



Tecnólogo en Análisis y Desarrollo de
Software
Ficha 2900177

Funciones JS

Manual JS

Funciones

María Isabel Tovar Pastrana

ADSO – 2900177



Tecnólogo en Análisis y Desarrollo de
Software
Ficha 2900177
Funciones JS

SALUDO

Nombre de la función: saludo()		Versión: 1.0
Descripción: Función que saluda		
saludar	Tipo de variable: Alfanumérico	
Código: <pre>// Con parámetros function saludo(psaludo){ let saludar=psaludo; return saludar; }</pre>		

Nombre de la función: const saludoExp = function(psaludo)		Versión: 2.0
Descripción: Función que saluda		
saludar	Tipo de variable: String	
Código: <pre>// Como expresión const saludoExp=function(psaludo){ let saludar=psaludo; return saludar; }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS



OPERACIONES

Nombre de la función: suma(pnumUno, pnumDos)		Versión: 1.0
Descripción: Función que suma dos números		
numUno	Tipo de variable: int	
numDos	Tipo de variable: int	
sumar	Tipo de variable: int	
Código:		
<pre>let numUno; let numDos; // Con parámetros function suma(pnumUno, pnumDos){ let sumar; numUno = pnumUno; numDos = pnumDos; sumar = numUno + numDos; return sumar; }</pre>		

Nombre de la función: const sumaExp = function(pnumUno, pnumDos)		Versión: 2.0
Descripción: Función que suma dos números		
numUno	Tipo de variable: int	
numDos	Tipo de variable: int	
sumar	Tipo de variable: int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Como expresión
const sumaExp = function(pnumUno, pnumDos){
    let sumar;
    numUno = pnumUno;
    numDos = pnumDos;
    sumar = numUno + numDos;
    return sumar;
}
```



Nombre de la función: resta(pnumUno, pnumDos)		Versión: 1.0
Descripción:		
Función que resta dos números		
numUno	Tipo de variable: int	
numDos	Tipo de variable: int	
restar	Tipo de variable: int	
Código:		
<pre>function resta (pnumUno, pnumDos){ let restar; numUno = pnumUno; numDos = pnumDos; restar = numUno - numDos; return restar; }</pre>		

Nombre de la función: const restaExp = function(pnumUno, pnumDos)	Versión: 2.0
Descripción:	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Función que resta dos números

numUno	Tipo de variable: int
numDos	Tipo de variable: int
restar	Tipo de variable: int

Código:

```
const restaExp = function(pnumUno, pnumDos){  
  let restar;  
  numUno = pnumUno;  
  numDos = pnumDos;  
  restar = numUno - numDos;  
  return restar;  
}
```

Nombre de la función:

multiplicacion(pnumUno, pnumDos)

Versión: 1.0

Descripción:

Función que multiplica dos números

numUno	Tipo de variable: int
numDos	Tipo de variable: int
multiplicar	Tipo de variable: int

Código:

```
function multiplicacion (pnumUno, pnumDos){  
  let multiplicar;  
  numUno = pnumUno;  
  numDos = pnumDos;  
  multiplicar = numUno * numDos;  
  return multiplicar;  
}
```

Nombre de la función: const

multiplicacionExp= function(pnumUno, pnumDos)

Versión: 2.0

Descripción:

Función que multiplica dos números

numUno	Tipo de variable: int
numDos	Tipo de variable: int
multiplicar	Tipo de variable: int

Código:



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const multiplicacionExp = function(pnumUno, pnumDos){  
    let multiplicar;  
    numUno = pnumUno;  
    numDos = pnumDos;  
    multiplicar = numUno * numDos;  
    return multiplicar;  
}
```

Nombre de la función: division(pnumUno, pnumDos)		Versión: 1.0
Descripción: Función que divide dos números		
numUno	Tipo de variable: int	
numDos	Tipo de variable: int	
dividir	Tipo de variable: int	
Código: <pre>function division (pnumUno, pnumDos){ let dividir; numUno = pnumUno; numDos = pnumDos; dividir = numUno / numDos; return dividir; }</pre>		

Nombre de la función: const divisionExp= function(pnumUno, pnumDos)		Versión: 2.0
Descripción: Función que divide dos números		
numUno	Tipo de variable: int	
numDos	Tipo de variable: int	
dividir	Tipo de variable: int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const divisionExp = function(pnumUno, pnumDos){  
    let dividir;  
    numUno = pnumUno;  
    numDos = pnumDos;  
    dividir = numUno / numDos;  
    return dividir;  
}
```

Nombre de la función: operaciones(poprador , pnumUno , pnumDos)		Versión: 1.0
Descripción: Función que elige la operacion		
operador	Tipo de variable: alfanumérico	
numUno	Tipo de variable: int	
numDos	Tipo de variable: int	
Código:		
<pre>function operaciones (poprador, pnumUno, pnumDos){ let operador = poprador; numUno = pnumUno; numDos = pnumDos; if(operador == "suma"){ return suma(numUno, numDos); } else if(operador == "resta"){ return resta(numUno, numDos); } else if(operador == "multiplicacion"){ return multiplicacion(numUno, numDos); } else if(operador == "division"){ return division(numUno, numDos); } else{ return "Error!! no se reconoce el operador,"; } }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Nombre de la función: const operacionesExp= function(poperador,pnumUno, pnumDos)

Versión: 2.0

Descripción:

Función que elige la operación

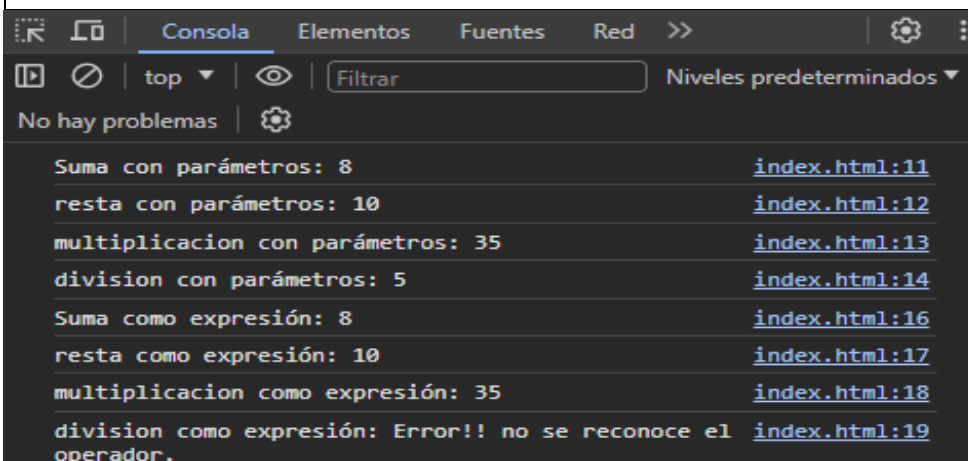
operador Tipo de variable: alfanumérico

numUno Tipo de variable: int

numDos Tipo de variable: int

Código:

```
const operacionesExp = function(poperador, pnumUno, pnumDos){
  let operador = poperador;
  numUno = pnumUno;
  numDos = pnumDos;
  if(operador == "suma"){
    return sumaExp(numUno, numDos);
  }
  else if(operador == "resta"){
    return restaExp(numUno, numDos);
  }
  else if(operador == "multiplicacion"){
    return multiplicacionExp(numUno, numDos);
  }
  else if(operador == "division"){
    return divisionExp(numUno, numDos);
  }
  else{
    return "Error!! no se reconoce el operador,";
  }
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

PORCENTAJE DE UN NÚMERO

Nombre de la función: porcentaje(pnum)		Versión: 1.0
Descripción: Función que calcula el porcentaje de un número		
numero	Tipo de variable: float	
resultado	Tipo de variable: float	
Código:		
<pre>let numero; let resultado; // Con parámetros function porcentaje(pnum){ numero = pnum; let resultado; resultado = numero /100; return resultado; }</pre>		

Nombre de la función: const porcentaje = function(pnum)		Versión: 2.0
Descripción: Función que calcula el porcentaje de un número		
numero	Tipo de variable: float	
resultado	Tipo de variable: float	
Código: <pre>// Como expresión const porcentajeExp = function(pnum){ numero = pnum; let resultado; resultado = numero /100; return resultado; }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
Consola  Elementos  Fuentes  Red  >>
top  Filtar  Niveles predeterminados
No hay problemas
Con parámetros: index.html:12
El porcentaje de: 6 Es: 0.06
Como expresión: index.html:13
El porcentaje de: 6 Es: 0.06
```

PROMEDIO DE TRES NOTAS

Nombre de la función: promedio(pnota1,pnota2,pnota3)		Versión: 1.0
Descripción: Función que calcula el porcentaje de tres notas y su promedio total		
nota1	Tipo de variable: float	
nota2	Tipo de variable: float	
nota3	Tipo de variable: float	
resultado	Tipo de variable: float	
Código: <pre>let nota1; let nota2; let nota3; // Con parámetros function promedio(pnota1,pnota2,pnota3){ nota1 = pnota1; nota2 = pnota2; nota3 = pnota3; let resultado; resultado = (nota1 + nota2 + nota3)/3; return resultado; }</pre>		
Nombre de la función: const promedioExp = function (pnota1,pnota2,pnota3)		Versión: 2.0
Descripción: Función que calcula el porcentaje de tres notas y su promedio total		
nota1	Tipo de variable: float	
nota2	Tipo de variable: float	
nota3	Tipo de variable: float	
resultado	Tipo de variable: float	



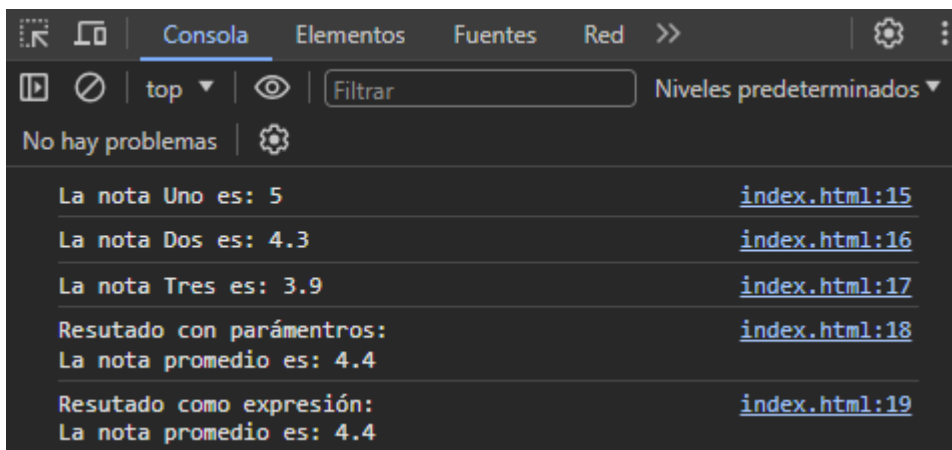
Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Código:

