



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		
saludo	Tipo de variable: Alfanumérico	
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		
saludo	Tipo de variable: Alfanumérica	
Código: <pre>const saludoExpresion = function(pSaludo){ let saludar = pSaludo; return saludar; //Valor }</pre>		



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		
suma	Tipo de variable: numérica	
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		
saludo	Tipo de variable: numérica	
Código:	<pre>const sumaExpresion = function(pNumUno, pNumDos){ let numUno = pNumUno let numDos = pNumDos let sumar sumar = numUno + numDos return sumar }</pre>	



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

03 - Operaciones

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



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)

operaciones

Tipo de variable: numérica

Código:

```
//Como Expresión
const sumaExpresion = function(pNumUno, pNumDos){
    let sumar;
    numUno = pNumUno;
    numDos = pNumDos;
    sumar = numUno + numDos

    return sumar
}

const restaExpresion = function(pNumUno, pNumDos){
    let restar;
    numUno = pNumUno;
    numDos = pNumDos;
    restar = numUno - numDos

    return restar
}
```

```
const multiplicacionExpresion = function(pNumUno, pNumDos){
    let multiplicar;
    numUno = pNumUno;
    numDos = pNumDos;
    multiplicar = numUno * numDos

    return multiplicar
}

const divisionExpresion = function(pNumUno, pNumDos){
    let dividir;
    numUno = pNumUno;
    numDos = pNumDos;
    dividir = numUno / numDos

    return dividir
}
```

```
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'
    }
}
```



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



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
Descripción: Imprime el promedio de tres notas		
promedio	Tipo de variable: Alfanumérico	
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		
promedio	Tipo de variable: Alfanumérica	
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>		



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%		
promedio	Tipo de variable: Alfanumérico	
Código: <pre>function sumarPorcentaje(pNota, pPorcentaje){ let nota = pNota let porcentaje = pPorcentaje let resultadoPorcentaje resultadoPorcentaje = nota * porcentaje return resultadoPorcentaje }</pre>		

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



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

07 – Área de Figuras Geométricas

Nombre de la función: function areaFiguras()		Versión: 2.0
Descripción: Calcula el área de un cuadrado, rectángulo y triángulo		
Área figuras	Tipo de variable: Alfanumérico	
Código:		
<pre>function areaCuadrado(pLado){ let lado = pLado let area area = lado * lado return area } function areaRectangulo(pBase, pAltura){ let base = pBase let altura = pAltura let area area = base * altura return area }</pre>		
<pre>function areaTriangulo(pBase, pAltura){ let base = pBase let altura = pAltura let area area = (base * altura)/2 return area }</pre>		

Nombre de la función: <code>const <u>areaFigurasExpresion</u> = function()</code>		Versión: 3.0
Descripción:		
Área figuras	Tipo de variable: Alfanumérica	
Código:		
<div><pre>const areaCuadradoExpresion = function(pLado){ let lado = pLado let area area = lado * lado return area } const areaRectanguloExpresion = function(pBase, pAltura){ let base = pBase let altura = pAltura let area area = base * altura return area }</pre></div> <div><pre>const areaTrianguloExpresion = function(pBase, pAltura){ let base = pBase let altura = pAltura let area area = (base * altura)/2 return area }</pre></div>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

08 – Sueldo de Persona

Nombre de la función: function calcularSueldoTotal	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)	
Salario persona	Tipo de variable: Alfanumérico

Código:

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

function calcularPension(pSueldo){
    let pen;
    pen = pSueldo * 0.16;
    return pen;
}
```

```
function calcularSalud(pSueldo){
    let salud;
    salud = pSueldo * 0.12;
    return salud;
}

function calcularARL(pSueldo){
    let arl;
    arl = (pSueldo * 0.16);
    return arl;
}
```

