



# Control de Flujo en JavaScript

## Clase 5

KOICA

IGU HANDONG GLOBAL  
UNIVERSITY



UNA

# OBJETIVOS DE LA CLASE 5

---

- Repasar los conceptos de estructuras repetitivas.
- Implementar los códigos de ejemplos propuestos en clase.



# Estructura Repetitiva (While)

---

- Hasta ahora hemos empleado estructuras SECUENCIALES y CONDICIONALES. Existe otro tipo de estructuras tan importantes como las anteriores que son las estructuras REPETITIVAS.
- Una estructura repetitiva permite ejecutar una instrucción o un conjunto de instrucciones varias veces.
- Una ejecución repetitiva de sentencias se caracteriza por:
  - La o las sentencias que se repiten.
  - El test o prueba de condición antes de cada repetición, que motivará que se repitan o no las sentencias.

# Funcionamiento del while

---

- En primer lugar se verifica la condición, si la misma resulta verdadera se ejecutan las operaciones que indicamos entre las llaves que le siguen al while.
- En caso que la condición sea Falsa continúa con la instrucción siguiente al bloque de llaves.
- El bloque se repite MIENTRAS la condición sea Verdadera.
- Importante: Si la condición siempre retorna verdadero estamos en presencia de un ciclo repetitivo infinito. Dicha situación es un error de programación, nunca finalizará el programa.

# Ejemplo

— — —

Realizar un programa  
que imprima en pantalla  
los números del 1 al  
100.

```
<!DOCTYPE html>
<html>
<head>
  <title>Ejemplo de JavaScript</title>
  <meta charset="UTF-8">
</head>

<body>
  <script>
    let x;
    x = 1;
    while (x <= 100) {
      document.write(x);
      document.write('<br>');
      x = x + 1;
    }
  </script>
</body>

</html>
```

# Funcionamiento del while

---

- Para que se impriman los números, uno en cada línea, agregamos la etiqueta HTML de `<br>`.
- Es muy importante analizar este programa:
- La primera operación inicializa la variable `x` en 1, seguidamente comienza la estructura repetitiva `while` y disponemos la siguiente condición ( `x <= 100`), se lee MIENTRAS la variable `x` sea menor o igual a 100.
- Es importante decir que NO debe haber un punto y coma al final del `while`, si hacemos esto estamos en presencia de un error lógico.
- Al ejecutarse la condición, retorna VERDADERO, porque el contenido de `x` (1) es menor o igual a 100.

# Funcionamiento del while

---

- Al ser la condición verdadera se ejecuta el bloque de instrucciones que contiene la estructura while. El bloque de instrucciones contiene dos salidas al documento y una operación.
- Se imprime el contenido de x y seguidamente se incrementa la variable x en uno.
- La operación  $x = x + 1$  se lee como "en la variable x se guarda el contenido de x más 1". Es decir, si x contiene 1 luego de ejecutarse esta operación se almacenará en x un 2.

# Funcionamiento del while

---

- Al finalizar el bloque de instrucciones que contiene la estructura repetitiva, se verifica nuevamente la condición de la estructura repetitiva y se repite el proceso explicado anteriormente.
- Mientras la condición retorne verdadero, se ejecuta el bloque de instrucciones; al retornar falso la verificación de la condición, se sale de la estructura repetitiva y continúa el algoritmo, en este caso, finaliza el programa.



# Funcionamiento del while

---

- Lo más difícil es la definición de la condición de la estructura while y qué bloque de instrucciones se va a repetir. Observar que si, por ejemplo, disponemos la condición  $x \geq 100$  ( si  $x$  es mayor o igual a 100) no provoca ningún error sintáctico pero estamos en presencia de un error lógico porque al evaluarse por primera vez la condición retorna falso y no se ejecuta el bloque de instrucciones que queríamos repetir 100 veces.
- No existe una RECETA para definir una condición de una estructura repetitiva, sino que se logra con una práctica continua, solucionando problemas.

# Funcionamiento del while

---

- Una vez planteado el programa debemos verificar si el mismo es una solución válida al problema (en este caso se deben imprimir los números del 1 al 100 en la página), para ello podemos hacer un seguimiento del flujo y los valores que toman las variables a lo largo de la ejecución:

```
x
1
2
3
4
.
.
100
```

# Funcionamiento del while

---

- La variable x recibe el nombre de CONTADOR. Un contador es un tipo especial de variable que se incrementa o decrementa con valores constantes durante la ejecución del programa.
- El contador x nos indica en cada momento la cantidad de valores impresos en la página.
- Importante: Podemos observar que el bloque repetitivo puede no ejecutarse si la condición retorna falso la primera vez.

