



Funciones en JavaScript

Clase 7

KOICA

IGU HANDONG GLOBAL
UNIVERSITY



UNA

OBJETIVOS DE LA CLASE 7

- Manejar los conceptos de funciones, funciones con parámetros y funciones que retornan un valor.
- Implementar los códigos de ejemplos propuestos en clase.



Funciones

- En programación es muy frecuente que un determinado procedimiento de cálculo definido por un grupo de sentencias tenga que repetirse varias veces, ya sea en un mismo programa o en otros programas, lo cual implica que se tenga que escribir tantos grupos de aquellas sentencias como veces aparezca dicho proceso.
- La herramienta más potente con que se cuenta para facilitar, reducir y dividir el trabajo en programación, es escribir aquellos grupos de sentencias una sola y única vez bajo la forma de una FUNCIÓN.

Funciones

- Un programa es una cosa compleja de realizar y por lo tanto es importante que esté bien ESTRUCTURADO y también que sea inteligible para las personas. Si un grupo de sentencias realiza una tarea bien definida, entonces puede estar justificado el aislar estas sentencias formando una función, aunque resulte que sólo se le llame o use una vez.
- Hasta ahora hemos visto cómo resolver un problema planteando un único algoritmo.

Funciones

- Con funciones podemos segmentar un programa en varias partes. Frente a un problema, planteamos un algoritmo, éste puede constar de pequeños algoritmos.
- Una función es un conjunto de instrucciones que resuelven una parte del problema y que puede ser utilizado (llamado) desde diferentes partes de un programa.
- Consta de un nombre y parámetros. Con el nombre llamamos a la función, es decir, hacemos referencia a la misma. Los parámetros son valores que se envían y son indispensables para la resolución del mismo. La función realizará alguna operación con los parámetros que le enviamos. Podemos cargar una variable, consultarla, modificarla, imprimirla, etc.

Funciones

- Incluso los programas más sencillos tienen la necesidad de fragmentarse. Las funciones son los únicos tipos de subprogramas que acepta JavaScript. Tienen la siguiente estructura:

```
function <nombre de función>(argumento1, argumento2, ..., argumento n)
{
    <código de la función>
}
```

Funciones

- Debemos buscar un nombre de función que nos indique cuál es su objetivo (Si la función recibe un string y lo centra, tal vez deberíamos llamarla `centrarTitulo`). Veremos que una función puede variar bastante en su estructura, puede tener o no parámetros, retornar un valor, etc.

Funciones

- Ejemplo: Mostrar un mensaje que se repita 3 veces en la página con el siguiente texto:

'Cuidado'

'Ingrese su documento correctamente'

'Cuidado'

'Ingrese su documento correctamente'

'Cuidado'

'Ingrese su documento correctamente'

La solución sin emplear funciones es:

— — —

```
<!DOCTYPE html>
<html>

<head>
  <title>Ejemplo de JavaScript </title>
  <meta charset="UTF-8">
</head>

<body>

  <script>
    document.write( "Cuidado<br>" );
    document.write( "Ingrese su documento
correctamente<br>" );
    document.write( "Cuidado<br>" );
    document.write( "Ingrese su documento
correctamente<br>" );
    document.write( "Cuidado<br>" );
    document.write( "Ingrese su documento
correctamente<br>" );
  </script>

</body>
</html>
```

Empleando una función:

— — —

```
<!DOCTYPE html>
<html>

<head>
  <title>Ejemplo de JavaScript </title>
  <meta charset="UTF-8">
</head>

<body>

  <script>
    function mostrarMensaje() {
      document.write( "Cuidado<br>" );
      document.write( "Ingrese su documento
correctamente<br>" );
    }

    mostrarMensaje();
    mostrarMensaje();
    mostrarMensaje();
  </script>

</body>
</html>
```

Recordemos que

- JavaScript es sensible a mayúsculas y minúsculas. Si fijamos como nombre a la función `mostrarMensaje` (es decir la segunda palabra con la primera letra en mayúscula) debemos respetar este nombre cuando la llamemos a dicha función.
- Es importante notar que para que una función se ejecute debemos llamarla desde fuera por su nombre (en este ejemplo: `mostrarMensaje()`).
- Cada vez que se llama una función se ejecutan todas las líneas contenidas en la misma. Si no se llama a la función, las instrucciones de la misma nunca se ejecutarán.
- A una función la podemos llamar tantas veces como necesitemos. Las funciones nos ahorran escribir código que se repite con frecuencia y permite que nuestro programa sea más entendible.

