



Tecnólogo en Análisis y Desarrollo de
Software
Ficha

Funciones JS



TECNÓLOGO EN ANÁLISIS Y DESARROLLO DE SOFTWARE

Ficha: 2900177

Aprendiz: Brayan Santiago Guerrero Mendez

Instructor: Andrés Moreno

Neiva-Huila



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Contenido

| | | |
|-----|---|----|
| 1. | Saludo | 3 |
| 2. | Operaciones aritméticas..... | 5 |
| 3. | Calcular porcentaje..... | 15 |
| 4. | Promedio de tres notas | 17 |
| 5. | Porcentaje de tres notas | 19 |
| 6. | Calcular área de figuras | 21 |
| 7. | Calcular sueldo | 27 |
| 8. | Mayor de dos números | 37 |
| 9. | Fecha nacimiento | 39 |
| 10. | Mayor igual de dos números | 41 |
| 11. | Mayor igual de tres números | 43 |
| 12. | Área de tres cuadrados | 45 |
| 13. | Fecha nacimiento de tres personas | 49 |
| 14. | Sueldo de una persona..... | 55 |
| 15. | Porcentaje y suma de tres notas..... | 69 |
| 16. | Contador 5 while | 73 |
| 17. | Factorial 5 while | 75 |
| 18. | Tabla del 5 while | 77 |
| 19. | Tabla del 9 while | 79 |
| 20. | Contar de 1 a 5 for | 83 |
| 21. | Factorial de 5 for | 85 |
| 22. | Tabla del 5 for | 87 |
| 23. | Tabla del 9 for | 89 |
| 24. | Tabla 1 a 5 for..... | 93 |
| 25. | Tabla 1 a 5 while..... | 97 |

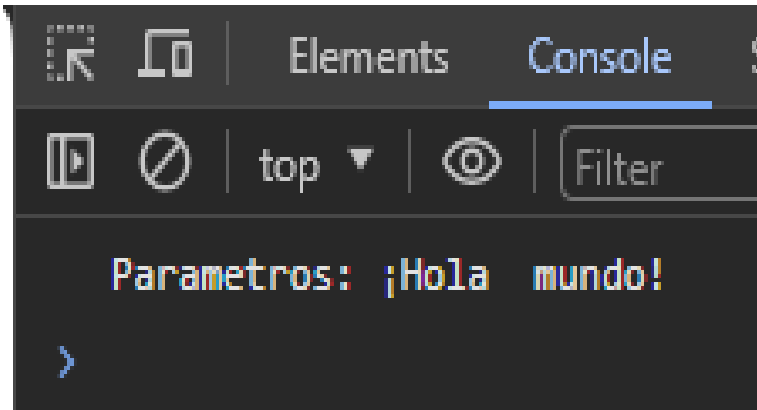


Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

1. Saludo

| | | |
|---|--------------------------------|--------------|
| Nombre de la función: saludo(psaludo) | | Versión: 1.0 |
| Descripción: Función que saluda | | |
| saludar | Tipo de variable: Alfanumérico | |
| Código: | | |
| <pre><script> let saludar = "¡Hola mundo!" console.log("Parametros: "+saludo(saludar)); </script></pre> | | |
| <pre>function saludo(psaludo){//Como parametro let saludar = psaludo; return saludar; }</pre> | | |
|  | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const saludoExp = function(psaludo)`

Versión: 2.0

Descripción:

Función que saluda

saludar

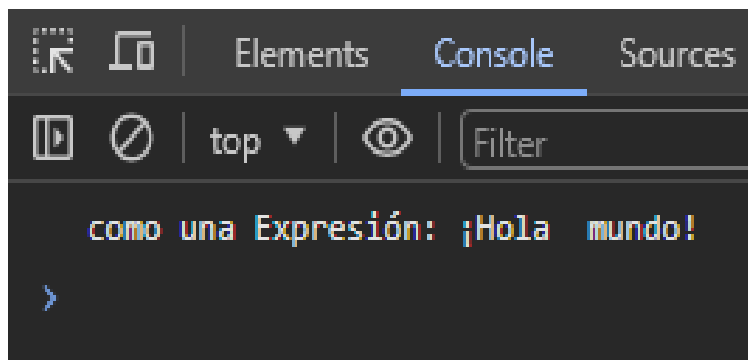
Tipo de variable: Alfanumérico

Código:

```
<script>
  let saludar = "¡Hola mundo!"

  console.log("como una Expresión: " + saludoExp(saludar));
</script>
```

```
const saludoExp = function(psaludo){//como expresión
  let saludar = psaludo;
  return saludar;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

2. Operaciones aritméticas

| | |
|--|---------------------|
| Nombre de la función: sumar(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:

```
<script>

    console.log("Suma con parametros " + operaciones("suma",1,2));

</script>
```

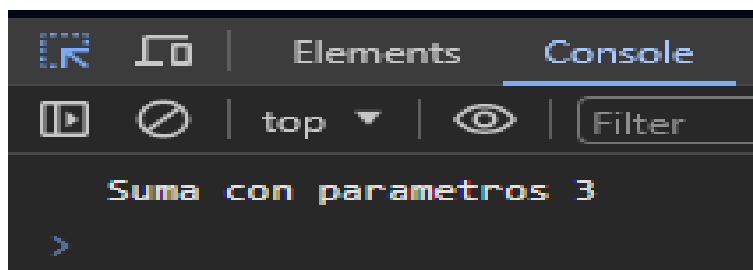
```
// con parametros

function suma(pnumUno,pnumDos){

    let numUno;
    let numDos;
    let sumar;
    numUno = pnumUno;
    numDos =pnumDos

    sumar = numUno + numDos;

    return sumar;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const sumarExp = 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:

```
<script>

    console.log("Suma con Expresión " + operacionesExp("sumaExp",3,4));

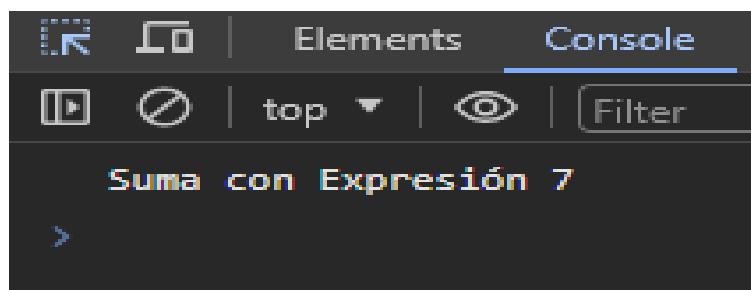
</script>
```

```
//Expresion

const sumaExp = function(pnumUno,pnumDos){
    let numUno;
    let numDos;
    let sumar;
    numUno = pnumUno;
    numDos =pnumDos

    sumar = numUno + numDos;

    return sumar;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
restar(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:

```
<script>

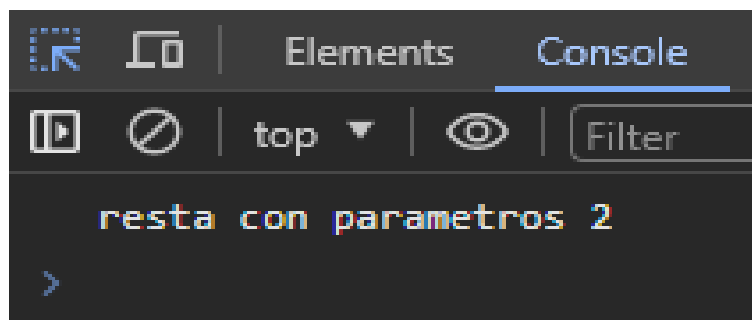
    console.log("resta con parametros " + operaciones("resta",8,6));

</script>
```

```
function resta(pnumUno,pnumDos){
    let numUno;
    let numDos;
    let restar;
    numUno = pnumUno;
    numDos =pnumDos

    restar = numUno - numDos;

    return restar;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const restarExp= function (pnumUno,pnumDos)`

Versión: 2.0

Descripción:

Función que resta dos números

numUno

Tipo de variable: INT

numDos

Tipo de variable: INT

restar

Tipo de variable: INT

Código:

```
<script>

    console.log("resta con Expresion " + operacionesExp("restaExp",9,8));

</script>
```

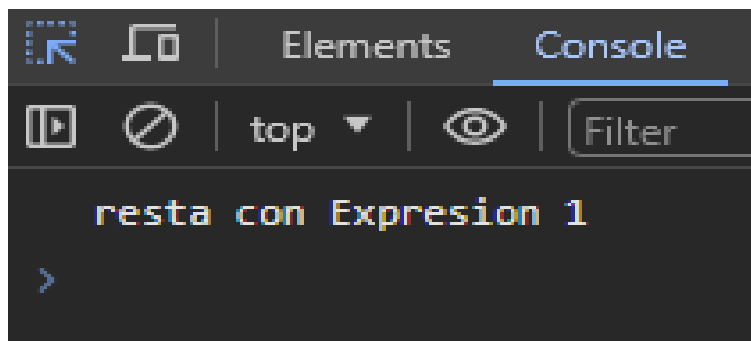
```
const restarExp = function(pnumUno,pnumDos){

    let numUno;
    let numDos;
    let restar;
    numUno = pnumUno;
    numDos =pnumDos

    restar = numUno - numDos;

    return restar;

}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
multiplicar(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:

```
<script>

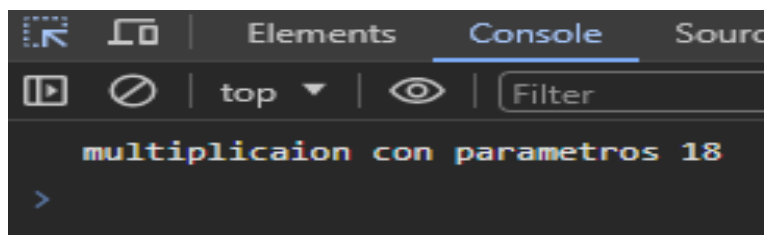
    console.log("multiplicaion con parametros " + operaciones("multiplicacion",6,3));

</script>
```

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

    multiplicar = numUno * numDos;

    return multiplicar;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const multiplicarExp = 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:

```
<script>

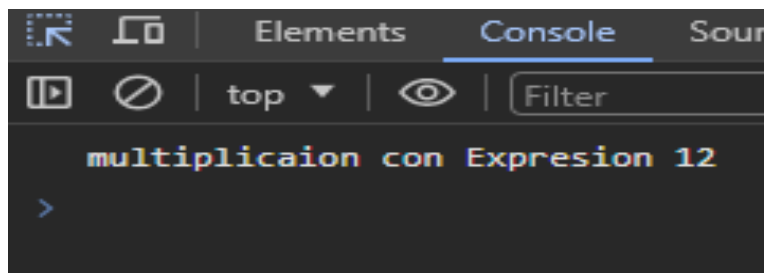
    console.log("multiplicaion con Expresion " + operacionesExp("multiplicacionExp",2,6));

</script>
```

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

    multiplicar = numUno * numDos;

    return multiplicar;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
dividir(pnumUno,pnumDos)

Versión: 1.0

Descripción:

Función que divide dos números

numUno Tipo de variable: FLOAT

numDos Tipo de variable: FLOAT

dividir Tipo de variable: FLOAT

Código:

```
<script>

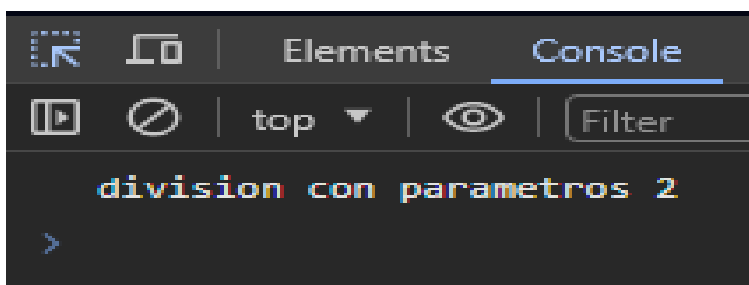
    console.log("division con parametros " + operaciones("division",6,3));

</script>
```

```
function division(pnumUno,pnumDos){
    let numUno;
    let numDos;
    let dividir;
    numUno = pnumUno;
    numDos =pnumDos

    dividir = numUno / numDos;

    return dividir;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const dividirExp= function (pnumUno,pnumDos)

Versión: 2.0

Descripción:

Función que divide dos números

numUno

Tipo de variable: FLOAT

numDos

Tipo de variable: FLOAT

dividir

Tipo de variable: FLOAT

Código:

```
<script>

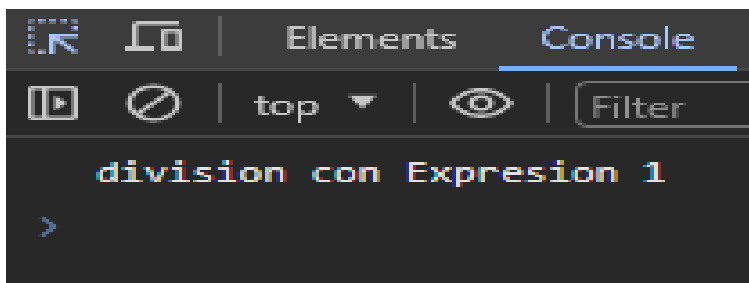
    console.log("division con Expresion " + operacionesExp("divisionExp",6,6));

</script>
```

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

    dividir = numUno / numDos;

    return dividir;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

operaciones(poperador,pnumUno,pnumDos)

Versión: 1.0

Descripción:

Función que ayuda a realizar Operaciones aritméticas

numUno

Tipo de variable: INT

numDos

Tipo de variable: INT

operador

Tipo de variable: string

Código:

```
function operaciones(poerador,pnumUno,pnumDos){
  let numUno;
  let numDos;
  let operador = poerador;
  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 "operacion no valida";
  }
}
```

```
<script>

console.log("Suma con parametros " + operaciones("suma",1,2));

console.log("resta con parametros " + operaciones("resta",8,6));

console.log("multiplicaion con parametros " + operaciones("multiplicacion",6,3));

console.log("division con parametros " + operaciones("division",6,3));

</script>
```

Suma con parametros 3

resta con parametros 2

multiplicaion con parametros 18

division con parametros 2



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const operacionesExp = function(poerador,pnumUno,pnumDos)