# Funcionamiento del while

---

- La variable x debe estar inicializada con algún valor antes que se ejecute la operación  $x = x + 1$ .
- Probemos algunas modificaciones de este programa y veamos qué cambios se deberían hacer para:
  - 1 - Imprimir los números del 1 al 500.
  - 2 - Imprimir los números del 50 al 100.
  - 3 - Imprimir los números del -50 al 0.
  - 4 - Imprimir los números del 2 al 100 pero de 2 en 2 (2,4,6,8 ....100).

# Problema

— — —

Realizar un programa que imprima 25 términos de la serie 11 - 22 - 33 - 44, etc. (No se ingresan valores por teclado).

```
<!DOCTYPE html>
<html>

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

<body>

  <script>
    let serie;
    serie = 11;
    let x;
    x = 1;
    while (x <= 25) {
      document.write(serie);
      document.write('<br>');
      x = x + 1;
      serie = serie + 11;
    }
  </script>

</body>

</html>
```

# Problema

— — —

Mostrar los múltiplos de 8 hasta el valor 500. Debe aparecer en pantalla 8 -16 -24, etc.

```
<!DOCTYPE html>
<html>

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

<body>

  <script>
    let multiplo8;
    multiplo8 = 8;
    while (multiplo8 <= 500) {
      document.write(multiplo8);
      document.write('<br>');
      multiplo8 = multiplo8 + 8;
    }
  </script>

</body>

</html>
```

# Concepto de Acumulador

— — —

Explicaremos el concepto de un acumulador con un ejemplo.

# Ejemplo

---

Desarrollar un programa que permita la carga de 5 valores por teclado y nos muestre posteriormente la suma.

```
<!DOCTYPE html>
<html>

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

<body>

  <script>
    let x = 1;
    let suma = 0;
    let valor;
    while (x <= 5) {
      valor = parseInt(prompt('Ingrese valor:'));
      suma = suma + valor;
      x = x + 1;
    }
    document.write('La suma de los valores es ' + suma
+ '<br>');
  </script>

</body>

</html>
```



# Concepto de Acumulador

---

- Podemos definir e inmediatamente inicializar la variable:

```
let x = 1;
```

- En este problema, a semejanza de los anteriores, llevamos un CONTADOR llamado x que nos sirve para contar las vueltas que debe repetir el while.
- También aparece el concepto de ACUMULADOR (un acumulador es un tipo especial de variable que se incrementa o decrementa con valores variables durante la ejecución del programa)

# Concepto de Acumulador

---

- Hemos dado el nombre de suma a nuestro acumulador. Cada ciclo que se repita la estructura repetitiva, la variable suma se incrementa con el contenido ingresado en la variable valor.
- La prueba se realiza dándole valores a las variables:

valor	suma	x
0	0	1
5	5	2
16	21	3
7	28	4
10	38	5
2	40	6

(Antes de entrar a la estructura repetitiva estos son los valores).

# Concepto de Acumulador

---

- Este es un seguimiento del programa planteado. Los números que toma la variable valor dependerá de qué cifras cargue el operador durante la ejecución del programa.
- Hay que tener en cuenta que cuando en la variable valor se carga el primer número (en éste ejemplo es el valor 5), al cargarse el segundo valor (16), el valor anterior 5 se pierde, por ello la necesidad de ir almacenando en la variable suma el valor acumulado de los valores ingresados.

# Concepto de Acumulador

---

Cada una de las tres variables tiene un objetivo distinto:

- valor : tiene por objetivo cargar valores por teclado.
- x : nos sirva para contar cuantas veces se ha repetido el while, sabemos que debemos cortar cuando x toma el valor 6.
- suma : almacena la suma de valores ingresados hasta ese momento en la variable valor.

# Problema

---

Escribir un programa que lea 10 notas de alumnos y nos informe cuántos tienen notas mayores o iguales a 7 y cuántos menores.

```
<script>
    let x = 0;
    let cant1 = 0;
    let cant2 = 0;
    while (x < 10) {
        let nota;
        nota = parseInt(prompt('Ingrese
nota'));

        if (nota >= 7) {
            cant1 = cant1 + 1;
        } else {
            cant2 = cant2 + 1;
        }
        x = x + 1;
    }
    document.write('Cantidad de alumnos con
notas mayores o iguales a 7:' + cant1);
    document.write('<br>');
    document.write('Cantidad de alumnos con
notas menores a 7:' + cant2);
</script>
```

# Problema

— — —

Se ingresan un conjunto de 5 alturas de personas por teclado. Mostrar la altura promedio de las personas.

```
<script>
    let x = 0;
    let suma = 0;
    while (x < 5) {
        let altura;
        altura = parseInt(prompt('Ingrese la
altura en centímetros(Ej. 175)'));
        suma = suma + altura;
        x = x + 1;
    }
    let promedio = suma / 5;
    document.write('La altura promedio de las
cinco personas es:' + promedio);
</script>
```

