

JavaScript

Ejercicios resueltos (2^a parte)

Programación web con ECMAScript 6



JavaScript

Ejercicios resueltos (2^a parte)

Programación web con ECMAScript 6



Ejercicios resueltos (2º parte)

Programación en Javascript (ECMAScript6)

Este libro de ejercicios es continuación de la parte1 , donde se inicia a programar y se dan los contenidos necesarios para poder realizar esta segunda parte.

Para más información de los conceptos de esta lección, puede visitarse [w3scholls](#) (esta en inglés)

Para cualquier aclaración, pueden ponerse en contacto con el autor, a través de [www.ibserveis.com](#)

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

ECMASCRIPT 6: PATTERNS

Object Interface - CommonJS - ES6 páginas 4

ARRAYS páginas 25

jQuery páginas 45

JSON páginas 67



ECMASCRIPT 6: PATTERNS

Clases 1A) Patrón ECMA6

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8" />
<title>ECMA6 - Clase Ficha</title>
<!-- https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Clases -->
<script>

class ficha {

    constructor() {
        this.alumno = {
            nombre: "Caterina",
            edad: 18
        };
    } // fin constructor

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

    verEdad () {
        let resultado = this.alumno.edad;
        return resultado;
    }
} //fin class ficha

function programa() {
    var 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>
```



Clases 1B) Patrón COMMON JS

```
<!DOCTYPE html>
<html>
<head><meta charset="utf-8"/>
<title>Modules - CommonJS</title>
<!-- https://medium.freecodecamp.com/javascript-modules-a-beginner-s-guide-783f7d7a5fcc#.dcganpttg -->
<script>

function ficha() {

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

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

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

function programa() {

    var 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()" />
</form>
</body>
</html>
```



Clases 1C) Patrón OBJECT INTERFACE

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

var ficha = (function () {

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

    return {

        verNombre() {
            return alumno.nombre;
        },
        verEdad() {
            var resultado = alumno.edad;
            return resultado;
        }
    }
}) // fin return

})(); //fin 'clase' ficha

function programa() {

    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>
```



Clases 2)

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

class Animal {
    constructor(name) {
        this.name = name;
    }

    decirNombre() {
        alert("Yo soy " + this.name);
    }
}

var objVivo1 = new Animal("Leon");
objVivo1.decirNombre();
var objVivo2 = new Animal("Jirafa");
objVivo2.decirNombre();

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



Clases 3) Uso funciones get , set para asignar y leer propiedad de la clases.
OJO! con la barra baja, es obligatoria para que funcione: _edad

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

class Animal{

    constructor(nombre) {
        this.nombre = nombre;
        this._edad = 0;
    }

    get edad() {
        return this._edad;
    }

    set edad(value) {
        if (value < 0) {
            alert("Eso no es posible");
        }
        this._edad = value;
    }

    verNombre() {
        console.log("Yo soy " + this.nombre);
    }
}

var objVivo = new Animal("Hormiga");
objVivo.verNombre();
objVivo.edad = 5;
console.log(objVivo.edad);

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

Clases 4)

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

class Calculadora {
    constructor() {
        this._resultado = 0;
        this._num1 =0; this._num2=0;
    }

    set resultado(oper) {
        switch (oper) {
            case 1:
                this._resultado = this._num1 + this._num2;
                break;
            case 2:
                this._resultado = this._num1 - this._num2;
                break;
        }
    }
    get resultado() {
        var cadena = "El resultado es: " + this._resultado;
        document.getElementById("salida").innerHTML = cadena;
    }

    introducirDatos() {
        let valor1, valor2;
        valor1 = parseInt(prompt("Introduce primer número ?", "0"));
        valor2 = parseInt(prompt("Introduce segundo número ?", "0"));
        if (isNaN(valor1 && valor2)==false){
            if (valor1>0 && valor1<100) this._num1=valor1;
            else prompt("Primer número no esta entre 0 y 100", "0");
            if (valor2>0 && valor2<100) this._num2=valor2;
            else prompt("Segundo número no esta entre 0 y 100", "0");
        }
        else prompt("No son números", "0");
    }
}

function programa(operacion) {
    var objCalcula = new Calculadora();
    objCalcula.introducirDatos();
    objCalcula.resultado=operacion;
    objCalcula.resultado;
}