Acotaciones

- Hemos estado haciendo uso de funciones existentes por defecto en JavaScript como son:

`parseInt`

`parseFloat`

Problema

— — —

Confeccionar una función que solicite el ingreso de un entero y nos muestre la tabla de multiplicar de dicho valor. Llamar luego una vez a la función.

```
<!DOCTYPE html>
<html>

<head>
  <title>Ejemplo de JavaScript</title>
  <meta charset="UTF-8">
</head>

<body>

  <script>
    function calcularTabla() {
      let valor = parseInt(prompt("Ingrese qué tabla de
multiplicar quiere obtener"))
      for (let f = 1; f <= 10; f++) {
        let tabla = f * valor;
        document.write(f + " * " + valor + " = " + tabla +
'<br>')

      }
    }

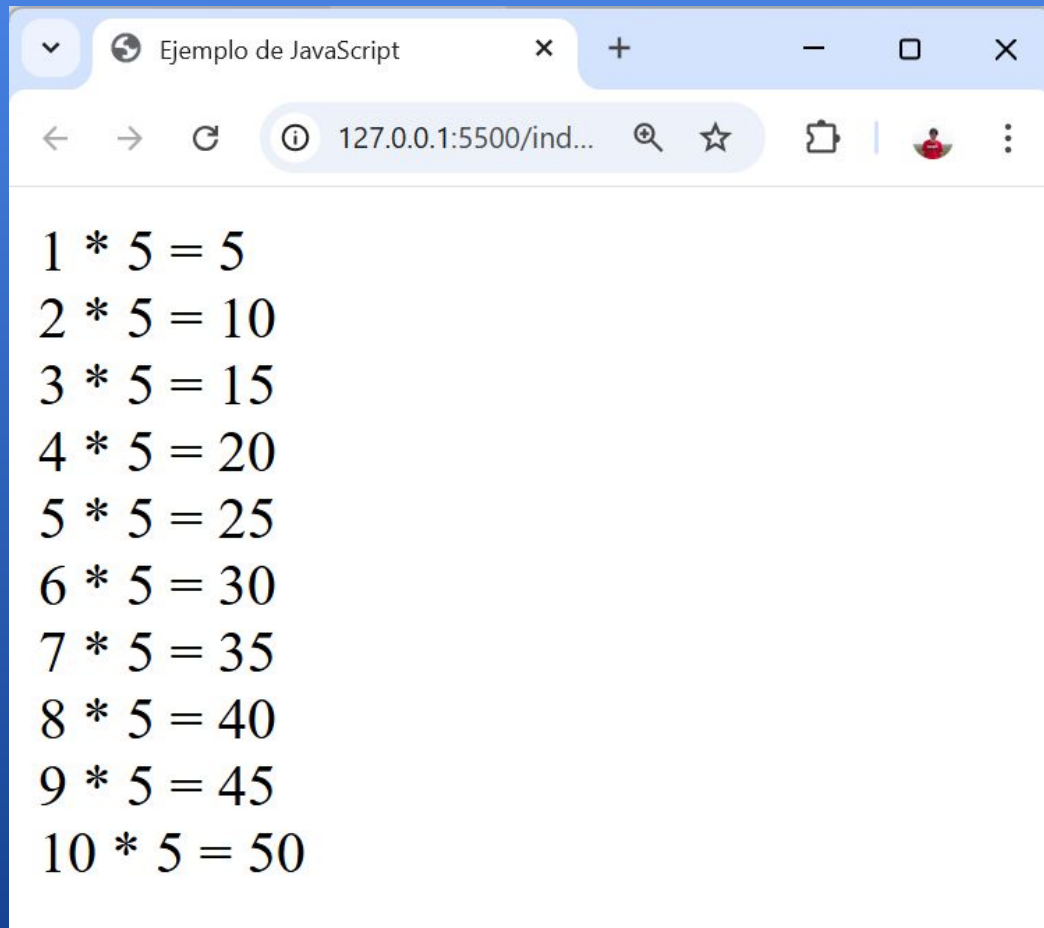
    calcularTabla();
  </script>

</body>
</html>
```

Acotaciones

Dentro de una función podemos definir variables locales como 'valor' y 'f'. También podemos ingresar datos por teclado.