```
function calcularSueldoTotal() {
    let sueldoCalculado = calcularSueldo(180000, 22);
    let pen = calcularPension(sueldoCalculado);
    let salud = calcularSalud(sueldoCalculado);
    let arl = calcularARL(sueldoCalculado);
    let desc = pen + salud + arl;
    let sueldoTotal = sueldoCalculado - desc;
    return sueldoTotal;
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función: const calcularSueldoTotalExpresion = function()

Versión: 3.0

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

Salario persona

Tipo de variable: Alfanumérica

Código:

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

const calcularPensionExpresion = function(pSueldo){
  let pen;
  pen = pSueldo * 0.16;
  return pen;
}
```

```
const calcularSaludExpresion = function(pSueldo){
  let salud;
  salud = pSueldo * 0.12;
  return salud;
}

const calcularARLExpresion = function(pSueldo){
  let arl;
  arl = (pSueldo * 0.052);
  return arl;
}
```

```
const calcularSueldoTotalExpresion = function() {
  let sueldoCalculado = calcularSueldoExpresion(180000, 22);
  let pen = calcularPensionExpresion(sueldoCalculado);
  let salud = calcularSaludExpresion(sueldoCalculado);
  let arl = calcularARLExpresion(sueldoCalculado);
  let desc = pen + salud + arl;
  let sueldoTotal = sueldoCalculado - desc;
  return sueldoTotal;
}
```



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		
Número mayor	Tipo de variable: Alfanumérico	
Código: <pre>function numeroMayor(pNum1, pNum2){ let num1 = pNum1; let num2 = pNum2; if(num1 > 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		
Número mayor	Tipo de variable: Alfanumérica	
Código: <pre>const numeroMayorExpresion = function(pNum1, pNum2){ let num1 = pNum1; let num2 = pNum2; if(num1 > num2){ return num1; }else{ return num2; } }</pre>		



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		
Mayor de Edad	Tipo de variable: Alfanumérico	
Código:		

```
function mayorEdad(pFechaN, pYearActual){
    let fechaN = pFechaN;
    let yearActual = pYearActual
    let edad
    edad = yearActual - fechaN

    if(edad > 17){
        return "Es Mayor de Edad"
    }else{
        return "Es Menor de Edad"
    }
}
```

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		
Mayor Edad	Tipo de variable: Alfanumérica	
Código: <pre>const mayorEdadExpresion = function(pFechaN, pYearActual){ let fechaN = pFechaN; let yearActual = pYearActual let edad edad = yearActual - fechaN if(edad > 17){ return "Es Mayor de Edad" }else{ return "Es Menor de Edad" } }</pre>		



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		
	Tipo de variable: Alfanumérico	
Código: <pre>function numeroMayor(pNum1, pNum2){ let num1 = pNum1; let num2 = pNum2; if(num1 == num2){ return "Los números son iguales "; }else if(num1 > 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		
Número Mayor	Tipo de variable: Alfanumérica	
Código: <pre>const numeroMayorExpresion = function(pNum1, pNum2){ let num1 = pNum1; let num2 = pNum2; if(num1 == num2){ return "Los números son iguales "; }else if(num1 > num2){ return "El " + num1 + " es mayor"; }else{ return "El " + num2 + " es mayor"; } }</pre>		



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		
Número Mayor	Tipo de variable: Alfanumérico	
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 > 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; } }</pre>		



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

Número Mayor

Tipo de variable: Alfanumérica

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;  
        }  
    }  
}
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

13 - Área de Tres Cuadrados

Nombre de la función: function areaCuadrados(plado1, plado2, plado3)		Versión: 2.0
Descripción: Calcular el área 3 de cuadrados e imprimir el mayor		
Calcula Área	Tipo de variable: Alfanumérico	
Código:		
<pre>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"; } } }</pre>		



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

Calcula Area

Tipo de variable: Alfanumérica

Código:

```
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 == cu3 || 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

14 – Mayor de Tres Edades

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

Calcular Edad	Tipo de variable: Alfanumérico
---------------	--------------------------------

