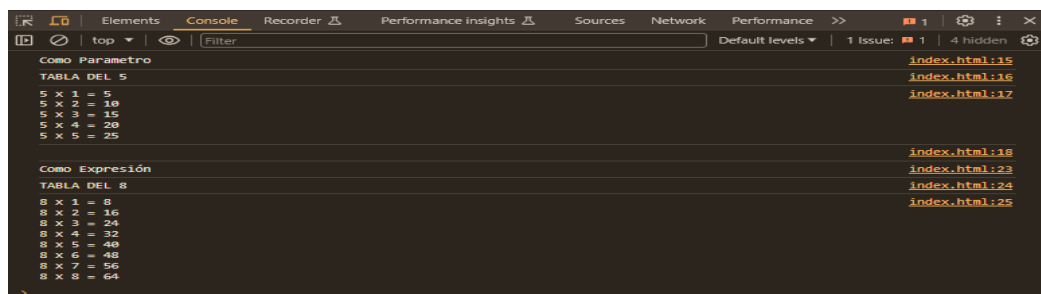
### Funciones JS

#### 19 – Tabla del 5

Nombre de la función: function tabla(pNumero)		Versión: 2.0
<b>Descripción:</b> Imprime la tabla del 5		
Variable	Tipo de variable	
numero	Int	
mul	Int	
resultado	String	
contar	Int	
Código:	<pre>function tabla(pNumero){   let numero = pNumero   let contar = 0;   let mul = 1   let resultado = ""   while(contar &lt; numero){     contar = contar + 1;     mul = numero * contar     resultado += `\${numero} x \${contar} = \${mul} \n`   }   return resultado }</pre>	

Nombre de la función: const tablaExpresion = function(pNumero){		Versión: 3.0
<b>Descripción:</b> Imprime la tabla del 5		
Variable	Tipo de variable	
numero	Int	
mul	Int	
resultado	String	
contar	Int	
Código:	<pre>const tablaExpresion = function(pNumero){   let numero = pNumero   let contar = 0;   let mul = 1   let resultado = ""   while(contar &lt; numero){     contar = contar + 1;     mul = numero * contar     resultado += `\${numero} x \${contar} = \${mul} \n`   }   return resultado }</pre>	

#### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

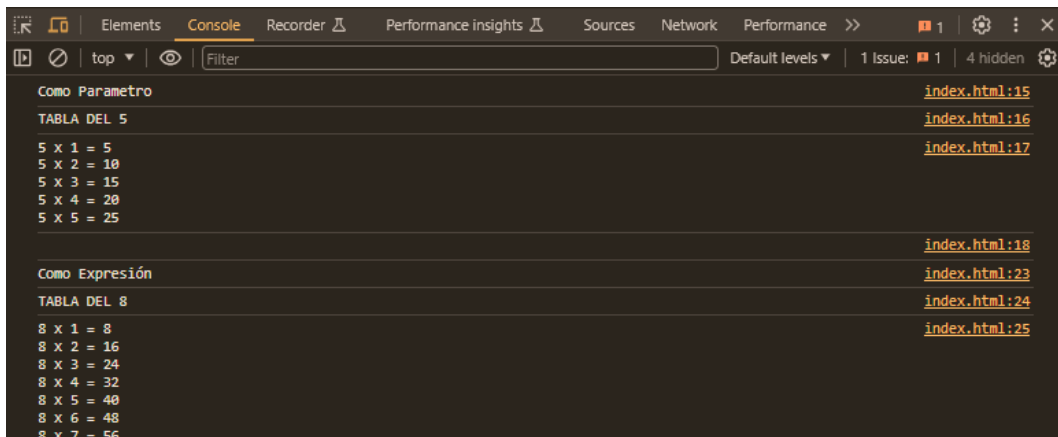
## Ficha

### Funciones JS

## 20 – Tabla del 9 par e impar

Nombre de la función: function tabla(pNumero)		Versión: 2.0
Descripción: Imprimir la tabla del 9 con los números pares e impares		
Variable	Tipo de variable	
numero	Int	
mul	Int	
resultado	String	
contar	Int	
Código:		
<pre>function tabla(pNumero){     let numero = pNumero     let contar = 0;     let mul = 1     resultado = ""     while(contar &lt; 5){         contar = contar + 1;         mul = numero * contar         if(mul % 2 == 0){             resultado += `\${numero} x \${contar} = \${mul} Es Par \n`         }else{             resultado += `\${numero} x \${contar} = \${mul} Es Impar \n`         }     }     return resultado }</pre>		

## PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: `const tablaExpresion = function(pNumero)`

Versión: 3.0

#### Descripción:

Imprimir la tabla del 9 con los números pares e impares

Variable	Tipo de variable
numero	Int
mul	Int
resultado	String
contar	Int

#### Código:

```
const tablaExpresion = function(pNumero){
    let numero = pNumero
    let contar = 0;
    let mul = 1
    resultado = ""
    while(contar < 5){
        contar = contar + 1;
        mul = numero * contar
        if(mul % 2 == 0){
            resultado += `${numero} x ${contar} = ${mul} Es Par \n`
        }else{
            resultado += `${numero} x ${contar} = ${mul} Es Impar \n`
        }
    }
    return resultado
}
```

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

## 26 – Tablas del 1 al 5

Nombre de la función: function tablas(pNumero, pTabla)		Versión: 2.0
<b>Descripción:</b> Imprimir las cinco primeras tablas, sacar cantidad de pares e impares e imprimir buzz para par y bass impar		
Variable	Tipo de variable	
numero	Int	
tabla	Int	
impar	Int	
par	Int	
mul	Int	
result	Int	
contar	Int	

Código:

```
function tablas(pNumero, pTabla){
    let numero = pNumero
    let tabla = pTabla;
    let contar = 0;
    let impar = 0
    let par = 0
    let mul
    let resul
    let resultado = ""

    while (contar < tabla) {
        contar = contar + 1;
        console.log("Tabla del " + contar)
        mul = 0;
        while (mul < numero) {
            mul = mul + 1;
            resul = contar * mul;
            if(resul % 2 == 0){
                resultado += `${contar} x ${mul} = ${resul} Buzz \n`
                // console.log(contar + " x " + mul + " = " + resul + " Buzz");
                par = par + 1
            }else{
                resultado += `${contar} x ${mul} = ${resul} Bass \n`
                // console.log(contar + " x " + mul + " = " + resul + " Bass");
                impar = impar + 1
            }
        }
        console.log("\n")
    }
    // return {par, impar}
    return `${resultado}La Cantidad de Pares es ${par}\nLa cantidad de Impares es ${impar}`
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

**Nombre de la función:** const tablasExpresion = function(pNumero, pTabla)

**Versión:** 3.0

**Descripción:**

Imprimir las cinco primeras tablas, sacar cantidad de pares e impares e imprimir buzz para par y bass impar

Variable

Tipo de variable

numero

Int

tabla

Int

impar

Int

par

Int

mul

Int

result

Int

contar

Int

Código:

```
const tablasExpresion = function(pNumero, pTabla){
    let numero = pNumero
    let tabla = pTabla;
    let contar = 0;
    let impar = 0
    let par = 0
    let mul
    let resul