# Problema

— — —

En una empresa trabajan 5 empleados cuyos sueldos oscilan entre \$100 y \$500, realizar un programa que lea los sueldos que cobra cada empleado e informe cuántos empleados cobran entre \$100 y \$300 y cuántos cobran más de \$300. Además el programa deberá informar el importe que gasta la empresa en sueldos al personal.

```
<script>
    let cont1 = 0;
    let cont2 = 0;
    let total = 0;
    let sueldo;
    let x = 0;
    while (x < 5) {
        sueldo = parseInt(prompt('Ingrese el
sueldo'));
        if (sueldo <= 300) {
            cont1 = cont1 + 1;
        } else {
            cont2 = cont2 + 1;
        }
        total = total + sueldo;
        x = x + 1;
    }
    document.write('Cantidad de empleados que cobran
300 o menos:' + cont1);
    document.write('<br>');
    document.write('Cantidad de empleados que cobran
más de 300:' + cont2);
    document.write('<br>');
    document.write('Gastos en sueldos en la empresa:'
+ total);
</script>
```

# Problema

— — —

Realizar un programa que imprima 20 términos de la serie 5 - 10 - 15 - 20, etc. (No se ingresan valores por teclado)

```
<!DOCTYPE html>
<html>

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

<body>

  <script>
    let serie = 5;
    let x = 0;
    while (x < 20) {
      document.write(serie + ' ');
      x = x + 1;
      serie = serie + 5;
    }
  </script>

</body>

</html>
```



# Problema

— — —

Mostrar los múltiplos de 10 hasta el valor 1500.

Debe aparecer en pantalla 10 - 20 -30 etc.

```
<script>
    document.write('Múltiplos de 10 hasta
el 1500<br>');
    let multiplo = 10;
    while (multiplo <= 1500) {
        document.write(multiplo + ' ');
        multiplo = multiplo + 10;
    }
</script>
```

# Problema

---

Realizar un programa que permita cargar dos listas de 3 valores cada una. Informar con un mensaje cuál de las dos listas tiene un valor acumulado mayor (mensajes 'Lista 1 mayor', 'Lista 2 mayor', 'Listas iguales')

Tener en cuenta que puede haber dos o más estructuras repetitivas en un algoritmo.

```
<script>

    let total1 = 0;
    let x = 0;
    let nro;
    while (x < 3) {
        nro = parseInt(prompt('Ingrese valor de la primera lista:'));
        total1 = total1 + nro;
        x = x + 1;
    }
    let total2 = 0;
    x = 0;
    while (x < 3) {
        nro = parseInt(prompt('Ingrese valor de la segunda lista:',
        ''));

        total2 = total2 + nro;
        x = x + 1;
    }
    if (total1 > total2) {
        document.write('Tiene mayor valor la lista1');
    } else {
        if (total1 < total2) {
            document.write('Tiene mayor valor la lista2');
        } else {
            document.write('Tienen el mismo valor acumulado las dos
listas');
        }
    }
}

</script>
```

# Problema

---

Desarrollar un programa que permita cargar 5 números enteros y luego nos informe cuántos valores fueron pares y cuántos impares.

Emplear el operador "%" en la condición de la estructura condicional.

```
if (valor%2==0)
```

El operador "%" retorna el resto de la división  $\text{valor} / 2$ . Por ejemplo:  $12 \% 2$ , retorna 0;  $13 \% 2$ , retorna 1, porque el resto de dividir 13 en 2 es 1.

```
<script>

    let cantpares = 0;
    let cantimpares = 0;
    let x = 0;
    let valor;
    while (x < 5) {
        valor = parseInt(prompt( 'Ingrese un valor' ));
        if (valor % 2 == 0) {
            cantpares = cantpares + 1;
        } else {
            cantimpares = cantimpares + 1;
        }
        x = x + 1;
    }
    document.write( 'Cantidad de valores pares
ingresados:' + cantpares);
    document.write( '<br>');
    document.write( 'Cantidad de valores impares
ingresados:' + cantimpares);
</script>
```

# Estructura Repetitiva (Do/While)

---

- La sentencia do/while es otra estructura repetitiva, la cual ejecuta al menos una vez su bloque repetitivo, a diferencia del while que puede no ejecutar el bloque.
- Esta estructura repetitiva se utiliza cuando conocemos de antemano que por lo menos una vez se ejecutará el bloque repetitivo.
- La condición de la estructura está abajo del bloque a repetir, a diferencia del while que está en la parte superior.
- Finaliza la ejecución del bloque repetitivo cuando la condición retorna falso, es decir igual que el while.