Código:

```
function mayorEdadUno(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 }
}

function mayorEdadDos(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 }
}

function mayorEdadTres(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 }
}

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)
    }
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función: const promedioEdadesExpresion = 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

Calcular Edad

Tipo de variable: Alfanumérica

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 }
}

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 }
}
```

```
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 }
}

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)
    }
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

15 – Sueldo Persona 2

Nombre de la función: function calcularSueldoTotal()		Versión: 2.0
Descripción: Calcular el sueldo de una persona todo incluido		
Sueldo Persona	Tipo de variable: Alfanumérico	
Código:		
<pre>function calcularSueldo(pValorD, pDiasT){ let valorD = pValorD; let diasT = pDiasT; let sueldo sueldo = valorD * diasT; return sueldo; } function calcularSubsidioTransporte(pSalarioM){ let salarioM = pSalarioM let subsidioT = calcularSueldo(43333, 30) < salarioM * 2 ? 114000 : 0; return subsidioT } function calcularPension(pSueldo){ let pen; pen = pSueldo * 0.16; return pen; } function calcularSalud(pSueldo){ let salud; salud = pSueldo * 0.12; return salud; } function calcularARL(pSueldo){ let arl; arl = (pSueldo * 0.052); return arl; } function calcularRetencion(pSalarioM){ let salarioM = pSalarioM let retencion = calcularSueldo(43333, 30) < salarioM * 4 ? 0 : sueldoP * 0.04; return retencion } function calcularSueldoTotal() { let sueldoCalculado = calcularSueldo(43333, 30); let subsidioT = calcularSubsidioTransporte(1300000) let pen = calcularPension(sueldoCalculado); let salud = calcularSalud(sueldoCalculado); let arl = calcularARL(sueldoCalculado); let retencion = calcularRetencion(1300000) let desc = pen + salud + arl; let sueldoTotal = ((sueldoCalculado + subsidioT) - (retencion + desc)).toFixed(0) return sueldoTotal; }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función: <code>const calcularSueldoTotalExpresion = function()</code>		Versión: 3.0
Descripción: Calcular el sueldo de una persona todo incluido		
Sueldo Persona	Tipo de variable: Alfanumérica	

Código:

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

const calcularSubsidioTransporteExpresion = function(pSalarioM){
    let salarioM = pSalarioM
    let subsidioT = calcularSueldoExpresion(180000, 30) < salarioM * 2 ? 114000 : 0;
    return subsidioT
}

const calcularPensionExpresion = function(pSueldo){
    let pen;
    pen = pSueldo * 0.16;
    return pen;
}
```

```
const calcularSaludExpresion = function(pSueldo){
    let salud;
    salud = pSueldo * 0.12;
    return salud;
}

const calcularARLEExpresion = function(pSueldo){
    let arl;
    arl = (pSueldo * 0.052);
    return arl;
}

const calcularRetencionExpresion = function(pSalarioM){
    let salarioM = pSalarioM
    let retencion = calcularSueldoExpresion(180000, 30) < salarioM * 4 ? 0 : calcularSueldoExpresion(180000, 30) * 0.04;
    return retencion
}

const calcularSueldoTotalExpresion = function() {
    let sueldoCalculado = calcularSueldoExpresion(180000, 30);
    let subsidioT = calcularSubsidioTransporteExpresion(1300000)
    let pen = calcularPensionExpresion(sueldoCalculado);
    let salud = calcularSaludExpresion(sueldoCalculado);
    let arl = calcularARLEExpresion(sueldoCalculado);
    let retencion = calcularRetencionExpresion(1300000)
    let desc = pen + salud + arl;
    let sueldoTotal = ((sueldoCalculado + subsidioT) - (retencion + desc)).toFixed(0)
    return sueldoTotal;
}
```