```
// Como expresión
const promedioExp = function(pnota1,pnota2,pnota3){
    nota1 = pnota1;
    nota2 = pnota2;
    nota3 = pnota3;
    let resultado;
    resultado = (nota1 + nota2 + nota3)/3;
    return resultado;
}
```



PORCENTAJE DE CADA NOTA Y PROMEDIO

Nombre de la función: porcNota(pnota,pporcentaje)		Versión: 1.0
Descripción: Función que calcula el porcentaje de tres notas, dependiendo su valor y su promedio total		
nota	Tipo de variable: float	
porcentaje	Tipo de variable: float	
resultado	Tipo de variable: float	
Código:		

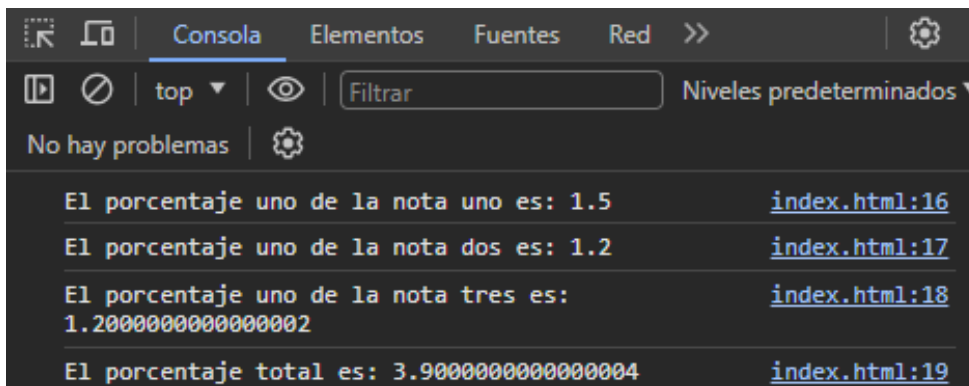


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
let nota;  
let porcentaje;  
  
// Con parámetros  
function porcNota(pnota,pporcentaje){  
    nota=pnota;  
    porcentaje=pporcentaje;  
    let resultado;  
    resultado=nota*porcentaje;  
    return resultado;  
}
```



Nombre de la función: const porcNotaExp = function(pnota,pporcentaje)		Versión: 2.0
Descripción: Función que calcula el porcentaje de tres notas, dependiendo su valor y su promedio total		
nota	Tipo de variable: float	
porcentaje	Tipo de variable: float	
resultado	Tipo de variable: float	
Código:		

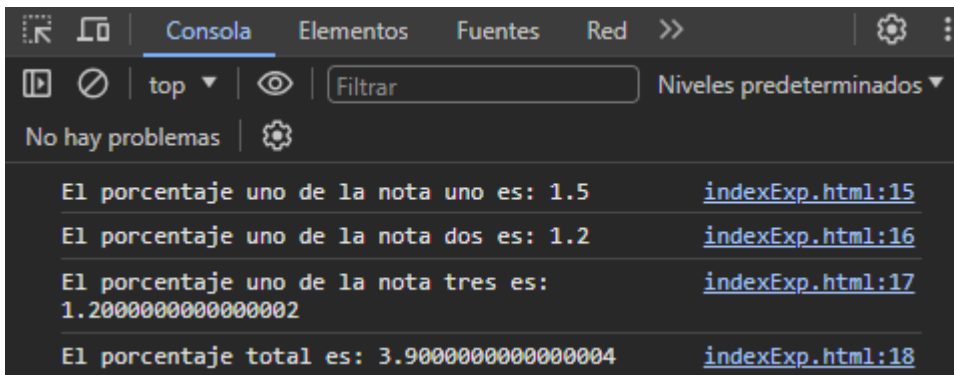


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Como expresión
const porcNotaExp = function(pnota,pporcentaje){
  nota=pnota;
  porcentaje=pporcentaje;
  let resultado;
  resultado=nota*porcentaje;
  return resultado;
}
```



ÁREA DE LAS FIGURAS GEOMÉTRICAS

Nombre de la función: areaC(pladoCuad)		Versión: 1.0
Descripción:		
Función que calcula el área del cuadrado		
ladoCuadrado	Tipo de variable: float	
cuadrado	Tipo de variable: float	
Código:		
<pre>// Con parámetros function areaC(pladoCuad){ let ladoCuadrado = pladoCuad; let cuadrado; cuadrado = ladoCuadrado*ladoCuadrado; return cuadrado ; }</pre>		

Nombre de la función: const areaCExp = function (pladoCuad)	Versión: 2.0
Descripción:	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Función que calcula el área del cuadrado

ladoCuadrado Tipo de variable: float

cuadrado Tipo de variable: float

Código:

```
// Como expresión
const areaCExp = function (pladoCua){
  let ladoCuadrado = pladoCua;
  let cuadrado;
  cuadrado = ladoCuadrado*ladoCuadrado;
  return cuadrado ;
}
```

Nombre de la función:

areaR(pbaseRec,palturaRec)

Versión: 1.0

Descripción:

Función que calcula el área del rectángulo

baseRectangulo Tipo de variable: float

alturaRectangulo Tipo de variable: float

rectangulo Tipo de variable: float

Código:

```
function areaR(pbaseRec,palturaRec){
  let baseRectangulo = pbaseRec;
  let alturaRectangulo = palturaRec;
  let rectangulo;
  rectangulo = (baseRectangulo*alturaRectangulo)/2;
  return rectangulo;
}
```

Nombre de la función: const areaRExp =
function(pbaseRec,palturaRec)

Versión: 2.0

Descripción:

Función que calcula el área del rectángulo

baseRectangulo Tipo de variable: float

alturaRectangulo Tipo de variable: float

rectangulo Tipo de variable: float

Código:



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const areaRExp = function(pbaseRec,palturaRec){  
    let baseRectangulo = pbaseRec;  
    let alturaRectangulo = palturaRec;  
    let rectangulo;  
    rectangulo = (baseRectangulo*alturaRectangulo)/2;  
    return rectangulo;  
}
```

Nombre de la función: areaT(pbaseTri,palturaTri)		Versión: 1.0
Descripción: Función que calcula el área del triangulo		
baseTriangulo	Tipo de variable: float	
alturaTriangulo	Tipo de variable: float	
truangulo	Tipo de variable: float	
Código:		
<pre>function areaT(pbaseTri,palturaTri){ let baseTriangulo = pbaseTri; let alturaTriangulo = palturaTri; let triangulo; triangulo = (baseTriangulo*alturaTriangulo)/2; return triangulo; }</pre>		

Nombre de la función: const areaTExp = function(pbaseTri,palturaTri)		Versión: 2.0
Descripción: Función que calcula el área del triangulo		
baseTriangulo	Tipo de variable: float	
alturaTriangulo	Tipo de variable: float	
truangulo	Tipo de variable: float	
Código:		

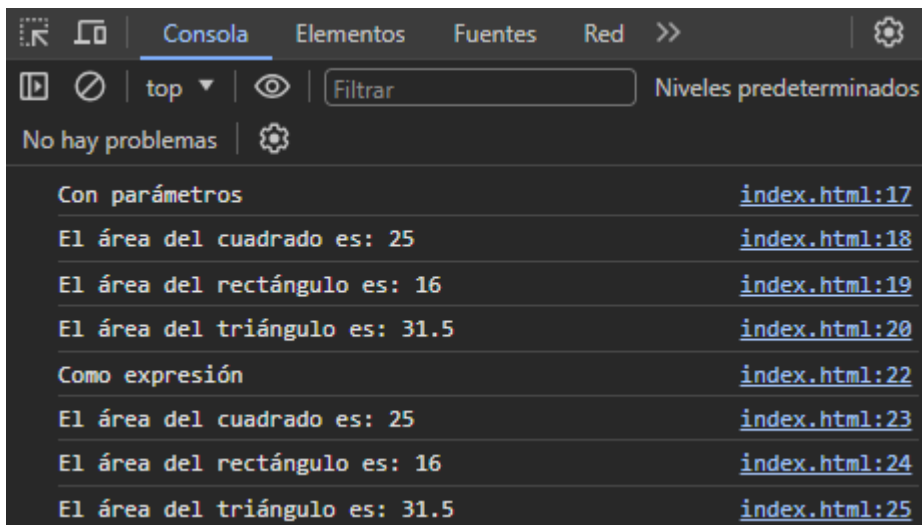


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const areaTExp = function(pbaseTri,palturaTri){  
    let baseTriangulo = pbaseTri;  
    let alturaTriangulo = palturaTri;  
    let triangulo;  
    triangulo = (baseTriangulo*alturaTriangulo)/2;  
    return triangulo;  
}
```



CALCULAR EL SALARIO DE UNA PERSONA

Nombre de la función: salario(pdias,pvalor)		Versión: 1.0
Descripción: Función que calcula el sueldo de una persona		
diasTrabaj	Tipo de variable: int	
valor	Tipo de variable: float	
sueldo	Tipo de variable: float	
Código:		
<pre>// Con parámetros function salario(pdias,pvalor){ let diasTrabaj = pdias; let valor = pvalor; let sueldo; sueldo = diasTrabaj*valor; return sueldo; }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Nombre de la función: const salarioExp = function(pdias,pvalor)		Versión: 2.0
Descripción:		
Función que calcula el sueldo de una persona		
diasTrabaj	Tipo de variable: int	
valor	Tipo de variable: float	
sueldo	Tipo de variable: float	
Código:		
<pre>// Como expresión const salarioExp = function(pdias,pvalor){ let diasTrabaj = pdias; let valor = pvalor; let sueldo; sueldo = diasTrabaj*valor; return sueldo; }</pre>		

Nombre de la función: salud(pdias,pvalor)		Versión: 1.0
Descripción: Función que calcula el valor de la salud		
sueldo	Tipo de variable: float	
saludT	Tipo de variable: float	
Código:		
<pre>function salud(pdias,pvalor){ let sueldo = salario(pdias,pvalor); let saludT; saludT=(sueldo*0.12); return saludT; }</pre>		

Nombre de la función: const saludExp = function(pdias,pvalor)		Versión: 2.0
Descripción: Función que calcula el valor de la salud		
sueldo	Tipo de variable: float	
saludT	Tipo de variable: float	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const saludExp = function(pdias,pvalor){  
    let sueldo = salario(pdias,pvalor);  
    let saludT;  
    saludT=(sueldo*0.12);  
    return saludT;  
}
```