Versión: 2.0

Descripción:

Función que ayuda a realizar Operaciones aritméticas

numUno

Tipo de variable: INT

numDos

Tipo de variable: INT

operador

Tipo de variable: string

Código:

```
const operacionesExp = function(poerador,pnumUno,pnumDos){
  let numUno;
  let numDos;
  let operador = poerador;
  numUno = pnumUno;
  numDos = pnumDos;

  if(operador == sumaExp){
    return sumaExp(numUno,numDos);
  }else if(operador == restaExp){
    return restaExp(numUno,numDos);
  }else if( operador == multiplicacionExp ){
    return multiplicacionExp(numUno,numDos);
  }else if(operador == divisionExp){
    return divisionExp(numUno,numDos);
  }else{
    return "operacion no valida";
  }
}
```

```
<script>

console.log("Suma con Expresión " + operacionesExp("sumaExp",3,4));

console.log("resta con Expresion " + operacionesExp("restaExp",9,8));

console.log("multiplicaion con Expresion " + operacionesExp("multiplicacionExp",2,6));

console.log("division con Expresion " + operacionesExp("divisionExp",6,6));

</script>
```

```
top | Filter

Suma con Expresión 7
resta con Expresion 1
multiplicaion con Expresion 12
division con Expresion 1
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

3. Calcular porcentaje

| | | |
|--|-------------------------|---------------------|
| Nombre de la función: porcentajes(pnumero,pporcentaje) | | Versión: 1.0 |
| Descripción: Función que calcula el porcentaje de un número | | |
| numero | Tipo de variable: INT | |
| porcentaje | Tipo de variable: float | |
| Código: <pre><script> numero = 10; console.log("El porcentaje con Parámetro es: "+porcentajesExp(numero) + "%"); </script></pre> <pre>//Como parámetro function porcentajes(pnumero){ let numero; let porcentaje; numero = pnumero; porcentaje = numero / 100; return porcentaje; }</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: <code>const porcentajesExp = function(pnumero,pporcentaje)</code> | Versión: 2.0 |
|---|---------------------|

| |
|---|
| Descripción: Función que calcula el porcentaje de un numero |
|---|

| | |
|--------|-----------------------|
| numero | Tipo de variable: INT |
|--------|-----------------------|

| | |
|------------|-------------------------|
| porcentaje | Tipo de variable: float |
|------------|-------------------------|

Código:

```
<script>
  numero = 20;
  console.log("El porcentaje con Expresión es: "+porcentajesExp(numero) + "%");
</script>
```

```
//Como Expresión
function porcentajesExp(pnumero){
  let numero;
  let porcentaje;
  numero = pnumero;

  porcentaje = numero / 100;
  return porcentaje;
}
```

```
top ▾ | Filter
El porcentaje con Expresión es: 0.2%
>
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

4. Promedio de tres notas

| | | |
|---|-------------------------|---------------------|
| Nombre de la función: calcularNota(pnotaUno,pnotaDos,pnotaTres) | | Versión: 1.0 |
| Descripción: Función que calcula el porcentaje de tres notas | | |
| notaUno | Tipo de variable: float | |
| notaDos | Tipo de variable: float | |
| notaTres | Tipo de variable: float | |
| promedio | Tipo de variable: float | |
| Código: <pre><script> notaUno = 4.0; notaDos = 1.0; notaTres = 4.0; console.log("El promedio de las tres notas es: " + calcularNota(notaUno, notaDos, notaTres)); </script></pre> <pre>//con parametros function calcularNota(pnotaUno,pnotaDos,pnotaTres){ let notaUno = pnotaUno; let notaDos = pnotaDos; let notaTres = pnotaTres; let promedio; promedio = (notaUno + notaDos + notaTres) / 3; return promedio; }</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|--|---------------------|
| Nombre de la función: <code>const calcularNotaExp = function(pnotaUno,pnotaDos,pnotaTres)</code> | Versión: 2.0 |
|--|---------------------|

| |
|--|
| Descripción: Función que calcula el porcentaje de tres notas |
|--|

| | |
|----------|-------------------------|
| notaUno | Tipo de variable: float |
| notaDos | Tipo de variable: float |
| notaTres | Tipo de variable: float |
| promedio | Tipo de variable: float |

| |
|--|
| <p>Código:</p> <pre><script> notaUno = 2.0; notaDos = 3.0; notaTres = 4.0; console.log("El promedio de las tres notas como Expresión es: " + calcularNotaExp(notaUno, notaDos, notaTres)); </script></pre> <pre>//Como Expresión const calcularNotaExp = function(pnotaUno,pnotaDos,pnotaTres){ let notaUno = pnotaUno; let notaDos = pnotaDos; let notaTres = pnotaTres; let promedio; promedio = (notaUno + notaDos + notaTres) / 3; return promedio; }</pre> <pre>El promedio de las tres notas como Expresión es: 3 ></pre> |
|--|



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

5. Porcentaje de tres notas

| | | |
|--|-------------------------|--------------|
| Nombre de la función: calcularProm(psuma,pnotas) | | Versión: 1.0 |
| Descripción: Función que calcula el promedio de tres notas | | |
| suma | Tipo de variable: float | |
| porcentaje | Tipo de variable: float | |
| resultado | Tipo de variable: float | |
| Código: | | |
| <pre><script> let nota1 = calcularPromExp(4.0,0.3); let nota2 = calcularPromExp(3.0,0.3); let nota3 = calcularPromExp(5.0,0.4); let suma; console.log("El porcentaje de la nota 1 es: " + nota1 + "%"); console.log("El porcentaje de la nota 2 es: " + nota2 + "%"); console.log("El porcentaje de la nota 3 es: " + nota3 + "%"); suma = (nota1 + nota2 + nota3); console.log("La suma del porcentaje de sus notas es de: "+suma + "%"); </script></pre> | | |
| <pre>//como parámetro function calcularProm(pnota,pporcentaje){ let nota = pnota; let porcentaje = pporcentaje; let resultado = nota * porcentaje; return resultado; }</pre> | | |
| <pre>El porcentaje de la nota 1 es: 1.2% El porcentaje de la nota 2 es: 0.8999999999999999% El porcentaje de la nota 3 es: 2% La suma del porcentaje de sus notas es de: 4.1% ></pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | | |
|---|-------------------------|---------------------|
| Nombre de la función: const calcularPromExp = function(psuma,pnotas) | | Versión: 2.0 |
| Descripción: Función que calcula el promedio de tres notas | | |
| Suma | Tipo de variable: float | |
| notas | Tipo de variable: float | |
| resultados | Tipo de variable: float | |
| Código: | | |
| <pre><script> let nota1 = calcularPromExp(5.0,0.3); let nota2 = calcularPromExp(3.0,0.3); let nota3 = calcularPromExp(4.0,0.4); let suma; console.log("El porcentaje de la nota 1 como Expresión es: " + nota1 + "%"); console.log("El porcentaje de la nota 2 como Expresión es: " + nota2 + "%"); console.log("El porcentaje de la nota 3 como Expresión es: " + nota3 + "%"); suma = (nota1 + nota2 + nota3); console.log("La suma del porcentaje de sus notas es de: "+suma + "%"); </script></pre> | | |
| <pre>//Como Expresión const calcularPromExp = function(pnota,pporcentaje){ let nota = pnota; let porcentaje = pporcentaje; let resultado = nota * porcentaje; return resultado; }</pre> | | |
| <pre>El porcentaje de la nota 1 como Expresión es: 1.5% El porcentaje de la nota 2 como Expresión es: 0.8999999999999999% El porcentaje de la nota 3 como Expresión es: 1.6% La suma del porcentaje de sus notas es de: 4% ></pre> | | |

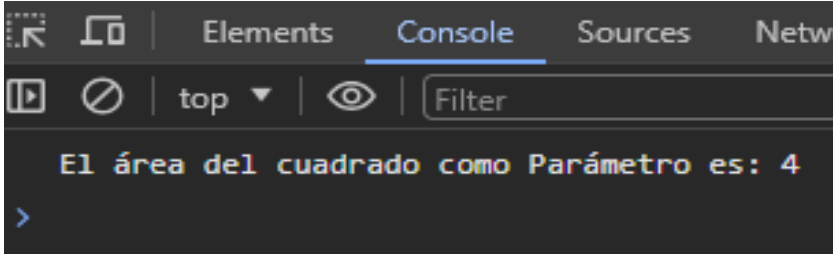


Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

6. Calcular área de figuras

| | | |
|--|-----------------------|---------------------|
| Nombre de la función: areasCuadrado(plado) | | Versión: 1.0 |
| Descripción: Función que calcula el área de un cuadrado | | |
| lado | Tipo de variable: Int | |
| areaCu | Tipo de variable: Int | |
| Código: <pre><script> console.log("El área del cuadrado como Parámetro es: "+areasCuadrado(2)); </script> //como parámetro function areasCuadrado(plado){ let lado; let areaCu; //area Cuadrado lado = plado; areaCu = lado*lado; return areaCu; }</pre>  | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
`const areasCuadradoExp = function(plado)`

Versión: 2.0

Descripción:

Función que calcula el área de un cuadrado

lado

Tipo de variable: Int

areaCu

Tipo de variable: Int

Código:

```
<script>

  console.log("El área del cuadrado como Expresión es: "+areasCuadradoExp(4));

</script>
```

```
//Como Expresión

const areasCuadradoExp = function(plado){

  let lado;
  let areaCu;      //area Cuadrado
  lado = plado;
  areaCu = lado*lado
  return areaCu;

}
```

The screenshot shows the browser's developer console with the 'Console' tab selected. It displays the output of the JavaScript code: 'El área del cuadrado como Expresión es: 16'. The console interface includes standard icons for opening the console, disabling it, and a filter input field.



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

areaRectangulo(pbaseRe,palturaRe)

Versión: 1.0

Descripción:

Función que calcula el área de un rectángulo

baseRe

Tipo de variable: Int

alturaRe

Tipo de variable: Int

areaRe

Tipo de variable: Int

Código:

```
<script>

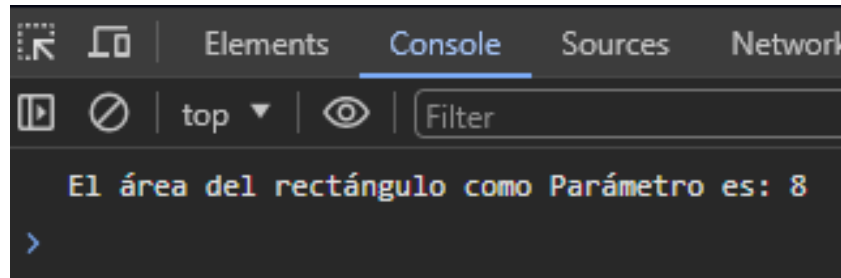
    console.log("El área del rectángulo como Parámetro es: "+ areaRectangulo(2,4));

</script>
```

```
function areaRectangulo(pbaseRe,palturaRe){

    let baseRe;      //basa rectangulo
    let alturaRe;    //altura rectangulo
    let areaRe;      //area Rectangulo
    baseRe = pbaseRe;
    alturaRe = palturaRe;

    areaRe = baseRe * alturaRe;
    return areaRe;
}
```





Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const areaRectanguloExp = function(pbaseRe,palturaRe)

Versión: 2.0

Descripción:

Función que calcula el área de un rectángulo

baseRe

Tipo de variable: Int

alturaRe

Tipo de variable: Int

areaRe

Tipo de variable: Int

Código:

```
<script>

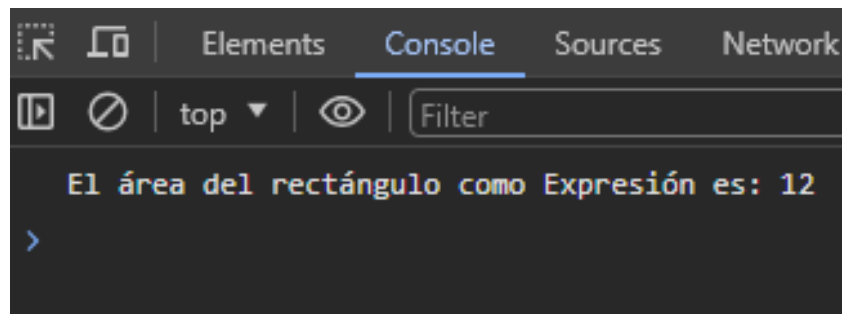
  console.log("El área del rectángulo como Expresión es: "+ areaRectanguloExp(3,4));

</script>
```

```
const areaRectanguloExp = function(pbaseRe,palturaRe){

  let baseRe;      //basa rectangulo
  let alturaRe;    //altura rectangulo
  let areaRe;      //area Rectangulo
  baseRe = pbaseRe;
  alturaRe = palturaRe;

  areaRe = baseRe * alturaRe;
  return areaRe;
}
```





Tecnólogo en Análisis y Desarrollo de Software

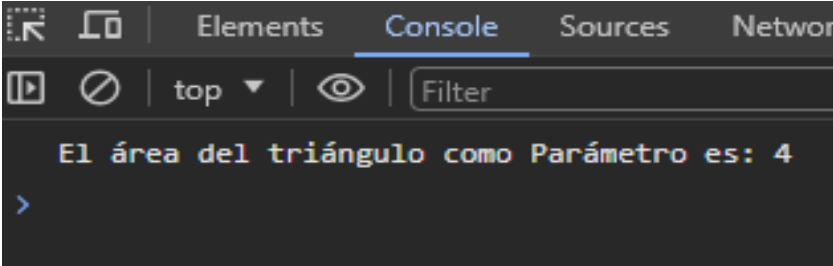
Ficha

Funciones JS

| | |
|--|---------------------|
| Nombre de la función: areaTriangulo(pbaseTri,palturaTri) | Versión: 1.0 |
|--|---------------------|

| |
|--|
| Descripción: Función que calcula el área de un triangulo |
|--|

| | |
|-----------|-----------------------|
| baeTri | Tipo de variable: Int |
| alturaTri | Tipo de variable: Int |
| areaTri | Tipo de variable: Int |

| |
|---|
| <p>Código:</p> <pre><script> console.log("El área del triángulo como Parámetro es: "+ areaTriangulo(4,2)); </script></pre> <pre>function areaTriangulo(pbaseTri,palturaTri){ let baseTri; //base Triangulo let alturaTri; //altura Triangulo let areaTri; //area Triangulo baseTri = pbaseTri; alturaTri = palturaTri; areaTri = (baseTri * alturaTri)/2 return areaTri; }</pre>  |
|---|



