



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

1.

Nombre de la función: saludar(psaludo)		Versión: 1.0
Descripción: Función de saludo		
Saludar	Tipo de variable:text	
Código:		
<pre>function saludar(psaludar){ let saludar=psaludar; return saludar }</pre>		<div>Hola mundo</div> <div>></div>

Hola mundo



Nombre de la función: <code>saludarExp(psaludar)</code>		Versión: 2.0
Descripción: Función de saludar como expresión		
saludar	Tipo de variable: text	
Código:		
<pre>const saludarExp=function(psaludar){ let saludar=psaludar; return saludar }</pre>		<div>Hola mundo</div> <div>></div>

Hola mundo





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

2.

Nombre de la función: suma(pnum1,pnum2)		Versión: 1.0
Descripción: Función de sumar		
num1	Tipo de variable: Int	
num2	Tipo de variable: Int	
suma	Tipo de variable: Int	
Código:		
<pre>function suma(pnum1,pnum2){ let num1=pnum1; let num2=pnum2; let suma; suma=num1+num2; return suma; }</pre>		
La suma de 12 y 23 es: 35		

La suma de 12 y 23 es: 35



Nombre de la función: sumaExp(pnum1,pnum2)		Versión: 2.0
Descripción: Función de sumar		
num1	Tipo de variable: Int	
num2	Tipo de variable: Int	
suma	Tipo de variable: Int	
Código: <pre>const sumaExp=function(pnum1,pnum2){ let num1=pnum1; let num2=pnum2; let suma; suma=num1+num2; return suma; }</pre>		

La suma de 12 y 23 es: 35

>

La suma de 12 y 23 es: 35





Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

3.

Nombre de la función: resta(pnum1,pnum2)		Versión: 1.0
Descripción: Función de resta		
pnum1	Tipo de variable: Int	
pnum2	Tipo de variable: Int	
restar	Tipo de variable: Int	
Código:		
<pre>function resta(pnum1,pnum2){ let restar; pnum1 = pnum1; pnum2 = pnum2; restar=pnum1-pnum2; return restar; }</pre>		
<div>La resta de 9 - 8 es: 1</div> <div>La multiplicacion de 9 x 8 es: 72</div> <div>La division de 19 / 11 es: 1.7272727272727273</div>		

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.7272727272727273

> |

Nombre de la función: restaExp(pnum1,pnum2)		Versión: 2.0
Descripción: Función de resta		
pnum1	Tipo de variable: Int	
pnum2	Tipo de variable: Int	
restar	Tipo de variable: Int	
Código:		
<pre>const restaExp=function(pnum1,pnum2){ let restar; pnum1 = pnum1; pnum2 = pnum2; restar = pnum1 - pnum2; return restar; }</pre>		
<div>La resta de 9 - 8 es: 1</div> <div>La multiplicacion de 9 x 8 es: 72</div> <div>La division de 19 / 11 es: 1.7272727272727273</div>		

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.7272727272727273

> |

Nombre de la función: multiplicacion(pnum1,pnum2)		Versión: 1.0
Descripción: Función de multiplicación		
pnum1	Tipo de variable: Int	
pnum2	Tipo de variable: Int	
multiplicar	Tipo de variable: Int	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Código:

```
function multiplicacion(pnum1,pnum2){  
    let multiplicar;  
    pnum1 = pnum1;  
    pnum2 = pnum2;  
    multiplicar=pnum1*pnum2;  
    return multiplicar;  
}
```

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.7272727272727273

> |

Nombre de la función: multiplicaciónExp(pnum1,pnum2)

Versión: 2.0

Descripción:

Función de multiplicación

pnum1

Tipo de variable: Int

pnum2

Tipo de variable: Int

multiplicar

Tipo de variable: Int

Código:

```
const multiplicacionExp=function(pnum1,pnum2){  
    let multiplicar;  
    pnum1 = pnum1;  
    pnum2 = pnum2;  
    multiplicar = pnum1 * pnum2;  
    return multiplicar;  
}
```

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.7272727272727273

> |

Nombre de la función: division(pnum1,pnum2)

Versión: 1.0

Descripción:

Función de división

pnum1

Tipo de variable: Int

pnum2

Tipo de variable: Int

dividir

Tipo de variable: Int

Código:



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
function division(pnum1,pnum2){  
    let dividir;  
    pnum1 = pnum1;  
    pnum2 = pnum2;  
    dividir=pnum1/pnum2;  
    return dividir;  
}
```

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.7272727272727273

> |

Nombre de la función: divisionExp(pnum1,pnum2)

Versión: 2.0

Descripción:

Función de división

pnum1

Tipo de variable: Int

pnum2

Tipo de variable: Int

dividir

Tipo de variable: Int

Código:

```
const divisionExp=function(pnum1,pnum2){  
    let divisor;  
    pnum1 = pnum1;  
    pnum2 = pnum2;  
    divisor = pnum1 / pnum2;  
    return divisor;  
}
```

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.7272727272727273

> |

Nombre de la función: operacione(poperador,pnum1,pnum2)

Versión: 1.0

Descripción:

Función de operaciones(resta,multiplicación,división)

operador

Tipo de variable:text

pnum1

Tipo de variable: Int

pnum2

Tipo de variable: Int

resultado

Tipo de variable: Int

Código:



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
function operacion(poperador,pnum1,pnum2){  
  let operador=poperador;  
  pnum1 = pnum1;  
  pnum2 = pnum2;  
  if(operador=="resta"){  
    resultado=resta(pnum1,pnum2);  
  }else if(operador=="multiplicacion"){  
    resultado=multiplicacion(pnum1,pnum2);  
  }else if(operador=="division"){  
    resultado=division(pnum1,pnum2);  
  }else{  
    resultado="Error";  
  }  
  return resultado;  
}
```