# Ejemplo

---

Escribir un programa que solicite la carga de un número entre 0 y 999, y nos muestre un mensaje de cuántos dígitos tiene el mismo. Finalizar el programa cuando se cargue el valor 0.

```
<script>
    let valor;
    do {
        valor = parseInt(prompt('Ingrese un
valor entre 0 y 999:', ''));
        document.write('El valor ' + valor + '
tiene ');

        if (valor < 10) {
            document.write('Tiene 1 digitos');
        } else {
            if (valor < 100) {
                document.write('Tiene 2
digitos');
            } else {
                document.write('Tiene 3
digitos');
            }
        }

        document.write('<br>');
    } while (valor != 0);
</script>
```

# Estructura Repetitiva (Do/While)

---

- En este problema por lo menos se carga un valor. Si se carga un valor menor a 10 se trata de un número de una cifra, si es mayor a 10 pero menor a 100 se trata de un valor de dos dígitos, en caso contrario se trata de un valor de tres dígitos. Este bloque se repite mientras se ingresa en la variable 'valor' un número distinto a 0.

# Problema

— — —

Realizar un programa que acumule (sume) valores ingresados por teclado hasta ingresa el 9999 (no sumar dicho valor, solamente indica que ha finalizado la carga). Imprimir el valor acumulado e informar si dicho valor es cero, mayor a cero o menor a cero.

```
<script>

    let valor;
    let suma = 0;
    do {
        valor = parseInt(prompt('Ingrese un valor (9999 para
finalizar)'));
        if (valor != 9999) {
            suma = suma + valor;
        }
    } while (valor != 9999);
    document.write('Valor acumulado total:' + suma);
    document.write('<br>');
    if (suma > 0) {
        document.write('El valor acumulado es mayor a cero');
    } else {
        if (suma == 0) {
            document.write('El valor acumulado es cero');
        } else {
            document.write('El valor acumulado es menor a
cero');
        }
    }
}

</script>
```

# Problema

En un banco se procesan datos de las cuentas corrientes de sus clientes. De cada cuenta corriente se conoce: número de cuenta, nombre del cliente y saldo actual. El ingreso de datos debe finalizar al ingresar un valor negativo en el número de cuenta.

Se pide confeccionar un programa que lea los datos de las cuentas corrientes e informe:

a) De cada cuenta: número de cuenta, nombre del cliente y estado de la cuenta según su saldo, sabiendo que:

Estado de la cuenta      'Acreedor' si el saldo es >0.

                                 'Deudor' si el saldo es <0.

                                 'Nulo' si el saldo es =0.

b) La suma total de los saldos acreedores.

```
<script>

let nrocuenta;
let nombre;
let saldo = 0;
let saldoacre = 0;
do {
    nrocuenta = parseInt(prompt('Ingrese nro de cuenta:'));
    if (nrocuenta >= 0) {
        nombre = prompt('Nombre del cliente:');
        saldo = parseFloat(prompt('Saldo actual:'));
        if (saldo > 0) {
            saldoacre = saldoacre + saldo;
            document.write(nombre + ' tiene saldo acreedor<br>');
        } else {
            if (saldo < 0) {
                document.write(nombre + ' tiene saldo
deudor<br>');
            } else {
                document.write(nombre + ' tiene saldo nulo<br>');
            }
        }
    }
} while (nrocuenta > 0);
document.write('Suma total de saldos acreedores:' + saldoacre);

</script>
```



# Problema

Se realizó un censo provincial y se desea procesar la información obtenida en dicho censo. De cada una de las personas censadas se tiene la siguiente información: número de documento, edad y sexo ('femenino' o 'masculino')

Se pide confeccionar un programa que lea los datos de cada persona censada (para finalizar ingresar el valor cero en el número de documento) e informar:

- a) Cantidad total de personas censadas.
- b) Cantidad de varones.
- c) Cantidad de mujeres.
- d) Cantidad de varones cuya edad varía entre 16 y 65 años.

```
<script>

    let documento;
    let edad;
    let totalpersonas = 0;
    let cantvarones = 0;
    let cantmujeres = 0;
    let cantvaronesgrandes = 0;
    do {
        documento = parseInt(prompt( 'Ingrese nro de documento (0 para
finalizar):' ));
        if (documento > 0) {
            edad = parseInt(prompt( 'Ingrese la edad:' ));
            sexo = prompt( 'Ingrese el sexo (masculino/femenino):' );
            if (sexo == 'masculino') {
                cantvarones = cantvarones + 1;
                if (edad >= 16 && edad <= 65) {
                    cantvaronesgrandes = cantvaronesgrandes + 1;
                }
            }
            if (sexo == 'femenino') {
                cantmujeres = cantmujeres + 1;
            }
            totalpersonas = totalpersonas + 1;
        }
    } while (documento != 0);
    document.write( 'Total de personas censadas: ' + totalpersonas +
'<br>' );
    document.write( 'Cantidad de varones: ' + cantvarones + '<br>' );
    document.write( 'Cantidad de mujeres: ' + cantmujeres + '<br>' );
    document.write( 'Cantidad de varones entre 16 y 65 años: ' +
cantvaronesgrandes + '<br>' );
</script>
```