Tecnólogo en Análisis y Desarrollo de Software

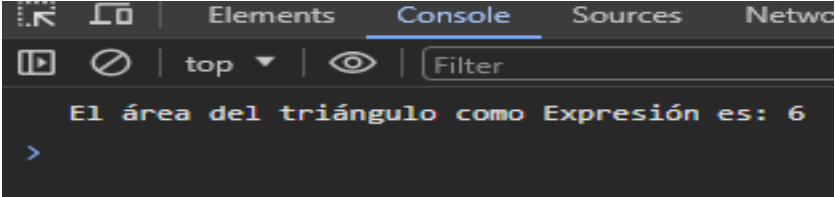
Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: const areaTrianguloExp = function(pbaseTri,palturaTri) | Versión: 2.0 |
|---|---------------------|

| |
|--|
| Descripción: Función que calcula el área de un triángulo |
|--|

| | |
|-----------|-----------------------|
| baseTri | Tipo de variable: Int |
| alturaTri | Tipo de variable: Int |
| areaTri | Tipo de variable: Int |

| |
|--|
| Código: <pre><script> console.log("El área del triángulo como Expresión es: "+ areaTriangulo(3,4)); </script></pre> <pre>const areaTrianguloExp = function(pbaseTri,palturaTri){ let baseTri; //base Triangulo let alturaTri; //altura Triangulo let areaTri; //area Triangulo baseTri = pbaseTri; alturaTri = palturaTri; areaTri = (baseTri * alturaTri)/2; return areaTri; }</pre>  |
|--|



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

7. Calcular sueldo

| | |
|---|--------------|
| Nombre de la función: function sueldo(pdiaT, pvalorD) | Versión: 1.0 |
|---|--------------|

Descripción:
Función que calcula el sueldo de una persona

| | |
|--------|-----------------------|
| diaT | Tipo de variable: Int |
| valorD | Tipo de variable: Int |
| pago | Tipo de variable: Int |

Código:

```
<script>
  valorD = 43333;
  diaT = 10;

  let sueldo = sueldo(diaT, valorD);

  console.log(
    "Su pago con Parámetro es de: " + sueldo
  );
</script>
```

```
//Como parámetro
function sueldo(pdiaT, pvalorD) {
  let diaT;
  let valorD;
  let pago;

  diaT = pdiaT;
  valorD = pvalorD;
  pago = diaT * valorD;
  return pago;
}
```

```
top ▾ | Filter
Su pago con Parámetro es de: 433330
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const sueldoExp = function(pdiaT, pvalorD)

Versión: 2.0

Descripción:

Función que calcula el sueldo de una persona

diaT

Tipo de variable: Int

valorD

Tipo de variable: Int

pago

Tipo de variable: Int

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);

  console.log(
    "Su pago con Expresión es de: " + sueldo + "."
  );
</script>
```

```
const sueldoExp = function(pdiaT, pvalorD) {
  let diaT;
  let valorD;
  let pago;

  diaT = pdiaT;
  valorD = pvalorD;
  pago = diaT * valorD;
  return pago;
}
```

```
Su pago con Expresión es de: $866660
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function salud(pago)

Versión: 1.0

Descripción:

Función que calcula el porcentaje para la salud de una persona

saludP

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 10;

  let sueldo = sueldo(diaT, valorD);
  let salud1 = salud(sueldo);

  console.log(
    "El porcentaje para la salud con Parámetro es del " + salud1 + "%"
  );
</script>
```

```
function salud(pago) {
  let saludP = pago * 0.12;
  return saludP;
}
```

top Filter Default levels

El porcentaje para la salud con Parámetro es del 51999.6%



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
`const saludExp = function(pago)`

Versión: 2.0

Descripción:

Función que calcula el sueldo de una persona

salud

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);
  let salud1 = saludExp(sueldo);

  console.log(
    "El porcentaje para la salud con Expresión es del " + salud1 + "%."
  );
</script>
```

```
const saludExp = function(pago) {
  let salud = pago * 0.12;
  return salud;
}
```

```
top ▾ | Filter | Defa
El porcentaje para la salud con Expresión es del 103999.2%
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function pension(pago)

Versión: 1.0

Descripción:

Función que calcula el porcentaje para la pensión de una persona

pensionP

Tipo de variable: float

Código:

```
<script>
valorD = 43333;
diaT = 10;

let sueldo = sueldo(diaT, valorD);

let pension1 = pension(sueldo);

console.log(
  "El porcentaje para la pensión con Parámetro es del " + pension1 + "%."
);
</script>
```

```
function pension(pago) {
  let pensionP = pago * 0.16;
  return pensionP;
}
```

El porcentaje para la pensión con Parámetro es del 69332.8%



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const pensionExp = function(pago)`

Versión: 2.0

Descripción:

Función que calcula el sueldo de una persona

Pension

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 20;
  let pension1 = pensionExp(sueldo);

  console.log(
    "El porcentaje para la pensión con Expresión es del " + pension1 + "%."
  );
</script>
```

```
const pensionExp = function(pago) {
  let pension = pago * 0.16;
  return pension;
}
```

```
top Filter Default
El porcentaje para la pensión con Expresión es del 138665.6%
>
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function arl(pago)

Versión: 1.0

Descripción:

Función que calcula el porcentaje para el arl de una persona

arlP

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 10;

  let sueldo = sueldo(diaT, valorD);
  let arl1 = arl(sueldo);

  console.log(
    "El porcentaje para el arl con Parámetro es del " + arl1 + "%."
  );
</script>
```

```
function arl(pago) {
  let arlP = pago * 0.052;
  return arlP;
}
```

top Filter De

El porcentaje para el arl con Parámetro es del 22533.16%

>



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
`const arlExp = function(pago)`

Versión: 2.0

Descripción:

Función que calcula el sueldo de una persona

arl

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);
  let arl1 = arlExp(sueldo);

  console.log(
    "El porcentaje para el arl con Expresión es del " + arl1 + "%."
  );
</script>
```

```
const arlExp = function(pago) {
  let arl = pago * 0.052;
  return arl;
}
```

El porcentaje para el arl con Expresión es del 45066.32%



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function pagoTotal(pago)

Versión: 1.0

Descripción:

Función que calcula el pago total de una persona

saludP Tipo de variable: float

pensionP Tipo de variable: float

arlP Tipo de variable: float

total Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 10;

  let sueldo = sueldo(diaT, valorD);
  let salud1 = salud(sueldo);
  let pension1 = pension(sueldo);
  let arl1 = arl(sueldo);
  let pagoT = pagoTotal(sueldo);

  console.log(
    "Su pago con Parámetro es de: " + sueldo + ".\n" +
    "El porcentaje para la salud con Parámetro es del " + salud1 + "%.\n" +
    "El porcentaje para la pensión con Parámetro es del " + pension1 + "%.\n" +
    "El porcentaje para el arl con Parámetro es del " + arl1 + "%.\n" +
    "El total de su pago con Parámetro es de " + pagoT + "\n"
  );
</script>
```

```
function pagoTotal(pago) {
  let saludP = salud(pago);
  let pensionP = pension(pago);
  let arlP = arl(pago);
  let total = pago - (saludP + pensionP + arlP);
  return total;
}
```

```
top Filter Default
Su pago con Parámetro es de: 433330.
El porcentaje para la salud con Parámetro es del 51999.6%.
El porcentaje para la pensión con Parámetro es del 69332.8%.
El porcentaje para el arl con Parámetro es del 22533.16%.
El total de su pago con Parámetro es de 289464.44
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const pagoTotalExp = function(pago)

Versión: 2.0

Descripción:

Función que calcula el sueldo de una persona

| | |
|--------|-------------------------|
| saludE | Tipo de variable: float |
|--------|-------------------------|

| | |
|----------|-------------------------|
| pensionE | Tipo de variable: float |
|----------|-------------------------|

| | |
|------|-------------------------|
| arlE | Tipo de variable: float |
|------|-------------------------|

| | |
|-------|-------------------------|
| total | Tipo de variable: float |
|-------|-------------------------|

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);
  let salud1 = saludExp(sueldo);
  let pension1 = pensionExp(sueldo);
  let arl1 = arlExp(sueldo);
  let pagoT = pagoTotalExp(sueldo);

  console.log(
    "Su pago con Expresión es de: " + sueldo + ".\n" +
    "El porcentaje para la salud con Expresión es del " + salud1 + "%.\n" +
    "El porcentaje para la pensión con Expresión es del " + pension1 + "%.\n" +
    "El porcentaje para el arl con Expresión es del " + arl1 + "%.\n" +
    "El total de su pago con Expresión es de " + pagoT + "\n"
  );
</script>
```

```
const pagoTotalExp = function(pago) {
  let saludE = salud(pago);
  let pensionE = pension(pago);
  let arlE = arl(pago);
  let total = pago - (saludE + pensionE + arlE);
  return total;
}
```

```
top Filter Default le

Su pago con Expresión es de: $866660.
El porcentaje para la salud con Expresión es del 103999.2%.
El porcentaje para la pensión con Expresión es del 138665.6%.
El porcentaje para el arl con Expresión es del 45066.32%.
El total de su pago con Expresión es de 578928.88

>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

8. Mayor de dos números

| | | |
|--|-----------------------|--------------|
| Nombre de la función: function validar(pnumUno, pnumDos) | | Versión: 1.0 |
| Descripción: Función que calcula el mayor de dos números | | |
| numUno | Tipo de variable: Int | |
| numDos | Tipo de variable: Int | |
| Código: | | |
| <pre><script> numUno = 5; numDos = 4; console.log("El número uno como Parámetro es: " + numUno); console.log("El número dos como Parámetro es: "+numDos); console.log(validar(numUno,numDos)); </script></pre> | | |
| <pre>function validar(pnumUno, pnumDos) { let numUno; let numDos; numUno = pnumUno; numDos = pnumDos; if (numUno > numDos) { return "El número uno es mayor " + numUno; } else { return "El número dos es mayor " + numDos; } }</pre> | | |
| <pre>El número uno como Parámetro es: 5 El número dos como Parámetro es: 4 El número uno es mayor 5</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
`const validarExp = function (pnumUno, pnumDos)`

Versión: 2.0

Descripción:

Función que calcula el mayor de dos números

numUno Tipo de variable: Int

numDos Tipo de variable: Int

Código:

```
<script>

    numUno = 15;
    numDos = 24;

    console.log("El número uno como Expresión es: " + numUno);
    console.log("El número dos como Expresión es: "+numDos );
    console.log(validarExp(numUno,numDos));

</script>
```

```
const validarExp = function (pnumUno, pnumDos) {
    let numUno;
    let numDos;

    numUno = pnumUno;
    numDos = pnumDos;

    if (numUno > numDos) {
        return "El número uno es mayor " + numUno;
    } else {
        return "El número dos es mayor " + numDos;
    }
};
```

```
El número uno como Expresión es: 15
El número dos como Expresión es: 24
El número dos es mayor 24
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

9. Fecha nacimiento

| | | |
|--|------------------------|---------------------|
| Nombre de la función: edades(pfechaAc,pfechaNac) | | Versión: 1.0 |
| Descripción: Función que calcula la edad de una persona | | |
| fechaAc | Tipo de variable: Date | |
| FechaNac | Tipo de variable: Date | |
| diferencia | Tipo de variable: Int | |
| edad | Tipo de variable: Int | |
| Código: <pre><script> fechaAc = new Date(); fechaNac = new Date("07/08/2006"); console.log("Tú edad como parámetros es: "+edades(fechaAc,fechaNac) + " años"); // 17 años </script> let fechaAc; let fechaNac; function edades(pfechaAc, pfechaNac) { fechaAc = pfechaAc; fechaNac = pfechaNac; let diferencia; let edad; diferencia = fechaAc - fechaNac; edad = Math.floor(diferencia / (1000 * 60 * 60 * 24 * 365.25)); return edad; }</pre>  | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función: const edadesExp = function(pfechaAc,pfechaNac)

Versión: 2.0

Descripción:

Función que calcula la edad de una persona

fechaAc Tipo de variable: Date

fechaNac Tipo de variable: Date

diferencia Tipo de variable: Int

edad Tipo de variable: Int

Código:

```
<script>

  fechaAc = new Date();
  fechaNacExp = new Date("07/08/2000");

  console.log("Tú edad como Expresión es: "+edadesExp(fechaAc,fechaNacExp) + " años"); // 23 años

</script>
```

```
//Como Expresión

const edadesExp = function (pfechaAc, pfechaNac) {
  let fechaAc;
  let fechaNac;
  fechaAc = pfechaAc;
  fechaNac = pfechaNac;

  let diferencia;
  let edad;

  diferencia = fechaAc - fechaNac;
  edad = Math.floor(diferencia / (1000 * 60 * 60 * 24 * 365.25));
  return edad;
}
```

```
top ▾ | Filter

Tú edad como Expresión es: 23 años

>
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

10. Mayor igual de dos números

| | | |
|--|-----------------------|---------------------|
| Nombre de la función: validar(pnumUno,pnumDos) | | Versión: 1.0 |
| Descripción: Función que calcula mayor igual de dos números | | |
| numUno | Tipo de variable: Int | |
| numDos | Tipo de variable: Int | |
| Código: <pre><script> numUno = 5; numDos = 4; console.log("El número uno como Parámetro es: " + numUno); console.log("El número dos como Parámetro es: " + numDos); console.log(validar(numUno,numDos)); </script></pre> <pre>function validar(pnumUno, pnumDos) { let numUno; let numDos; numUno = pnumUno; numDos = pnumDos; if (numUno == numDos) { return "Los número son iguales"; } else if (numUno > numDos) { return "El número uno es el mayor " + numUno; } else { return "El número dos es el mayor " + numUno; } }</pre> <pre>El número uno como Parámetro es: 5 El número dos como Parámetro es: 4 El número uno es el mayor 5</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const validarExp = function(pnumUno,pnumDos)`

Versión: 2.0

Descripción:

Función que calcular mayor e igual de dos números

numUno

Tipo de variable: Int

numDos

Tipo de variable: Int

Código:

```
<script>

  numUno = 4;
  numDos = 4;

  console.log("El número uno como Expresión es: " + numUno);
  console.log("El número dos como Expresión es: " + numDos);
  console.log( validarExp(numUno,numDos));

</script>
```

```
const validarExp = function (pnumUno, pnumDos) {
  let numUno;
  let numDos;

  numUno = pnumUno;
  numDos = pnumDos;

  if (numUno == numDos) {
    return "Los número son iguales";
  } else if (numUno > numDos) {
    return "El número uno es el mayor " + numUno;
  } else {
    return "El número dos es el mayor " + numUno;
  }
}
```

```
top ▼ | Filter

El número uno como Expresión es: 4
El número dos como Expresión es: 4
Los número son iguales
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

11. Mayor igual de tres números

| | | |
|--|-----------------------|---------------------|
| Nombre de la función: function validar(pnumUno, pnumDos, pnumTres) | | Versión: 1.0 |
| Descripción: Función que calcula mayor e igual de tres números | | |
| numUno | Tipo de variable: Int | |
| numDos | Tipo de variable: Int | |
| numTres | Tipo de variable: Int | |
| Código: <pre><script> numUno = 2; numDos = 2; numTres = 2; console.log("El número uno como Parámetro es: "+numUno); console.log("El número dos como Parámetro es: "+numDos); console.log("El número tres como Parámetro es: "+numTres); console.log(validar(numUno,numDos,numTres)); </script></pre> <pre>function validar(pnumUno, pnumDos, pnumTres) { let numUno; let numDos; let numTres; numUno = pnumUno; numDos = pnumDos; numTres = pnumTres; if (numUno == numDos && numUno == numTres && numDos == numTres) { return "Los tres numeros son iguales"; } else if (numUno > numDos && numUno > numTres) { return "El número uno es el mayor " + numUno; } else if (numDos > numUno && numDos > numTres) { return "El número dos es el mayor " + numTres; } else { return "El número tres es el mayor " + numTres; } }</pre> <pre>El número uno como Parámetro es: 2 El número dos como Parámetro es: 2 El número tres como Parámetro es: 2 Los tres numeros son iguales ></pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const validarExp = function (pnumUno, pnumDos, pnumTres)

Versión: 2.0

Descripción:

Función que calcula mayor igual de tres números

numUno

Tipo de variable: Int

numDos

Tipo de variable: Int

NumTres

Tipo de variable: Int

Código:

```
<script>

  numUno = 3;
  numDos = 2;
  numTres = 5;

  console.log("El número uno como Expresión es: "+numUno);
  console.log("El número dos como Expresión es: "+numDos);
  console.log("El número tres como Expresión es: "+numTres);
  console.log(validarExp(numUno,numDos,numTres));
</script>
```

```
const validarExp = function (pnumUno, pnumDos, pnumTres) {
  let numUno;
  let numDos;
  let numTres;

  numUno = pnumUno;
  numDos = pnumDos;
  numTres = pnumTres;

  if (numUno == numDos && numUno == numTres && numDos == numTres) {
    return "Los tres numeros son iguales";
  } else if (numUno > numDos && numUno > numTres) {
    return "El número uno es el mayor " + numUno;
  } else if (numDos > numUno && numDos > numTres) {
    return "El número dos es el mayor " + numTres;
  } else {
    return "El número tres es el mayor " + numTres;
  }
}
```

```
El número uno como Expresión es: 3
El número dos como Expresión es: 2
El número tres como Expresión es: 5
El número tres es el mayor 5
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

12. Área de tres cuadrados

| | | |
|--|-----------------------|--------------|
| Nombre de la función: function areas(plado) | | Versión: 1.0 |
| Descripción: Función que calcula el área de tres cuadrados | | |
| lado | Tipo de variable: Int | |
| area | Tipo de variable: Int | |
| Código: | | |
| <pre><script> let ladoUno = 5; let ladoDos = 10; let ladoTres = 6; areaUno = areas(ladoUno); areaDos = areas(ladoDos); areaTres = areas(ladoTres); console.log("El área del primer cuadrado como Parámetro es: " + areaUno); console.log("El área del segundo cuadrado como Parámetro es: " + areaDos); console.log("El área del tercer cuadrado como Parámetro es: " + areaTres); </script></pre> | | |
| <pre>//Como parámetro function areas(plado) { let lado; let area; lado = plado; area = lado * lado; return area; }</pre> | | |
| <div>🔍 🔍 top ▼ 🔍 Filter</div> <div>El área del primer cuadrado como Parámetro es: 25</div> <div>El área del segundo cuadrado como Parámetro es: 100</div> <div>El área del tercer cuadrado como Parámetro es: 36</div> <div>> </div> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const areasExp = function (plado)

Versión: 2.0

Descripción:

Función que calcula el área de tres cuadrados

lado

Tipo de variable: Int

area

Tipo de variable: Int

Código:

```
<script>
  let ladoUno = 5;
  let ladoDos = 5;
  let ladoTres = 5;

  areaUno = areasExp(ladoUno);
  areaDos = areasExp(ladoDos);
  areaTres = areasExp(ladoTres);

  console.log("El área del primer cuadrado como Expresión es: " + areaUno);
  console.log("El área del segundo cuadrado como Expresión es: " + areaDos);
  console.log("El área del tercer cuadrado como Expresión es: " + areaTres);
</script>
```

```
//Como Expresiónb
const areasExp = function (plado) {
  let lado;
  let area;

  lado = plado;
  area = lado * lado;
  return area;
}
```

```
top | Filter
El área del primer cuadrado como Expresión es: 25
El área del segundo cuadrado como Expresión es: 25
El área del tercer cuadrado como Expresión es: 25
> |
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: function validar(pareaUno, pareaNos, pareaNres) | Versión: 1.0 |
|---|---------------------|

Descripción:
Función que calcula el área de tres cuadrados y retorna que área es mayor

| | |
|----------|-----------------------|
| areaUno | Tipo de variable: Int |
| areaDos | Tipo de variable: Int |
| areaTres | Tipo de variable: Int |

Código:

```
<script>
  let ladoUno = 5;
  let ladoDos = 10;
  let ladoTres = 6;

  areaUno = areas(ladoUno);
  areaDos = areas(ladoDos);
  areaTres = areas(ladoTres);

  console.log("El área del primer cuadrado como Parámetro es: " + areaUno);
  console.log("El área del segundo cuadrado como Parámetro es: " + areaDos);
  console.log("El área del tercer cuadrado como Parámetro es: " + areaTres);

  console.log(validar(areaUno,areaDos,areaTres));
</script>
```

```
function validar(pareaUno, pareaNos, pareaNres) {
  let areaUno;
  let areaDos;
  let areaTres;
  areaUno = pareaNUno;

  areaDos = pareaNos;
  areaTres = pareaNres;

  if (areaUno == areaDos && areaUno == areaTres && areaDos == areaTres) {
    return "Las áreas son iguales";
  } else if (areaUno > areaDos && areaUno > areaTres) {
    return "El área del primer cuadrado es mayor. " + areaUno;
  } else if (areaDos > areaUno && areaDos > areaTres) {
    return "El área del segundo cuadrado es mayor." + areaDos;
  } else {
    return "El área del tercer cuadrado es mayor." + areaTres;
  }
}
```

```
El área del primer cuadrado como Parámetro es: 25
El área del segundo cuadrado como Parámetro es: 100
El área del tercer cuadrado como Parámetro es: 36
El área del segundo cuadrado es mayor.100
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const validarExp = function (pareaUno, pareaDos, pareaTres)

Versión: 2.0

Descripción:

Función que calcula el área de tres cuadrados y retorna cual es mayor

areaUno Tipo de variable: Int

areaDos Tipo de variable: Int

areaTres Tipo de variable: Int

Código:

```
<script>
  let ladoUno = 5;
  let ladoDos = 5;
  let ladoTres = 5;

  areaUno = areasExp(ladoUno);
  areaDos = areasExp(ladoDos);
  areaTres = areasExp(ladoTres);

  console.log("El área del primer cuadrado como Expresión es: " + areaUno);
  console.log("El área del segundo cuadrado como Expresión es: " + areaDos);
  console.log("El área del tercer cuadrado como Expresión es: " + areaTres);

  console.log(validarExp(areaUno,areaDos,areaTres));
</script>
```

```
const validarExp = function (pareaUno, pareaDos, pareaTres) {
  let areaUno;
  let areaDos;
  let areaTres;

  areaUno = pareaUno;
  areaDos = pareaDos;
  areaTres = pareaTres;
  if (areaUno == areaDos && areaUno == areaTres && areaDos == areaTres) {
    return "Las áreas son iguales";
  } else if (areaUno > areaDos && areaUno > areaTres) {
    return "El área del primer cuadrado es mayor." + areaUno;
  } else if (areaDos > areaUno && areaDos > areaTres) {
    return "El área del segundo cuadrado es mayor." + areaDos;
  } else {
    return "El área del tercer cuadrado es mayor." + areaTres;
  }
}
```

```
El área del primer cuadrado como Expresión es: 25
El área del segundo cuadrado como Expresión es: 25
El área del tercer cuadrado como Expresión es: 25
Las áreas son iguales
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

13. Fecha nacimiento de tres personas

| | | |
|---|------------------------|--------------|
| Nombre de la función: function edades(pfchaAc, pfchaNac) | | Versión: 1.0 |
| Descripción: Función que la edad de tres personas | | |
| fechaAc | Tipo de variable: Date | |
| fechaNac | Tipo de variable: Date | |
| diferencia | Tipo de variable: Int | |
| edad | Tipo de variable: Int | |
| Código: | | |
| <pre><script> fechaAc = new Date(); fechaNac1 = new Date("07/08/2006"); fechaNac2 = new Date("07/08/2000"); fechaNac3 = new Date("07/08/2012"); edadUno = edades(fechaAc, fechaNac1); edadDos = edades(fechaAc, fechaNac2); edadTres = edades(fechaAc, fechaNac3); console.log("La edad de la primera persona como Parámetro es: " + edadUno + " años"); console.log("La edad de la segunda persona como Parámetro es: " + edadDos + " años"); console.log("La edad de la tercera persona como Parámetro es: " + edadTres + " años"); </script></pre> | | |
| <pre>//como parámetro function edades(pfchaAc, pfchaNac) { let fechaAc; let fechaNac; fechaAc = pfchaAc; fechaNac = pfchaNac; let diferencia; let edad; diferencia = fechaAc - fechaNac; edad = Math.floor(diferencia / (1000 * 60 * 60 * 24 * 365.25)); return edad; }</pre> | | |
| <pre>La edad de la primera persona como Parámetro es: 17 años La edad de la segunda persona como Parámetro es: 23 años La edad de la tercera persona como Parámetro es: 11 años</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: const edadesExp = function (pfechaAc, pfechaNac) | Versión: 2.0 |
|---|---------------------|

Descripción:
Función que la edad de tres personas

| | |
|------------|------------------------|
| fechaAc | Tipo de variable: Date |
| fechaNac | Tipo de variable: Date |
| diferencia | Tipo de variable: Int |
| edad | Tipo de variable: Int |

Código:

```
<script>
    fechaAc = new Date();
    fechaNac1 = new Date("07/08/2008");
    fechaNac2 = new Date("07/08/1987");
    fechaNac3 = new Date("07/08/2010");

    edadUno = edadesExp(fechaAc, fechaNac1);
    edadDos = edadesExp(fechaAc, fechaNac2);
    edadTres = edadesExp(fechaAc, fechaNac3);

    console.log("La edad de la primera persona como Expresión es: " + edadUno + " años");
    console.log("La edad de la segunda persona como Expresión es: " + edadDos + " años");
    console.log("La edad de la tercera persona como Expresión es: " + edadTres + " años");
</script>
```

```
//Comom Expresión
const edadesExp = function (pfechaAc, pfechaNac) {
    let fechaAc;
    let fechaNac;
    let diferencia;
    let edad;

    fechaAc = pfechaAc;
    fechaNac = pfechaNac;

    diferencia = fechaAc - fechaNac;
    edad = Math.floor(diferencia / (1000 * 60 * 60 * 24 * 365.25));
    return edad;
}
```

```
top | Filter | De

La edad de la primera persona como Expresión es: 15 años
La edad de la segunda persona como Expresión es: 36 años
La edad de la tercera persona como Expresión es: 13 años
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function validar(pedadP)

Versión: 1.0

Descripción:

Función que la edad de tres personas y me retorna si la persona es o no mayor de edad

edadP

Tipo de variable: Int

Código:

```
<script>
    fechaAc = new Date();
    fechaNac1 = new Date("07/08/2006");
    fechaNac2 = new Date("07/08/2000");
    fechaNac3 = new Date("07/08/2012");

    edadUno = edades(fechaAc, fechaNac1);
    edadDos = edades(fechaAc, fechaNac2);
    edadTres = edades(fechaAc, fechaNac3);

    console.log("La edad de la primera persona como Parámetro es: " + edadUno + " años");
    console.log("La edad de la segunda persona como Parámetro es: " + edadDos + " años");
    console.log("La edad de la tercera persona como Parámetro es: " + edadTres + " años");

    console.log("La primera persona es " + validar(edadUno));
    console.log("La segunda persona es " + validar(edadDos));
    console.log("La tercera persona es " + validar(edadTres));
</script>
```

```
function validar(pedadP) {
    let edadP;
    edadP = pedadP;

    if (edadP >= 18) {
        return "Mayor de edad";
    } else {
        return "Menor de edad";
    }
}
```

```
La edad de la primera persona como Parámetro es: 17 años
La edad de la segunda persona como Parámetro es: 23 años
La edad de la tercera persona como Parámetro es: 11 años
La primera persona es Menor de edad
La segunda persona es Mayor de edad
La tercera persona es Menor de edad
```

>



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const validarExp = function (pedadP)`

Versión: 2.0

Descripción:

Función que la edad de tres personas y me retorna si la persona es o no mayor de edad

edadP

Tipo de variable: Int

Código:

```
<script>
    fechaAc = new Date();
    fechaNac1 = new Date("07/08/2008");
    fechaNac2 = new Date("07/08/1987");
    fechaNac3 = new Date("07/08/2010");

    edadUno = edadesExp(fechaAc, fechaNac1);
    edadDos = edadesExp(fechaAc, fechaNac2);
    edadTres = edadesExp(fechaAc, fechaNac3);

    console.log("La edad de la primera persona como Expresión es: " + edadUno + " años");
    console.log("La edad de la segunda persona como Expresión es: " + edadDos + " años");
    console.log("La edad de la tercera persona como Expresión es: " + edadTres + " años");

    console.log("La primera persona es " + validarExp(edadUno));
    console.log("La segunda persona es " + validarExp(edadDos));
    console.log("La tercera persona es " + validarExp(edadTres));
</script>
```

```
const validarExp = function(pedadP) {
    let edadP;
    edadP = pedadP;

    if (edadP >= 18) {
        return "Mayor de edad";
    } else {
        return "Menor de edad";
    }
}
```

```
La edad de la primera persona como Expresión es: 15 años
La edad de la segunda persona como Expresión es: 36 años
La edad de la tercera persona como Expresión es: 13 años
La primera persona es Menor de edad
La segunda persona es Mayor de edad
La tercera persona es Menor de edad
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: function promedio(psuma, pnumEdad) | Versión: 1.0 |
|---|---------------------|

| |
|---|
| Descripción: Función que calcula el promedio de tres edades |
|---|

| | |
|----------|-------------------------|
| suma | Tipo de variable: Int |
| numEdad | Tipo de variable: Int |
| promedio | Tipo de variable: float |

| |
|---|
| Código: <pre><script> fechaAc = new Date(); fechaNac1 = new Date("07/08/2006"); fechaNac2 = new Date("07/08/2000"); fechaNac3 = new Date("07/08/2012"); edadUno = edades(fechaAc, fechaNac1); edadDos = edades(fechaAc, fechaNac2); edadTres = edades(fechaAc, fechaNac3); console.log("La edad de la primera persona como Parámetro es: " + edadUno + " años"); console.log("La edad de la segunda persona como Parámetro es: " + edadDos + " años"); console.log("La edad de la tercera persona como Parámetro es: " + edadTres + " años"); console.log("La primera persona es " + validar(edadUno)); console.log("La segunda persona es " + validar(edadDos)); console.log("La tercera persona es " + validar(edadTres)); suma = edadUno + edadDos + edadTres; console.log(promedio(suma, 3)); </script> function promedio(psuma, pnumEdad) { let suma; let numEdad; let promedio; suma = psuma; numEdad = pnumEdad; promedio = suma / pnumEdad; if (promedio >= 18) { return "El promedio de edad se encuentra dentro de la mayoría de edad " + promedio; } else { return "El promedio de edad no se encuentra dentro de la mayoría de edad " + promedio; } } La edad de la primera persona como Parámetro es: 17 años La edad de la segunda persona como Parámetro es: 23 años La edad de la tercera persona como Parámetro es: 11 años La primera persona es Menor de edad La segunda persona es Mayor de edad La tercera persona es Menor de edad El promedio de edad no se encuentra dentro de la mayoría de edad 17 ></pre> |
|---|



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: const promedioExp = function (psuma, pnumEdad) | Versión: 2.0 |
|---|---------------------|

Descripción:
Función que calcula el promedio de tres edades

| | |
|----------|-------------------------|
| suma | Tipo de variable: Int |
| numEdad | Tipo de variable: Int |
| promedio | Tipo de variable: float |

Código:

```
<script>
    fechaAc = new Date();
    fechaNac1 = new Date("07/08/2008");
    fechaNac2 = new Date("07/08/1987");
    fechaNac3 = new Date("07/08/2010");

    edadUno = edadesExp(fechaAc, fechaNac1);
    edadDos = edadesExp(fechaAc, fechaNac2);
    edadTres = edadesExp(fechaAc, fechaNac3);

    console.log("La edad de la primera persona como Expresión es: " + edadUno + " años");
    console.log("La edad de la segunda persona como Expresión es: " + edadDos + " años");
    console.log("La edad de la tercera persona como Expresión es: " + edadTres + " años");

    console.log("La primera persona es " + validarExp(edadUno));
    console.log("La segunda persona es " + validarExp(edadDos));
    console.log("La tercera persona es " + validarExp(edadTres));

    suma = edadUno + edadDos + edadTres

    console.log(promedioExp(suma,3))
</script>
```

```
const promedioExp = function (psuma, pnumEdad) {
    let suma;
    let numEdad;
    let promedio;

    suma = psuma;
    numEdad = pnumEdad;

    promedio = suma / pnumEdad;

    if (promedio >= 18) {
        return (
            "El promedio de edad se encuentra dentro de la mayoría de edad " + promedio
        );
    } else {
        return (
            "El promedio de edad no se encuentra dentro de la mayoría de edad " +
            promedio
        );
    }
}
```

```
top | Filter | Default le

La edad de la primera persona como Expresión es: 15 años
La edad de la segunda persona como Expresión es: 36 años
La edad de la tercera persona como Expresión es: 13 años
La primera persona es Menor de edad
La segunda persona es Mayor de edad
La tercera persona es Menor de edad
El promedio de edad se encuentra dentro de la mayoría de edad
21.333333333333332
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

14. Sueldo de una persona

| | | |
|---|-----------------------|---------------------|
| Nombre de la función: function sueldo(pdiaT, pvalorD) | | Versión: 1.0 |
| Descripción: Función que calcula el sueldo de una persona | | |
| diaT | Tipo de variable: Int | |
| valorD | Tipo de variable: Int | |
| pago | Tipo de variable: Int | |

Código:

```
<script>
  diaT = 10;
  valorD = 43333;

  let pago = sueldo(diaT, valorD);
  console.log("El pago como Parámetro es: $" + pago);
</script>
```

```
//Como parámetro
function sueldo(pdiaT, pvalorD) {
  let diaT;
  let valorD;
  let pago;

  diaT = pdiaT;
  valorD = pvalorD;
  pago = diaT * valorD;
  return pago;
}
```

```
top
Filter

El pago como Parámetro es: $433330
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const sueldoExp = function(pdiaT, pvalorD)`

Versión: 2.0

Descripción:

Función que calcula el sueldo de una persona

diaT Tipo de variable: Int

valorD Tipo de variable: Int

pago Tipo de variable: Int

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);

  console.log(
    "Su pago con Expresión es de: $" + sueldo
  );
</script>
```

```
//como Expresión

const sueldoExp = function(pdiaT, pvalorD) {
  let diaT;
  let valorD;
  let pago;

  diaT = pdiaT;
  valorD = pvalorD;
  pago = diaT * valorD;
  return pago;
}
```

```
Su pago con Expresión es de: $866660
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function salud(pago)

Versión: 1.0

Descripción:

Función que calcula el porcentaje para la salud de una persona

saludP

Tipo de variable: float

Código:

```
<script>
  diaT = 10;
  valorD = 43333;

  let pago = sueldo(diaT, valorD);
  console.log("El pago como Parámetro es: $" + pago);

  let salud1 = salud(pago);
  console.log("El porcentaje para la salud como Parámetro es: " + salud1 + "%");
</script>
```

```
function salud(pago) {
  let saludP = pago * 0.12;
  return saludP;
}
```

```
top ▾ | Filter
El pago como Parámetro es: $433330
El porcentaje para la salud como Parámetro es: 51999.6%
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
`const saludExp = function(pago)`

Versión: 2.0

Descripción:

Función que calcula el porcentaje para la salud de una persona

salud

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);
  let salud1 = saludExp(sueldo);

  console.log(
    "El porcentaje para la salud con Expresión es del " + salud1 + "%"
  );
</script>
```

```
const saludExp = function(pago) {
  let salud = pago * 0.12;
  return salud;
}
```

```
El porcentaje para la salud con Expresión es del 103999.2%
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function pension(pago)

Versión: 1.0

Descripción:

Función que calcula el porcentaje para la pensión de una persona

pensionP

Tipo de variable: float

Código:

```
<script>
  diaT = 10;
  valorD = 43333;

  let pago = sueldo(diaT, valorD);
  console.log("El pago como Parámetro es: $" + pago);

  let pension1 = pension(pago);
  console.log("El porcentaje para la pensión como Parámetro es: " + pension1 + "%");
</script>
```

```
function pension(pago) {
  let pensionP = pago * 0.16;
  return pensionP;
}
```

```
top ▼ | Filter | De
El pago como Parámetro es: $433330
El porcentaje para la pensión como Parámetro es: 69332.8%
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const pensionExp = function(pago)`

Versión: 2.0

Descripción:

Función que calcula el porcentaje para la pensión de una persona

Pension

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);

  let pension1 = pensionExp(sueldo);

  console.log(
    "El porcentaje para la pensión con Expresión es del " + pension1 + "%"
  );
</script>
```

```
const pensionExp = function(pago) {
  let pension = pago * 0.16;
  return pension;
}
```

```
top Filter Default
El porcentaje para la pensión con Expresión es del 138665.6%
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function arl(pago)

Versión: 1.0

Descripción:

Función que calcula el porcentaje para el arl de una persona

arlP

Tipo de variable: float

Código:

```
<script>
  diaT = 10;
  valorD = 43333;

  let pago = sueldo(diaT, valorD);
  console.log("El pago como Parámetro es: $" + pago);

  let arl1 = arl(pago);
  console.log("El porcentaje para el arl como Parámetro es: " + arl1 + "%");
</script>
```

```
function arl(pago) {
  let arlP = pago * 0.052;
  return arlP;
}
```

```
top | Filter
El pago como Parámetro es: $433330
El porcentaje para el arl como Parámetro es: 22533.16%
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
`const arlExp = function(pago)`

Versión: 2.0

Descripción:

Función que calcula el porcentaje para el arl de una persona

arl

Tipo de variable: float

Código:

```
<script>
  valorD = 43333;
  diaT = 20;

  let sueldo = sueldoExp(diaT, valorD);

  let arl1 = arlExp(sueldo);

  console.log(
    "El porcentaje para el arl con Expresión es del " + arl1 + "%"
  );
</script>
```

```
const arlExp = function(pago) {
  let arl = pago * 0.052;
  return arl;
}
```

top Filter De

El porcentaje para el arl con Expresión es del 45066.32%



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function subTras(pago)

Versión: 1.0

Descripción:

Función que calcula el subsidio de transporte

salarioM

Tipo de variable: Int

trans

Tipo de variable: Int

Código:

```
<script>
  diaT = 10;
  valorD = 43333;

  let pago = sueldo(diaT, valorD);
  console.log("El pago como Parámetro es: $" + pago);

  let trasportep = subTras(pago)
  console.log("El subsidio de transporte como Parámetro es: $" + trasportep);
</script>
```

```
function subTras(pago) {
  let salarioM = 1300000;
  let trans;
  if (pago <= salarioM * 2) {
    trans = 114000;
  } else {
    trans = 0;
  }
  return trans;
}
```

```
top ▾ | Filter
El pago como Parámetro es: $433330
El subsidio de transporte como Parámetro es: $114000
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const subTrasExp = function(pago)`

Versión: 2.0

Descripción:

Función que calcula el subsidio de transporte

salarioM

Tipo de variable: Int

trasn

Tipo de variable: Int

Código:

```
<script>
    diaT = 20;
    valorD = 43333;

    let pago = sueldoExp(diaT, valorD);
    console.log("El pago como Expresión es: $" + pago);
    let trasportep = subTrasExp(pago)
    console.log("El subsidio de transporte como Expresión es: $" + trasportep);
</script>
```

```
const subTrasExp = function(pago) {
    let salarioM = 1300000;
    let trans;
    if (pago <= salarioM * 2) {
        trans = 114000;
    } else {
        trans = 0;
    }
    return trans;
}
```

```
top | Filter
El pago como Expresión es: $866660
El subsidio de transporte como Expresión es: $114000
>
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function reten(pago)

Versión: 1.0

Descripción:

Función que calcula la retención del salario de una persona

salarioM

Tipo de variable: Int

reten

Tipo de variable: float

Código:

```
<script>
    diaT = 10;
    valorD = 43333;

    let pago = sueldo(diaT, valorD);
    console.log("El pago como Parámetro es: $" + pago);

    let retencion = reten(pago);
    console.log("La retencion como Parámetro es: $" + retencion);
</script>
```

```
function reten(pago) {
    let salarioM = 1300000;
    let retencion;
    if (pago > salarioM * 4) {
        retencion = pago * 0.04;
    } else {
        retencion = 0;
    }
    return retencion;
}
```

```
top Filter
El pago como Parámetro es: $433330
La retencion como Parámetro es: $0
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const retenExp = function(pago)

Versión: 2.0

Descripción:

Función que calcula la retención del salario de una persona

salarioM

Tipo de variable: Int

retencion

Tipo de variable: float

Código:

```
<script>
    diaT = 20;
    valorD = 43333;

    let pago = sueldoExp(diaT, valorD);
    console.log("El pago como Expresión es: $" + pago);

    let retencion = retenExp(pago);
    console.log("La retencion como Expresión es: $" + retencion);
</script>
```

```
const retenExp = function(pago) {
    let salarioM = 1300000;
    let retencion;
    if (pago > salarioM * 4) {
        retencion = pago * 0.04;
    } else {
        retencion = 0;
    }
    return retencion;
}
```

```
El pago como Expresión es: $866660
La retencion como Expresión es: $0
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function pagoT(pago)

Versión: 1.0

Descripción:

Función que calcula el pago total de una persona

| | |
|---------------|-------------------------|
| salario | Tipo de variable: Int |
| saludE | Tipo de variable: float |
| pensionE | Tipo de variable: float |
| arlE | Tipo de variable: float |
| retencion | Tipo de variable: float |
| subTransporte | Tipo de variable: Int |
| descuento | Tipo de variable: float |
| tatalSalario | Tipo de variable: float |