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.72727272727273

> |

Nombre de la función: operacionesExp(poperacion,pnum1,pnum2)		Versión: 2.0
Descripción:		
Función de operaciones(resta,multiplicación,división)		
peración	Tipo de variable:text	
pnum1	Tipo de variable: Int	
pnum2	Tipo de variable: Int	
resultado	Tipo de variable: Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const operacionExp = function(poperador,pnum1,pnum2){  
  let operador = poperador;  
  pnum1 = pnum1;  
  pnum2 = pnum2;  
  if (operador == "resta") {  
    operacionResultado = restaExp(pnum1,pnum2);  
  }else if (operador == "multiplicacion") {  
    operacionResultado = multiplicacionExp(pnum1,pnum2);  
  }else if (operador == "division") {  
    operacionResultado = divisionExp(pnum1,pnum2);  
  }else{  
    operacionResultado = "Error!!! no se puede operar";  
  }  
  return operacionResultado;  
}
```

La resta de 9 - 8 es: 1

La multiplicacion de 9 x 8 es: 72

La division de 19 / 11 es: 1.7272727272727273

> |

4.

Nombre de la función: porcentajes(pnum1)		Versión: 1.0
Descripción: Función de porcentaje		
porciento	Tipo de variable:float	
num1	Tipo de variable:Int	
Código:		
<pre>function porcentajes(pnum1){ let num1=pnum1; let porciento; porciento=num1/100; return porciento; }</pre>		<div>El porcentaje de 23 es: 0.23</div> <div>></div>

Nombre de la función: porcentajeExp(pnum1)		Versión: 2.0
Descripción: Función de porcentaje		
porcentaje	Tipo de variable:float	
num1	Tipo de variable: Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const porcentajeExp=function(pnum1){  
  let num1=pnum1;  
  let porcentaje;  
  porcentaje=num1/100;  
  return porcentaje;  
}
```

El porcentaje de 23 es: 0.23

>

5.

Nombre de la función: promedio(pnota1,pnota2,pnota3)		Versión: 1.0
Descripción: Función de promedio de tres notas		
promedio	Tipo de variable: float	
suma	Tipo de variable: float	
nota1	Tipo de variable: float	
nota2	Tipo de variable: float	
nota3	Tipo de variable: float	
Código: <pre>function promedio(pnota1,pnota2,pnota3){ let nota1=pnota1; let nota2=pnota2; let nota3=pnota3; let suma; let promedio; suma=nota1+nota2+nota3; promedio=suma/3; return promedio; }</pre>		
El promedio de las tres notas es: 3.8333333333333335		

Nombre de la función: promedioExp(pnota1,pnota2,pnota3)		Versión: 2.0
Descripción: Función de promedio de tres notas		
promedio	Tipo de variable: float	
suma	Tipo de variable: float	
nota1	Tipo de variable: float	
nota2	Tipo de variable: float	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

nota3	Tipo de variable: float
Código: <pre>const promedioExp=function(pnota1,pnota2,pnota3){ let nota1=pnota1; let nota2=pnota2; let nota3=pnota3; let suma; let promedio; suma=nota1+nota2+nota3; promedio=suma/3; return promedio; }</pre>	
El promedio de las tres notas es: 3.8333333333333335	

6.

Nombre de la función:porcentajes(pnota,pporcentaje)	Versión: 1.0
Descripción: Función de porcentaje de tres notas	
porcentaje	Tipo de variable:float
nota	Tipo de variable:float
respuesta	Tipo de variable:float
Código: <pre>function porcentajes(pnota,pporcentaje){ let nota=pnota; let porcentaje=pporcentaje; let respuesta; respuesta=nota*porcentaje; return respuesta; }</pre>	
El porcentaje de la nota 3.4 es: 1.02 El porcentaje de la nota 2.5 es: 0.75 El porcentaje de la nota 4.3 es: 1.72 La suma de los porcentajes es: 3.49	

Nombre de la función: porcentajeExp(pnota,pporcentaje)	Versión: 2.0
Descripción: Función de porcentajes de varias notas	
porcentaje	Tipo de variable:float



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

nota	Tipo de variable:float
respuesta	Tipo de variable:float
<p>Código:</p> <pre>const porcentajeExp=function(pnota,pporcentaje){ let nota=pnota; let porcentaje=pporcentaje; let respuesta; respuesta=nota*porcentaje; return respuesta; }</pre> <p>El porcentaje de la nota 3.4 es: 1.02 El porcentaje de la nota 2.5 es: 0.75 El porcentaje de la nota 4.3 es: 1.72 La suma de los porcentajes es: 3.49</p>	

7.

Nombre de la función:figuras(pFigura,pBase,pAltura)	Versión: 1.0
Descripción: Funcion de las áreas de tres figuras	
figuras	Tipo de variable:text
base	Tipo de variable:Int
altura	Tipo de variable:Int
operacion	Tipo de variable:Int
Código:	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
function figuras(pFiguras,pBase,pAltura){  
  let figuras=pFiguras;  
  let base=pBase;  
  let altura=pAltura;  
  let operacion;  
  if(figuras=="cuadrado"){  
    operacion=base*base;  
    return operacion;  
  }else if(figuras=="rectangulo"){  
    operacion=base*altura;  
    return operacion;  
  }else if(figuras=="triangulo"){  
    operacion=(base*altura)/2;  
    return operacion;  
  }else{  
    return "figura no encontrada";  
  }  
}
```

Area del cuadrado 1 es: 4

Area del rectangulo 1 es: 72

Area del triangulo 1 es: 10



Nombre de la función: figurasExp(poperador,pbase,paltura)		Versión: 2.0
Descripción: Función de áreas de figuras		
operador	Tipo de variable:text	
base	Tipo de variable:Int	
altura	Tipo de variable:Int	
operacion	Tipo de variable:Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const figurasExp=function(pFiguras,pBase,pAltura){  
  let figuras=pFiguras;  
  let base=pBase;  
  let altura=pAltura;  
  let operacion;  
  if(figuras=="cuadrado"){  
    operacion=base*base;  
    return operacion;  
  }else if(figuras=="rectangulo"){  
    operacion=base*altura;  
    return operacion;  
  }else if(figuras=="triangulo"){  
    operacion=(base*altura)/2;  
    return operacion;  
  }else{  
    return "figura no encontrada";  
  }  
}
```

Area del cuadrado 1 es: 4

Area del rectangulo 1 es: 72

Area del triangulo 1 es: 10



8.