# Estructura Repetitiva (For)

— — —

- Cualquier problema que requiera una estructura repetitiva se puede resolver empleando la estructura while. Pero hay otra estructura repetitiva cuyo planteo es más sencillo en ciertas situaciones.
- Esta estructura se emplea en aquellas situaciones en las cuales CONOCEMOS la cantidad de veces que queremos que se ejecute el bloque de instrucciones. Ejemplo: cargar 10 números, ingresar 5 notas de alumnos, etc. Conocemos de antemano la cantidad de veces que queremos que el bloque se repita.

# Estructura Repetitiva (For)

---

- Por último, hay que decir que la ejecución de la sentencia break dentro de cualquier parte del bucle provoca la salida inmediata del mismo.

Sintaxis:

```
for (<Inicialización> ; <Condición> ; <Incremento o Decremento>)
```

```
{
```

```
    <Instrucciones>
```

```
}
```

# Estructura Repetitiva (For)

---

- Esta estructura repetitiva tiene tres argumentos: variable de inicialización, condición y variable de incremento o decremento.
- Este tipo de estructura repetitiva se utiliza generalmente cuando sabemos la cantidad de veces que deseamos que se repita el bloque.

# Funcionamiento

---

- 1 - Primero se ejecuta por única vez el primer argumento <Inicialización>. Por lo general se inicializa una variable.
- 2 - El segundo paso es evaluar la (Condición), en caso de ser verdadera se ejecuta el bloque, en caso contrario continúa el programa.
- 3 - El tercer paso es la ejecución de las instrucciones.
- 4 - El cuarto paso es ejecutar el tercer argumento (Incremento o Decremento).
- 5 - Luego se repiten sucesivamente del Segundo al Cuarto Paso.

# Ejemplo

— — —

Mostrar por pantalla los números del 1 al 10.

```
<!DOCTYPE html>
<html>

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

<body>

  <script>
    for (let f = 1; f <= 10; f++) {
      document.write(f + " ");
    }
  </script>

</body>

</html>
```

# Estructura Repetitiva (For)

---

- Podemos definir la variable `f` directamente dentro del `for` si no se la requiere a dicha variable después del `for`. Como la condición se verifica como verdadera ( $1 \leq 10$ ) se ejecuta el bloque del `for` (en este caso mostramos el contenido de la variable `f` y un espacio en blanco). Luego de ejecutar el bloque pasa al tercer argumento del `for` (en este caso con el operador `++` se incrementa en uno el contenido de la variable `f`, existe otro operador `--` que decrementa en uno una variable), hubiera sido lo mismo poner `f=f+1` pero este otro operador matemático nos simplifica las cosas.

# Estructura Repetitiva (For)

---

- Importante: Tener en cuenta que no lleva punto y coma al final de los tres argumentos del for.
- El disponer un punto y coma provoca un error lógico y no sintáctico, por lo que el navegador no avisará.



# Acotaciones

---

Acá ya podemos mostrar una ventaja de utilizar la palabra clave 'let' para definir la variable en lugar de la sintaxis antigua de JavaScript de utilizar la palabra clave 'var'.

Si utilizamos la palabra clave 'var' en principio el resultado es lo mismo:

```
<script>
    for (var f = 1; f <= 10; f++) {
        document.write(f + " ");
    }
</script>
```

# Acotaciones

---

Pero si nos equivocamos y volvemos a mostrar la variable 'f' fuera del for:

```
<script>
  for (var f = 1; f <= 10; f++) {
    document.write(f + " ");
  }
  document.write(f + " ");
</script>
```

# Acotaciones

---

Como podemos comprobar la variable 'f' sigue existiendo y tiene almacenado el valor '10' (gracias a dicho valor sale del for).