Código:

```
<script>
  diaT = 10;
  valorD = 43333;

  let pago = sueldo(diaT, valorD);
  console.log("El pago como Parámetro es: $" + pago);

  let salud1 = salud(pago);
  console.log("El porcentaje para la salud como Parámetro es: " + salud1 + "%");

  let pension1 = pension(pago);
  console.log("El porcentaje para la pensión como Parámetro es: " + pension1 + "%");

  let arl1 = arl(pago);
  console.log("El porcentaje para el arl como Parámetro es: " + arl1 + "%");

  let trasporte = subTras(pago);
  console.log("El subsidio de transporte como Parámetro es: $" + trasporte);

  let retencion = reten(pago);
  console.log("La retencion como Parámetro es: $" + retencion);

  let pagoTotal = pagoT(pago);
  console.log("Su pago total será de: $" + pagoTotal);
</script>
```

```
function pagoT(pago) {
  let salario = pago;
  let saludE = salud(pago);
  let pensionE = pension(pago);
  let arlE = arl(pago);

  let retencion = reten(pago);
  let subTrasporte = subTras(pago);
  let descuento = saludE + pensionE + arlE;

  let totalSalario = salario + subTrasporte - (retencion + descuento);

  return totalSalario;
}
```

```
top Filter De
El pago como Parámetro es: $433330
El porcentaje para la salud como Parámetro es: 51999.6%
El porcentaje para la pensión como Parámetro es: 69332.8%
El porcentaje para el arl como Parámetro es: 22533.16%
El subsidio de transporte como Parámetro es: $114000
La retencion como Parámetro es: $0
Su pago total será de: $403464.44
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

Const pagoTExp = function(pago)

Versión: 2.0

Descripción:

Función que calcula el pago total de una persona

| | |
|---------------|-------------------------|
| salario | Tipo de variable: Int |
| saludE | Tipo de variable: float |
| pensionE | Tipo de variable: float |
| arlE | Tipo de variable: float |
| retención | Tipo de variable: float |
| subTransporte | Tipo de variable: Int |
| descuento | Tipo de variable: float |
| totalsalario | Tipo de variable: float |

Código:

```
<script>
    diaf = 28;
    valor0 = 43333;

    let pago = sueldoExp(diaf, valor0);
    console.log("El pago como Expresión es: $" + pago);

    let salud1 = saludExp(pago);
    console.log("El porcentaje para la salud como Expresión es: " + salud1 + "%");

    let pension1 = pensionExp(pago);
    console.log("El porcentaje para la pensión como Expresión es: " + pension1 + "%");

    let arl1 = arlExp(pago);
    console.log("El porcentaje para el arl como Expresión es: " + arl1 + "%");

    let trasporte = subTrasExp(pago);
    console.log("El subsidio de transporte como Expresión es: $" + trasporte);

    let retencion = retenExp(pago);
    console.log("La retencion como Expresión es: $" + retencion);

    let pagoTotal = pagoTExp(pago);
    console.log("Su pago total será de: $" + pagoTotal);

</script>
```

```
const pagoTExp = function(pago) {
    let salario = pago;
    let saludE = salud(pago);
    let pensionE = pension(pago);
    let arlE = arl(pago);

    let retencion = reten(pago);
    let subTrasporte = subTras(pago);
    let descuento = saludE + pensionE + arlE;

    let totalSalario = salario + subTrasporte - (retencion + descuento);

    return totalSalario;
}
```

```
top Filter Defa

El pago como Expresión es: $866660
El porcentaje para la salud como Expresión es: 103999.2%
El porcentaje para la pensión como Expresión es: 138665.6%
El porcentaje para el arl como Expresión es: 45066.32%
El subsidio de transporte como Expresión es: $114000
La retencion como Expresión es: $0
Su pago total será de: $692928.88
>
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

15. Porcentaje y suma de tres notas

| | | |
|---|-------------------------|--------------|
| Nombre de la función: notas(pnota,pporcentaje) | | Versión: 1.0 |
| Descripción: Función que calcula el porcentaje de tres notas | | |
| nota | Tipo de variable: float | |
| porcentaje | Tipo de variable: float | |
| operacion | Tipo de variable: float | |
| Código: | | |
| <pre><script> notaUno = 4.5; notaDos = 5.0; notaTres = 4.0; let promedioUno = notas(notaUno, 0.2); let promedioDos = notas(notaDos, 0.35); let promedioTres = notas(notaTres, 0.45); suma = promedioUno + promedioDos + promedioTres; console.log("La nota uno es: " + notaUno + ", que equivale al " + promedioUno + "% del examen."); console.log("La nota dos es: " + notaDos + ", que equivale al " + promedioDos + "% del examen."); console.log("La nota tres es: " + notaTres + ", que equivale al " + promedioTres + "% del examen."); </script></pre> | | |
| <pre>//Como parámetro function notas(pnota, pporcentaje) { let nota; let porcentaje; let operacion; nota = pnota; porcentaje = pporcentaje; operacion = nota * porcentaje; return operacion; }</pre> | | |
| <pre>La nota uno es: 4.5, que equivale al 0.9% del examen. La nota dos es: 5, que equivale al 1.75% del examen. La nota tres es: 4, que equivale al 1.8% del examen.</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: const notasExp = function (pnota, pporcentaje) | Versión: 2.0 |
|---|---------------------|

| |
|--|
| Descripción: Función que calcula el porcentaje de tres notas |
|--|

| | |
|------------|-------------------------|
| nota | Tipo de variable: float |
| porcentaje | Tipo de variable: float |
| Operación | Tipo de variable: float |

| |
|---|
| <p>Código:</p> <pre><script> notaUno = 2.0; notaDos = 4.0; notaTres = 3.0; let promedioUno = notasExp(notaUno, 0.2); let promedioDos = notasExp(notaDos, 0.35); let promedioTres = notasExp(notaTres, 0.45); suma = promedioUno + promedioDos + promedioTres; console.log("La nota uno como Expresión es: " + notaUno + ", que equivale al " + promedioUno + "% del examen."); console.log("La nota dos como Expresión es: " + notaDos + ", que equivale al " + promedioDos + "% del examen."); console.log("La nota tres como Expresión es: " + notaTres + ", que equivale al " + promedioTres + "% del examen."); </script></pre> <pre>//Como Expresión const notasExp = function (pnota, pporcentaje) { let nota; let porcentaje; let operacion; nota = pnota; porcentaje = pporcentaje; operacion = nota * porcentaje; return operacion; }</pre> |
|---|



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
function validar(psuma)

Versión: 1.0

Descripción:

Función que valida la suma del porcentaje de las tres notas

suma

Tipo de variable: float

Código:

```
<script>
notaUno = 4.5;
notaDos = 5.0;
notaTres = 4.0;

let promedioUno = notas(notaUno, 0.2);
let promedioDos = notas(notaDos, 0.35);
let promedioTres = notas(notaTres, 0.45);

suma = promedioUno + promedioDos + promedioTres;

console.log(
  "La nota uno como Parámetro es: " + notaUno + ", que equivale al " + promedioUno + "% del examen."
);
console.log(
  "La nota dos como Parámetro es: " + notaDos + ", que equivale al " + promedioDos + "% del examen."
);
console.log(
  "La nota tres como Parámetro es: " + notaTres + ", que equivale al " + promedioTres + "% del examen."
);

suma = promedioUno + promedioDos + promedioTres;
console.log(validar(suma))
</script>
```

```
function validar(psuma) {
  let suma;
  suma = psuma;
  if (suma > 4.5) {
    return "La calificación es Superior " + suma;
  } else if (suma <= 4.5 && suma > 3.5) {
    return "La calificación es Buena " + suma;
  } else if (suma >= 3 && suma <= 3.5) {
    return "La calificación es Media " + suma;
  } else {
    return "la calificación es Mala" + suma;
  }
}
```

top Filter Default levels No logs

```
La nota uno como Parámetro es: 4.5, que equivale al 0.9% del examen.
La nota dos como Parámetro es: 5, que equivale al 1.75% del examen.
La nota tres como Parámetro es: 4, que equivale al 1.8% del examen.
La calificación es Buena 4.45
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const validarExp = function (psuma)

Versión: 2.0

Descripción:

Función que valida la suma del porcentaje de las tres notas

suma

Tipo de variable: float

Código:

```
<script>
notaUno = 2.0;
notaDos = 4.0;
notaTres = 3.0;

let promedioUno = notasExp(notaUno, 0.2);
let promedioDos = notasExp(notaDos, 0.35);
let promedioTres = notasExp(notaTres, 0.45);

suma = promedioUno + promedioDos + promedioTres;

console.log(
  "La nota uno como Expresión es: " + notaUno + ", que equivale al " + promedioUno + "% del examen."
);
console.log(
  "La nota dos como Expresión es: " + notaDos + ", que equivale al " + promedioDos + "% del examen."
);
console.log(
  "La nota tres como Expresión es: " + notaTres + ", que equivale al " + promedioTres + "% del examen."
);

suma = promedioUno + promedioDos + promedioTres;
console.log(validarExp(suma))
</script>
```

```
const validarExp = function (psuma) {
  let suma;
  suma = psuma;
  if (suma > 4.5) {
    return "La calificación es Superior " + suma;
  } else if (suma <= 4.5 && suma > 3.5) {
    return "La calificación es Buena " + suma;
  } else if (suma >= 3 && suma <= 3.5) {
    return "La calificación es Media " + suma;
  } else {
    return "la calificación es Mala" + suma;
  }
};
```

top Filter Default levels

```
La nota uno como Expresión es: 2, que equivale al 0.4% del examen.
La nota dos como Expresión es: 4, que equivale al 1.4% del examen.
La nota tres como Expresión es: 3, que equivale al 1.35% del examen.
La calificación es Media 3.15
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

16. Contador 5 while

| | | |
|---|--------------------------|---------------------|
| Nombre de la función: contar(pnumero) | | Versión: 1.0 |
| Descripción: Función que cuenta hasta 5 con while | | |
| numero | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| resultado | Tipo de variable: String | |
| Código: | | |
| <pre>let numero; //Como Parametros function contar(pnumero) { numero = pnumero; let contador = 0; let resultado = ""; while (contador < numero) { contador = contador + 1; resultado += contador + "\n"; } return resultado; }</pre> | | |
| <pre><script> numero = 5; console.log("Como parámetro: \n"+contar(numero)); </script></pre> | | |
| <pre>Como parámetro: 1 2 3 4 5</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const contarExp = function(pnumero)

Versión: 2.0

Descripción:

Función que cuenta hasta 5 con while

numero

Tipo de variable: Int

contador

Tipo de variable: Int

resultado

Tipo de variable: String

Código:

//Como Expresión

```
const contarExp = function (pnumero) {  
  numero = pnumero;  
  let contador = 0;  
  let resultado = "";  
  
  while (contador < numero) {  
    contador = contador + 1;  
    resultado += contador + "\n";  
  }  
  
  return resultado;  
}
```

```
<script>  
  numero = 5;  
  console.log("Como Expresión: \n"+contarExp(numero));  
</script>
```

Como Expresión:

1
2
3
4
5

>



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

17. Factorial 5 while

| | | |
|--|-----------------------|--------------|
| Nombre de la función: factor(pnumero) | | Versión: 1.0 |
| Descripción: Función que calcula el factorial de 5 con while | | |
| numero | Tipo de variable: Int | |
| factorial | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| Código: | | |
| <pre><script> numero = 5; console.log("Como parámetro: \n"+factor(numero)); </script></pre> | | |
| <pre>let numero; //como parámetro function factor(pnumero){ numero = pnumero; let factorial; let contador; contador = 0; factorial = 1; while(contador < numero){ contador = contador+1; factorial =factorial*contador; } return factorial; }</pre> | | |
| <pre>Como parámetro: 120</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función: const factorExp = function(pnumero)

Versión: 2.0

Descripción:

Función que calcula el factorial de 5 con while

| | |
|---------|--------------------------------|
| saludar | Tipo de variable: Alfanumérico |
|---------|--------------------------------|

| | |
|---------|--------------------------------|
| saludar | Tipo de variable: Alfanumérico |
|---------|--------------------------------|

| | |
|---------|--------------------------------|
| saludar | Tipo de variable: Alfanumérico |
|---------|--------------------------------|

Código:

```
<script>
  numero = 5;
  console.log("Como Expresión: \n"+factorExp(numero));
</script>
```

```
//Como Expresión

const factorExp = function (pnumero){
  numero = pnumero;
  let factorial;
  let contador;

  contador = 0;
  factorial = 1;

  while(contador < numero){
    contador = contador+1;
    factorial =factorial*contador;
  }
  return factorial;
}
```

```
Como Expresión:
120
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

18. Tabla del 5 while

| | | |
|--|--------------------------|--|
| Nombre de la función: tabla(ptablas,plimite) | | Versión: 1.0 |
| Descripción: Función que haga la tabla del 5 con while | | |
| tablas | Tipo de variable: Int | |
| limite | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| resultado | Tipo de variable: Int | |
| msg | Tipo de variable: String | |
| Código: | | |
| <pre><script> tablas = 5; limite = 5; console.log("Como parámetro: \n"+tabla(tablas,limite)); </script></pre> | | <pre>Como parámetro: 5 x 1 = 5 5 x 2 = 10 5 x 3 = 15 5 x 4 = 20 5 x 5 = 25</pre> |
| <pre>let tablas; let limite; //Como Parametro function tabla(ptablas,plimite){ tablas = ptablas; limite = plimite; let contador = 0; let resultado; let msg = ""; while (contador < limite) { contador = contador +1; resultado = tablas * contador; msg += `\${tablas} x \${contador} = \${resultado}\n` } return msg; }</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: const tablaExp = function(ptablas,plimite) | Versión: 2.0 |
|---|---------------------|

| |
|--|
| Descripción: Función que haga las tablas del 5 con while |
|--|

| | |
|-----------|--------------------------|
| tablas | Tipo de variable: Int |
| limite | Tipo de variable: Int |
| contador | Tipo de variable: Int |
| resultado | Tipo de variable: Int |
| msg | Tipo de variable: String |

Código:

```
<script>
  tablas = 5;
  limite = 5;
  console.log("Como Expresión: \n"+tablaExp(tablas,limite));
</script>
```

Como Expresión:

```
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
```

```
//Como Expresión

const tablaExp = function (ptablas,plimite){

  tablas = ptablas;
  limite = plimite

  let contador = 0;
  let resultado;
  let msg = "";

  while (contador < limite) {

    contador = contador +1;
    resultado = tablas * contador;

    msg += `${tablas} x ${contador} = ${resultado}\n`
  }

  return msg;
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

19. Tabla del 9 while

| | | |
|---|--------------------------|--|
| Nombre de la función: tabla(ptablas,plimite) | | Versión: 1.0 |
| Descripción: Función que hace la tabla del 9 y me retorna si el resultado es par o impar | | |
| tablas | Tipo de variable: Int | |
| limite | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| resultado | Tipo de variable: Int | |
| msg | Tipo de variable: String | |
| Código: | | |
| <pre><script> tablas = 9; limite = 5; console.log("Como parámetro: \n"+tabla(tablas,limite)); </script></pre> | | <pre>Como parámetro: 9 x 1 = 9 impar 9 x 2 = 18 par 9 x 3 = 27 impar 9 x 4 = 36 par 9 x 5 = 45 impar</pre> |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|---|---------------------|
| Nombre de la función: <code>const tablaExp = function(ptablas,plimite)</code> | Versión: 2.0 |
|---|---------------------|

| |
|---|
| Descripción: Función que hace la tabla del 9 y retorna si el resultado es par e impar |
|---|

| | |
|-----------|--------------------------|
| tablas | Tipo de variable: Int |
| limite | Tipo de variable: Int |
| contador | Tipo de variable: Int |
| resultado | Tipo de variable: Int |
| msg | Tipo de variable: String |

Código:

```
<script>
  tablas = 9;
  limite = 5;
  console.log("Como Expresión: \n"+tablaExp(tablas,limite));
</script>
```

Como Expresión:

| | |
|------------|-------|
| 9 x 1 = 9 | impar |
| 9 x 2 = 18 | par |
| 9 x 3 = 27 | impar |
| 9 x 4 = 36 | par |
| 9 x 5 = 45 | impar |

```
//Como Expresión
const tablaExp = function (ptablas,plimite){
  let tablas;
  let limite
  tablas = ptablas;
  limite = plimite
  let contador;
  let msg="";
  let resultado;
  contador = 0;
  msg = "";

  while (contador<limite) {
    contador = contador +1
    resultado = tablas * contador;

    msg += `${tablas} x ${contador} = ${resultado} `;

    msg += validarExp(resultado);
  }

  return msg;
}
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
tabla(ptablas,plimite)

Versión: 1.0

Descripción:

Función que valida la tabla del 9 y me retorna si el resultado es par o impar

resultado

Tipo de variable: Int

Código:

```
<script>
  tablas = 9;
  limite = 5;
  console.log("Como parámetro: \n"+tabla(tablas,limite));
</script>
```

Como parámetro:
9 x 1 = 9 impar
9 x 2 = 18 par
9 x 3 = 27 impar
9 x 4 = 36 par
9 x 5 = 45 impar

```
function validar(resultado){
    if(resultado %2==0){
        return "par\n";
    }else{
        return "impar\n"
    }
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const tablaExp = function(ptablas,plimite)`

Versión: 2.0

Descripción:

Función que valida la tabla del 9 y retorna si el resultado es par e impar

resultado

Tipo de variable: Int

Código:

```
<script>
  tablas = 9;
  limite = 5;
  console.log("Como Expresión: \n"+tablaExp(tablas,limite));
</script>
```

Como Expresión:

9 x 1 = 9 impar
9 x 2 = 18 par
9 x 3 = 27 impar
9 x 4 = 36 par
9 x 5 = 45 impar

```
function validarExp(resultado){
    if(resultado %2==0){
        return "par\n";
    }else{
        return "impar\n"
    }
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

20. Contar de 1 a 5 for

| | | |
|---|--------------------------|---|
| Nombre de la función: contar(pnumero) | | Versión: 1.0 |
| Descripción: Función que cuenta hasta 5 con for | | |
| numero | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| resultado | Tipo de variable: String | |
| Código: | | |
| <pre><script> numero = 5; console.log("Como parámetros: \n"+contar(numero)); </script></pre> | | <div>Como parámetros:</div> <div>1.</div> <div>2.</div> <div>3.</div> <div>4.</div> <div>5.</div> <div>></div> |
| <pre>let numero; //Como parámetros function contar(pnumero){ numero = pnumero; let contador; let resultado = ""; for(contador = 1; numero >= contador; contador++){ resultado += contador+".\n"; } return resultado; }</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|--|---------------------|
| Nombre de la función: const contarExp = function(pnumero) | Versión: 2.0 |
|--|---------------------|

| | |
|---|--|
| Descripción: Función que cuenta hasta 5 con for | |
|---|--|

| | |
|--------|-----------------------|
| numero | Tipo de variable: Int |
|--------|-----------------------|

| | |
|-----------|--------------------------|
| resultado | Tipo de variable: String |
|-----------|--------------------------|

| | |
|-----------|--------------------------|
| resultado | Tipo de variable: String |
|-----------|--------------------------|

Código:

```
<script>
    numero = 5;
    console.log("Como Expresión: \n"+contarExp(numero));
</script>
```

Como Expresión:

- 1.
- 2.
- 3.
- 4.
- 5.

```
// Como Expresión

const contarExp = function (pnumero){
    numero = pnumero;
    let contador;
    let resultado = "";

    for(contador = 1; numero >= contador; contador++){

        resultado += contador+".\n";
    }
    return resultado;
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

21. Factorial de 5 for

| | | |
|--|-----------------------|-----------------------------------|
| Nombre de la función: factor(pnumero) | | Versión: 1.0 |
| Descripción: Función que calcula la factorial de 5 con for | | |
| numero | Tipo de variable: Int | |
| factorial | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| Código: | | |
| <pre><script> numero = 5; console.log("Como parámetro: \n"+factor(numero)); </script></pre> | | <p>Como parámetro:</p> <p>120</p> |
| <pre>let numero; //Como parámetro function factor(pnumero){ numero = pnumero; let factorial = 1; let contador; for(contador = 1; numero >= contador; contador++){ factorial = factorial * contador; } return factorial; }</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|--|---------------------|
| Nombre de la función: <code>const factorExp = function(pnumero)</code> | Versión: 2.0 |
|--|---------------------|

| |
|--|
| Descripción: Función que calcula el factorial de 5 |
|--|

| | |
|-----------|-----------------------|
| numero | Tipo de variable: Int |
| factorial | Tipo de variable: Int |
| contador | Tipo de variable: Int |

Código:

```
<script>
  numero = 5;
  console.log("Como Expresión: \n"+factorExp(numero));
</script>
```

Como Expresión:
120

```
//Como Expresión

const factorExp = function (pnumero){

  numero = pnumero;
  let factorial = 1;
  let contador;

  for(contador = 1; numero >= contador; contador++){

    factorial = factorial * contador;
  }

  return factorial;
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

22. Tabla del 5 for

| | |
|--|---------------------|
| Nombre de la función: tablas(ptabla, plimite) | Versión: 1.0 |
|--|---------------------|

Descripción:

Función que hace la tabla de 5 con for

| | |
|-----------|--------------------------|
| tabla | Tipo de variable: Int |
| Limite | Tipo de variable: Int |
| contador | Tipo de variable: Int |
| resultado | Tipo de variable: Int |
| msg | Tipo de variable: String |

Código:

```
<script>
  tabla = 5;
  limite = 5;

  console.log("Como parámetro: \n"+tablas(tabla,limite));
</script>
```

Como parámetro:

```
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
```

```
let tabla;
let limite;
//Como parámetro

function tablas(ptablas,plimite){
  tabla = ptablas;
  limite = plimite;
  let contador;
  let resultado;
  let msg = "";

  for(contador=1; limite >= contador; contador ++ ){

    resultado = tabla * contador;

    msg += `${tabla} x ${contador} = ${resultado}\n`;
  }
  return msg;
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const tablasExp = function(ptabla, plimite)

Versión: 2.0

Descripción:

Función que hace la tabla del 5 con for

| | |
|-------|-----------------------|
| tabla | Tipo de variable: Int |
|-------|-----------------------|

| | |
|--------|-----------------------|
| limite | Tipo de variable: Int |
|--------|-----------------------|

| | |
|----------|-----------------------|
| contador | Tipo de variable: Int |
|----------|-----------------------|

| | |
|-----------|-----------------------|
| resultado | Tipo de variable: Int |
|-----------|-----------------------|

| | |
|-----|--------------------------|
| msg | Tipo de variable: String |
|-----|--------------------------|

Código:

```
<script>
  tabla = 5;
  limite = 5;

  console.log("Como Expresión: \n"+tablasExp(tabla,limite));
</script>
```

Como Expresión:

```
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
```

```
//Como Expresión
const tablasExp = function (ptablas,plimite){
  tabla = ptablas;
  limite = plimite;
  let contador;
  let resultado;
  let msg = "";

  for(contador=1; limite >= contador; contador ++ ){

    resultado = tabla * contador;

    msg += `${tabla} x ${contador} = ${resultado}\n`;
  }
  return msg;
}
```




Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

23. Tabla del 9 for

| | |
|--|---------------------|
| Nombre de la función: tablas(ptabla, plimite) | Versión: 1.0 |
|--|---------------------|

Descripción:

Función que hace la tabla del 9 y retorna si el resultado es par o impar

| | |
|-----------|--------------------------|
| limite | Tipo de variable: Int |
| tabla | Tipo de variable: Int |
| contador | Tipo de variable: Int |
| Resultado | Tipo de variable: Int |
| msg | Tipo de variable: String |

Código:

```
<script>
  tabla=9;
  limite = 5
  console.log("Como parámetro: \n"+tablas(tabla,limite));
</script>
```

Como parámetro:

```
9 x 1 = 9 El resultado es impar
9 x 2 = 18 El resultado es par
9 x 3 = 27 El resultado es impar
9 x 4 = 36 El resultado es par
9 x 5 = 45 El resultado es impar
```

```
let tabla;
let limite;

// como parámetros

function tablas(ptablas,plimite){
  tabla = ptablas;
  limite = plimite;
  let contador;
  let resultado;
  let msg = "";

  for(contador=1; limite >= contador; contador ++ ){

    resultado = tabla * contador;

    msg += `${tabla} x ${contador} = ${resultado} `;
    msg+= validar(resultado);
  }
  return msg;
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const tablasExp = function(ptabla, plimite)`

Versión: 2.0

Descripción:

Función que hace la tabla del 9 y retorna si los resultados son pares e impares

Msg

par

impar

Tipo de variable: numérico

Código:

```
<script>
  tabla=9;
  limite = 5
  console.log("Como Expresión: \n"+tablasExp(tabla,limite));
</script>
```

Como Expresión:

9 x 1 = 9 Este es impar
9 x 2 = 18 Este es un par
9 x 3 = 27 Este es impar
9 x 4 = 36 Este es un par
9 x 5 = 45 Este es impar

```
//Como Expresión

const tablasExp = function (ptablas,plimite){
  tabla = ptablas;
  limite = plimite
  let contador;
  let resultado;
  let msg = "";

  for(contador=1; limite >= contador; contador ++ ){

    resultado = tabla * contador;

    msg += `${tabla} x ${contador} = ${resultado}`;

    msg += validarExp(resultado);

  }
  return msg;
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
tabla(ptablas,plimite)

Versión: 1.0

Descripción:

Función que valida la tabla del 9 y me retorna si el resultado es par o impar

resultado

Tipo de variable: Int

Código:

```
<script>
  tablas = 9;
  limite = 5;
  console.log("Como parámetro: \n"+tabla(tablas,limite));
</script>
```

Como parámetro:

```
9 x 1 = 9 El resultado es impar
9 x 2 = 18 El resultado es par
9 x 3 = 27 El resultado es impar
9 x 4 = 36 El resultado es par
9 x 5 = 45 El resultado es impar
```

```
function validar(resultado){

    if(resultado%2==0){
        return "El resultado es par\n";
    }else{
        return"El resultado es impar\n";
    }

}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const tablaExp = function(ptablas,plimite)`

Versión: 2.0

Descripción:

Función que valida la tabla del 9 y retorna si el resultado es par e impar

resultado

Tipo de variable: Int

Código:

```
<script>
  tablas = 9;
  limite = 5;
  console.log("Como Expresión: \n"+tablaExp(tablas,limite));
</script>
```

Como Expresión:

```
9 x 1 = 9 Este es impar
9 x 2 = 18 Este es un par
9 x 3 = 27 Este es impar
9 x 4 = 36 Este es un par
9 x 5 = 45 Este es impar
```

```
const validarExp = function(resultado){

    if(resultado%2==0){
        return " El resultado es par\n";
    }else{
        return " El resultado es impar\n";
    }
}
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

24. Tabla 1 a 5 for

| | | |
|--|--------------------------|--|
| Nombre de la función: tablas(ptabla, plimite, plimiteT) | | Versión: 1.0 |
| Descripción: Función que hace las tablas del 1 al 5 y me dice si es par”buzz” o impar”bass” | | |
| limite | Tipo de variable: Int | |
| limiteT | Tipo de variable: Int | |
| tabla | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| msg | Tipo de variable: String | |
| Código: | | |
| <pre><script> tabla = 1; // inicio de la tabla limite = 5; //Hasta que tabla quiere que multiplique limiteT = 5; //Hasta que número desea multiplicar cada tabla console.log("Como Parámetro"+tablas(tabla,limite,limiteT)); </script></pre> | | <pre>Como Parámetro 1 x 1 = 1 Bass 1 x 2 = 2 Buzz 1 x 3 = 3 Bass 1 x 4 = 4 Buzz 1 x 5 = 5 Bass 2 x 1 = 2 Buzz 2 x 2 = 4 Buzz 2 x 3 = 6 Buzz 2 x 4 = 8 Buzz 2 x 5 = 10 Buzz 3 x 1 = 3 Bass 3 x 2 = 6 Buzz 3 x 3 = 9 Bass 3 x 4 = 12 Buzz 3 x 5 = 15 Bass 4 x 1 = 4 Buzz 4 x 2 = 8 Buzz 4 x 3 = 12 Buzz 4 x 4 = 16 Buzz 4 x 5 = 20 Buzz 5 x 1 = 5 Bass 5 x 2 = 10 Buzz 5 x 3 = 15 Bass 5 x 4 = 20 Buzz 5 x 5 = 25 Bass total de pares: 16 total de impares 9</pre> |
| <pre>// Con parámetros let par = 0; let impar = 0; let resultado; function tablas(ptabla, plimite,plimiteT) { let limite; let limiteT; let tabla; let contador; let msg = ""; tabla = ptabla; limite = plimite; limiteT = plimiteT for (tabla = 1; limite >= tabla; tabla++) { for (contador = 1; limiteT >= contador; contador++) { resultado = tabla * contador; msg += ` \n \${tabla} x \${contador} = \${resultado}`; msg += validar(resultado); } msg += "\n"; } return msg + "total de pares: " + par + "\n" + "total de impares " + impar; }</pre> | | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

const tablasExp = function(ptabla, plimite, plimiteT)