Nombre de la función: subsidios(psubsidios,pdias,pvdia)		Versión: 1.0
Descripción: Función de subsidios para un pago total		
subsidios	Tipo de variable:text	
días	Tipo de variable:Int	
vdia	Tipo de variable:Int	
salario	Tipo de variable:Int	
salud	Tipo de variable:Int	
pension	Tipo de variable:Int	
arl	Tipo de variable:Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
function subsidios(psubsidio, pdias, pvdia){  
    let subsidio=psubsidio;  
    let dias=pdias;  
    let vdia=pvdia;  
    let salario,salud,pension,arl;  
    salario=dias*vdia;  
    if(subsidio=="salud"){  
        salud=salario*0.12;  
        return salud;  
    }else if(subsidio=="pension"){  
        pension=salario*0.16;  
        return pension;  
    }else if(subsidio=="arl"){  
        arl=salario*0.052;  
        return arl;  
    }else{  
        return "No hay mas subsidios";  
    }  
}
```

El subsidio de salud es: 828000

El subsidio de pension es: 1104000

El subsidio de el arl es: 358800

El pago total de la persona con su respectivo deducible es: 4609200

Nombre de la función: subsidioExp(psubsidio,pdias,pvdia)

Versión: 2.0

Descripción:

Función de pago total

subsidios	Tipo de variable:text
días	Tipo de variable:Int
vdia	Tipo de variable:Int
salario	Tipo de variable:Int
salud	Tipo de variable:Int
pension	Tipo de variable:Int
arl	Tipo de variable:Int

Código:



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const subsidioExp=function(psubsidio,pdias,pvdia){  
  let subsidio=psubsidio;  
  let dias=pdias;  
  let vdia=pvdia;  
  let salario,salud,pension,arl;  
  salario=dias*vdia;  
  if(subsidio=="salud"){  
    salud=salario*0.12;  
    return salud;  
  }else if(subsidio=="pension"){  
    pension=salario*0.16;  
    return pension;  
  }else if(subsidio=="arl"){  
    arl=salario*0.052;  
    return arl;  
  }else{  
    return "No hay mas subsidios";  
  }  
}
```

El subsidio de salud es: 828000

El subsidio de pension es: 1104000

El subsidio de el arl es: 358800

El pago total de la persona con su respectivo deducible es: 4609200

Nombre de la función: pagoTotal(pdias,pvdia)		Versión: 1.0
Descripción:		
Función de el pago total de una persona		
pagTot	Tipo de variable: Int	
deducible	Tipo de variable: Int	
dias	Tipo de variable: Int	
vdia	Tipo de variable: Int	
salario	Tipo de variable: Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
function pagoTotal(pdias,pvdia){  
  let pagoTot;  
  let deducible;  
  let dias=pdias;  
  let vdia=pvdia;  
  let salario=dias*vdia;  
  deducible=subsidios("salud",pdias,pvdia)+subsidios("pension",pdias,pvdia)+subsidios("ar1",pdias,pvdia);  
  pagoTot=salario-deducible;  
  return pagoTot;  
}
```

El subsidio de salud es: 828000

El subsidio de pension es: 1104000

El subsidio de el ar1 es: 358800

El pago total de la persona con su respectivo deducible es: 4609200

Nombre de la función: pagoTotalExp(pdias,pvdia)		Versión: 1.0
Descripción: Función de el pago total de una persona		
pagTot	Tipo de variable: Int	
deducible	Tipo de variable: Int	
dias	Tipo de variable: Int	
vdia	Tipo de variable: Int	
salario	Tipo de variable: Int	
Código:		
<pre>const pagoTotalExp=function(pdias,pvdia){ let pagoTot; let deducible; let dias=pdias; let vdia=pvdia; let salario=dias*vdia; deducible=subsidios("salud",pdias,pvdia)+subsidios("pension",pdias,pvdia)+subsidios("ar1",pdias,pvdia); pagoTot=salario-deducible; return pagoTot; }</pre>		
El subsidio de salud es: 828000		
El subsidio de pension es: 1104000		
El subsidio de el ar1 es: 358800		
El pago total de la persona con su respectivo deducible es: 4609200		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

9.

Nombre de la función: mayorEdad(pedad)		Versión: 1.0
Descripción: Función de si es mayor o menor de edad		
edad	Tipo de variable: Int	
Código:		
<pre>function mayorEdad(pedad){ let edad=pedad; if(edad>17){ return "Edad: "+"(+edad+)" +"Usted es mayor de edad"; }else{ return "Edad: "+"(+edad+)" +"Usted es menor de edad"; } }</pre>		
Edad: (14) Usted es menor de edad		
Edad: (18) Usted es mayor de edad		
>		

Nombre de la función:mayorEdadExp(pedad)		Versión: 2.0
Descripción: Función de si es mayor o menor de edad		
Edad	Tipo de variable:Int	
Código:		
<pre>function mayorEdadExp(pedad){ let edad=pedad; if(edad>17){ return "Edad: "+"(+edad+)" +"Usted es mayor de edad"; }else{ return "Edad: "+"(+edad+)" +"Usted es menor de edad"; } }</pre>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Edad: (14) Usted es menor de edad
Edad: (18) Usted es mayor de edad

>

10.

Nombre de la función: edad(pNacimiento)		Versión: 1.0
Descripción: Función de si es mayor o menor de edad pero con el año de nacimiento		
nacimiento	Tipo de variable: Int	
actual	Tipo de variable: Int	
edad	Tipo de variable: Int	
Código: <pre>function edadFecha(pNacimiento){ let nacimiento=pNacimiento; let actual=2024; let edad; edad=actual-nacimiento; if(edad>17){ return "Edad: "+"("+edad+"") "+"usted es mayor de edad"; }else{ return "Edad: "+"("+edad+"") "+"usted es menor de edad"; } }</pre>		
Edad: (19) usted es mayor de edad		

Nombre de la función: edadFechaExp(pNacimiento)		Versión: 2.0
Descripción: Función de si es mayor o menor de edad pero con el año de nacimiento		
nacimiento	Tipo de variable: Int	
actual	Tipo de variable: Int	
edad	Tipo de variable: Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const edadFechaExp=function(pNacimiento){  
  let nacimiento=pNacimiento;  
  let actual=2024;  
  let edad;  
  edad =actual-nacimiento;  
  if(edad>17){  
    return "Edad: "+"("+edad+") "+"usted es mayor de edad";  
  }else{  
    return "Edad: "+"("+edad+") "+"usted es menor de edad";  
  }  
}
```

Edad: (19) usted es mayor de edad

11.