    while (contar < tabla) {
        contar = contar + 1;
        console.log("Tabla del " + contar)
        mul = 0;
        while (mul < numero) {
            mul = mul + 1;
            resul = contar * mul;
            if(resul % 2 == 0){
                console.log(contar + " x " + mul + " = " + resul + " Buzz");
                par = par + 1
            }else{
                console.log(contar + " x " + mul + " = " + resul + " Bass");
                impar = impar + 1
            }
        }
        console.log("\n")
    }
    return {par, impar}
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### PRUEBA DE ESCRITORIO

```
Como Parametro                                index.html:16
Tabla del 1                                    tablas.js:19
1 x 1 = 1 Bass                                tablas.js:28
1 x 2 = 2 Buzz                                tablas.js:25
1 x 3 = 3 Bass                                tablas.js:28
1 x 4 = 4 Buzz                                tablas.js:25
1 x 5 = 5 Bass                                tablas.js:28
                                                tablas.js:33
Tabla del 2                                    tablas.js:19
2 x 1 = 2 Buzz                                tablas.js:25
2 x 2 = 4 Buzz                                tablas.js:25
2 x 3 = 6 Buzz                                tablas.js:25
2 x 4 = 8 Buzz                                tablas.js:25
2 x 5 = 10 Buzz                               tablas.js:25
                                                tablas.js:33
Tabla del 3                                    tablas.js:19
3 x 1 = 3 Bass                                tablas.js:28
3 x 2 = 6 Buzz                                tablas.js:25
3 x 3 = 9 Bass                                tablas.js:28
3 x 4 = 12 Buzz                               tablas.js:25
3 x 5 = 15 Bass                               tablas.js:28
```

```
Tabla del 4                                    tablas.js:19
4 x 1 = 4 Buzz                                tablas.js:25
4 x 2 = 8 Buzz                                tablas.js:25
4 x 3 = 12 Buzz                               tablas.js:25
4 x 4 = 16 Buzz                               tablas.js:25
4 x 5 = 20 Buzz                               tablas.js:25
                                                tablas.js:33
Tabla del 5                                    tablas.js:19
5 x 1 = 5 Bass                                tablas.js:28
5 x 2 = 10 Buzz                               tablas.js:25
5 x 3 = 15 Bass                               tablas.js:28
5 x 4 = 20 Buzz                               tablas.js:25
5 x 5 = 25 Bass                               tablas.js:28
                                                tablas.js:33
▶ {par: 16, impar: 9}                          index.html:17
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

## CICLO FOR

### 21 – Contar al 5

Nombre de la función: function contador(pNumero)		Versión: 2.0
<b>Descripción:</b> Imprimir los números del 1 al 5		
Variable	Tipo de variable	
numero	Int	
resultado	String	
contar	Int	
Código:	<pre>function contador(pNumero){     let numero = pNumero;     let resultado = "";     for(let contador = 1; contador &lt;= numero; contador++){         resultado += `\${contador}\n`;     }     return resultado; }</pre>	

Nombre de la función: const contadorExpresion = function(pNumero)		Versión: 3.0
<b>Descripción:</b> Imprimir los números del 1 al 5		
Variable	Tipo de variable	
numero	Int	
resultado	String	
contar	Int	
Código:	<pre>const contadorExpresion = function(pNumero){     let numero = pNumero;     let resultado = "";     for(let contador = -5; contador &lt;= numero; contador++){         resultado += `\${contador}\n`;     }     return resultado; }</pre>	

### PRUEBA DE ESCRITORIO