Luego de ingresar el valor 5 por teclado tenemos como resultado en pantalla:



Problema

Confeccionar una función que solicite la carga de dos enteros (primero el menor y luego el mayor) y nos muestre desde el menor hasta el mayor de uno en uno.

Por ejemplo si ingresamos los valores 4 y 10 luego se debe mostrar por pantalla:

4 5 6 7 8 9 10

```
<!DOCTYPE html>
<html>

<head>
  <title>Ejemplo de JavaScript</title>
  <meta charset="UTF-8">
</head>

<body>

  <script>
    function mostrarRango() {
      let inicio = parseInt(prompt('Ingrese valor inicial'))
      let fin = parseInt(prompt('Ingrese valor final'))
      for (let f = inicio; f <= fin; f++) {
        document.write(f + " - ")
      }
    }

    mostrarRango();
  </script>

</body>

</html>
```

Funciones con parámetros

— — —

- Explicaremos con un ejemplo, una función que tiene datos de entrada.

Ejemplo

Confeccionar una función que reciba dos números y muestre en la página los valores comprendidos entre ellos de uno en uno. Cargar por teclado esos dos valores.

```
<!DOCTYPE html>
<html>

<head>
  <title>Ejemplo de JavaScript</title>
  <meta charset="UTF-8">
</head>

<body>

  <script>
    function mostrarComprendidos(inicio, fin) {
      for (let f = inicio; f <= fin; f++) {
        document.write(f + ' ');
      }
    }

    let valor1 = parseInt(prompt('Ingrese valor inferior:'));
    let valor2 = parseInt(prompt('Ingrese valor superior:'));
    mostrarComprendidos(valor1, valor2);
  </script>

</body>

</html>
```

Funciones con parámetros

- El programa de JavaScript empieza a ejecutarse donde definimos las variables valor1 y valor2 y no donde se define la función.
- Luego de cargar los dos valores por teclado se llama a la función mostrarComprendidos y le enviamos las variables valor1 y valor2. Los parámetros inicio y fin reciben los contenidos de las variables valor1 y valor2.
- Es importante notar que a la función la podemos llamar la cantidad de veces que la necesitemos.
- Los nombres de los parámetros, en este caso se llaman inicio y fin, no necesariamente se deben llamar igual que las variables que le pasamos cuando la llamamos a la función, en este caso le pasamos las variable valor1 y valor2.

Problema

Elaborar una función a la cual le enviemos tres enteros y muestre el menor.

```
<!DOCTYPE html>
<html>

<head>
  <title>Ejemplo de JavaScript </title>
  <meta charset="UTF-8">
</head>

<body>

  <script>
    //Elaborar una función a la cual le enviemos tres enteros y muestre el
    menor.
    function mostrarMenor(x1, x2, x3) {
      if (x1 < x2 && x1 < x3) {
        document.write( 'El menor de los tres valores es :' + x1);
      } else {
        if (x2 < x3) {
          document.write( 'El menor de los tres valores es :' + x2);
        } else {
          document.write( 'El menor de los tres valores es :' + x3);
        }
      }
    }

    let valor1 = parseInt(prompt( 'Ingrese primer valor:' ));
    let valor2 = parseInt(prompt( 'Ingrese segundo valor:' ));
    let valor3 = parseInt(prompt( 'Ingrese tercer valor:' ));
    document.write( 'Los tres valores ingresados son ' + valor1 + ' ' + valor2 +
    ' ' + valor3 + '<br>');
    mostrarMenor(valor1, valor2, valor3);
  </script>

</body>

</html>
```