Nombre de la función: pension(pdias,pvalor)		Versión: 1.0
Descripción:		
Función que calcula el valor de la pension		
sueldo	Tipo de variable: float	
pensionT	Tipo de variable: float	
Código:		
<pre>function pension(pdias,pvalor) { let sueldo = salario(pdias,pvalor); let pensionT; pensionT=(sueldo*0.16); return pensionT; }</pre>		

Nombre de la función: const pensionExp = function(pdias,pvalor)		Versión: 2.0
Descripción: Función que calcula el valor de la pension		
sueldo	Tipo de variable: float	
pensionT	Tipo de variable: float	
Código: <pre>const pensionExp = function(pdias,pvalor) { let sueldo = salario(pdias,pvalor); let pensionT; pensionT=(sueldo*0.16); return pensionT; }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Nombre de la función: arl(pdias,pvalor)		Versión: 1.0
Descripción:		
Función que calcula el valor del arl		
sueldo	Tipo de variable: float	
arlT	Tipo de variable: float	
Código:		
<pre>function arl(pdias,pvalor){ let sueldo = salario(pdias,pvalor); let arlT; arlT=(sueldo*0.052); return arlT; }</pre>		

Nombre de la función: const arlExp = function(pdias,pvalor)		Versión: 2.0
Descripción: Función que calcula el valor del arl		
sueldo	Tipo de variable: float	
arlT	Tipo de variable: float	
Código: <pre>const arlExp = function(pdias,pvalor){ let sueldo = salario(pdias,pvalor); let arlT; arlT=(sueldo*0.052); return arlT; }</pre>		

Nombre de la función: sueldoTotal(pdias,pvalor)		Versión: 1.0
Descripción: Función que calcula el valor total del sueldo		
sueldo	Tipo de variable: float	
saludT	Tipo de variable: float	
pensionT	Tipo de variable: float	
arlT	Tipo de variable: float	
descuento	Tipo de variable: float	
total	Tipo de variable: float	
Código:		

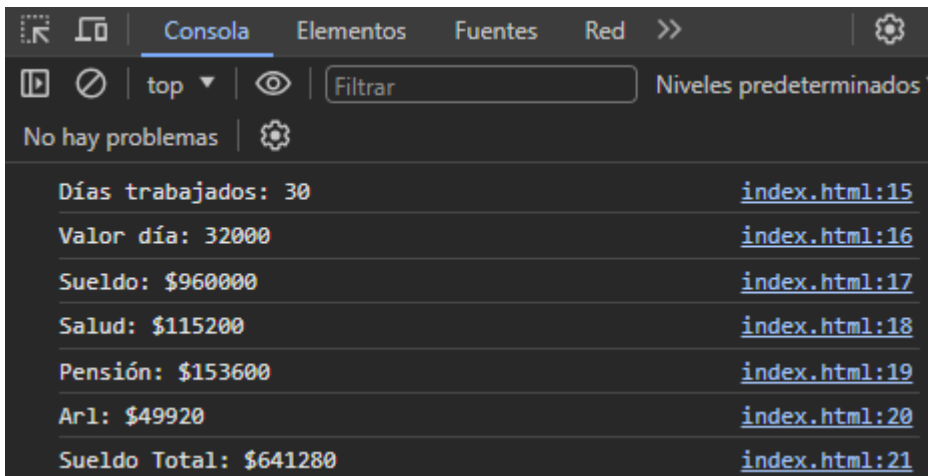


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
function sueldoTotal(pdias,pvalor) {  
    let sueldo = salario(pdias,pvalor);  
    let saludT = salud(sueldo);  
    let pensionT = pension(sueldo);  
    let arlT = arl(sueldo);  
    let total;  
    total=sueldo-(saludT+pensionT+arlT);  
    return total;  
}
```



Nombre de la función: sueldoTotalExp(pdias,pvalor)		Versión: 2.0
Descripción: Función que calcula el valor total del sueldo		
sueldo	Tipo de variable: float	
saludT	Tipo de variable: float	
pensionT	Tipo de variable: float	
arlT	Tipo de variable: float	
descuento	Tipo de variable: float	
total	Tipo de variable: float	
Código:		

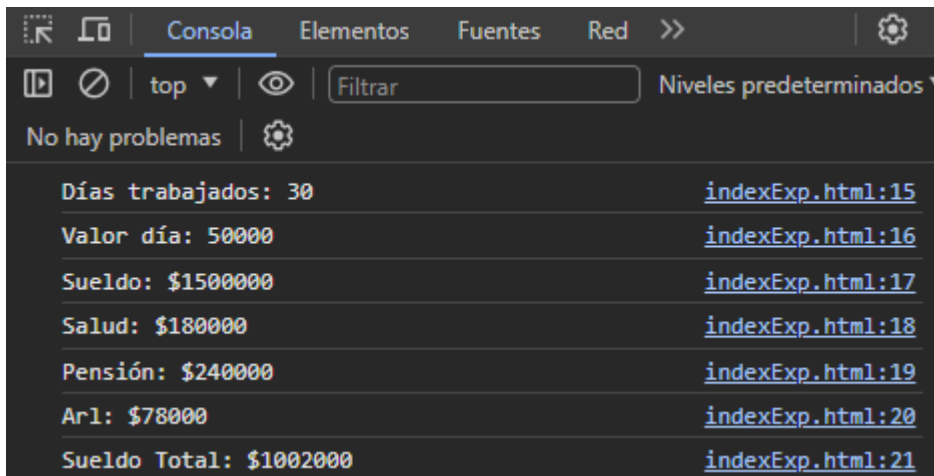


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const sueldoTotalExp = function(pdias,pvalor) {  
    let sueldo = salario(pdias,pvalor);  
    let saludT = salud(sueldo);  
    let pensionT = pension(sueldo);  
    let arlT = arl(sueldo);  
    let total;  
    total=sueldo-(saludT+pensionT+arlT);  
    return total;  
}
```



CON CONDICIONALES

Nombre de la función: condicionTrans(pdias,pvalor)		Versión: 1.0
Descripción: Función que dependiendo el valor del sueldo calcula el valor de transporte		
SMMLV	Tipo de variable: float	
sueldo	Tipo de variable: float	
subTransporte	Tipo de variable: float	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
function condicionTrans(pdias,pvalor){  
    let SMMLV = 1300000;  
    let sueldo = salario(pdias,pvalor);  
    let subTransporte;  
    if(sueldo<=2*SMMLV){  
        return subTransporte=114000;  
    }else{  
        return subTransporte=0;  
    }  
}
```

Nombre de la función: condicionTransExp(pdias,pvalor)		Versión: 2.0
Descripción: Función que dependiendo el valor del sueldo calcula el valor de transporte		
SMMLV	Tipo de variable: float	
sueldo	Tipo de variable: float	
subTransporte	Tipo de variable: float	
Código:		
<pre>const condicionTransExp = function (pdias,pvalor){ let SMMLV = 1300000; let sueldo = salario(pdias,pvalor); let subTransporte; if(sueldo<=2*SMMLV){ return subTransporte=114000; }else{ return subTransporte=0; } }</pre>		

Nombre de la función: condicionRetencion(pdias,pvalor)		Versión: 1.0
Descripción: Función que dependiendo el valor del sueldo calcula el valor de retencion		
SMMLV	Tipo de variable: float	
sueldo	Tipo de variable: float	
retencion	Tipo de variable: float	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Código:

```
function condicionRetencion(pdias,pvalor) {  
    let SMMLV = 1300000;  
    let sueldo = salario(pdias,pvalor);  
    let retencion;  
    if(sueldo>=4*SMMLV){  
        return retencion=sueldo*0.04;  
    }  
    else{  
        return retencion=0;  
    }  
}
```

Nombre de la función:

condicionRetencionExp(pdias,pvalor)

Versión: 2.0

Descripción:

Función que dependiendo el valor del sueldo calcula el valor de retencion

SMMLV Tipo de variable: float

sueldo Tipo de variable: float

retencion Tipo de variable: float

Código:

```
const condicionRetencionExp = function (pdias,pvalor) {  
    let SMMLV = 1300000;  
    let sueldo = salario(pdias,pvalor);  
    let retencion;  
    if(sueldo>=4*SMMLV){  
        return retencion=sueldo*0.04;  
    }  
    else{  
        return retencion=0;  
    }  
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Nombre de la función: descuentos(pdias,pvalor)		Versión: 1.0
Descripción: Función que calcula el valor de los descuentos		
saludT	Tipo de variable: float	
pensionT	Tipo de variable: float	
arlT	Tipo de variable: float	
retencion	Tipo de variable: float	
descuento	Tipo de variable: float	
Código: <pre>const descuentos = function(pdias,pvalor){ let saludT = salud(pdias,pvalor); let pensionT = pension(pdias,pvalor); let arlT = arl(pdias,pvalor); let retencion = condicionRetencion(pdias,pvalor); let descuento; descuento=saludT+pensionT+arlT+retencion; return descuento; }</pre>		