</script>
</head>
<body>
    <div id="salida"></div>
    <input type="button" value="Suma" onclick="programa(1)" />
    <input type="button" value="Resta" onclick="programa(2)" />
</body>
</html>
```

Clases 4B)

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

var objCalculadora = (function (inicio) {
    // variables privadas (solo accesibles desde el mismo objeto)
    var resultado = 0;
    var num1 = 0; var num2=0;

    // función privada
    salirpantalla = function () {
        var cadena = "El resultado es: " + resultado;
        document.getElementById("salida").innerHTML = cadena;
    };

    return {
        // funciones públicas
        introduce: function (dato1,dato2) {
            num1 = parseInt(dato1);
            num2 = parseInt(dato2);
        },
        suma: function () {
            resultado = num1 + num2;
            salirpantalla ();
        }
    }; //fin return
})(); // fin objCalcula

function calcula (var1,var2){
    objCalculadora.introduce(var1,var2);
    var final = objCalculadora.suma();
}

</script>
</head>

<body>
<div id="contenedor">
<form name="formDatos">
    <input type="text" name="txtValor1" value="0" />
    + <input type="text" name="txtValor2" value="0"/>
    <input type="button" value="Suma" onclick="calcula(txtValor1.value, txtValor2.value)" />
</form>
<div id="salida"></div>
</div>

</body>
</html>
```



Clases 5) Calcula AREA con ES6

```
<!DOCTYPE html>
<html>
<head><meta charset="utf-8"/>
<title>Calculo area y perímetro de un triangulo</title>
<script>

class Rectangulo {
    constructor(){
        this.resultado=0;
    }
    calculaArea(A,B){
        this.resultado = A*B;
    }
    calculaPerimetro(A,B){
        this.resultado = 2*(A+B);
    }

    resultadofinal(){
        return this.resultado;
    }
}

function programa() {

    var num1 = parseFloat(document.fdatos.entradaA.value);
    var num2 = parseFloat(document.fdatos.entradaB.value);

    var obj = new Rectangulo();
    obj.calculaArea (num1,num2);
    document.writeln(`El area es ${obj.resultadofinal()} <br`');

    obj.calculaPerimetro (num1,num2);
    document.writeln(`El perimetro es ${obj.resultadofinal()} `);
}

</script>
</head>
<body>
    <h2>Calculo Area y perimetro rectangulo</h2>
    <form name="fdatos">
        Escribe Base: <input type="text" size="2"  name="entradaA"/>
        Escribe Altura: <input type="text" size="2"  name="entradaB"/>
        <input type="button" value="Pulsa aquí" onclick="programa()"  />
    </form>

    <div id="Zona1"></div>
    <div id="Zona2"></div>
    <div id="Zona3"></div>
</body>
</html>
```

Clases 5B) Calcula AREA con Object Interface

```
<!DOCTYPE html>
<html><head><meta charset="utf-8"/>
<title>Modules - Object Interface</title>

<script>

var rectangulo = (function () {

    // Variable privada interna
    var numero=0;

    return {
        perimetro: function(altura, base) {
            var calculoperimetro =  2*(altura+base);
            return 'El perimetro es ' + calculoperimetro;
        },
        area: function(altura,base) {
            var calculoarea = altura*base;
            numero++;
            return 'El area es ' + calculoarea;
        },
        numeroveces: function() {
            return numero;
        }
    }
} // fin return
})(); //fin function

function programa() {

    var num1 = parseFloat(document.fdatos.entradaA.value);
    var num2 = parseFloat(document.fdatos.entradaB.value);

    document.writeln("El area es "+ rectangulo.area(num1,num2) + "<br>");
    document.writeln("El perimetro es "+ rectangulo.perimetro(num1,num2) + "<br>");
    document.writeln("número veces "+ rectangulo.numeroveces());
}

</script>
</head>

<body>
<h2>Calculo Area y perimetro rectangulo</h2>
<form name="fdatos">
    Escribe Base: <input type="text" name="entradaA"/>
    Escribe Altura: <input type="text" name="entradaB"/>
    <input type="button" value="Calcula" onclick="programa()" />
</form>
</body>
</html>
```



Clases 6) Realizar programa donde el usuario puede obtener area y perímetro de triangulos, rectángulos y cuadrados.