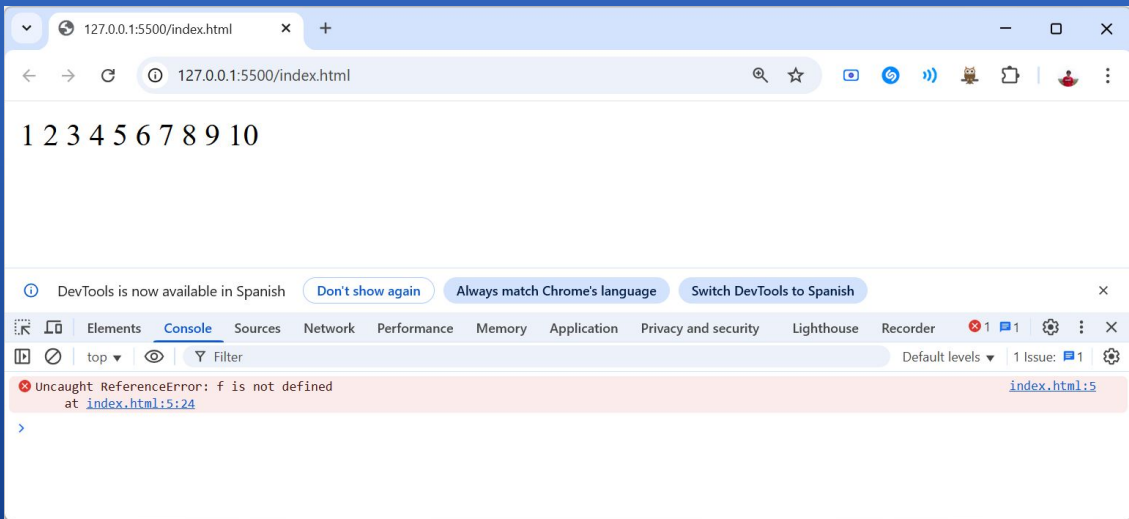
Si utilizamos la palabra clave 'let' la variable 'f' tiene existencia dentro del ámbito donde se la definió, en nuestro caso solo existe dentro del for, luego si codificamos:

```
<script>
    for (let f = 1; f <= 10;
f++) {
        document.write(f + "
");
    }
    document.write(f + " ");
</script>
```

# Acotaciones

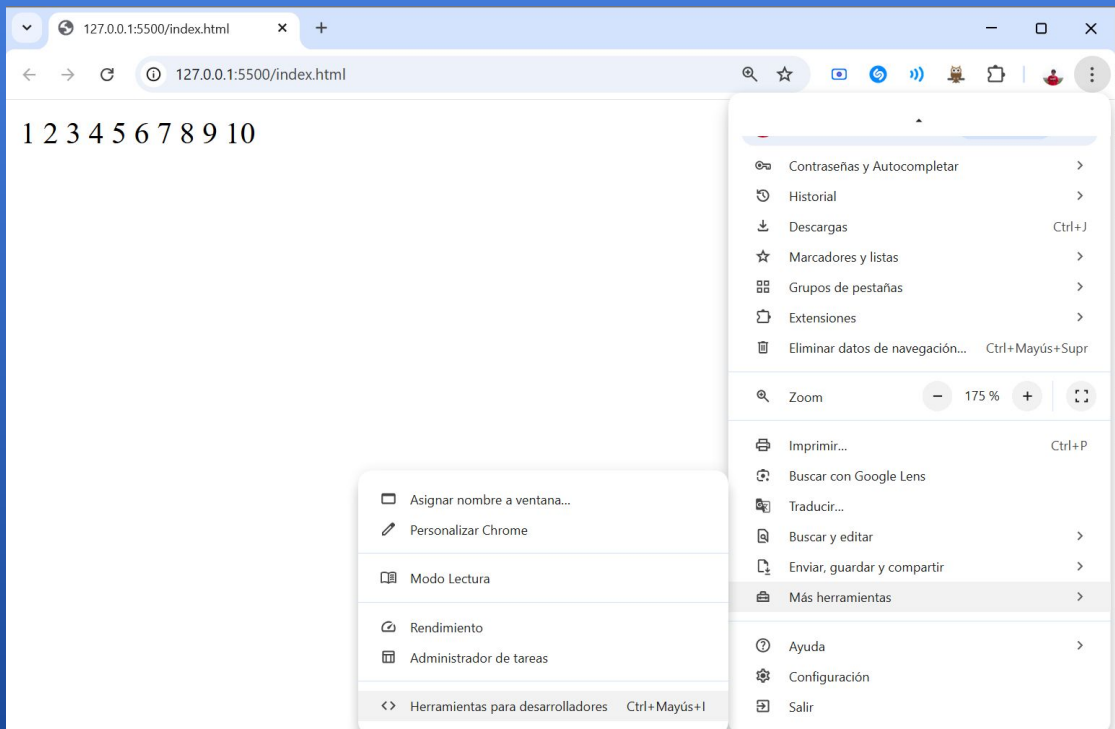
— — —

Podemos ver que el intérprete de JavaScript detecta como un error el acceso a la variable 'f' fuera del for:



# Acotaciones

La ventana de "Herramientas para desarrolladores" es de fundamental importancia para detectar los errores sintácticos y lógicos de nuestra aplicación. La podemos abrir presionando la tecla 'F12' o desde el menú de opciones:



Acotar el ámbito para el acceso a una variable es una muy buena práctica que nos va a evitar dolores de cabeza cuando tenemos programas grandes (cientos o miles de líneas)