Nombre de la función: descuentosExp(psueldo)		Versión: 2.0
Descripción: Función que calcula el valor de los descuentos		
saludT	Tipo de variable: float	
pensionT	Tipo de variable: float	
arlT	Tipo de variable: float	
retencion	Tipo de variable: float	
descuento	Tipo de variable: float	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const descuentosExp = function (pdias,pvalor){  
    let saludT = salud(pdias,pvalor);  
    let pensionT = pension(pdias,pvalor);  
    let arlT = arl(pdias,pvalor);  
    let retencion = condicionRetencion(pdias,pvalor);  
    let descuento;  
    descuento=saludT+pensionT+arlT+retencion;  
    return descuento;  
}
```

Nombre de la función: sueldoTotal(pdias,pvalor)		Versión: 1.0
Descripción: Función que calcula el valor de total del sueldo		
sueldo	Tipo de variable: float	
descuento	Tipo de variable: float	
subTransporte	Tipo de variable: float	
total	Tipo de variable: float	
Código:		
<pre>function sueldoTotal(pdias,pvalor) { let sueldo = salario(pdias,pvalor); let descuento = descuentos(pdias,pvalor); let subTransporte = condicionTrans(pdias,pvalor); let total; total=(sueldo-descuento)+subTransporte; return total; }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
Consola Elementos Fuentes Red >>
top Filtar Niveles predeterminados
No hay problemas
Días trabajados: 30 index.html:15
Valor día: 32000 index.html:16
Sueldo: $960000 index.html:17
Salud: $115200 index.html:18
Pensión: $153600 index.html:19
Arl: $49920 index.html:20
Transporte: 114000 index.html:21
Retencion: 0 index.html:22
Descuentos: $318720 index.html:23
Sueldo Total: $755280 index.html:24
```

Nombre de la función: sueldoTotalExp(pdias,pvalor)		Versión: 2.0
Descripción: Función que calcula el valor de total del sueldo		
sueldo	Tipo de variable: float	
descuento	Tipo de variable: float	
subTransporte	Tipo de variable: float	
total	Tipo de variable: float	
Código:		

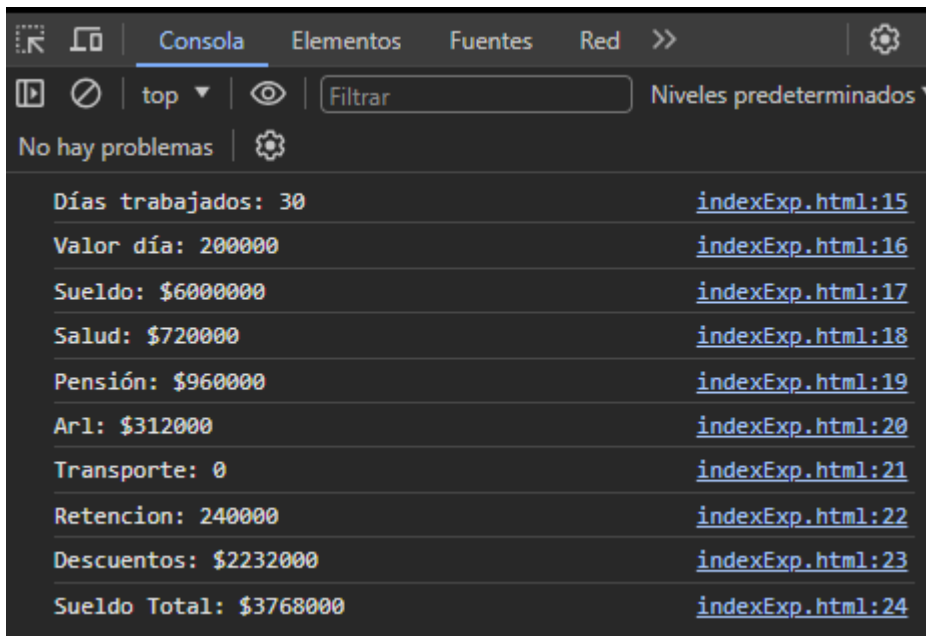


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
const sueldoTotalExp = function (pdias,pvalor) {  
    let sueldo = salario;  
    let descuento = descuentos(pdias,pvalor);  
    let subTransporte = condicionTrans(pdias,pvalor);  
    let total;  
    total=(sueldo-descuento)+subTransporte;  
    return total;  
}
```



NÚMERO MAYOR DE DOS NÚMEROS

Nombre de la función: numMayor(pnum1,pnum2)		Versión: 1.0
Descripción: Función que muestra el número mayor de dos números		
num1	Tipo de variable: int	
num2	Tipo de variable: int	
Código:		



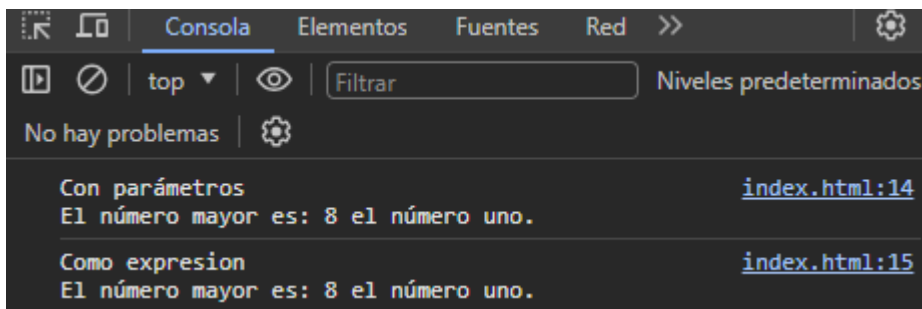
Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
let num1;
let num2;
// Con parámetros
function numMayor(pnum1,pnum2){
    num1 = pnum1;
    num2 = pnum2;
    if (num1 > num2) {
        return "El número mayor es: "+num1+ " el número uno.";
    } else {
        return "El número mayor es: "+num2+ " el número dos.";
    }
}
```

Nombre de la función: const numMayorExp = function(pnum1,pnum2)		Versión: 2.0
Descripción: Función que muestra el número mayor de dos números		
num1	Tipo de variable: int	
num2	Tipo de variable: int	
Código:		
<pre>const numMayorExp = function(pnum1,pnum2){ num1 = pnum1; num2 = pnum2; if (num1 > num2) { return "El número mayor es: "+num1+ " el número uno."; } else { return "El número mayor es: "+num2+ " el número dos."; } }</pre>		





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

NÚMERO MAYOR O IGUAL DE DOS NÚMEROS

Nombre de la función: numMayor(pnum1,pnum2)		Versión: 1.0
Descripción: Función que muestra el número mayor de dos números o si son iguales		
num1	Tipo de variable: int	
num2	Tipo de variable: int	
Código:		
<pre>let num1; let num2; // Con parámetros function numMayor(pnum1,pnum2){ num1 = pnum1; num2 = pnum2; if (num1 != num2) { if (num1 > num2) { return "El número mayor es: "+num1+ " el número uno."; } else { return "El número mayor es: "+num2+ " el número dos."; } }else{ return "Los números son iguales"; } }</pre>		

Nombre de la función: const numMayorExp = function(pnum1,pnum2)		Versión: 2.0
Descripción: Función que muestra el número mayor de dos números o si son iguales		
num1	Tipo de variable: int	
num2	Tipo de variable: int	
Código:		

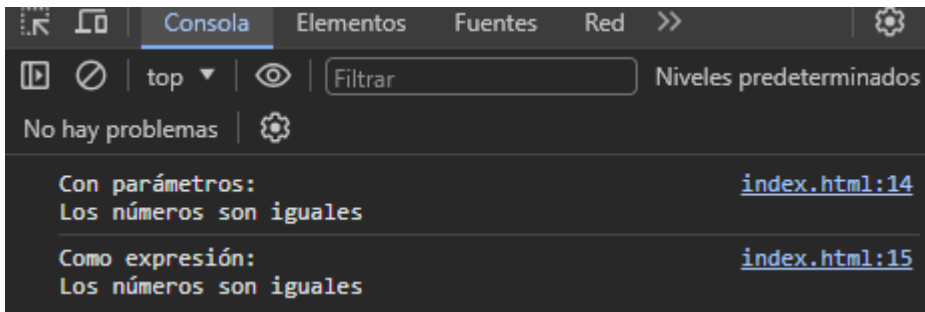


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Como expresion
const numMayorExp = function(pnum1,pnum2){
    num1 = pnum1;
    num2 = pnum2;
    if (num1 != num2) {
        if (num1 > num2) {
            return "El número mayor es: "+num1+ " el número uno.";
        } else {
            return "El número mayor es: "+num2+ " el número dos.";
        }
    }else{
        return "Los números son iguales";
    }
}
```



CALCULAR SI UNA PERSONA ES MAYOR DE EDAD

Nombre de la función: edad(pfechaNac)		Versión: 1.0
Descripción: Función que calcula si una persona es mayor de edad		
fechaAct	Tipo de variable: int	
fechaNac	Tipo de variable: int	
edad	Tipo de variable: int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Con parámetros
function edad(pfechaNac){
    let fechaAct = 2024;
    let fechaNac = pfechaNac;
    let edad;
    edad = fechaAct - fechaNac;
    if(edad>=18){
        return "Es mayor de edad, tiene "+edad+ " años.";
    }else{
        return "Es menor de edad, tiene "+edad+ " años.";
    }
}
```

Nombre de la función: const edadExp =
function(pfechaNac)

Versión: 2.0

Descripción:

Función que calcula si una persona es mayor de edad

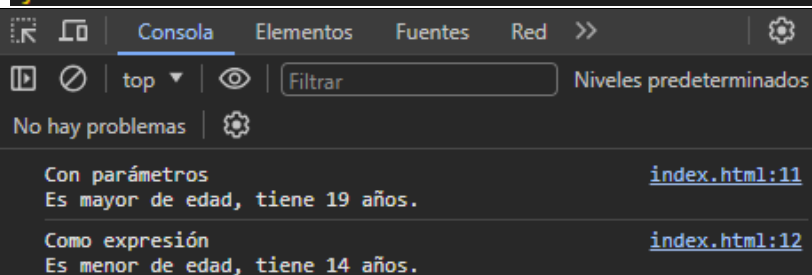
fechaAct Tipo de variable: int

fechaNac Tipo de variable: int

edad Tipo de variable: int

Código:

```
// Como expresion
const edadExp = function(pfechaNac) {
    let fechaAct = 2024;
    let fechaNac = pfechaNac;
    let edad;
    edad = fechaAct - fechaNac;
    if(edad>=18){
        return "Es mayor de edad, tiene "+edad+ " años.";
    }else{
        return "Es menor de edad, tiene "+edad+ " años.";
    }
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

NÚMERO MAYOR O IGUAL DE TRES NÚMEROS

Nombre de la función: numMayor(pnum1,pnum2,pnum3)		Versión: 1.0
Descripción: Función que muestra el número mayor de tres números o si son iguales		
num1	Tipo de variable: int	
num2	Tipo de variable: int	
num3	Tipo de variable: int	
Código: <pre>let num1; let num2; let num3; // Con parámetros function numMayor(pnum1,pnum2,pnum3){ num1 = pnum1; num2 = pnum2; num3 = pnum3; if (num1 != num2 && num2 != num3 && num1 != num3) { if (num1 > num2 && num1 > num3) { return "El número mayor es: "+num1+ " el número uno."; } else if (num2>num1 && num2>num3) { return "El número mayor es: "+num2+ " el número dos."; } else{ return "El número mayor es: "+num3 + " el número tres."; } }else{ return "Los números son iguales"; } }</pre>		

Nombre de la función: const numMayorExp = function(pnum1,pnum2,pnum3)		Versión: 2.0
Descripción: Función que muestra el número mayor de tres números o si son iguales		
num1	Tipo de variable: int	
num2	Tipo de variable: int	
num3	Tipo de variable: int	
Código:		

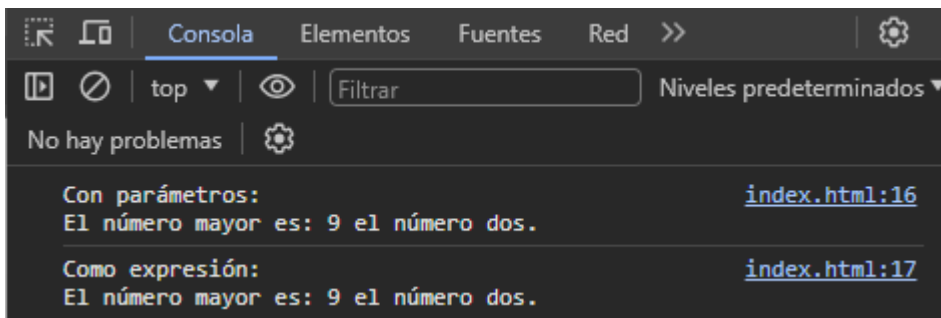


Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Como expresion
const numMayorExp = function(pnum1,pnum2,pnum3){
    num1 = pnum1;
    num2 = pnum2;
    num3 = pnum3;
    if (num1 != num2 && num2 != num3 && num1 != num3) {
        if (num1 > num2 && num1 > num3) {
            return "El número mayor es: "+num1+ " el número uno.";
        } else if (num2>num1 && num2>num3) {
            return "El número mayor es: "+num2+ " el número dos.";
        }
        else{
            return "El número mayor es: "+num3 + " el número tres.";
        }
    }else{
        return "Los números son iguales";
    }
}
```



ÁREA MAYOR O IGUAL DE TRES CUADRADOS

Nombre de la función: areaCuadrado(plado)		Versión: 1.0
Descripción: Función que calcula el área de tres cuadrados y decir cual es el mayor o si son iguales		
lado	Tipo de variable: float	
cuad	Tipo de variable: float	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Con parámetros
function areaCuadrado(plado){
    let lado = plado;
    let cuad;
    cuad = lado**2;
    return cuad ;
}
function cuadMayor(cuadUno,cuadDos,cuadTres) {
    if(cuadUno!=cuadDos && cuadDos!=cuadTres && cuadUno!=cuadTres){
        if(cuadUno>cuadDos && cuadUno>cuadTres){
            return ("El área del cuadrado uno "+cuadUno+" es mayor.");
        }
        else{
            if (cuadDos>cuadUno && cuadDos>cuadTres) {
                return ("El área del cuadrado dos "+cuadDos+" es mayor.");
            } else {
                return ("El área del cuadrado tres "+cuadTres+" es mayor.")
            }
        }
    }
    else{
        return ("Las tres áreas son iguales.");
    }
}
```

Nombre de la función: const areaCuadradoExp = function (plado)		Versión: 2.0
Descripción: Función que calcula el área de tres cuadrados y decir cual es el mayor o si son iguales		
lado	Tipo de variable: float	
cuad	Tipo de variable: float	
Código:		



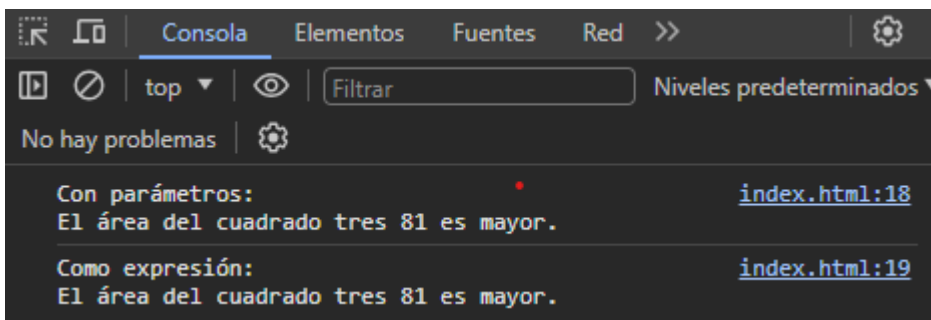
Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Como expresión
const areaCuadradoExp = function (plado){
  let lado = plado;
  let cuad;
  cuad = lado**2;
  return cuad ;
}

const cuadMayorExp = function (cuadUno,cuadDos,cuadTres) {
  if(cuadUno!=cuadDos && cuadDos!=cuadTres && cuadUno!=cuadTres){
    if(cuadUno>cuadDos && cuadUno>cuadTres){
      return ("El área del cuadrado uno "+cuadUno+" es mayor.");
    }
    else{
      if (cuadDos>cuadUno && cuadDos>cuadTres) {
        return ("El área del cuadrado dos "+cuadDos+" es mayor.");
      } else {
        return ("El área del cuadrado tres "+cuadTres+" es mayor.")
      }
    }
  }
  else{
    return ("Las tres áreas son iguales.");
  }
}
```



CALCULAR LA EDAD DE TRES PERSONAS

Nombre de la función: edad(pfechaNac)	Versión: 1.0
Descripción: Función que calcula la edad de una persona	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

fechaAct	Tipo de variable: int
fechaNac	Tipo de variable: int
edad	Tipo de variable: int
Código:	
<pre>// Con parámetros function edad(pfechaNac){ let fechaAct = 2024; let fechaNac = pfechaNac; let edad; edad = fechaAct - fechaNac; return edad; }</pre>	

Nombre de la función: const edadExp = function (pfechaNac)		Versión: 2.0
Descripción:		
Función que calcula la edad de una persona		
fechaAct	Tipo de variable: int	
fechaNac	Tipo de variable: int	
edad	Tipo de variable: int	
Código:		
<pre>// Como expresión const edadExp = function (pfechaNac){ let fechaAct = 2024; let fechaNac = pfechaNac; let edad; edad = fechaAct - fechaNac; return edad; }</pre>		

Nombre de la función: mayor(edadp)		Versión: 1.0
Descripción:		
Función que calcula si una persona es mayor de edad		
edad	Tipo de variable: int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
function mayor(edadp){  
  let edad = edadp;  
  if(edad>=18){  
    return "Es mayor de edad, tiene "+edad+ " años.";  
  }else{  
    return "Es menor de edad, tiene "+edad+ " años.";  
  }  
}
```

Nombre de la función: mayorExp(edadp)		Versión: 2.0
Descripción:		
Función que calcula si una persona es mayor de edad		
edad	Tipo de variable: int	
Código:		
<pre>const mayorExp = function (edadp){ let edad = edadp; if(edad>=18){ return "Es mayor de edad, tiene "+edad+ " años."; }else{ return "Es menor de edad, tiene "+edad+ " años."; } }</pre>		

Nombre de la función: promedioEdad(psuma)		Versión: 1.0
Descripción: Función que calcula si el promedio de tres edades es mayor de edad		
suma	Tipo de variable: int	
promedio	Tipo de variable: int	
Código:		



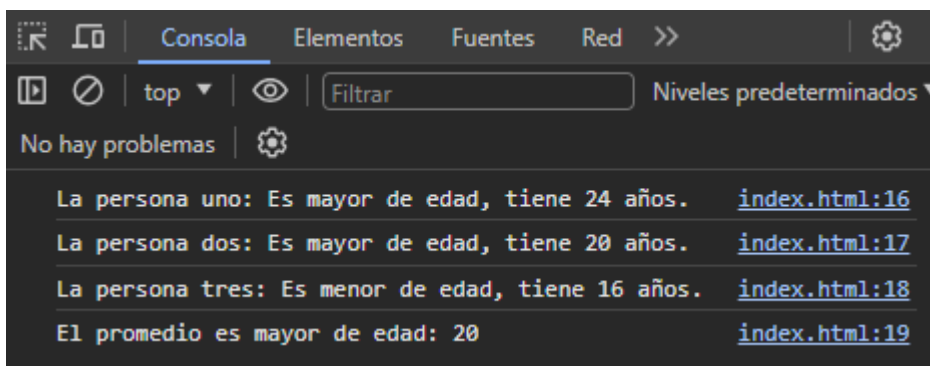
Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
function promedioEdad(psuma){  
  let suma = psuma;  
  let promedio = suma/3;  
  if(promedio>=18){  
    return ("El promedio es mayor de edad: "+promedio);  
  }else{  
    return ("El promedio es menor de edad: "+promedio);  
  }  
}
```

Nombre de la función: promedioEdadExp(psuma)		Versión: 2.0
Descripción: Función que calcula si el promedio de tres edades es mayor de edad		
suma	Tipo de variable: int	
promedio	Tipo de variable: int	
Código:		
<pre>const promedioEdadExp = function (psuma){ let suma = psuma; let promedio = suma/3; if(promedio>=18){ return ("El promedio es mayor de edad: "+promedio); }else{ return ("El promedio es menor de edad: "+promedio); } }</pre>		





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

CALCULAR EL PORCENTAJE DE TRES NOTAS

Nombre de la función: porcNota(pnota,pporcentaje)		Versión: 1.0
Descripción: Función que calcula el porcentaje de cada nota		
nota	Tipo de variable: float	
porcentaje	Tipo de variable: float	
resultado	Tipo de variable: float	
Código: <pre>// Con parámetros function porcNota(pnota,pporcentaje){ let nota=pnota; let porcentaje=pporcentaje; let resultado; resultado=nota*porcentaje; return resultado; }</pre>		