Problema

Confeccionar una función a la cual le envíe tres enteros y los muestre ordenados de menor a mayor.

```
<script>

//Confeccionar una función a la cual le envíe tres
// enteros y los muestre ordenados de menor a mayor.
function mostrarOrdenados(x1, x2, x3) {
  if (x1 < x2 && x1 < x3) {
    document.write(x1 + ' ');
    if (x2 < x3) {
      document.write(x2 + ' ' + x3);
    } else {
      document.write(x3 + ' ' + x2);
    }
  } else if (x2 < x3) {
    document.write(x2 + ' ');
    if (x1 < x3) {
      document.write(x1 + ' ' + x3);
    } else {
      document.write(x3 + ' ' + x1);
    }
  } else {
    document.write(x3 + ' ');
    if (x1 < x2) {
      document.write(x1 + ' ' + x2);
    } else {
      document.write(x2 + ' ' + x1);
    }
  }
}

let valor1 = parseInt(prompt( 'Ingrese primer valor:' ));
let valor2 = parseInt(prompt( 'Ingrese segundo valor:' ));
let valor3 = parseInt(prompt( 'Ingrese tercer valor:' ));
document.write( 'Los tres valores ingresados son ' + valor1 + ' ' + valor2 + ' ' + valor3 +
'<br>');
document.write( 'Los valores ordenados de menor a mayor son: ' );
mostrarOrdenados(valor1, valor2, valor3);

</script>
```

Funciones que retornan un valor

- Son comunes los casos donde una función, luego de hacer un proceso, retorne un valor.

Ejemplo

— — —

Confeccionar una función que reciba un valor entero comprendido entre 1 y 5. Luego retornar en castellano el valor recibido.

```
<script>

function convertirCastellano(x) {
    if (x == 1) {
        return "uno";
    } else {
        if (x == 2) {
            return "dos";
        } else {
            if (x == 3) {
                return "tres";
            } else {
                if (x == 4) {
                    return "cuatro";
                } else {
                    if (x == 5) {
                        return "cinco";
                    } else {
                        return "valor incorrecto";
                    }
                }
            }
        }
    }
}

let valor = parseInt(prompt("Ingrese un valor entre 1 y 5"));
let r = convertirCastellano(valor);
document.write(r);

</script>
```

Funciones que retornan un valor

- Podemos ver que el valor retornado por una función lo indicamos por medio de la palabra clave return. Cuando se llama a la función, debemos asignar el nombre de la función a una variable, ya que la misma retorna un valor.
- Una función puede tener varios parámetros, pero sólo puede retornar un único valor.

Funciones que retornan un valor

La estructura condicional if de este ejemplo puede ser reemplazada por la instrucción switch, la función queda codificada de la siguiente manera:

```
function convertirCastellano(x) {  
    switch (x) {  
        case 1:  
            return "uno";  
        case 2:  
            return "dos";  
        case 3:  
            return "tres";  
        case 4:  
            return "cuatro";  
        case 5:  
            return "cinco";  
        default:  
            return "valor incorrecto";  
    }  
}
```


Funciones que retornan un valor

- Esta es una forma más elegante que una serie de if anidados. La instrucción switch analiza el contenido de la variable x con respecto al valor de cada caso. En la situación de ser igual, ejecuta el bloque seguido de los 2 puntos hasta que encuentra la instrucción return o break.

Ejemplo

Confeccionar una función que reciba una fecha con el formato de día, mes y año y retorne un string con un formato similar a: "Hoy es 27 de enero de 2021".

```
<script>

function formatearFecha(dia, mes, año) {
    let s = 'Hoy es ' + dia + ' de ';
    switch (mes) {
        case 1:
            s = s + 'enero ';
            break;
        case 2:
            s = s + 'febrero ';
            break;
        case 3:
            s = s + 'marzo ';
            break;
        case 4:
            s = s + 'abril ';
            break;
        case 5:
            s = s + 'mayo ';
            break;
        case 6:
            s = s + 'junio ';
            break;
        case 7:
            s = s + 'julio ';
            break;
```

```
        case 9:
            s = s + 'septiembre ';
            break;
        case 10:
            s = s + 'octubre ';
            break;
        case 11:
            s = s + 'noviembre ';
            break;
        case 12:
            s = s + 'diciembre ';
            break;
    } //fin del switch
    s = s + 'de ' + año;
    return s;
}
```

```
    document.write(formatearFecha(27,
1, 2021));
</script>
```

Analicemos un poco la función `formatearFecha`

- Llegan tres parámetros con el día, mes y año.

Definimos e inicializamos una variable con:

```
let s = 'Hoy es ' + dia + ' de ';
```

- Luego le concatenamos o sumamos el mes:

```
s = s + 'enero ';
```

- Esto, si el parámetro `mes` tiene un uno. Observemos cómo acumulamos lo que tiene `s` más el string `'enero '`. En caso de hacer `s = 'enero '` perderíamos el valor previo que tenía la variable `s`.

