

# Curso práctico de programación

- Javascript 2017 -



## Ejercicios de iniciación a la programación.

Después de conocer muchos cursos de programación, se puede encontrar un común denominador en muchos de ellos: faltan ejercicios realmente prácticos que ayuden a asimilar todos los conceptos. Por otro lado, también pueden encontrarse bastantes ejemplos y ejercicios de programación en internet, pero muchas veces no aportan conocimientos fundamentales.

Tras varios años impartiendo cursos de programación, he ido acumulando, puliendo y creando una serie de ejercicios que consiguen que cualquier persona, con interés y dedicando el tiempo oportuno, consiga realizar proyectos que realmente funcionan en la web.

Es conveniente tener conocimientos de HTML y de algún entorno de desarrollo web para depurar y ejecutar los siguientes ejercicios.

También es conveniente seguir algún curso "teórico" donde se expliquen con más profundidad los conceptos que se aplican en estos ejercicios.

Para más información de los conceptos de estos ejercicios, se enlazan los ejercicios con la información aportada por: [w3scholls](#) (esta en inglés)

Está prohibido cualquier tipo de distribución sin permiso del autor.

Está prohibida la reproducción de este curso para su utilización comercial.

Está permitido el uso en común en aulas físicas donde se enseñen asignaturas de programación.



## ÍNDICE

Iniciación .....	pág.3
Condicionales .....	pág.10
Funciones .....	pág.18
Formularios .....	pág. 23
Bucles For .....	pág. 29
Resumen .....	pág. 33
Avance Parte 2 .....	pág. 44



# INICIACIÓN A LA PROGRAMACIÓN

## Iniciación y variables

En esta lección se aprende a escribir un primer programa “*Hola mundo*” .

Esto significa que se maneja correctamente un editor de texto, en este caso **BRACKETS** <http://www.vozidea.com/editor-brackets-para-desarrollo-web> , para sacar por pantalla, en el navegador web, una pequeña frase.

El concepto de *Variable* también se asimila fácilmente a partir de ejemplos muy reveladores.

## Inicio con Javascript (ECMAScript6) : variables



### ES6 Inicio 1)

- Abrir Brackets ( o cualquier otro editor de HTML)
- Archivo Nuevo
- Copiar el código que empieza por `<!DOCTYPE html>` siguiente.
- Menú... Archivo... Guardar como: Ejercicio1.html
- Menú... Archivo... Vista previa dinámica

```
<!DOCTYPE html>
<html>
<head>
  <title>Ejercicio 1 </title>
</head>

<body>

<script>
  document.writeln("Hola Mundo!");
</script>

</body>
</html>
```

Si no aparece nada en el explorador... puede haber un fallo copiando el texto.

### Solución:

En Firefox y Chrome , pulsar CTRL + Mayúsculas + J , para abrir consola de errores. Fijarse en el último error que señala. Corregirlo y GUARDAR de nuevo el archivo Ej1

**ES6 Inicio 2) Crear archivo nuevo 'Ej2.htm' y copiar el siguiente código.****Una vez guardado, ejecutar el programa: Launch in Firefox (Chrome ...)**

```
<!DOCTYPE html>
<html>
<body>

<script>

  let a, b;
  a = 10; b = 5;
  resultado = a + b;

  alert (`La suma de a+b es ${resultado} ` );

</script>

</body>
</html>
```

A la variable **a** se le asigna el valor 10  
A la variable **b** se le asigna el valor 5  
La variable **resultado** contiene la **suma** de a y b

**ES6 Inicio 3) Crear archivo nuevo 'Ej3.htm' y copiar el siguiente código.****Una vez guardado, ejecutar el programa: Launch in Firefox (Chrome ...)**

```
<!DOCTYPE html>
<html>
<body>

<script>

  let num1=0;
  let num2=0;

  num1 = num1 + 1;
  num2 = num2 + 5;

  alert (`El primer numero ahora es ${num1} `);
  alert (`El segundo numero ahora es ${num2} ` );

</script>
</body>
</html>
```

Se añade 1 al valor que contiene **num1**  
Se añade 5 al valor que contiene **num2**

## Javascript (ES6): Condicionales.



Un programa consta de datos, lógica y presentación. La parte de lógica esta formada por condicionales, decisiones que toma el programa en base al diseño del programador.

Los condicionales simples son fáciles de entender y aplicar, pero anidar (ir sumando) varios de ellos necesita una estructura para optimizar el número de decisiones que toma el programa.