Nombre de la función: porcNotaExp(pnota,pporcentaje)		Versión: 2.0
Descripción: Función que calcula el porcentaje de cada nota		
nota	Tipo de variable: float	
porcentaje	Tipo de variable: float	
resultado	Tipo de variable: float	
Código:		
<pre>// Como expresión const porcNotaExp = function (pnota,pporcentaje) { let nota=pnota; let porcentaje=pporcentaje; let resultado; resultado=nota*porcentaje; return resultado; }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Nombre de la función:

condicionporcentaje(porcentajeTotalp)

Versión: 1.0

Descripción:

Función que muestra cual es el desempeño de cada nota

porcTotal

Tipo de variable: float

Código:

```
function condicionporcentaje(porcentajeTotalp){
  let porcTotal = porcentajeTotalp;
  if(porcTotal>4.5){
    return (" La nota promedio es Superior.");
  }
  else{
    if(porcTotal<=4.5 && porcTotal>3.5) {
      return (" La nota promedio es Buena.");
    }
    else{
      if(porcTotal<=3.5 && porcTotal>=3){
        return (" La nota promedio es Media.");
      }
      else{
        return (" La nota promedio es Mala.");
      }
    }
  }
}
```

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

```
Se tienen tres notas: index.html:20
Nota 1: 5
Nota 2: 4
Nota 3: 3

Los porcentajes son los siguientes: index.html:21
Porcentaje Nota 1: 1
Porcentaje Nota 2: 1.4
Porcentaje Nota 3: 1.35

Distribuyendo sus porcentajes su promedio es: 3.75 index.html:23
La nota promedio es Buena.
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Nombre de la función:
condicionporcentajeExp(porcentajeTotalp)

Versión: 2.0

Descripción:

Función que muestra cual es el desempeño de cada nota

porcTotal

Tipo de variable: float

Código:

```
const condicionporcentajeExp = function (porcentajeTotalp){  
  let porcTotal = porcentajeTotalp;  
  if(porcTotal>4.5){  
    return (" La nota promedio es Superior.");  
  }  
  else{  
    if(porcTotal<=4.5 && porcTotal>3.5) {  
      return (" La nota promedio es Buena.");  
    }  
    else{  
      if(porcTotal<=3.5 && porcTotal>=3){  
        return (" La nota promedio es Media.");  
      }  
      else{  
        return (" La nota promedio es Mala.");  
      }  
    }  
  }  
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
Consola Elementos Fuentes Red >>
top Filtar Niveles predeterminados
No hay problemas
Se tienen tres notas: indexExp.html:20
Nota 1: 3
Nota 2: 4
Nota 3: 5
Los porcentajes son los siguientes: indexExp.html:21
Porcentaje Nota 1: 0.6000000000000001
Porcentaje Nota 2: 1.4
Porcentaje Nota 3: 2.25
Distribuyendo sus porcentajes su promedio es: indexExp.html:23
4.25 La nota promedio es Buena.
```

CONTAR DE 1 A 5 CICLO WHILE

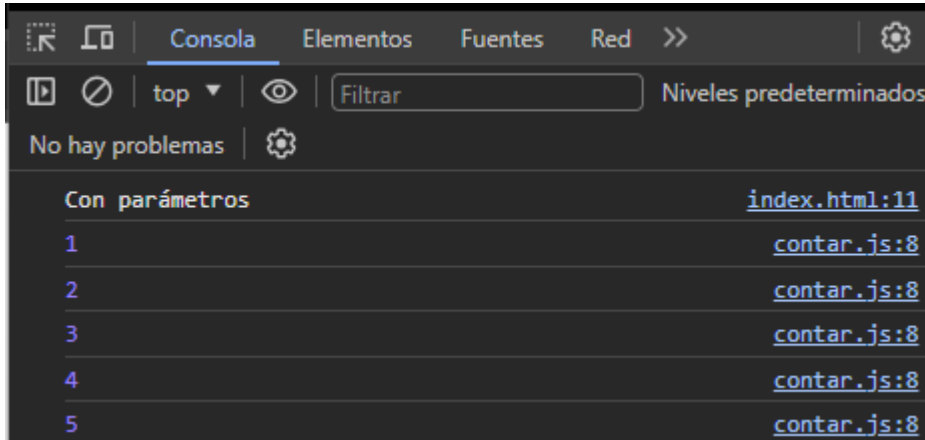
Nombre de la función: funcionContar(pnum)		Versión: 1.0
Descripción:		
Función que cuenta de 1 a 5 con el ciclo while		
num	Tipo de variable: int	
contar	Tipo de variable: int	
Código:		
<pre>// Con parámetros function funcionContar(pnum) { let contar = 0; let num = pnum; while (contar < num) { contar++; console.log(contar); } }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS



Nombre de la función:
funcionContarExp(pnum)

Versión: 2.0

Descripción:

Función que cuenta de 1 a 5 con el ciclo while

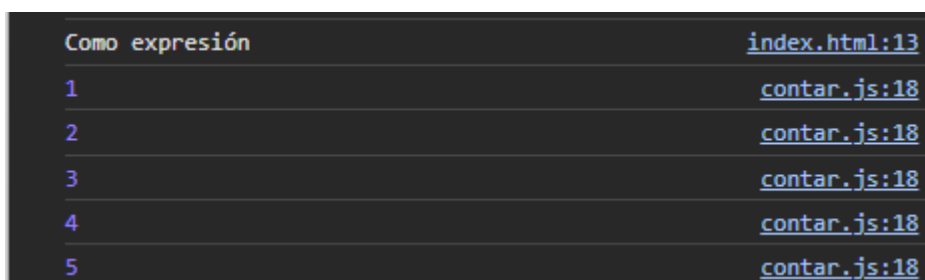
num Tipo de variable: int

contar Tipo de variable: int

Código:

```
// Como expresión
const funcionContarExp = function (pnum) {
  let contar = 0;
  let num = pnum;

  while (contar < num) {
    contar++;
    console.log(contar);
  }
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

FACTORIAL DE 5 CICLO WHILE

Nombre de la función: funcionFactorial(pnum)		Versión: 1.0
Descripción: Función que calcula la factorial de 5		
contador	Tipo de variable: int	
numero	Tipo de variable: int	
factorial	Tipo de variable: int	
Código: <pre>// Con parámetros function funcionFactorial(pnum){ let contador = 0; let numero = pnum; let factorial=1; while(contador<numero){ contador++; factorial=factorial*contador; } return factorial }</pre>		

Nombre de la función: funcionFactorialExp(pnum)		Versión: 2.0
Descripción: Función que calcula la factorial de 5		
contador	Tipo de variable: int	
numero	Tipo de variable: int	
factorial	Tipo de variable: int	
Código: <pre>//Funcion con expresion const funcionFactorialExp = function (pnum){ let contador = 0; let numero = pnum; let factorial=1; while(contador<numero){ contador++; factorial=factorial*contador; } return factorial }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

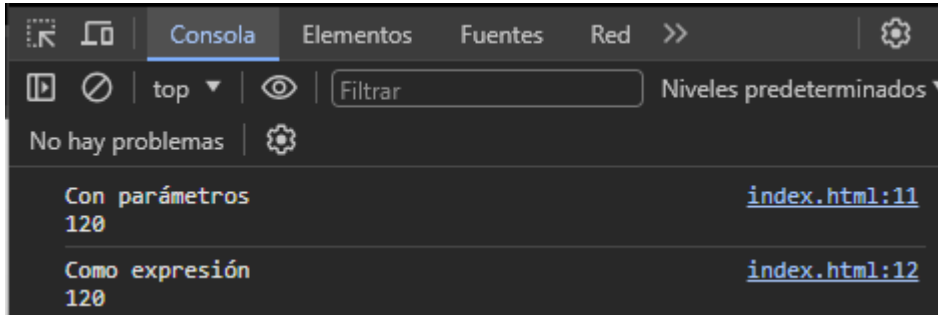


TABLA DEL 5 CICLO WHILE

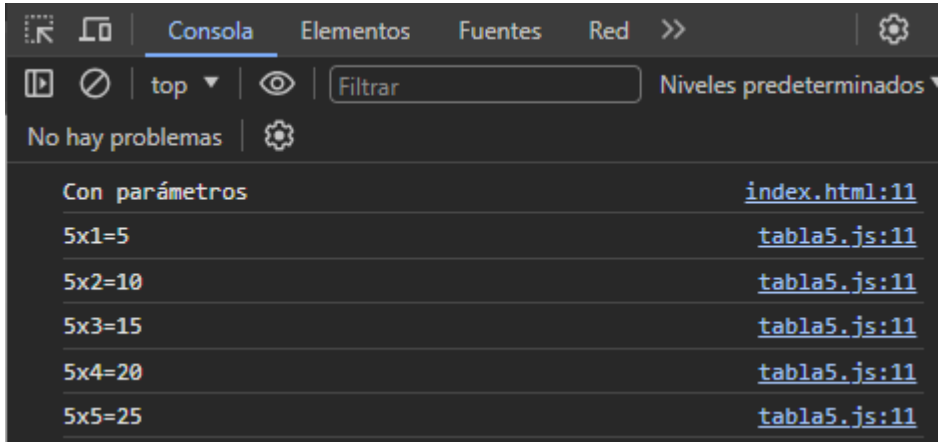
Nombre de la función: funcionTabla(pnum, pmult)		Versión: 1.0
Descripción: Función que calcula la tabla del 5 y multiplica hasta 5		
contador	Tipo de variable: int	
tabla	Tipo de variable: int	
numero	Tipo de variable: int	
resultado	Tipo de variable: int	
Código:		
<pre>// Con parámetros function funcionTabla(pnum, pmult) { let contador = 0; let tabla = pmult; let numero = pnum; let resultado = 0; while (contador < numero) { contador++; resultado = tabla * contador; console.log(tabla + "x" + contador + "=" + resultado); } }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS



Nombre de la función: funcionTablaExp
(pnum, pmult)

Versión: 2.0

Descripción:

Función que calcula la tabla del 5 y multiplica hasta 5

contador	Tipo de variable: int
tabla	Tipo de variable: int
numero	Tipo de variable: int
resultado	Tipo de variable: int