Analicemos un poco la función `formatearFecha`

Por último concatenamos el año:

```
s=s + 'de ' + año;
```

- Cuando se llama a la función directamente, al valor devuelto se lo enviamos a la función `write` del objeto `document`. Esto último lo podemos hacer en dos pasos:

```
let fec = formatearFecha(27, 1, 2021);
```

```
document.write(fec);
```

- Guardamos en la variable `'fec'` el string devuelto por la función.

Problema

Confeccionar una función a la cual le envíe tres enteros y retorne el mayor de ellos.

```
<script>

    //Confeccionar una función a la cual le envíe tres enteros
    //y retorne el mayor de ellos.

    function retornarMayor(x1, x2, x3) {
        if (x1 > x2 && x1 > x3) {
            return x1;
        } else {
            if (x2 > x3) {
                return x2;
            } else {
                return x3;
            }
        }
    }

    let valor1 = parseInt(prompt('Ingrese primer valor:'));
    let valor2 = parseInt(prompt('Ingrese segundo valor:'));
    let valor3 = parseInt(prompt('Ingrese tercer valor:'));
    document.write('Los tres valores ingresados son ' + valor1 +
        ' ' + valor2 + ' ' + valor3 + '<br>');
    let may = retornarMayor(valor1, valor2, valor3);
    document.write('El mayor de los tres es :'+ may);

</script>
```

Problema

— — —

Elaborar una función a la cual le envíe el valor del lado de un cuadrado y me retorne su perímetro.

```
<script>

    //Elaborar una función a la cual le envíe el
valor del lado de un cuadrado
    //y me retorne su perímetro.

    function retornarPerimetro(lado) {
        let perimetro;
        perimetro = lado * 4;
        return perimetro;
    }

    let lado = parseInt(prompt('Ingrese la
medida del lado de un cuadrado:'));
    document.write('El perímetro del cuadrado
es:' + retornarPerimetro(lado));
</script>
```

Problema

— — —

Desarrollar una función que retorne la cantidad de dígitos que tiene una variable entera positiva de hasta 5 dígitos.

```
<script>

function cantidadDigitos(x) {
    if (x < 10) {
        return 1;
    } else {
        if (x < 100) {
            return 2;
        } else {
            if (x < 1000) {
                return 3;
            } else {
                if (x < 10000) {
                    return 4;
                } else {
                    if (x < 100000) {
                        return 5;
                    }
                }
            }
        }
    }
}

let valor = parseInt(prompt('Ingrese un valor positivo de hasta 5 dígitos:'));

document.write('La cantidad de dígitos del valor ingresado es:' + cantidadDigitos(valor));

</script>
```

Problema

— — —

Elaborar una función que reciba tres enteros y retorne el promedio.

```
<script>

    //Elaborar una función que reciba tres enteros y
    retorne el promedio.

    function promediar(x1, x2, x3) {
        let promedio = (x1 + x2 + x3) / 3;
        return promedio;
    }

    let valor1 = parseInt(prompt('Ingrese primer valor:'));
    let valor2 = parseInt(prompt('Ingrese segundo
valor:'));
    let valor3 = parseInt(prompt('Ingrese tercer valor:'));
    document.write('Los tres valores ingresados son ' +
valor1 + ' ' + valor2 + ' ' + valor3 + '<br>');
    let pro;
    pro = promediar(valor1, valor2, valor3);
    document.write('El promedio es :' + pro);
</script>
```


Problema

— — —

Confeccionar una función que solicite la carga de 5 valores por teclado y retorne su suma.

```
<script>
    //Confeccionar una función que solicite la carga de 5
valores por teclado
    //y retorne su suma.

    function cargar5Valores() {
        let suma = 0;
        for (let f = 1; f <= 5; f++) {
            let valor = parseInt(prompt('Ingrese valor:'));
            suma = suma + valor;
        }
        return suma;
    }

    let s = cargar5Valores();
    document.write('La suma de los 5 valores es: ' + s);
</script>
```

EJERCICIOS ADICIONALES PROPUESTOS

— — —





**¡MUCHAS
GRACIAS!**