## ES6 Condicionales 1) ALGORITMO (NO programa) con introducción de 2 números.

Sacar por pantalla cual es el mayor.

Cambiar en un papel y con lápiz, el pseudocódigo donde aparece XXX

inicio

```
num1 ← 0: num2 ← 0
Introducir "Introduce primer número"; num1
Introducir "Introduce segundo número"; num2
```

```

Si (num1 > num2)
    Escribir "El mayor es el número ", XXX
sino
    Escriure "El mayor es el número ", XXX
Fin si
```

Fin

## ES6 Condicionales 2) Ejecutar el siguiente programa.

Introducción de 2 números y el programa indica cuál es el mayor.

```
<!DOCTYPE html>
<html><head><meta charset="utf-8"></head>
<script>

    let dato1,dato2,num1, num2, resultado;

    dato1 = window.prompt("Primer número?", "0");
    num1 = parseInt(dato1);

    dato2 = window.prompt("Segundo número?", "0");
    num2 = parseInt(dato2);

    if (num1 > num2)
    {
        resultado = " el primero";
    }
    else
    {
        resultado = " el segundo";
    }
    document.write(` <h3>El mayor es ${resultado} </h3> `);

</script>
</head>
</html>
```

*Si **num1** es mayor que **num2**, entonces resultado es "El primero"*  
**SINO** resultado es "El segundo"

Más información sobre condicionales: [http://www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp)

Realizar programa en el que el usuario introduce el número del mes (1 al 12) y el código responde si ese mes tiene 30 o 31 días.



**ES6 Condicionales 15)** Ejecutar el siguiente programa.

```
<!DOCTYPE html>
<html>
<head><meta charset="utf-8">
<script>

  let dia,mes,año;
  mes=prompt('Introduzca el número del mes:', '');
  mes=parseInt(mes);

  if (mes==1 || mes==3 || mes==5 || mes==7 || mes==8 || mes ==10 || mes==12)
  {
    document.write('Este mes tiene 31 dias. ');
  }
  else
  {
    if (mes==2)
      document.write('Este mes puede tener 30 dias. ');
    else
      document.write('Este mes tiene 30 dias. ');
  }

</script>

</head>
</html>
```

**Es6 Ejercicio 16)** Agregar código al programa que muestra el número de días de un mes, para que el usuario solo pueda introducir números del 1 al 12.

# Javascript (ES6) : Funciones

Los programas crecen, y cuando alcanzan cierto tamaño la única forma de estructurarlos es dividirlos en secciones. Ese es el objetivo de las funciones, tener el código ordenado y en módulos lo más independientes entre sí (preparando las futuras clases encapsuladas y autónomas).

Información sobre funciones: [http://www.w3schools.com/js/js\\_functions.asp](http://www.w3schools.com/js/js_functions.asp)

## Funciones 1)

Ejecutar el siguiente código. ¿En qué momento se “dispara” la función?

```
<!DOCTYPE html>
<html><head>
<meta content="text/html; charset=utf-8">

<script>

    function saludo(nombre)
    {
        document.write(`Hola ${nombre} `);
    }

</script>

</head>
<body>

<input type="button" onclick="saludo('Ana')" value="Pulsa aquí">

</body>
</html>
```

## Funciones 2)

Colocar un punto de interrupción en la primera línea 8 de código del ejercicio anterior (`document.write (`Hola ${nombre} `)`) y otro punto de interrupción en la línea 16 (`<input type`).

Empezar la depuración e ir ejecutando “paso a paso por procedimientos” (F10)

Ver el orden en que se ejecutan las instrucciones. Pulsar el botón “Llamada a función” para llegar al punto de interrupción de la función.

## Javascript (ES6) : Forms

Para introducir datos desde campos de texto, se necesita una marca `<form>` que delimita todas las posibles entradas de datos. Esta marca crea un objeto al cual podemos acceder a cada uno de sus elementos de entrada de datos.

**Forms1) Introducción datos en FORMS**

```
<!DOCTYPE html>
<html>
<head>
<title> Datos en Formularios </title>
<script>
  function saludo()
  {
    let nom_usuario = document.fdatos.entrada.value
    document.writeln(`Hola, bon dia  ${nom_usuario} `)
  }
</script>
</head>

<body>
  <form name="fdatos">
    Escribe tu nombre: <input type="text" size="10" name="entrada"/>
    <input type="button" value="Pulsa aquí" onclick="saludo()" />
  </form>
</body>
</html>
```

**Forms 2)** Realizar programa en el cual el usuario introduce un valor en un FORM y por pantalla sale como resultado su 10%.