Clases 7) Realizar programa donde introduciendo la edad de un niño y su altura, el programa indica la estatura aproximada que tendrá de adulto.

Clases 8) Un Cuadrado en movimiento - Version ES6

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

<style type="text/css">
#contenedor{
    position: relative;
    background-color: #C0C0C0;
}
#cuadrado1{
    background-color: #00FF00;
    width: 100px; height: 100px;
    position: relative;
    left: 100px; top: 100px;
}
</style>
<script>

class Cuadrado {
    constructor() {
        this.posX = Cuadrado.incrementa;
        console.log(this.posX);
    }

    static get incrementa() {
        Cuadrado.numero = (Cuadrado.numero || 100) + 20;
        return Cuadrado.numero;
    }

    establecerPosX1(name) {
        document.getElementById(name).style.left = this.posX + "px";
    }

    salirPantalla(name) {
        let cadena = "Posición:" + this.posX
        document.getElementById(name).innerHTML = cadena;
    }
}

function incrementar (nom){
    objCuadrado = new Cuadrado();
    objCuadrado.establecerPosX1(nom);
    objCuadrado.salirPantalla(nom);
}

</script>
</head>
<body>
<div id="contenedor">
    <div id="cuadrado1" onclick="incrementar('cuadrado1')"></div>
</div>
</body>
</html>
```

Clases 8B) Un Cuadrado en movimiento - Version Object Interface

```
<!DOCTYPE html>
<html><head><meta charset="utf-8"/>
<style type="text/css">
#contenedor{
    position: relative;
    background-color: #C0C0C0;
}
#cuadrado1{
    background-color: #00FF00;
    width: 100px; height: 100px;
    position: relative;
    left: 100px; top: 100px;
}
</style>
<script>
var objCuadrado = (function () {
    var posX1 = 100;

    return {

        incrementarposicion() {
            return posX1 = posX1 + 20;
        },
        getposicionX(name){
            return (document.getElementById(name).style.left);
        },
        setposicionX(name) {
            document.getElementById(name).style.left = posX1 + "px";
        },
        salirpantalla(name) {
            let cadena = "Posición:" + posX1
            document.getElementById(name).innerHTML = cadena;
        }
    };
})();

function incrementar (nom){
    objCuadrado.incrementarposicion();
    objCuadrado.setposicionX(nom);
    objCuadrado.salirpantalla(nom);
}

</script>
</head>

<body>
<div id="contenedor">
    <div id="cuadrado1" onclick="incrementar('cuadrado1')"></div>
</div>
</body>
</html>
```

Clases 8+) Realizar programa donde usuario maneja un gráfico de comecocos por toda la pantalla , derecha-izquierda y arriba-abajo.



Clases 9) Contador : Object Interface

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

let objContador = (function (inicio) {
    let contador = 0;

    return {
        incrementarcontador: function () {
            return contador++;
        },
        salirpantalla: function () {
            let cadena = "El contador es: " + contador;
            document.getElementById("salida").innerHTML = cadena;
        },
        resetcontador: function () {
            console.log( "contador antes de reset: " + contador );
            contador = 0;
        }
    };
})();

function incrementar (){
objContador.incrementarcontador();
objContador.salirpantalla();
}

</script>
</head>

<body>
<div id="salida"></div>

<input type="button" value="Incrementa contador" onclick="incrementar()" >

</body>
</head>
</html>
```



Clases 9B) Contador ES6

```
<!DOCTYPE html>
<html><head><meta charset="utf-8"/>
<script>
class Contador {
    constructor() {
        this.total = Contador.inicio;
        console.log(this.valor);
    }
    incrementarcontador() {
        Contador.incrementa;
    }
    get valor() {
        return this.total;
    }

    static get incrementa() {
        Contador.numero = (Contador.numero || 0) + 1;
        return Contador.numero;
    }
    static get inicio() {
        Contador.numero = (Contador.numero || 0);
        return Contador.numero;
    }
    salirpantalla() {
        let cadena = "El contador es: " + this.valor;
        document.getElementById("salida").innerHTML = cadena;
    }
    resetcontador() {
        console.log( "contador antes de reset: " + this.valor );
        Contador.numero = 0;
    }
}

function programa(menu) {
var objContador = new Contador;
if (menu==2) {
    objContador.incrementarcontador();
    objContador.salirpantalla();
}
if (menu==3) {
    objContador.resetcontador();
}
}
</script>
</head>
<body>
<div id="salida"></div>
<input type="button" value="Incrementa contador" onclick="programa(2)" >
<input type="button" value="Resetea contador" onclick="programa(3)" >
</body>
</html>
```