Nombre de la función: comparar(pnum1,pnum2)

Versión: 1.0

Descripción:

Función de comparar si son iguales o cual es mayor

num1

Tipo de variable: Int

num2

Tipo de variable: Int

Código:

```
function comparar(pnum1,pnum2){  
  let num1=pnum1;  
  let num2=pnum2;  
  if(num1==num2){  
    return "el numero "+num1+" y el numero "+num2+" son iguales";  
  }else if(num1>num2){  
    return "El numero "+num1+" es mayor a "+num2;  
  }else{  
    return "El numero "+num2+" es mayor a "+num1;  
  }  
}
```

El numero 24 es mayor a 13

el numero 2 y el numero 2 son iguales

El numero 12 es mayor a 4

>



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Nombre de la función: compararExp(pnum1,pnum2)		Versión: 2.0
Descripción:		
Función de comparar dos números y cual es mayor o si son iguales		
num1	Tipo de variable: Int	
num2	Tipo de variable: Int	
Código:		
<pre>const compararExp=function(pnum1,pnum2){ let num1=pnum1; let num2=pnum2; if(num1==num2){ return "El numero "+num1+" y el numero "+num2+" son iguales"; }else if(num1>num2){ return "El numero "+num1+" es mayor a "+num2; }else{ return "El numero "+num2+" es mayor a "+num1; } }</pre>		
<div>El numero 24 es mayor a 13</div> <div>el numero 2 y el numero 2 son iguales</div> <div>El numero 12 es mayor a 4</div> <div>></div>		

12.

Nombre de la función: areaCudad(plado)		Versión: 1.0
Descripción: Función de áreas de tres cuadrados		
Lado	Tipo de variable: Int	
cuadrado	Tipo de variable: Int	
Código:		
<pre>function areaCudad(plado){ let lado=plado; let cuadrado=lado*lado; return cuadrado; }</pre>		<div>El area de un cuadrado con un lado de tres es: 9</div> <div>El area de un cuadrado con un lado de doce es: 144</div> <div>El area de un cuadrado con un lado de trece es: 169</div>

Nombre de la función: areasComp(parea1,parea2,parea3)		Versión: 1.0
Descripción: Función de comparar las áreas de tres cuadrados		
area1	Tipo de variable: Int	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

area2	Tipo de variable: Int
area3	Tipo de variable: Int
Código: <pre>function areasComp(parea1,parea2,parea3){ let area1=parea1; let area2=parea2; let area3=parea3; if(area1==area2&&area2==area3){ return "El area de todos los cuadrados es la misma"; }else if(area1>area2&&area2>area3){ return "El area del cuadrado 1 es mayor"; }else if(area1<area2&&area2>area3){ return "El area del cuadrado 2 es mayor"; }else{ return "El area del cuadrado 3 es mayor"; } }</pre>	

El area de un cuadrado con un lado de tres es: 9
El area de un cuadrado con un lado de doce es: 144
El area de un cuadrado con un lado de trece es: 169
El area del cuadrado 3 es mayor

Nombre de la función: areaCuadExp(plado)		Versión: 2.0
Descripción: Función de áreas de tres cuadrados		
Lado	Tipo de variable: Int	
cuadrado	Tipo de variable: Int	
Código: <pre>const areaCuadExp=function(plado){ let lado=plado; let cuadrado=lado*lado; return cuadrado; }</pre>		El area de un cuadrado con un lado de tres es: 9 El area de un cuadrado con un lado de doce es: 144 El area de un cuadrado con un lado de trece es: 169

Nombre de la función: areasComp(parea1,parea2,parea3)		Versión: 2.0
Descripción: Función de comparar las áreas de tres cuadrados		
area1	Tipo de variable: Int	
area2	Tipo de variable: Int	
area3	Tipo de variable: Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const areasCompExp=function(parea1,parea2,parea3){  
  let area1=parea1;  
  let area2=parea2;  
  let area3=parea3;  
  
  if(area1==area2&&area2==area3){  
    return "El area de todos los cuadrados es la misma";  
  }else if(area1>area2&&area2>area3){  
    return "El area del cuadrado 1 es mayor";  
  }else if(area1<area2&&area2>area3){  
    return "El area del cuadrado 2 es mayor";  
  }else{  
    return "El area del cuadrado 3 es mayor";  
  }  
}
```

El area de un cuadrado con un lado de tres es: 9
El area de un cuadrado con un lado de doce es: 144
El area de un cuadrado con un lado de trece es: 169
El area del cuadrado 3 es mayor

13.

Nombre de la función: edades(pedad1,pedad2,pedad3)		Versión: 1.0
Descripción: Función de edades, saber cual es mayor o si son iguales.		
edad1	Tipo de variable: Int	
edad2	Tipo de variable: Int	
edad3	Tipo de variable: Int	
Código:		
<pre>function edades(pedad1,pedad2,pedad3){ let edad1=pedad1; let edad2=pedad2; let edad3=pedad3; if(edad1==edad2&&edad2==edad3){ return "Las edades son iguales"; }else if(edad1>edad2&&edad2>edad3){ return "La edad: "+edad1+" es la edad mas grande"; }else if(edad1<edad2&&edad2>edad3){ return "La edad: "+edad2+" es la edad mas grande"; }else{ return "La edad: "+edad3+" es la edad mas grande"; } }</pre>		
La edad: 23 es la edad mas grande		
La suma de las edades es: 57 y cumple con la mayoria de edad		

Nombre de la función: edadesExp(pedad1,pedad2,pedad3)	Versión: 2.0
Descripción: Función de edades, saber cual es mayor o si son iguales.	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

edad1	Tipo de variable: Int
edad2	Tipo de variable: Int
edad3	Tipo de variable: Int

Código:

```
const edadesExp=function(pedad1,pedad2,pedad3){  
  let edad1=pedad1;  
  let edad2=pedad2;  
  let edad3=pedad3;  
  if(edad1==edad2&&edad2==edad3){  
    return "Las edades son iguales";  
  }else if(edad1>edad2&&edad2>edad3){  
    return "La edad: "+edad1+" es la edad mas grande";  
  }else if(edad1<edad2&&edad2<edad3){  
    return "La edad: "+edad2+" es la edad mas grande";  
  }else{  
    return "La edad: "+edad3+" es la edad mas grande";  
  }  
}
```

La edad: 23 es la edad mas grande

La suma de las edades es: 57 y cumple con la mayoría de edad

14.

Nombre de la función: salarioP(pdias,pvdia)		Versión: 1.0
Descripción: Función de Salario de una persona		
pdias	Tipo de variable: Int	
pvdia	Tipo de variable: Int	
totalSueldo	Tipo de variable: Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
function salarioP(pdias,pvdia){  
    let dias=pdias;  
    let vdia=pvdia;  
    let totalSueldo;  
    totalSueldo=dias*vdia;  
    return totalSueldo;  
}
```

Salario de la persona: 48000000

El subsidio de transporte de la persona es: 0

Salud: 5760000

Pension: 7680000

Arl: 2496000

Total persona: 32064000

Nombre de la función: subTransporteP(pdias,pvdia)		Versión: 1.0
Descripción:		
Función de Subsidio de transporte de una persona		
salarioMin	Tipo de variable: Int	
salarioTrans	Tipo de variable: Int	
subTransporte	Tipo de variable: Int	
Código:		
<pre>function subTransporteP(pdias,pvdia){ let salarioMin=1600000; let salarioTrans=salario(pdias,pvdia); let subTransporte; if(salarioTrans<=2*salarioMin){ subTransporte=114000; }else{ subTransporte=0; } return subTransporte; }</pre>		
Salario de la persona: 48000000		
El subsidio de transporte de la persona es: 0		
Salud: 5760000		
Pension: 7680000		
Arl: 2496000		
Total persona: 32064000		

Nombre de la función: salud(pdias,pvdia)		Versión: 1.0
Descripción: Función de subsidio de salud de una persona		
pagoSalud	Tipo de variable: Float	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Código:

```
function saludP(pdias,pvdia){  
    let pagoSalud;  
    pagoSalud=salario(pdias,pvdia)*0.12;  
    return pagoSalud;  
}
```

Salario de la persona: 48000000
El subsidio de transporte de la persona es: 0
Salud: 5760000
Pension: 7680000
Arl: 2496000
Total persona: 32064000

Nombre de la función: pensionP(pdias,pvdia)		Versión: 1.0
Descripción: Función de subsidio de pension de una persona		
pagoPension	Tipo de variable:Float	
Código:		
<pre>function pensionP(pdias,pvdia){ let pagoPension; pagoPension=salario(pdias,pvdia)*0.16; return pagoPension; }</pre>		
<div>Salario de la persona: 48000000</div> <div>El subsidio de transporte de la persona es: 0</div> <div>Salud: 5760000</div> <div>Pension: 7680000</div> <div>Arl: 2496000</div> <div>Total persona: 32064000</div>		

Nombre de la función: arlP(pdias,pvdia)		Versión: 1.0
Descripción: Función de subsidio del arl de una persona		
pagoArl	Tipo de variable:Float	
Código:		
<pre>function arlP(pdias,pvdia){ let pagoArl; pagoArl=salario(pdias,pvdia); return pagoArl; }</pre>		<div>Salario de la persona: 48000000</div> <div>El subsidio de transporte de la persona es: 0</div> <div>Salud: 5760000</div> <div>Pension: 7680000</div> <div>Arl: 2496000</div> <div>Total persona: 32064000</div>

Nombre de la función: deducibleP(pdias,pvdia)		Versión: 1.0
Descripción: Función del deducible		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

salarioReten	Tipo de variable: Int
salarioMin	Tipo de variable: Int
retencionPago	Tipo de variable: Int
Código: <pre>function deducibleP(pdias,pvdia){ let pagoDeducible; pagoDeducible=saludP(pdias,pvdia)+pensionP(pdias,pvdia)+arlP(pdias,pvdia); return pagoDeducible; }</pre>	
Salario de la persona: 48000000	
El subsidio de transporte de la persona es: 0	
Salud: 5760000	
Pension: 7680000	
Arl: 2496000	
Total persona: 32064000	

Nombre de la función: retencionP(pdias,pvdia)		Versión: 1.0
Descripción: Función de retención de sueldo		
salarioReten	Tipo de variable: Int	
salarioMin	Tipo de variable: Int	
retencionPago	Tipo de variable: Int	
Código: <pre>function retencionP(pdias,pvdia){ let salarioReten=salario(pdias,pvdia); let salarioMin=1600000; let retencionPago; if(salarioReten<=4*salarioMin){ retencionPago=114000; }else{ retencionPago=0; } return retencionPago; }</pre>		
Salario de la persona: 48000000		
El subsidio de transporte de la persona es: 0		
Salud: 5760000		
Pension: 7680000		
Arl: 2496000		
Total persona: 32064000		

Nombre de la función: pagoTotal(pdias,pvdia)		Versión: 1.0
Descripción: Función de Total a pagar de una persona		
pagoSueldo	Tipo de variable: Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
function pagoTotal(pdias,pvdia){  
    let pagoSueldo;  
    pagoSueldo=salario(pdias,pvdia)+subTransporte(pdias,pvdia)-deducible(pdias,pvdia);  
    return pagoSueldo;  
}
```

Salario de la persona: 48000000
El subsidio de transporte de la persona es: 0
Salud: 5760000
Pension: 7680000
Arl: 2496000
Total persona: 32064000

Nombre de la función: salario(pdias,pvdia)		Versión: 2.0
Descripción: Función de salario de una persona		
dias	Tipo de variable: Int	
vdia	Tipo de variable: Int	
totalSalario	Tipo de variable: Int	
Código:		
<pre>const salario=function(pdias,pvdia){ let dias=pdias; let vdia=pvdia; let totalSalario; totalSalario=dias*vdia; return totalSalario; }</pre>		<div>Salario de la persona: 48000000</div> <div>El subsidio de transporte de la persona es: 0</div> <div>Salud: 5760000</div> <div>Pension: 7680000</div> <div>Arl: 2496000</div> <div>Total persona: 32064000</div>

Nombre de la función: subTransporte(pdias,pvdia)		Versión: 2.0
Descripción: Función de subsidio de transporte de una persona		
salarioMin	Tipo de variable: Int	
salarioTrans	Tipo de variable: Int	
subTransporte	Tipo de variable: Int	
Código:		
<pre>const subTransporte=function(pdias,pvdia){ let salarioMin=1600000; let salarioTrans=salario(pdias,pvdia); let subTransporte; if(salarioTrans<=2*salarioMin){ subTransporte=114000; }else{ subTransporte=0; } return subTransporte; }</pre>		<div>Salario de la persona: 48000000</div> <div>El subsidio de transporte de la persona es: 0</div> <div>Salud: 5760000</div> <div>Pension: 7680000</div> <div>Arl: 2496000</div> <div>Total persona: 32064000</div>



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Nombre de la función: salario(pdias,pvdia)

Versión: 2.0

Descripción:

Función de subsidio de salud de una persona

pagoSalud

Tipo de variable:Float

Código:

```
const salud=function(pdias,pvdia){  
  let pagoSalud;  
  pagoSalud=salario(pdias,pvdia)*0.12;  
  return pagoSalud;  
}
```

Salario de la persona: 48000000

El subsidio de transporte de la persona es: 0

Salud: 5760000

Pension: 7680000

Arl: 2496000

Total persona: 32064000

Nombre de la función: pension(pdias,pvdia)

Versión: 2.0

Descripción:

Función de subsidio de pension de una persona

pagoPension

Tipo de variable:Float

Código:

```
const pension=function(pdias,pvdia){  
  let pagoPension;  
  pagoPension=salario(pdias,pvdia)*0.16;  
  return pagoPension;  
}
```

Salario de la persona: 48000000

El subsidio de transporte de la persona es: 0

Salud: 5760000

Pension: 7680000

Arl: 2496000

Total persona: 32064000

Nombre de la función: arl(pdias,pvdia)

Versión: 2.0

Descripción:

Función de subsidio de arl de una persona

pagoArl

Tipo de variable:Float

Código:



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const arl=function(pdias,pvdia){  
  let pagoArl;  
  pagoArl=salario(pdias,pvdia)*0.052;  
  return pagoArl;  
}
```

Salario de la persona: 48000000
El subsidio de transporte de la persona es: 0
Salud: 5760000
Pension: 7680000
Arl: 2496000
Total persona: 32064000

Nombre de la función: deducible(pdias,pvdia)		Versión: 2.0
Descripción:		
Función del deducible de una persona		
pagoDeducible	Tipo de variable: Int	
Código:		
<pre>const deducible=function(pdias,pvdia){ let pagoDeducible; pagoDeducible=salud(pdias,pvdia)+pension(pdias,pvdia)+arl(pdias,pvdia); return pagoDeducible; }</pre>		
Salario de la persona: 48000000		
El subsidio de transporte de la persona es: 0		
Salud: 5760000		
Pension: 7680000		
Arl: 2496000		
Total persona: 32064000		

Nombre de la función: retencion(pdias,pvdia)		Versión: 2.0
Descripción: Función de retención del sueldo de una persona		
salarioReten	Tipo de variable: Int	
salarioMin	Tipo de variable: Int	
retencionPago	Tipo de variable: Int	
Código:		
<pre>const retencion=function(pdias,pvdia){ let salarioReten=salario(pdias,pvdia); let salarioMin=1600000; let retencionPago; if(salarioReten<=4*salarioMin){ retencionPago=114000; }else{ retencionPago=0; } return retencionPago; }</pre>		<div>Salario de la persona: 48000000</div> <div>El subsidio de transporte de la persona es: 0</div> <div>Salud: 5760000</div> <div>Pension: 7680000</div> <div>Arl: 2496000</div> <div>Total persona: 32064000</div>



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Nombre de la función: pagoTotalExp(pdias,pvdia)

Versión: 2.0

Descripción:

Función de sueldo total de una persona

pagoSueldo

Tipo de variable: Int

Código:

```
const pagoTotalExp=function(pdias,pvdia){  
  let pagoSueldo;  
  pagoSueldo=salario(pdias,pvdia)+subTransporte(pdias,pvdia)-deducible(pdias,pvdia);  
  return pagoSueldo;  
}
```

Salario de la persona: 48000000

El subsidio de transporte de la persona es: 0

Salud: 5760000

Pension: 7680000

Arl: 2496000

Total persona: 32064000

15.

Nombre de la función: nota(pnota,pporcentaje)

Versión: 1.0

Descripción:

Función de porcentaje de notas

nota

Tipo de variable: float

porcentaje

Tipo de variable: float

operacion

Tipo de variable: float

Código:

```
function nota(pnota,pporcentaje){  
  let nota=pnota;  
  let porcentaje=pporcentaje;  
  let operacion=nota*porcentaje;  
  return operacion;  
}
```

El porcentaje de la primera nota es: 0.6000000000000001

El porcentaje de la segunda nota es: 1.2249999999999999

El porcentaje de la tercera nota es: 1.575

La nota es media

Nombre de la función: suma(pporcentaje1,pporcentaje2,pporcentaje3)

Versión: 1.0

Descripción:

Función de suma de porcentajes de notas y su aplica al rango

porcentaje1

Tipo de variable: Float

porcentaje2

Tipo de variable: Float

porcentaje3

Tipo de variable: Float



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

suma	Tipo de variable:Float
Código: <pre>function suma(pporcentaje1,pporcentaje2,pporcentaje3){ let porcentaje1=pporcentaje1; let porcentaje2=pporcentaje2; let porcentaje3=pporcentaje3; let suma; suma=porcentaje1+porcentaje2+porcentaje3; if(suma>4.5){ return "La nota es superior" }else if(suma>3.5){ return "La nota es buena"; }else if(suma>3){ return "La nota es media"; }else{ return "La nota es mala"; } }</pre>	
El porcentaje de la primera nota es: 0.6000000000000001	
El porcentaje de la segunda nota es: 1.2249999999999999	
El porcentaje de la tercera nota es: 1.575	
La nota es media	