Código:

```
// como expresión
const funcionTablaExp = function (pnum, pmult){
    let contador = 0;
    let tabla = pmult;
    let numero = pnum;
    let resultado = 0;

    while(contador<numero){
        contador++;
        resultado=tabla*contador;
        console.log(tabla+"x"+contador+"="+resultado);
    }
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Como expresión	index.html:13
5x1=5	tabla5.js:24
5x2=10	tabla5.js:24
5x3=15	tabla5.js:24
5x4=20	tabla5.js:24
5x5=25	tabla5.js:24

TABLA DEL 9 CICLO WHILE

Nombre de la función: funcionTabla(pmult,pnum)		Versión: 1.0
Descripción: Función que calcula la tabla del 9 y multiplica hasta 5 y dice si es par o impar		
contador	Tipo de variable: int	
tabla	Tipo de variable: int	
numero	Tipo de variable: int	
resultado	Tipo de variable: int	
Código: <pre>// Con parámetros function funcionTabla(pmult,pnum){ let contador = 0; let tabla = pmult; let numero = pnum; let resultado = 0; while(contador<numero){ contador++; resultado=tabla*contador; if(resultado%2==0){ console.log(tabla+"x"+contador+"="+resultado+" Es par"); }else{ console.log(tabla+"x"+contador+"="+resultado+" Es impar"); } } }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

		top ▾		Filtrar	Niveles predeterminados ▾
No hay problemas					
Con parámetros:				index.html:11	
9x1=9 Es impar				tabla9.js:13	
9x2=18 Es par				tabla9.js:11	
9x3=27 Es impar				tabla9.js:13	
9x4=36 Es par				tabla9.js:11	
9x5=45 Es impar				tabla9.js:13	

Nombre de la función:
funcionTablaExp(pmult,pnum)

Versión: 2.0

Descripción:

Función que calcula la tabla del 9 y multiplica hasta 5 y dice si es par o impar

contador	Tipo de variable: int
tabla	Tipo de variable: int
numero	Tipo de variable: int
resultado	Tipo de variable: int

Código:

```
// como expresión
const funcionTablaExp = function (pmult,pnum){
  let contador = 0;
  let tabla = pmult;
  let numero = pnum;
  let resultado = 0;
  while(contador<numero){
    contador++;
    resultado=tabla*contador;
    if(resultado%2==0){
      console.log(tabla+"x"+contador+"="+resultado+" Es par");
    }else{
      console.log(tabla+"x"+contador+"="+resultado+" Es impar");
    }
  }
}
```




Tecnólogo en Análisis y Desarrollo de Software

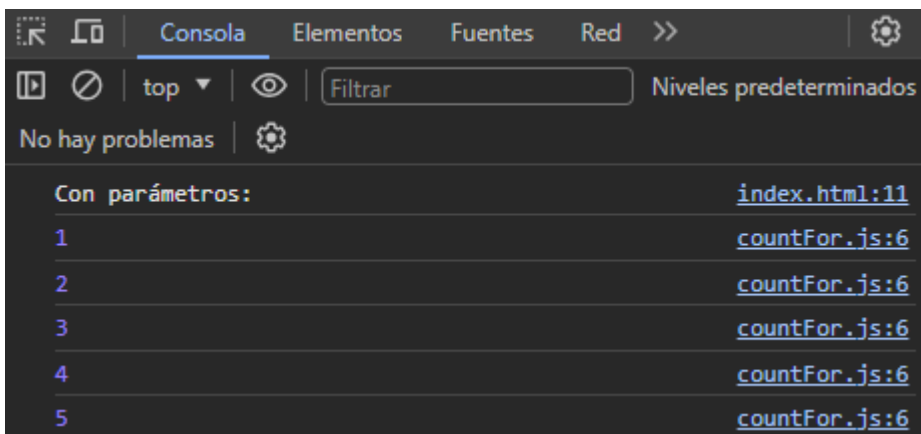
Ficha 2900177

Funciones JS

Como expresión:	index.html:13
9x1=9 Es impar	tabla9.js:29
9x2=18 Es par	tabla9.js:27
9x3=27 Es impar	tabla9.js:29
9x4=36 Es par	tabla9.js:27
9x5=45 Es impar	tabla9.js:29

CONTAR DE 1 A 5 CICLO FOR

Nombre de la función: contarFor(pnum)		Versión: 1.0
Descripción: Función que cuenta de 1 a 5 con el ciclo for		
num	Tipo de variable: int	
contar	Tipo de variable: int	
Código:		
<pre>// Con parámetros function contarFor(pnum){ let numero = pnum; for(let contar=1;contar <=numero; contar++){ console.log(contar); } }</pre>		





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Nombre de la función: contarForExp(pnum)

Versión: 2.0

Descripción:

Función que cuenta de 1 a 5 con el ciclo for

num Tipo de variable: int

contar Tipo de variable: int

Código:

```
// como expresión
const contarForExp = function (pnum){
    let numero = pnum;

    for(let contar=1;contar <=numero; contar++){
        console.log(contar);
    }
}
```

Con parámetros:

[index.html:13](#)

1

[countFor.js:14](#)

2

[countFor.js:14](#)

3

[countFor.js:14](#)

4

[countFor.js:14](#)

5

[countFor.js:14](#)

TABLA DEL 5 CICLO FOR

Nombre de la función: tablaFor(pnum, pmult)

Versión: 1.0

Descripción:

Función que calcula la tabla del 5 y multiplica hasta 5

contar Tipo de variable: int

tabla Tipo de variable: int

numero Tipo de variable: int

resultado Tipo de variable: int

Código:



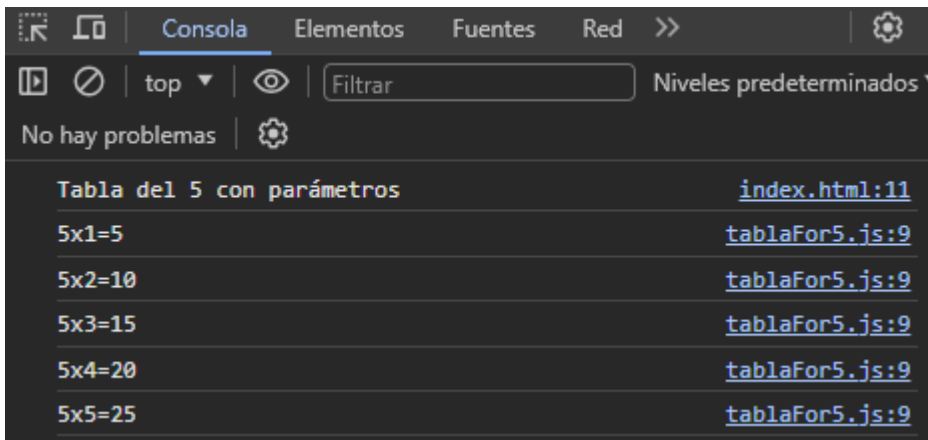
Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// con parámetros
function tablaFor(pmult,pnum){
    let tabla=pmult;
    let numero=pnum;
    let resultado=0;

    for(let contar=1; contar<=numero; contar++){
        resultado=tabla*contar;
        console.log(tabla+"x"+contar+"="+resultado);
    }
}
```



Nombre de la función: tablaForExp (pnum, pmult)		Versión: 2.0
Descripción: Función que calcula la tabla del 5 y multiplica hasta 5		
contar	Tipo de variable: int	
tabla	Tipo de variable: int	
numero	Tipo de variable: int	
resultado	Tipo de variable: int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// como expresión
const tablaForExp = function (pmult,pnum){
  let tabla=pmult;
  let numero=pnum;
  let resultado=0;

  for(let contar=1; contar<=numero; contar++){
    resultado=tabla*contar;
    console.log(tabla+"x"+contar+"="+resultado);
  }
}
```

Tabla del 5 como expresión

[index.html:13](#)

5x1=5

[tablaFor5.js:20](#)

5x2=10

[tablaFor5.js:20](#)

5x3=15

[tablaFor5.js:20](#)

5x4=20

[tablaFor5.js:20](#)

5x5=25

[tablaFor5.js:20](#)

TABLA DEL 9 CICLO FOR

Nombre de la función: tablaFor(pmult,pnum)		Versión: 1.0
Descripción: Función que calcula la tabla del 9 y multiplica hasta 5 y dice si es par o impar		
contar	Tipo de variable: int	
tabla	Tipo de variable: int	
numero	Tipo de variable: int	
resultado	Tipo de variable: int	
Código:		



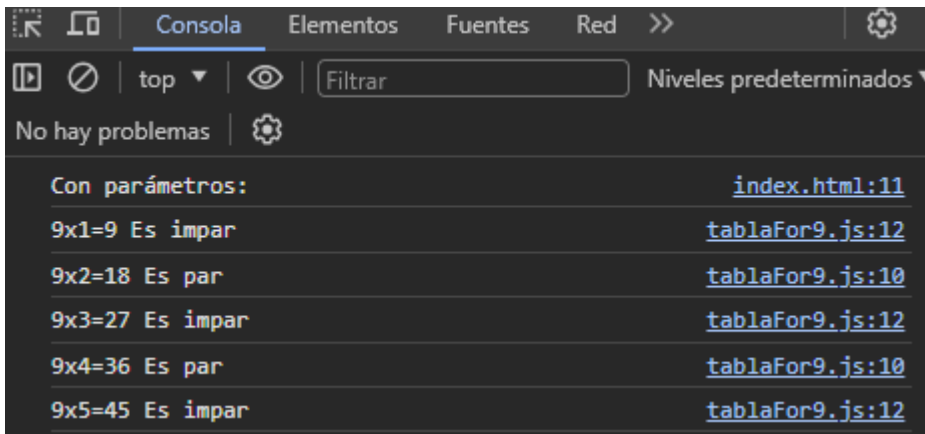
Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// Con parámetros
function TablaFor(pmult,pnum){
    let tabla = pmult;
    let numero = pnum;
    let resultado = 0;

    for(let contar=1; contar<=numero; contar++){
        resultado=tabla*contar;
        if(resultado%2==0){
            console.log(tabla+"x"+contar+"="+resultado+" Es par");
        }else{
            console.log(tabla+"x"+contar+"="+resultado+" Es impar");
        }
    }
}
```