Clases 10) Objeto para entrada de datos

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

<title>ECMA6 - Clases</title>
<!-- https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Clases -->
<script>

class EntradaDatos {
    constructor (){
        this.nombre = "admin";
        this.contrasenya = "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
        // if (!(valor == null || valor == "")) return 1
    }
    verNombre(){
        return this.nombre;
    }
}

function programa() {

    var 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>
```

Clases 11) Objeto Display

```
<!DOCTYPE html>
<html><head><meta charset="utf-8"/>
<title>ECMA6 - V2_miDisplay</title>
<style type="text/css">
#contenedor{
    position: relative;
    background-color: #C0C0C0;
}

#display{
    background-color: #00FF00;
    width: 100px; height: 100px;
    position: relative;
    left: 200px
}
</style>
<script>

class miDisplay {

    constructor (inicio){
        this.miObj = document.getElementById('display');
        this.miObjestilo = document.getElementById('display').style;
        this.valor= inicio;
    }

    posiciona (x, y){
        this.miObjestilo.left = x + "px";
        this.miObjestilo.top = y + "px";
    }

    asignaValor (result) {
        this.valor = result;
    }

    muestraValor () {
        let tam = document.formEntrada.txtTamano.value;
        this.miObjestilo.fontSize = tam + "px";
        this.miObj.innerHTML = this.valor;
    }
}

function programa(num) {
    let obj = new miDisplay(0);
    obj.posiciona (500,200);
    obj.asignaValor(num);
    obj.muestraValor();
}

</script>
</head>
```

```
<body>
<h3> Display </h3>
<div id="contenedor">
    <form name="formEntrada">
        Tamaño<input type="text" name="txtTamano" value="36" />
        Valor <input type="text" name="txtValor" value="0"/>
        <input type="button" value="Pulsa"
    onclick="programa(formEntrada.txtValor.value)" /><br>
    </form>

    <div id="display"></div>
</div>

</body>
</html>
```

Clases 11) Realizar programa conversor CELSIUS-FAHRENHEIT y viceversa , UTILIZANDO el objeto DISPLAY del ejercicio anterior para sacar resultado en pantalla.

Clases 12) Ejecutar el siguiente programa con PATTERN COMMON JS:

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

<title>Modules - CommonJS</title>
<!-- Teoria: https://medium.freecodecamp.com/javascript-modules-a-beginner-s-guide-783f7d7a5fcc#.dcganpttg -->

<script type="text/javascript">

function rectangulo() {

    var resultado;

    this.calculaArea = function (A,B)
    {
        resultado = A*B;
    }

    this.calculaPerimetro = function (A,B)
    {
        resultado = 2*(A+B);
    }

    this.resultadofinal = function ()
    {
        return resultado;
    }
}

// module.exports = rectangulo;

function programa() {

    //var rectangulo = require('rectangulo');

    var obj = new rectangulo();

    var num1 = parseFloat(document.fdatos.entradaA.value);
    var num2 = parseFloat(document.fdatos.entradaB.value);

    obj.calculaArea (num1,num2);
    document.writeln("El area es "+ obj.resultadofinal() + "<br>");

    obj.calculaPerimetro (num1,num2);
    document.writeln("El perimetro es "+ obj.resultadofinal());

}

</script>
</head>
```

Clases 13) Ejecuta el siguiente programa

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

<style type="text/css">
#contenedor
{
    position: relative;
    background-color: #C0C0C0;
}

#globo
{
    background-color: #00AAFF;
    width: 100px; height: 100px;
    position: relative;
    left: 500px; top: 500px;
}
</style>