# Problema

---

Confeccionar un programa que lea 3 pares de datos, cada par de datos corresponde a la medida de la base y la altura de un triángulo. El programa deberá informar:

a) De cada triángulo la medida de su base, su altura y su superficie.

b) La cantidad de triángulos cuya superficie es mayor a 12.

```
<script>
    let contal = 0;
    for (let f = 1; f <= 3; f++) {
        let base = parseInt(prompt('Ingrese la base:'));
        let altura = parseInt(prompt('Ingrese la
altura:'));
        let superficie = base * altura / 2;
        if (superficie > 12) {
            contal++;
        }
        document.write('Triángulo nro:' + f + '<br>');
        document.write('Base:' + base + '<br>');
        document.write('Altura:' + altura + '<br>');
        document.write('Superficie:' + superficie +
'<br>');
    }
    document.write('Cantidad de triángulos con superficie
mayor a 12:' + contal);
</script>
```

# Problema

— — —

Desarrollar un programa que solicite la carga de 10 números e imprima la suma de los últimos 5 valores ingresados.

```
<script>
/* Desarrollar un programa que solicite la carga
de 10 números
e imprima la suma de los últimos 5 valores
ingresados.*/
let suma = 0;
for (let f = 1; f <= 10; f++) {
    let valor = parseInt(prompt('Ingrese un
nro:'));
    if (f > 5) {
        suma = suma + valor;
    }
}
document.write('La suma de los últimos cinco
valores ingresados es: ' + suma);
</script>
```

# Problema

— — —

Desarrollar un programa que muestre la tabla de multiplicar del 5 (del 5 al 50).

```
<script>
    //Desarrollar un programa que muestre
    la tabla de multiplicar del 5 (del 5 al 50)
    let tabla = 5;
    for (let f = 1; f <= 10; f++) {
        document.write(tabla + '-');
        tabla = tabla + 5;
    }
</script>
```



# Problema

— — —

Confeccionar un programa que permita ingresar un valor del 1 al 10 y nos muestre la tabla de multiplicar del mismo (los primeros 12 términos)

Ejemplo: Si ingreso 3 deberá aparecer en pantalla los valores 3, 6, 9, hasta el 36.

```
<script>
    /*Confeccionar un programa que permita
    ingresar un valor del 1 al 10 y
    nos muestre la tabla de
    multiplicar del mismo (los primeros 12
    términos)

    Ejemplo: Si ingreso 3 deberá
    aparecer en pantalla los valores 3, 6, 9, hasta
    el 36.*/

    let tabla = parseInt(prompt('Ingrese un
    valor del 1 al 10:', ''));
    let conta = tabla;
    for (let f = 1; f <= 12; f++) {
        document.write(conta + ' ');
        conta = conta + tabla;
    }
</script>
```

# Problema

Realizar un programa que lea los lados de 4 triángulos, e informar:

a) De cada uno de ellos, qué tipo de triángulo es: equilátero (tres lados iguales), isósceles (dos lados iguales), o escaleno (ningún lado igual)

b) Cantidad de triángulos de cada tipo.

c) Tipo de triángulo del que hay menor cantidad.

```
<script>
    let cant1 = 0, cant2 = 0, cant3 = 0;
    for (let f = 1; f <= 4; f++) {
        let lado1 = parseInt(prompt(Ingrese primer lado:));
        let lado2 = parseInt(prompt(Ingrese segundo lado:));
        let lado3 = parseInt(prompt(Ingrese tercer lado:));
        if (lado1 == lado2 && lado1 == lado3) {
            document.write(Triángulo equilátero.<br>);
            cant1++;
        } else {
            if (lado1 == lado2 || lado1 == lado3 || lado2 == lado3) {
                document.write(Triángulo isósceles.<br>);
                cant2++;
            } else {
                document.write(Triángulo escaleno.<br>);
                cant3++;
            }
        }
    }
    document.write(<br>');
    document.write(Cantidad de triángulos equiláteros:' + cant1 + '<br>');
    document.write(Cantidad de triángulos isósceles:' + cant2 + '<br>');
    document.write(Cantidad de triángulos escalenos:' + cant3 + '<br>');
    if (cant1 < cant2 && cant1 < cant3) {
        document.write(Se ingresaron menos triángulos equiláteros');
    } else {
        if (cant2 < cant3) {
            document.write(Se ingresaron menos triángulos isósceles');
        } else {
            document.write(Se ingresaron menos triángulos escalenos');
        }
    }
}
</script>
```