Nombre de la función: tablaForExp(pmult,pnum)		Versión: 2.0
Descripción: Función que calcula la tabla del 9 y multiplica hasta 5 y dice si es par o impar		
contar	Tipo de variable: int	
tabla	Tipo de variable: int	
numero	Tipo de variable: int	
resultado	Tipo de variable: int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
// como expresión
const TablaForExp = function(pmult,pnum){
    let tabla = pmult;
    let numero = pnum;
    let resultado = 0;

    for(let contar=1; contar<=numero; contar++){
        resultado=tabla*contar;
        if(resultado%2==0){
            console.log(tabla+"x"+contar+"="+resultado+" Es par");
        }else{
            console.log(tabla+"x"+contar+"="+resultado+" Es impar");
        }
    }
}
```

Como expresión:	index.html:13
9x1=9 Es impar	tablaFor9.js:27
9x2=18 Es par	tablaFor9.js:25
9x3=27 Es impar	tablaFor9.js:27
9x4=36 Es par	tablaFor9.js:25
9x5=45 Es impar	tablaFor9.js:27

TABLAS DE MULTIPLICAR CON EL CICLO WHILE

Nombre de la función: tablasMult(limiteTabla,limiteMult)		Versión: 1.0
Descripción: Función que calcula las tablas de multiplicar y si es par o impar		
ctabla	Tipo de variable: int	
tabla	Tipo de variable: int	
cMult	Tipo de variable: int	
multiplicar	Tipo de variable: int	
resultado	Tipo de variable: int	
par	Tipo de variable: int	
impar	Tipo de variable: int	
Código:		



Tecnólogo en Análisis y Desarrollo de
Software
Ficha 2900177

Funciones JS

```
// Con parámetros
function tablasMult(limiteTabla,limiteMult){
    let ctabla = 0;
    let tabla = limiteTabla;
    let cMult;
    let multiplicar = limiteMult;
    let resultado = 0;
    let par = 0;
    let impar = 0;

    while(ctabla<tabla){
        cMult=0;
        ctabla=ctabla+1;
        while (cMult<multiplicar){
            cMult=cMult+1;
            resultado=ctabla*cMult;
            if (resultado%2==0){
                par=par+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es par. Buzz");
            }else{
                impar=impar+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es impar. Bass");
            }
        }
    }
    console.log("Hay "+par+" numeros pares y "+impar+" numeros impares.");
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

```
Consola Elementos Fuentes Red >>
top Filtar Niveles predeterminados
No hay problemas
1x1=1 Es impar. Bass tablas.js:22
1x2=2 Es par. Buzz tablas.js:19
1x3=3 Es impar. Bass tablas.js:22
1x4=4 Es par. Buzz tablas.js:19
1x5=5 Es impar. Bass tablas.js:22
2x1=2 Es par. Buzz tablas.js:19
2x2=4 Es par. Buzz tablas.js:19
2x3=6 Es par. Buzz tablas.js:19
2x4=8 Es par. Buzz tablas.js:19
2x5=10 Es par. Buzz tablas.js:19
3x1=3 Es impar. Bass tablas.js:22
3x2=6 Es par. Buzz tablas.js:19
3x3=9 Es impar. Bass tablas.js:22
3x4=12 Es par. Buzz tablas.js:19
3x5=15 Es impar. Bass tablas.js:22
4x1=4 Es par. Buzz tablas.js:19
4x2=8 Es par. Buzz tablas.js:19
4x3=12 Es par. Buzz tablas.js:19
4x4=16 Es par. Buzz tablas.js:19
4x5=20 Es par. Buzz tablas.js:19
5x1=5 Es impar. Bass tablas.js:22
5x2=10 Es par. Buzz tablas.js:19
5x3=15 Es impar. Bass tablas.js:22
5x4=20 Es par. Buzz tablas.js:19
5x5=25 Es impar. Bass tablas.js:22
Hay 16 numeros pares y 9 numeros impares. tablas.js:26
```

Nombre de la función: tablasMultExp(limiteTabla,limiteMult)		Versión: 2.0
Descripción: Función que calcula las tablas de multiplicar y si es par o impar		
ctabla	Tipo de variable: int	
tabla	Tipo de variable: int	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

cMult	Tipo de variable: int
multiplicar	Tipo de variable: int
resultado	Tipo de variable: int
par	Tipo de variable: int
impar	Tipo de variable: int

Código:

```
// Como expresión
const tablasMultExp = function (limiteTabla,limiteMult){
    let ctabla = 0;
    let tabla = limiteTabla;
    let cMult;
    let multiplicar = limiteMult;
    let resultado = 0;
    let par = 0;
    let impar = 0;

    while(ctabla<tabla){
        cMult=0;
        ctabla=ctabla+1;
        while (cMult<multiplicar){
            cMult=cMult+1;
            resultado=ctabla*cMult;
            if (resultado%2==0){
                par=par+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es par. Buzz");
            }else{
                impar=impar+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es impar. Bass")
            }
        }
    }

    console.log("Hay "+par+" numeros pares y "+impar+" numeros impares.");
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Tablas de multiplicar como una expresión

1x1=1 Es impar. Bass

1x2=2 Es par. Buzz

1x3=3 Es impar. Bass

1x4=4 Es par. Buzz

1x5=5 Es impar. Bass

2x1=2 Es par. Buzz

2x2=4 Es par. Buzz

2x3=6 Es par. Buzz

2x4=8 Es par. Buzz

2x5=10 Es par. Buzz

3x1=3 Es impar. Bass

3x2=6 Es par. Buzz

3x3=9 Es impar. Bass

3x4=12 Es par. Buzz

3x5=15 Es impar. Bass

4x1=4 Es par. Buzz

4x2=8 Es par. Buzz

4x3=12 Es par. Buzz

4x4=16 Es par. Buzz

4x5=20 Es par. Buzz

5x1=5 Es impar. Bass

5x2=10 Es par. Buzz

5x3=15 Es impar. Bass

5x4=20 Es par. Buzz

5x5=25 Es impar. Bass

Hay 16 numeros pares y 9 numeros impares.

TABLAS DE MULTIPLICAR CON EL CICLO FOR

Nombre de la función: tablasFor(limiteTabla,limiteMult)	Versión: 1.0
Descripción: Función que calcula las tablas de multiplicar y si es par o impar	
ctabla	Tipo de variable: int



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

tabla	Tipo de variable: int
cMult	Tipo de variable: int
multiplicar	Tipo de variable: int
resultado	Tipo de variable: int
par	Tipo de variable: int
impar	Tipo de variable: int

Código:

```
// con parámetros
function tablasFor(limiteTabla,limiteMult){
    let tabla = limiteTabla;
    let multiplicar = limiteMult;
    let resultado = 0;
    let par = 0;
    let impar = 0;

    for(let ctabla = 1; ctabla <= tabla; ctabla++){
        for(let cMult=1; cMult<=multiplicar; cMult++){
            resultado=ctabla * cMult;
            if (resultado%2==0){
                par=par+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es par. Buzz");
            }else{
                impar=impar+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es impar. Bass")
            }
        }
    }
    console.log("Hay "+par+" numeros pares y "+impar+" numeros impares.");
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Tablas con el ciclo For con parámetros:

1x1=1 Es impar. Bass

1x2=2 Es par. Buzz

1x3=3 Es impar. Bass

1x4=4 Es par. Buzz

1x5=5 Es impar. Bass

2x1=2 Es par. Buzz

2x2=4 Es par. Buzz

2x3=6 Es par. Buzz

2x4=8 Es par. Buzz

2x5=10 Es par. Buzz

3x1=3 Es impar. Bass

3x2=6 Es par. Buzz

3x3=9 Es impar. Bass

3x4=12 Es par. Buzz

3x5=15 Es impar. Bass

4x1=4 Es par. Buzz

4x2=8 Es par. Buzz

4x3=12 Es par. Buzz

4x4=16 Es par. Buzz

4x5=20 Es par. Buzz

5x1=5 Es impar. Bass

5x2=10 Es par. Buzz

5x3=15 Es impar. Bass

5x4=20 Es par. Buzz

5x5=25 Es impar. Bass

Hay 16 numeros pares y 9 numeros impares.

Nombre de la función: tablasForExp(limiteTabla,limiteMult)		Versión: 2.0
Descripción: Función que calcula las tablas de multiplicar y si es par o impar		
ctabla	Tipo de variable: int	
tabla	Tipo de variable: int	
cMult	Tipo de variable: int	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

multiplicar	Tipo de variable: int
resultado	Tipo de variable: int
par	Tipo de variable: int
impar	Tipo de variable: int

Código:

```
// como expresión
const tablasForExp = function (limiteTabla, limiteMult){
    let tabla = limiteTabla;
    let multiplicar = limiteMult;
    let resultado = 0;
    let par = 0;
    let impar = 0;

    for(let ctabla = 1; ctabla <= tabla; ctabla++){
        for(let cMult=1; cMult<=multiplicar; cMult++){
            resultado=ctabla * cMult;
            if (resultado%2==0){
                par=par+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es par. Buzz");
            }else{
                impar=impar+1;
                console.log(ctabla+"x"+cMult+"="+resultado+" Es impar. Bass")
            }
        }
    }
    console.log("Hay "+par+" numeros pares y "+impar+" numeros impares.");
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2900177

Funciones JS

Tablas con el ciclo For como expresión:

1x1=1 Es impar. Bass

1x2=2 Es par. Buzz

1x3=3 Es impar. Bass

1x4=4 Es par. Buzz

1x5=5 Es impar. Bass

2x1=2 Es par. Buzz

2x2=4 Es par. Buzz

2x3=6 Es par. Buzz

2x4=8 Es par. Buzz

2x5=10 Es par. Buzz

3x1=3 Es impar. Bass

3x2=6 Es par. Buzz

3x3=9 Es impar. Bass

3x4=12 Es par. Buzz

3x5=15 Es impar. Bass

4x1=4 Es par. Buzz

4x2=8 Es par. Buzz

4x3=12 Es par. Buzz

4x4=16 Es par. Buzz

4x5=20 Es par. Buzz

5x1=5 Es impar. Bass

5x2=10 Es par. Buzz

5x3=15 Es impar. Bass

5x4=20 Es par. Buzz

5x5=25 Es impar. Bass

Hay 16 numeros pares y 9 numeros impares.