Versión: 2.0

Descripción:

Función que hace las tablas del 1 al 5 y me dice si es par "buzz" o impar "bass" for

limite Tipo de variable: Int

limiteT Tipo de variable: Int

tabla Tipo de variable: Int

contador Tipo de variable: Int

msg Tipo de variable: String

Código:

```
<script>
  tabla = 1; // inicio de la tabla
  limite = 5; //Hasta que tabla quiere que multiplique
  limiteT = 5; //Hasta que número desea multiplicar cada tabla
  console.log("Como Expresión"+tablasExp(tabla,limite,limiteT));
</script>
```

```
//Como Expresión
const tablasExp = function(ptabla, plimite,plimiteT) {

  let limite;
  let limiteT;
  let tabla;
  let contador;
  let msg = "";

  tabla = ptabla;
  limite = plimite;
  limiteT = plimiteT

  for (tabla = 1; limite >= tabla; tabla++) {
    for (contador = 1; limiteT >= contador; contador++) {
      resultado = tabla * contador;

      msg += `\n ${tabla} x ${contador} = ${resultado}`;

      msg += validarExp(resultado);
    }
    msg += "\n";
  }
  return msg + "total de pares: " + par + "\n" + "total de impares " + impar;
}
```

Como Expresión

1 x 1 = 1 Bass
1 x 2 = 2 Buzz
1 x 3 = 3 Bass
1 x 4 = 4 Buzz
1 x 5 = 5 Bass

2 x 1 = 2 Buzz
2 x 2 = 4 Buzz
2 x 3 = 6 Buzz
2 x 4 = 8 Buzz
2 x 5 = 10 Buzz

3 x 1 = 3 Bass
3 x 2 = 6 Buzz
3 x 3 = 9 Bass
3 x 4 = 12 Buzz
3 x 5 = 15 Bass

4 x 1 = 4 Buzz
4 x 2 = 8 Buzz
4 x 3 = 12 Buzz
4 x 4 = 16 Buzz
4 x 5 = 20 Buzz

5 x 1 = 5 Bass
5 x 2 = 10 Buzz
5 x 3 = 15 Bass
5 x 4 = 20 Buzz
5 x 5 = 25 Bass
total de pares: 16
total de impares 9



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
validar(resultado)

Versión: 1.0

Descripción:

Función que valida si el resultado es par o impar y si es par escribe "buzz" o impar escribe "Bass"

par Tipo de variable: Int

impar Tipo de variable: Int

Código:

```
function validar(resultado){  
  if (resultado % 2 == 0){  
    par ++  
    return " Buzz";  
  }else{  
    impar++  
    return " Bass";  
  }  
}
```

Como Parámetro

| | |
|--------------------|------|
| 1 x 1 = 1 | Bass |
| 1 x 2 = 2 | Buzz |
| 1 x 3 = 3 | Bass |
| 1 x 4 = 4 | Buzz |
| 1 x 5 = 5 | Bass |
| | |
| 2 x 1 = 2 | Buzz |
| 2 x 2 = 4 | Buzz |
| 2 x 3 = 6 | Buzz |
| 2 x 4 = 8 | Buzz |
| 2 x 5 = 10 | Buzz |
| | |
| 3 x 1 = 3 | Bass |
| 3 x 2 = 6 | Buzz |
| 3 x 3 = 9 | Bass |
| 3 x 4 = 12 | Buzz |
| 3 x 5 = 15 | Bass |
| | |
| 4 x 1 = 4 | Buzz |
| 4 x 2 = 8 | Buzz |
| 4 x 3 = 12 | Buzz |
| 4 x 4 = 16 | Buzz |
| 4 x 5 = 20 | Buzz |
| | |
| 5 x 1 = 5 | Bass |
| 5 x 2 = 10 | Buzz |
| 5 x 3 = 15 | Bass |
| 5 x 4 = 20 | Buzz |
| 5 x 5 = 25 | Bass |
| | |
| total de pares: 16 | |
| total de impares 9 | |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const validarExp = function(resultado)`

Versión: 2.0

Descripción:

Función que valida si el resultado es par o impar y si es par escribe "buzz" o impar escribe "Bass"

par Tipo de variable: Int

impar Tipo de variable: Int

Código:

```
const validarExp = function(resultado){  
  
    if (resultado % 2 == 0){  
        par ++  
        return " Buzz";  
    }else{  
        impar++  
        return " Bass";  
    }  
}
```

Como Expresión

```
1 x 1 = 1 Bass  
1 x 2 = 2 Buzz  
1 x 3 = 3 Bass  
1 x 4 = 4 Buzz  
1 x 5 = 5 Bass
```

```
2 x 1 = 2 Buzz  
2 x 2 = 4 Buzz  
2 x 3 = 6 Buzz  
2 x 4 = 8 Buzz  
2 x 5 = 10 Buzz
```

```
3 x 1 = 3 Bass  
3 x 2 = 6 Buzz  
3 x 3 = 9 Bass  
3 x 4 = 12 Buzz  
3 x 5 = 15 Bass
```

```
4 x 1 = 4 Buzz  
4 x 2 = 8 Buzz  
4 x 3 = 12 Buzz  
4 x 4 = 16 Buzz  
4 x 5 = 20 Buzz
```

```
5 x 1 = 5 Bass  
5 x 2 = 10 Buzz  
5 x 3 = 15 Bass  
5 x 4 = 20 Buzz  
5 x 5 = 25 Bass
```

```
total de pares: 16  
total de impares 9
```

>



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

25. Tabla 1 a 5 while

| | | |
|---|---|---------------------|
| Nombre de la función: tablas(ptabla, plimite, plimiteT) | | Versión: 1.0 |
| Descripción: Función que hace las tablas del 1 al 5 y me dice si es par"buzz" o impar"bass" while | | |
| tabla | Tipo de variable: Int | |
| limite | Tipo de variable: Int | |
| limitieT | Tipo de variable: Int | |
| contador | Tipo de variable: Int | |
| resultado msg | Tipo de variable: Int Tipo de variable: String | |
| Código: | | |

```
<script>
  tabla = 0;
  limite = 5;
  limiteT = 5
  console.log("Como parámetros: \n"+tablas(tabla,limite,limiteT));
</script>
```

```
let par = 0;
let impar = 0;
// Como parámetros

function tablas(ptabla, plimite, plimiteT) {
  let tabla;
  let limite;
  let limiteT;

  let contador;
  let resultado;
  let msg = "";
  tabla = ptabla;
  limite = plimite;
  limiteT = plimiteT;

  while (tabla < limite) {
    contador = 0;
    tabla = tabla + 1;

    while (contador < limiteT) {
      contador = contador + 1;
      resultado = tabla * contador;
      msg += ` \n ${tabla} x ${contador} = ${resultado}`;
      msg += validar(resultado);
    }
    msg += "\n";
  }
  return (
    msg + "\n" + "total de pares: " + par + "\n" + "total de impares " + impar
  );
}
```

Como parámetros:

1 x 1 = 1 Bass
1 x 2 = 2 Buzz
1 x 3 = 3 Bass
1 x 4 = 4 Buzz
1 x 5 = 5 Bass

2 x 1 = 2 Buzz
2 x 2 = 4 Buzz
2 x 3 = 6 Buzz
2 x 4 = 8 Buzz
2 x 5 = 10 Buzz

3 x 1 = 3 Bass
3 x 2 = 6 Buzz
3 x 3 = 9 Bass
3 x 4 = 12 Buzz
3 x 5 = 15 Bass

4 x 1 = 4 Buzz
4 x 2 = 8 Buzz
4 x 3 = 12 Buzz
4 x 4 = 16 Buzz
4 x 5 = 20 Buzz

5 x 1 = 5 Bass
5 x 2 = 10 Buzz
5 x 3 = 15 Bass
5 x 4 = 20 Buzz
5 x 5 = 25 Bass

total de pares: 16
total de impares 9



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

| | |
|--|---------------------|
| Nombre de la función: const tablasExp = function(ptabla, plimite, plimiteT) | Versión: 2.0 |
|--|---------------------|

Descripción:
Función que hace las tablas del 1 al 5 y me dice si es par "buzz" o impar "bass" while

| | |
|------------------|---|
| tabla | Tipo de variable: Int |
| limite | Tipo de variable: Int |
| limiteT | Tipo de variable: Int |
| contador | Tipo de variable: Int |
| resultado msg | Tipo de variable: Int Tipo de variable: String |

Código:

```
<script>
  tabla = 0;
  limite = 5;
  limiteT = 5
  console.log("Como Expresión: \n"+tablas(tabla,limite,limiteT));
</script>
```

```
//Como Expresión
const tablasExp = function(ptabla, plimite, plimiteT) {
  let tabla;
  let limite;
  let limiteT;

  let contador;
  let resultado;
  let msg = "";
  tabla = ptabla;
  limite = plimite;
  limiteT = plimiteT;

  while (tabla < limite) {
    contador = 0;
    tabla = tabla + 1;

    while (contador < limiteT) {
      contador = contador + 1;
      resultado = tabla * contador;
      msg += "\n ${tabla} x ${contador} = ${resultado}";
      msg += validarExp(resultado);
    }
    msg += "\n";
  }
  return (
    msg + "\n" + "total de pares: " + par + "\n" + "total de impares " + impar
  );
};
```

Como Expresión:

```
1 x 1 = 1 Bass
1 x 2 = 2 Buzz
1 x 3 = 3 Bass
1 x 4 = 4 Buzz
1 x 5 = 5 Bass

2 x 1 = 2 Buzz
2 x 2 = 4 Buzz
2 x 3 = 6 Buzz
2 x 4 = 8 Buzz
2 x 5 = 10 Buzz

3 x 1 = 3 Bass
3 x 2 = 6 Buzz
3 x 3 = 9 Bass
3 x 4 = 12 Buzz
3 x 5 = 15 Bass

4 x 1 = 4 Buzz
4 x 2 = 8 Buzz
4 x 3 = 12 Buzz
4 x 4 = 16 Buzz
4 x 5 = 20 Buzz

5 x 1 = 5 Bass
5 x 2 = 10 Buzz
5 x 3 = 15 Bass
5 x 4 = 20 Buzz
5 x 5 = 25 Bass

total de pares: 16
total de impares 9
```



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:
validar(resultado)

Versión: 1.0

Descripción:

Función que valida si el resultado es par o impar y si es par escribe "Buzz" o impar escribe "Bass"

par Tipo de variable: Int

impar Tipo de variable: Int

Código:

```
function validar(resultado){  
  if (resultado % 2 == 0){  
    par ++  
    return " Buzz";  
  }else{  
    impar++  
    return " Bass";  
  }  
}
```

Como Parámetro

| | | | | | |
|---|---|---|---|----|--------------------|
| 1 | x | 1 | = | 1 | Bass |
| 1 | x | 2 | = | 2 | Buzz |
| 1 | x | 3 | = | 3 | Bass |
| 1 | x | 4 | = | 4 | Buzz |
| 1 | x | 5 | = | 5 | Bass |
| | | | | | |
| 2 | x | 1 | = | 2 | Buzz |
| 2 | x | 2 | = | 4 | Buzz |
| 2 | x | 3 | = | 6 | Buzz |
| 2 | x | 4 | = | 8 | Buzz |
| 2 | x | 5 | = | 10 | Buzz |
| | | | | | |
| 3 | x | 1 | = | 3 | Bass |
| 3 | x | 2 | = | 6 | Buzz |
| 3 | x | 3 | = | 9 | Bass |
| 3 | x | 4 | = | 12 | Buzz |
| 3 | x | 5 | = | 15 | Bass |
| | | | | | |
| 4 | x | 1 | = | 4 | Buzz |
| 4 | x | 2 | = | 8 | Buzz |
| 4 | x | 3 | = | 12 | Buzz |
| 4 | x | 4 | = | 16 | Buzz |
| 4 | x | 5 | = | 20 | Buzz |
| | | | | | |
| 5 | x | 1 | = | 5 | Bass |
| 5 | x | 2 | = | 10 | Buzz |
| 5 | x | 3 | = | 15 | Bass |
| 5 | x | 4 | = | 20 | Buzz |
| 5 | x | 5 | = | 25 | Bass |
| | | | | | |
| | | | | | total de pares: 16 |
| | | | | | total de impares 9 |



Tecnólogo en Análisis y Desarrollo de Software

Ficha

Funciones JS

Nombre de la función:

`const validarExp = function(resultado)`

Versión: 2.0

Descripción:

Función que valida si el resultado es par o impar y si es par escribe " Buzz" o impar escribe "Bass"

par Tipo de variable: Int

impar Tipo de variable: Int

Código:

```
const validarExp = function(resultado){  
  
    if (resultado % 2 == 0){  
        par ++  
        return " Buzz";  
    }else{  
        impar++  
        return " Bass";  
    }  
}
```

Como Expresión

```
1 x 1 = 1 Bass  
1 x 2 = 2 Buzz  
1 x 3 = 3 Bass  
1 x 4 = 4 Buzz  
1 x 5 = 5 Bass
```

```
2 x 1 = 2 Buzz  
2 x 2 = 4 Buzz  
2 x 3 = 6 Buzz  
2 x 4 = 8 Buzz  
2 x 5 = 10 Buzz
```

```
3 x 1 = 3 Bass  
3 x 2 = 6 Buzz  
3 x 3 = 9 Bass  
3 x 4 = 12 Buzz  
3 x 5 = 15 Bass
```

```
4 x 1 = 4 Buzz  
4 x 2 = 8 Buzz  
4 x 3 = 12 Buzz  
4 x 4 = 16 Buzz  
4 x 5 = 20 Buzz
```

```
5 x 1 = 5 Bass  
5 x 2 = 10 Buzz  
5 x 3 = 15 Bass  
5 x 4 = 20 Buzz  
5 x 5 = 25 Bass
```

```
total de pares: 16  
total de impares 9
```

>