Nombre de la función: notaExp(pnota,pporcentaje)		Versión: 2.0
Descripción: Función de porcentaje de notas		
nota	Tipo de variable:float	
porcentaje	Tipo de variable:float	
operacion	Tipo de variable:float	
Código:		
<pre>const notaExp=function(pnota,pporcentaje){ let nota=pnota; let porcentaje=pporcentaje; let operacion=nota*porcentaje; return operacion; }</pre>		
El porcentaje de la primera nota es: 0.6000000000000001		
El porcentaje de la segunda nota es: 1.2249999999999999		
El porcentaje de la tercera nota es: 1.575		
La nota es media		

Nombre de la función: sumaExp(pporcentaje1,pporcentaje2,pporcentaje3)	Versión: 2.0
---	---------------------



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Descripción:

Función de suma de porcentajes de notas y su aplica al rango

porcentaje1	Tipo de variable:Float
porcentaje2	Tipo de variable:Float
porcentaje3	Tipo de variable:Float
suma	Tipo de variable:Float

Código:

```
const sumaExp=function(pporcentaje1,pporcentaje2,pporcentaje3){  
  let porcentaje1=pporcentaje1;  
  let porcentaje2=pporcentaje2;  
  let porcentaje3=pporcentaje3;  
  let suma;  
  suma=porcentaje1+porcentaje2+porcentaje3;  
  if(suma>4.5){  
    return "La nota es superior"  
  }else if(suma>3.5){  
    return "La nota es buena";  
  }else if(suma>3){  
    return "La nota es media";  
  }else{  
    return "La nota es mala";  
  }  
}
```

El porcentaje de la primera nota es: 0.6000000000000001

El porcentaje de la segunda nota es: 1.2249999999999999

El porcentaje de la tercera nota es: 1.575

La nota es media

16.

Nombre de la función: contar(pinicio)		Versión: 1.0
Descripción:		
Función de contar hasta 5 (while)		
inicio	Tipo de variable: Int	
Código:		
<pre>function contar(pinicio){ let inicio=pinicio; return inicio; }</pre>		<div>1</div> <div>2</div> <div>3</div> <div>4</div> <div>5</div>

Nombre de la función: contarExp(pinicio)	Versión: 2.0
Descripción:	



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Función de contar hasta 5 (while)

inicio Tipo de variable: Int

Código:

```
const contarExp=function(pinicio){  
    let inicio=pinicio;  
    return inicio;  
}
```

1
2
3
4
5

17.

Nombre de la función: contador(pcontar)

Versión: 1.0

Descripción:

Función de contar hasta 5 (for)

Inicio Tipo de variable: Int

Código:

```
function contar(pinicio){  
    let inicio=pinicio;  
    return inicio;  
}
```

1
2
3
4
5

Nombre de la función: contaExp(pcontar)

Versión: 2.0

Descripción:

Función de contar hasta 5 (for)

Inicio Tipo de variable: Int

Código:

```
const contarExp=function(pinicio){  
    let inicio=pinicio;  
    return inicio;  
}
```

1
2
3
4
5



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

18.

Nombre de la función: contar(pinicio)		Versión: 1.0
Descripción: Función de contar hasta 5 y cual es par o impar (while)		
inicio	Tipo de variable: Int	
Código:		
<pre>function contar(pinicio){ let inicio=pinicio; if(inicio%2==0){ return inicio+" par"; }else{ return inicio+" impar"; } }</pre>		<div>1 impar</div> <hr/> <div>2 par</div> <hr/> <div>3 impar</div> <hr/> <div>4 par</div> <hr/> <div>5 impar</div>

Nombre de la función: contarExp(pinicio)		Versión: 2.0
Descripción:		
Función de contar hasta 5 y cual es par o impar (while)		
inicio	Tipo de variable: Int	
Código:		
<div><pre>const contarExp=function(pinicio){ let inicio=pinicio; if(inicio%2==0){ return inicio+" par"; }else{ return inicio+" impar"; } }</pre></div> <div><div>1 impar</div><div>2 par</div><div>3 impar</div><div>4 par</div><div>5 impar</div></div>		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

19.

Nombre de la función: contar(pinicio)		Versión: 1.0
Descripción:		
Función de contar hasta 5 y cual es par o impar (for)		
inicio	Tipo de variable: Int	
Código:		
<pre>function contar(pinicio){ let inicio=pinicio; if(inicio%2==0){ return inicio+" par"; }else{ return inicio+" impar"; } }</pre>		
	1 impar	
	2 par	
	3 impar	
	4 par	
	5 impar	

Nombre de la función: contarExp(pinicio)		Versión: 2.0
Descripción:		
Función de contar hasta 5 y cual es par o impar (for)		
inicio	Tipo de variable: Int	
Código:		
<pre>const contarExp=function(pinicio){ let inicio=pinicio; if(inicio%2==0){ return inicio+" par"; }else{ return inicio+" impar"; } }</pre>		<div>1 impar</div> <div>2 par</div> <div>3 impar</div> <div>4 par</div> <div>5 impar</div>



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

20.

Nombre de la función: tabla5(pinicio)		Versión: 1.0
Descripción: Función de la tabla del 5 (while)		
inicio	Tipo de variable: Int	
Código:		
<pre>function tabla5(pinicio){ let inicio=pinicio; inicio=inicio*5; return inicio; }</pre>		<div>5x1=5</div> <div>5x2=10</div> <div>5x3=15</div> <div>5x4=20</div> <div>5x5=25</div>

5x1=5
5x2=10
5x3=15
5x4=20
5x5=25

Nombre de la función: tabla5Exp(pinicio)		Versión: 2.0
Descripción: Función de la tabla del 5 (while)		
Inicio	Tipo de variable: Int	
Código:		
<pre>const tabla5Exp=function(pinicio){ let inicio=pinicio; inicio=inicio*5; return inicio; }</pre>		<div>5x1=5</div> <div>5x2=10</div> <div>5x3=15</div> <div>5x4=20</div> <div>5x5=25</div>

5x1=5
5x2=10
5x3=15
5x4=20
5x5=25



Tecnólogo en Análisis y Desarrollo de
Software
Ficha 2899747
Funciones JS

21.