<script>
let objGlobo = (function () {

    var posY; var tam;

    return {
        subir(obj) {

            if (posY>50) {
                posY = posY - 50;
                tam = tam + 20;
            }
            else{
                alert("explota");
                //obj.style.style.img = "explosion.jpg"
            }
        },
        obtenerValores(obj) {
            let valor = parseInt(obj.style.top);
            if (isNaN(valor)) valor = 400; // valor inicial
            posY = valor;

            let valor2 = parseInt(obj.style.width);
            if (isNaN(valor)) valor2 = 50; // valor inicial
            tam = valor2;
        },
        colocar(obj) {
            obj.style.top = posY + "px";
            obj.style.width = tam + "px";
        },
    };
});

```

```
salirTam(obj) {
    document.getElementById("salida").innerHTML = "Tamaño:" + tam;
}

};

})();

function programa (nom){
objGlobo.obtenerValores(nom);
objGlobo.subir(nom);
objGlobo.colocar(nom);
objGlobo.salirTam(nom);

}
</script>
</head>

<body>
<div id="contenedor">
    <div id="globo"></div>
</div>

<div id="salida">
<input type="button" value="Sube"
onclick="programa(document.getElementById('globo'))"  />
</div>
</body>
</html>
```

Clases 14) Crear programa del Juego "Mayor o menor", donde el usuario intenta adivinar el número elegido al azar por el ordenador y si no lo consigue un globo explota por elevarse demasiado.

Clases 15) Crear programa TAMAGOTCHI , con todas las funciones necesarias para ver crecer y desarrollarse a tu mascota virtual.

Javascript ES6: ARRAYS

Arrays 1) variable tipo array y su contenido en pantalla.

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

var dias = ["lunes", "martes", "miércoles", "jueves", "viernes"];

document.write("<h2>Días de la semana</h2>");

for (var i = 0; i < dias.length; i++)
    document.write(` ${dias[i]} <br>`);

</script>

</head>
</html>
```

Arrays 2) Recorrido inverso: Modificar siguiente programa para que muestre los días, jueves y martes.

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

var dias = ["lunes", "martes", "miércoles", "jueves", "viernes"];

document.write("<h2>Días de la semana</h2>");

final = (dias.length -1);

for (var i = final; i >1 ; i=i-2)
    document.write(` ${dias[i]} <br>`);

</script>

</head>
</html>
```

Arrays 3) Uso del índice

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

var dias = ["lunes", "martes", "miércoles", "jueves", "viernes", "sabado",
"domingo"];

document.write("<h2>Días de la semana</h2>");

dato = window.prompt("Introduce número del 0 al 6", "0");
num = parseInt(dato);

document.write(` <br> El dia introducido es: ${dias[num]} `);

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

Utilizando Node.js

Arrays 4) variable array y muestra de su contenido

```
var dias = ["lunes", "martes", "miércoles", "jueves", "viernes", "sabado", "domingo"];

function verCada(array) {
  for (var i = 0; i < array.length; i++)
    console.log(array[i]);
}
verCada(dias);
```

Arrays 5) Para cada elemento, ejecuta una función determinada.

```
var dias = ["lunes", "martes", "miércoles", "jueves", "viernes", "sabado", "domingo"];
```

```
function paraCada(array, action) {  
    for (var i = 0; i < array.length; i++)  
        action(array[i]);  
}  
paraCada(dias,console.log);
```

Arrays 6) Para cada elemento, ejecuta una función determinada.

```
var numeros =[100,200,300,400];
```

```
function paraCada(array, action) {  
    for (var i = 0; i < array.length; i++)  
        action(array[i]);  
}  
paraCada(numeros,alert);
```

Arrays 7) Para cada elemento, ejecuta una función determinada.