# Problema

— — —

Escribir un programa que pida ingresar coordenadas (x,y) que representan puntos en el plano. Informar cuántos puntos se han ingresado en el primer, segundo, tercer y cuarto cuadrante. Al comenzar el programa se pide que se ingrese la cantidad de puntos a procesar.

```
<script>

/*Escribir un programa que pida ingresar coordenadas (x,y) que representan puntos en el plano.
   Informar cuántos puntos se han ingresado en el primer, segundo, tercer y cuarto cuadrante.
   Al comenzar el programa se pide que se ingrese la cantidad de puntos a procesar.*/

let cuad1 = 0;
let cuad2 = 0;
let cuad3 = 0;
let cuad4 = 0;
let cant = parseInt(prompt(Cuantos puntos procesará:));
cant = parseInt(cant);
for (let f = 1; f <= cant; f++) {
    let x = parseInt(prompt(Ingrese coordenada x:));
    let y = parseInt(prompt(Ingrese coordenada y:));
    if (x > 0 && y > 0) {
        cuad1++;
    } else {
        if (x < 0 && y > 0) {
            cuad2++;
        } else {
            if (x < 0 && y < 0) {
                cuad3++;
            } else {
                if (x > 0 && y < 0) {
                    cuad4++;
                }
            }
        }
    }
}

document.write(Cantidad de puntos ingresados en el primer cuadrante:# cuad1 + '<br>');
document.write(Cantidad de puntos ingresados en el segundo cuadrante:# cuad2 + '<br>');
document.write(Cantidad de puntos ingresados en el tercer cuadrante:# cuad3 + '<br>');
document.write(Cantidad de puntos ingresados en el cuarto cuadrante:# cuad4 + '<br>');

</script>
```

# Problema

Se realiza la carga de 10 valores enteros por teclado. Se desea conocer:

- a) La cantidad de valores negativos ingresados.
- b) La cantidad de valores positivos ingresados.
- c) La cantidad de múltiplos de 15.
- d) El valor acumulado de los números ingresados que son pares.

```
<script>
    let cantnegativos = 0;
    let cantpositivos = 0;
    let mult15 = 0;
    let sumapares = 0;
    for (let f = 1; f <= 10; f++) {
        let valor = parseInt(prompt('Ingrese un valor:'));
        if (valor < 0) {
            cantnegativos++;
        } else {
            if (valor > 0) {
                cantpositivos++;
            }
        }
        if (valor % 15 == 0 && valor != 0) {
            mult15++;
        }
        if (valor % 2 == 0) {
            sumapares = sumapares + valor;
        }
    }
    document.write('Cantidad de valores negativos:' + cantnegativos +
'<br>');
    document.write('Cantidad de valores positivos:' + cantpositivos +
'<br>');
    document.write('Cantidad de múltiplos de 15:' + mult15 + '<br>');
    document.write('Suma de los valores pares ingresados:' + sumapares +
'<br>');
</script>
```

# Problema

---

Se cuenta con la siguiente información:

Las edades de 5 estudiantes del turno mañana.

Las edades de 6 estudiantes del turno tarde.

Las edades de 11 estudiantes del turno noche.

Las edades de cada estudiante deben ingresarse por teclado.

a) Obtener el promedio de las edades de cada turno (tres promedios).

b) Imprimir dichos promedios (promedio de cada turno).

c) Mostrar por pantalla un mensaje que indique cual de los tres turnos tiene un promedio de edades mayor.

```
<script>

let suma1 = 0;
for (let f = 1; f <= 5; f++) {
    let edad = parseInt(prompt( 'Edad de estudiante del turno mañana:' ));
    suma1 = suma1 + edad;
}

let suma2 = 0;
for (let f = 1; f <= 6; f++) {
    let edad = parseInt(prompt( 'Edad de estudiante del turno tarde:' ));
    suma2 = suma2 + edad;
}

let suma3 = 0;
for (let f = 1; f <= 11; f++) {
    let edad = parseInt(prompt( 'Edad de estudiante del turno noche:' ));
    suma3 = suma3 + edad;
}

let promedio1 = suma1 / 5;
let promedio2 = suma2 / 6;
let promedio3 = suma3 / 11;

document.write( 'Promedio de edades de alumnos del turno mañana:' + promedio1 +
'<br>');

document.write( 'Promedio de edades de alumnos del turno tarde:' + promedio2 +
'<br>');

document.write( 'Promedio de edades de alumnos del turno noche:' + promedio3 +
'<br>');

if (promedio1 > promedio2 && promedio1 > promedio3) {
    document.write( 'El turno mañana tiene un promedio mayor de edades' );
} else {
    if (promedio2 > promedio3) {
        document.write( 'El turno tarde tiene un promedio mayor de edades' );
    } else {
        document.write( 'El turno noche tiene un promedio mayor de edades' );
    }
}

}</script>
```

# EJERCICIOS ADICIONALES PROPUESTOS

— — —





**¡MUCHAS  
GRACIAS!**