```
Como Parametro index.html:15
1
2
3
4
5
Como Expresión index.html:20
-5
-4
-3
-2
-1
0
```





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

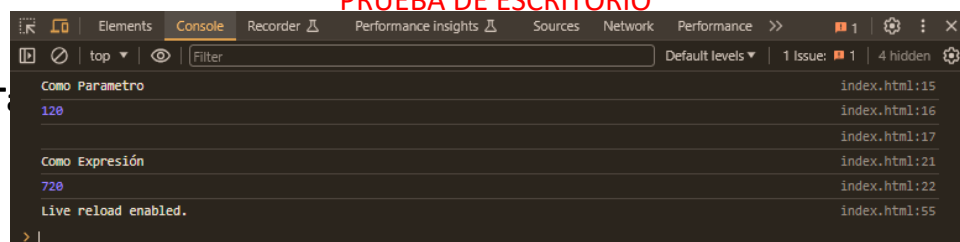
## 22 – Factorial de 5

Nombre de la función: function factorial(pNumero)		Versión: 2.0
<b>Descripción:</b> Imprimir la factorial de 5		
Variable	Tipo de variable	
numero	Int	
fac	Int	
resultado	String	
contar	Int	
Código:	<pre>function factorial(pNumero){     let numero = pNumero     let fac = 1     for(let contar = 1; contar &lt;= numero; contar++){         fac = fac * contar;     }     return fac }</pre>	

Nombre de la función: const factorialExpresion = function(pNumero){		Versión: 3.0
<b>Descripción:</b> Imprime la factorial de 5		
Variable	Tipo de variable	
numero	Int	
fac	Int	
resultado	String	
contar	Int	
Código:	<pre>const factorialExpresion = function(pNumero){     let numero = pNumero     let fac = 1     for(let contar = 1; contar &lt;= numero; contar++){         fac = fac * contar;     }     return fac }</pre>	

## PRUEBA DE ESCRITORIO

## 23 – T





# Tecnólogo en Análisis y Desarrollo de Software

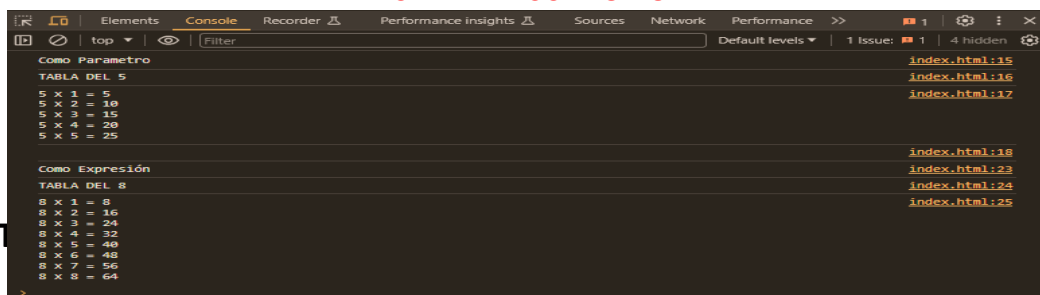
## Ficha

### Funciones JS

Nombre de la función: function tabla(pNumero)		Versión: 2.0
<b>Descripción:</b> Imprime la tabla del 5		
Variable	Tipo de variable	
numero	Int	
mul	Int	
resultado	String	
contar	Int	
Código:	<pre>function tabla(pNumero){     let numero = pNumero     let mul = 1     let resultado = ""     for(let contar = 1; contar &lt;= numero; contar++){         mul = numero * contar;         resultado += `\${numero} x \${contar} = \${mul} \n`     }     return resultado }</pre>	

Nombre de la función: const tablaExpresion = function(pNumero)		Versión: 3.0
<b>Descripción:</b> Imprime la tabla del 5		
Variable	Tipo de variable	
numero	Int	
mul	Int	
resultado	String	
contar	Int	
Código:	<pre>const tablaExpresion = function(pNumero){     let numero = pNumero     let mul = 1     let resultado = ""     for(let contar = 1; contar &lt;= numero; contar++){         mul = numero * contar;         resultado += `\${numero} x \${contar} = \${mul} \n`     }     return resultado }</pre>	

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

Nombre de la función: **function tabla(pNumero)**

Versión: **2.0**

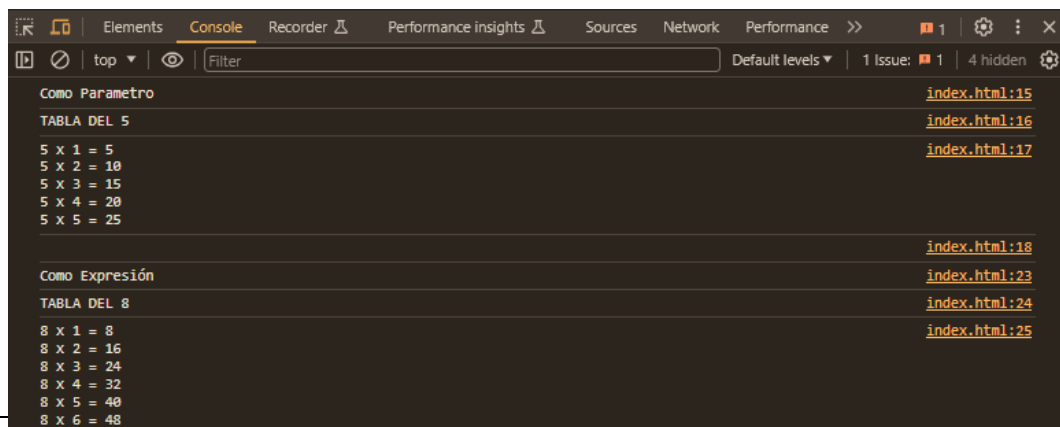
Descripción: Imprimir la tabla del 9 con los números pares e impares

Variable	Tipo de variable
numero	Int
mul	Int
resultado	String
contar	Int

Código:

```
function tabla(pNumero){
    let numero = pNumero
    let contar = 0;
    let mul = 1
    let resultado = ""
    for(contar = 1; contar <= 5; contar++){
        mul = numero * contar
        if(mul % 2 == 0){
            resultado += `${numero} x ${contar} = ${mul} Es Par \n`
        }else{
            resultado += `${numero} x ${contar} = ${mul} Es Impar \n`
        }
    }
    return resultado
}
```

### PRUEBA DE ESCRITORIO



Nombre de la función: **const tablaExpresión = function(pNumero){**

Versión: **3.0**



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

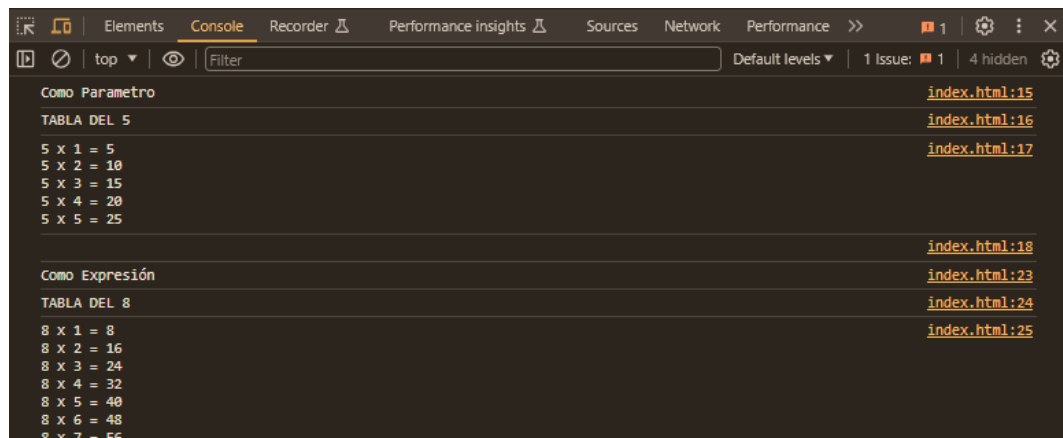
**Descripción:** Imprimir la tabla del 9 con los números pares e impares

Variable	Tipo de variable
numero	Int
mul	Int
resultado	String
contar	Int

Código:

```
const tablaExpresion = function(pNumero){
    let numero = pNumero
    let contar = 0;
    let mul = 1
    resultado = ""
    for(contar = 1; contar <= 5; contar++){
        mul = numero * contar
        if(mul % 2 == 0){
            resultado += `${numero} x ${contar} = ${mul} Es Par \n`
        }else{
            resultado += `${numero} x ${contar} = ${mul} Es Impar \n`
        }
    }
    return resultado
}
```

### PRUEBA DE ESCRITORIO





# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

#### 25 – Tablas del 1 al 5

Nombre de la función: function tablas(pNumero, pTabla)		Versión: 2.0
Descripción: Imprimir las cinco primeras tablas, sacar cantidad de pares e impares e imprimir buzz para par y bass impar		
Variable	Tipo de variable	
numero	Int	
tabla	Int	
impar	Int	
par	Int	
mul	Int	
result	Int	
contar	Int	

Código:

```
function tablas(pNumero, pTabla){
    let numero = pNumero
    let tabla = pTabla;
    let contar = 0;
    let impar = 0
    let par = 0
    let mul
    let resul
    let indicador = ""
    let resultado = ""

    for(contar = 1; contar <= tabla; contar++){
        mul = 0;
        // console.log("Tabla del " + contar)
        for(mul = 1; mul <= numero; mul++){
            resul = contar * mul;
            if(resul % 2 == 0){
                resultado += `${contar} x ${mul} = ${resul} Buzz \n`
                // console.log(contar + " x " + mul + " = " + resul + " Buzz")
                par = par + 1
            }else{
                resultado += `${contar} x ${mul} = ${resul} Bass \n`
                // console.log(contar + " x " + mul + " = " + resul + " Bass")
                impar = impar + 1
            }
        }
        // console.log("\n")
    }
    // return {par, impar}
    return `${resultado}La Cantidad de Pares es ${par}\nLa cantidad de Impares es ${impar} `
}
```



# Tecnólogo en Análisis y Desarrollo de Software

## Ficha

### Funciones JS

<b>Nombre de la función:</b> const tablasExpresion = function(pNumero, pTabla)		<b>Versión:</b> 3.0
<b>Descripción:</b> Imprimir las cinco primeras tablas, sacar cantidad de pares e impares e imprimir buzz para par y bass impar		
Variable	Tipo de variable	
numero	Int	
tabla	Int	
impar	Int	
par	Int	
mul	Int	
result	Int	
contar	Int	
Código:	<pre>const tablasExpresion = function(pNumero, pTabla){     let numero = pNumero     let tabla = pTabla;     let contar = 0;     let impar = 0     let par = 0     let mul     let resul      for(contar = 1; contar &lt;= tabla; contar++){         mul = 0;         console.log("Tabla del " + contar)         for(mul = 1; mul &lt;= numero; mul++){             resul = contar * mul;             if(resul % 2 == 0){                 console.log(contar + " x " + mul + " = " + resul + " Buzz")                 par = par + 1             }else{                 console.log(contar + " x " + mul + " = " + resul + " Bass")                 impar = impar + 1             }         }         console.log("\n")     }     return {par, impar} }</pre>	

## PRUEBA DE ESCRITORIO

Como Parametro	<a href="#">index.html:16</a>
Tabla del 1	<a href="#">tablas.js:19</a>
1 x 1 = 1 Bass	<a href="#">tablas.js:28</a>
1 x 2 = 2 BUZZ	<a href="#">tablas.js:25</a>
1 x 3 = 3 Bass	<a href="#">tablas.js:28</a>
1 x 4 = 4 BUZZ	<a href="#">tablas.js:25</a>
1 x 5 = 5 Bass	<a href="#">tablas.js:28</a>
	<a href="#">tablas.js:33</a>
Tabla del 2	<a href="#">tablas.js:19</a>
2 x 1 = 2 BUZZ	<a href="#">tablas.js:25</a>
2 x 2 = 4 BUZZ	<a href="#">tablas.js:25</a>
2 x 3 = 6 BUZZ	<a href="#">tablas.js:25</a>
2 x 4 = 8 BUZZ	<a href="#">tablas.js:25</a>
2 x 5 = 10 BUZZ	<a href="#">tablas.js:25</a>