Nombre de la función: tabla5(pinicio)		Versión: 1.0
Descripción: Función de la tabla del 5 (for)		
inicio	Tipo de variable: Int	
Código:		
<pre>function tabla5(pinicio){ let inicio=pinicio; inicio=inicio*5; return inicio; }</pre>		

5x1=5
5x2=10
5x3=15
5x4=20
5x5=25

Nombre de la función: tabla5Exp(presul)		Versión: 2.0
Descripción: Función de la tabla del 5 (for)		
inicio	Tipo de variable: Int	
Código:		
<pre>const tabla5Exp=function(pinicio){ let inicio=pinicio; inicio=inicio*5; return inicio; }</pre>		

5x1=5
5x2=10
5x3=15
5x4=20
5x5=25



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

22.

Nombre de la función: tabla9(pinicio)		Versión: 1.0
Descripción: Función de la tabla del 9 y cual es par o impar (while)		
Inicio	Tipo de variable: Int	
Código:		
<pre>function tabla9(pinicio){ let inicio=pinicio; inicio=inicio*9; if(inicio%2==0){ return inicio+" par" }else{ return inicio+" impar" } }</pre>		<div>9x1=9 impar</div> <div>9x2=18 par</div> <div>9x3=27 impar</div> <div>9x4=36 par</div> <div>9x5=45 impar</div>

9x1=9 impar
9x2=18 par
9x3=27 impar
9x4=36 par
9x5=45 impar

Nombre de la función: tabla9Exp(pinicio)		Versión: 2.0
Descripción: Función de la tabla del 9 y cual es par o impar (while)		
Inicio	Tipo de variable: Int	
Código:		
<pre>const tabla9Exp=function(pinicio){ let inicio=pinicio; inicio=inicio*9; if(inicio%2==0){ return inicio+" par"; }else{ return inicio+" impar"; } }</pre>		

9x1=9 impar
9x2=18 par
9x3=27 impar
9x4=36 par
9x5=45 impar



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

23.

Nombre de la función: tabla9(pinicio)		Versión: 1.0
Descripción:		
Función de la tabla del 9 y cual es par o impar (for)		
inicio	Tipo de variable: Int	
Código:		
<pre>function tabla9(pinicio){ let inicio=pinicio; inicio=inicio*9; if(inicio%2==0){ return inicio+" par" }else{ return inicio+" impar" } }</pre>		

9x1=9 impar
9x2=18 par
9x3=27 impar
9x4=36 par
9x5=45 impar

Nombre de la función: tabla9Exp(presul)		Versión: 2.0
Descripción:		
Función de la tabla del 9 y cual es par o impar (for)		
inicio	Tipo de variable: Int	
Código:		
<pre>const tabla9Exp=function(pinicio){ let inicio=pinicio; inicio=inicio*9; if(inicio%2==0){ return inicio+" par"; }else{ return inicio+" impar"; } }</pre>		

9x1=9 impar
9x2=18 par
9x3=27 impar
9x4=36 par
9x5=45 impar



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

24.

Nombre de la función: tablas1A5(pinicio,pinicio2)		Versión: 2.0
Descripción: Función de las tablas del 1 al 5 con bass y buzz y contando cuales son par e impar (while)		
Inicio1	Tipo de variable: Int	
Inicio2	Tipo de variable: Int	
Código:		
<pre>function tablas(pinicio1,pcinicio2){ let inicio1=pinicio1; let inicio2=pcinicio2; let resultado=inicio1*inicio2; return resultado; }</pre>		
<div>4x2=8 Buzz</div> <div>4x3=12 Buzz</div> <div>4x4=16 Buzz</div> <div>4x5=20 Buzz</div> <div>5x1=5 Bass</div> <div>5x2=10 Buzz</div> <div>5x3=15 Bass</div> <div>5x4=20 Buzz</div> <div>5x5=25 Bass</div> <div>La cantidad de Buzz: 16</div> <div>La cantidad de Bass: 9</div>		

Nombre de la función: tablasExp(pinicio,pinicio2)		Versión: 2.0
Descripción: Función de la tabla del 1 al 5 con bass y buzz y contando cuales son par e impar (while)		
Inicio1	Tipo de variable:Int	
Inicio2	Tipo de variable:Int	
Código:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

```
const tablasExp=function(pinicio1,pinicio2){  
  let inicio1=pinicio1;  
  let inicio2=pinicio2;  
  let resultado=inicio1*inicio2;  
  return resultado;  
}
```

4x2=8 Buzz
4x3=12 Buzz
4x4=16 Buzz
4x5=20 Buzz
5x1=5 Bass
5x2=10 Buzz
5x3=15 Bass
5x4=20 Buzz
5x5=25 Bass
La cantidad de Buzz: 16
La cantidad de Bass: 9

25.

Nombre de la función: tablas1A5(pinicio,pinicio2)		Versión: 2.0
Descripción: Función de las tablas del 1 al 5 con bass y buzz y contando cuales son par e impar (for)		
inicio1	Tipo de variable: Int	
inicio2	Tipo de variable: Int	
Código: <pre>function tablas(pinicio1,pinicio2){ let inicio1=pinicio1; let inicio2=pinicio2; let resultado=inicio1*inicio2; return resultado; }</pre> <p>4x2=8 Buzz 4x3=12 Buzz 4x4=16 Buzz 4x5=20 Buzz 5x1=5 Bass 5x2=10 Buzz 5x3=15 Bass 5x4=20 Buzz 5x5=25 Bass La cantidad de Buzz: 16 La cantidad de Bass: 9</p>		

Nombre de la función: tablas1A5(pinicio,pinicio2)		Versión: 2.0
Descripción:		



Tecnólogo en Análisis y Desarrollo de Software

Ficha 2899747

Funciones JS

Función de las tablas del 1 al 5 con bass y buzz y contando cuales son par e impar (for)

inicio1
inicio2

Tipo de variable: Int
Tipo de variable: Int

Código:

```
const tablasExp=function(pinicio1,pinicio2){  
  let inicio1=pinicio1;  
  let inicio2=pinicio2;  
  let resultado=inicio1*inicio2;  
  return resultado;  
}
```

4x2=8 Buzz

4x3=12 Buzz

4x4=16 Buzz

4x5=20 Buzz

5x1=5 Bass

5x2=10 Buzz

5x3=15 Bass

5x4=20 Buzz

5x5=25 Bass

La cantidad de Buzz: 16

La cantidad de Bass: 9