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		
Promedio Notas	Tipo de variable: Alfanumérico	
Código:		
<pre>//Como Parametro 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 } function validarNota(pNota1, pNota2, pNota3){ let sumPorc = calcularSumaPorcentaje(pNota1, pNota2, pNota3) if(sumPorc > 4.5){ return "El porcentaje de notas es superior" }else if (sumPorc <= 4.5 && sumPorc > 3.5){ return "El porcentaje de nota es buena" }else if (sumPorc <= 3.5 && sumPorc >= 3.0){ return "El porcentaje de nota es media" }else{ return "El porcentaje de nota es mala" } }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función: <code>const calcularSumaPorcentajeExpresion = function(pNota1, pNota2, pNota3)</code>	Versión: 3.0
---	--------------

Descripción:

Calcular el promedio de notas y demás

Promedio Notas

Tipo de variable: Alfanumérica

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
}

const validarNotaExpresion = function(pNota1, pNota2, pNota3){
  let sumPorc = calcularSumaPorcentajeExpresion(pNota1, pNota2, pNota3)
  if(sumPorc > 4.5){
    return "El porcentaje de notas es superior"
  }else if (sumPorc <= 4.5 && sumPorc > 3.5){
    return "El porcentaje de nota es buena"
  }else if (sumPorc <= 3.5 && sumPorc >= 3.0){
    return "El porcentaje de nota es media"
  }else{
    return "El porcentaje de nota es mala"
  }
}
```




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		
Contador	Tipo de variable: Alfanumérico	
Código: <pre>function contador(pNumero){ let numero = pNumero let contar = 0; while(contar < numero){ contar = contar + 1; console.log(contar); } return "" }</pre>		

Nombre de la función: const contadorExpresion = function(pNumero){		Versión: 3.0
Descripción: Imprimir los números del 1 al 5		
Contador	Tipo de variable: Alfanumérica	
Código: <pre>const contadorExpresion = function(pNumero){ let numero = pNumero let contar = -6; while(contar < numero){ contar = contar + 1; console.log(contar); } return "" }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

18 – Factorial de 5

Nombre de la función: function factorial(pNumero)		Versión: 2.0
Descripción: Imprimir la factorial de 5		
Factorial	Tipo de variable: Alfanumérico	
Código: <pre>function factorial(pNumero){ let numero = pNumero let fac = 1 let contar = 0; while(contar < 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		
Factorial	Tipo de variable: Alfanumérica	
Código:		
<pre>const factorialExpresion = function(pNumero){ let numero = pNumero let fac = 1 let contar = 0; while(contar < numero){ contar = contar + 1; fac = fac * contar } return fac }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

19 – Tabla del 5

Nombre de la función: function tabla(pNumero)		Versión: 2.0
Descripción: Imprime la tabla del 5		
tabla del multiplicar	Tipo de variable: Alfanumérico	
Código: <pre>function tabla(pNumero){ let numero = pNumero let contar = 0; let mul = 1 while(contar < numero){ contar = contar + 1; mul = numero * contar console.log(numero + " x " + contar + " = " + mul); } return "" }</pre>		

Nombre de la función: const tablaExpresion = function(pNumero){		Versión: 3.0
Descripción: Imprime la tabla del 5		
tabla de multiplicar	Tipo de variable: Alfanumérica	
Código: <pre>const tablaExpresion = function(pNumero){ let numero = pNumero let contar = 0; let mul = 1 while(contar < numero){ contar = contar + 1; mul = numero * contar console.log(numero + " x " + contar + " = " + mul); } return "" }</pre>		



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		
Tabla del Multiplicar	Tipo de variable: Alfanumérico	
Código:		
<pre>function tabla(pNumero){ let numero = pNumero let contar = 0; let mul = 1 while(contar < 5){ contar = contar + 1; mul = numero * contar if(mul % 2 == 0){ console.log(numero + " x " + contar + " = " + mul + " Es Par"); }else{ console.log(numero + " x " + contar + " = " + mul + " Es Impar"); } } return "" }</pre>		

Nombre de la función: const tablaExpresion = function(pNumero)		Versión: 3.0
Descripción: Imprimir la tabla del 9 con los números pares e impares		
Tabla del Multiplicar	Tipo de variable: Alfanumérica	
Código:		
<pre>const tablaExpresion = function(pNumero){ let numero = pNumero let contar = 0; let mul = 1 while(contar < 5){ contar = contar + 1; mul = numero * contar if(mul % 2 == 0){ console.log(numero + " x " + contar + " = " + mul + " Es Par"); }else{ console.log(numero + " x " + contar + " = " + mul + " Es Impar"); } } return "" }</pre>		



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

Descripción:

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

Tablas de Multiplicar

Tipo de variable: Alfanumérico

Código:

```
function tablas(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

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

Tablas de Multiplicar

Tipo de variable: Alfanumérica

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

CICLO FOR

21 – Contar al 5

Nombre de la función: function contador(pNumero)		Versión: 2.0
Descripción: Imprimir los números del 1 al 5		
Contador	Tipo de variable: Alfanumérico	
Código: <pre>function contador(pNumero){ let numero = pNumero let contador for(contador = 1; contador <= numero; contador++){ console.log(contador) } return "" }</pre>		

Nombre de la función: const contadorExpresion = function(pNumero)		Versión: 3.0
Descripción: Imprimir los números del 1 al 5		
Contador	Tipo de variable: Alfanumérica	
Código: <pre>const contadorExpresion = function(pNumero){ let numero = pNumero let contador for(contador = -5; contador <= numero; contador++){ console.log(contador) } return "" }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

22 – Factorial de 5

Nombre de la función: function factorial(pNumero)		Versión: 2.0
Descripción: Imprimir la factorial de 5		
Factorial	Tipo de variable: Alfanumérico	
Código: <pre>function factorial(pNumero){ let numero = pNumero let fac = 1 let contar for(contar = 1; contar <= numero; contar++){ fac = fac * contar; } return fac }</pre>		

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



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

23 – Tabla del 5

Nombre de la función: function tabla(pNumero)		Versión: 2.0
Descripción: Imprime la tabla del 5		
Tabla	Tipo de variable: Alfanumérico	
Código:		
<pre>function tabla(pNumero){ let numero = pNumero let contar let mul = 1 for(contar = 1; contar <= numero; contar++){ mul = numero * contar; console.log(numero + " x " + contar + " = " + mul); } return "" }</pre>		

Nombre de la función: const tablaExpresion = function(pNumero)		Versión: 3.0
Descripción: Imprime la tabla del 5		
Tabla	Tipo de variable: Alfanumérica	
Código:		
<pre>const tablaExpresion = function(pNumero){ let numero = pNumero let contar let mul = 1 for(contar = 1; contar <= numero; contar++){ mul = numero * contar; console.log(numero + " x " + contar + " = " + mul); } return "" }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

24 – 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		
Tabla	Tipo de variable: Alfanumérico	
Código:		
<pre>function tabla(pNumero){ let numero = pNumero let contar = 0; let mul = 1 for(contar = 1; contar <= 5; contar++){ mul = numero * contar if(mul % 2 == 0){ console.log(numero + " x " + contar + " = " + mul + " Es Par \n"); }else{ console.log(numero + " x " + contar + " = " + mul + " Es Impar \n"); } } return "" }</pre>		

Nombre de la función: const tablaExpresion = function(pNumero){		Versión: 3.0
Descripción: Imprimir la tabla del 9 con los números pares e impares		
Tabla	Tipo de variable: Alfanumérica	
Código:		
<pre>const tablaExpresion = function(pNumero){ let numero = pNumero let contar = 0; let mul = 1 for(contar = 1; contar <= 5; contar++){ mul = numero * contar if(mul % 2 == 0){ console.log(numero + " x " + contar + " = " + mul + " Es Par \n"); }else{ console.log(numero + " x " + contar + " = " + mul + " Es Impar \n"); } } return "" }</pre>		



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

Tablas

Tipo de variable: Alfanumérico

Código:

```
function tablas(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 <= tabla; contar++){
        mul = 0;
        console.log("Tabla del " + contar)
        for(mul = 1; mul <= 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}
}
```



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

Tablas

Tipo de variable: Alfanumérica

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

    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){
                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}
}
```