## Javascript (ES6): BUCLES

Para repetir instrucciones un número determinado de veces, se utiliza **for**.

Más adelante también se verá como utilizarlo sobre valores de datos en arrays y sus alternativas más actuales.

**Información de los bucles for :** [http://www.w3schools.com/js/js\\_loop\\_for.asp](http://www.w3schools.com/js/js_loop_for.asp)

**ES6 Bucles 1) Realizar seguimiento de las variables i , cadena (puntos interrupción)**

```
<!DOCTYPE html>
<html><body>
<p id="central"></p>
<script>

    let cadena="",i;

    for (i=0;i<5;i++)
    {
        cadena = cadena + `El numero es  ${i} <br />`;
    }

    document.getElementById("central").innerHTML= cadena;
</script>
</body>
</html>
```

**ES6 Bucles 2) Crear programa para que salgan en pantalla los siguiente números:**

```
54321
4321
321
21
1
```

```
<!DOCTYPE html>
<html>
<head>
<script>
    for (x = 5; x > 0; x--)
    {
        for (j = x; j >= 1; j--)
            document.write(+j);

        document.write("</br>");
    }
</script>
</head></html>
```

## AVANCE : PARTE2 - Curso práctico de programación 2017 -

Adelanto de la Parte2 de los Ejercicios resueltos.

Los apartados que se estan preparando en la parte 2 son los siguientes:

- Clases y Objetos
- Arrays
- JSON

## EJERCICIO1 DE CLASES - 1) CommonJS, 2) Object Interface , 3)Clases ECMAScript 6

```
<!DOCTYPE html>
<html>
<head><meta charset="utf-8"/>

<title>Modules - CommonJS</title>
<!-- Teoria:
https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Classes -->

<script>

function ficha() {

    let alumno = {
        nombre: "Caterina",
        edad: 18
    };

    this.verNombre = function ()
    {
        return alumno.nombre;
    }

    this.verEdad = function ()
    {
        var resultado = alumno.edad;
        return resultado;
    }
}

function programa() {

    let obj = new ficha();

    document.writeln(`El nombre es ${obj.verNombre()} <br>`);
    document.writeln(`La edad es ${obj.verEdad()} <br>`);
}

</script>
</head>
<body>
<h3>Clase ficha </h3>
    <input type="button" value="Pulsa aquí" onclick="programa()" />
</body>
</html>
```

```
<!DOCTYPE html>
<html>
<head><meta charset="utf-8"/>
<title>ObjectInterface - Clases1</title>
<script>

let ficha = (function () {

    let alumno;
    this.alumno = "Marga";

    return {

        inicio ()
        {
            this.alumno = "Norat"
        },

        verNombre()
        {
            return this.alumno;
        },

    } // fin return

})(); //fin 'clase'

function programa() {

ficha.inicio();
document.writeln(`El nombre es ${ficha.verNombre()} <br>`);

//document.writeln(`La edad es ${ficha.verEdad()} <br>`);
}

</script>
</head>

<body>
<h3>Clase ficha </h3>
    <input type="button" value="Pulsa aquí" onclick="programa()" />
</body>
</html>
```

```
<!DOCTYPE html>
<html><head><meta charset="utf-8"/>
<title>ECMA6 - Clases1</title>

<!-- Teoria: https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Clases -->
<script>

class EntradaDatos {
  constructor () {
    let nombre;
    let contrasenia;
    this.nombre = "admin";
    this.contrasenia = "1234"
  }

  introduceNombre ()
  {
    let valorEntrada = document.formEntrada.txtInicial.value;

    if (this.compruebaNombre(valorEntrada))
      this.nombre = valorEntrada;
    else
      this.nombre = "No valido";
  }

  compruebaNombre (valor)
  {
    let flag = true;
    if (valor == null || valor == "") flag = false;
    return flag
  }

  verNombre()
  {
    return this.nombre;
  }
}

function programa() {
  let obj = new EntradaDatos();
  obj.introduceNombre ();
  document.writeln(`El nombre es ${obj.verNombre()} <br>`);
}
</script>
</head>

<body>
<h3>Clase Entrada Datos </h3>
  <form name="formEntrada">
    <input type="text" name="txtInicial" />
    <input type="button" value="Pulsa aquí" onclick="programa()" />
  </form>
</body>
</html>
```