```
var dias = ["lunes", "martes", "miércoles", "jueves", "viernes", "sabado", "domingo"];  
dias.forEach(console.log);
```

Arrays 8) Uso de forEach

```
<!DOCTYPE html>  
<html>  
<head><meta charset="utf-8">  
<script>  
    /* forEach utiliza elementos de la Array que tengan un índice,  
     de esta manera evitamos problemas con indices vacíos.*/  
  
    var nuevosCoches = ["Seat", "Renault", "Alfa Romeo", "Peugeot"];  
  
    document.write("<h2>Marcas de Coches</h2>");  
  
    nuevosCoches.forEach(function(item, index){  
        document.write(` ${item} <br>`);  
    });  
  
</script>  
</head>  
<body></body>  
</html>
```

Arrays 8) Selección de valores en un array (mayores o menores) con PATTERNS

```
<!DOCTYPE html>
<html><head><meta charset="utf-8" />
<title>ECMA6 - Arrays1</title>
<script>
class ficha {
    constructor() {
        this.arrayNumeros = [10,20,30,40,50,60,70,80,90]
    }
    mostrarMayores50() {
        let texto = "";
        let datos = this.arrayNumeros;
        for( let i = 0; i < datos.length; i++) {
            if (datos[i]>50) texto += `<p> dato Mayor50: ${datos[i]} `;
        }
        document.getElementById("salidaDatos").innerHTML = texto ;
    }
    arrayMenoresX(cota) {
        let arrayMenores = [];
        let datos = this.arrayNumeros;
        for( let i = 0; i < datos.length; i++) {
            if (datos[i]<cota) arrayMenores[i] = datos[i];
        }
        document.getElementById("salidaDatos").innerHTML = arrayMenores ;
    }
}

function programa(valor) {
    var obj = new ficha();
    if (valor==1) obj.mostrarMayores50();
    else obj.arrayMenoresX(valor);
}

</script>
</head>
<body>
    <h3>Menores o Mayores </h3>
    <input type="button" value="Ver Mayores 50" onclick="programa(1)" /><br>
    <form>Ver menores que:<input type="text" size="10" name="txtMenor"/>
    <input type="button" value="Menores" onclick="programa(txtMenor.value)" />
</form>
    <div id="salidaDatos"></div>
</body>
</html>
```

2) Version 2 : foreach()

```
<title>ECMA6 - Arrays2</title>
<script>
class ficha {
    constructor() {
        this.arrayNumeros = [10,20,30,40,50,60,70,80,90]
    }

    arrayMenoresX() {
        let arrayMenores = [];
        let cota = parseInt(document.getElementById("txtMenor").value);
        let datos = this.arrayNumeros;
        for( let i = 0; i < datos.length; i++) {
            if (datos[i]<cota) arrayMenores[i] = datos[i];
        }
        document.getElementById("salidaDatos").innerHTML = arrayMenores ;
    }

    mostrarMayorX(cota) {
        var texto = "";
        let datos = this.arrayNumeros;
        datos.forEach(function(item, index){
            if (item>cota) texto += '<p> dato Mayor50: ${item}</p>';
        });
        document.getElementById("salidaDatos").innerHTML = texto ;
    }
}

function programa(valor) {
    var obj = new ficha();
    if (valor==2) obj.arrayMenoresX();
    if (valor==3) obj.mostrarMayorX(50)
}
</script>
</head>
<body>
<h3>Menores o Mayores </h3>
<form>Ver menores que:<input type="text" size="10" id="txtMenor"/>
<input type="button" value="Menores" onclick="programa(2)" /></form>
<input type="button" value="Mayores50" onclick="programa(3)" />
<div id="salidaDatos"></div>
</body>
</html>
```

3) Version 3 (node.js)

```
//Definición del objeto PattNum
var PattNum = function(){
    this.numeros = [10, 20, 30, 40, 50, 60, 70, 80, 90];
}

//Método "mayores" de PattNum
PattNum.prototype.mayores = function(){
    var mayores = [];
    for (var i in this.numeros){
        if (this.numeros[i] > 50){
            mayores.push(this.numeros[i]);
        }
    }
    return mayores;
}

//Método "menores que" de PattNum
PattNum.prototype.menoresQue = function(num_lim){
    var menores = [];
    for (var i in this.numeros){
        if (this.numeros[i] < num_lim){
            menores.push(this.numeros[i]);
        }
    }
    return menores;
}

//Definiciones de sistema //Entrada de datos
var stdin = process.stdin;
//Instancia de PattNum => PatternTest
var PatternTest = new PattNum();
//Menú
console.log('Programa de números mayores y menores.\n');
console.log('Parámetros:');
console.log('1 - Muestra números mayores que 50');
console.log('2 - Muestra números menores que un número elegido');
console.log('0 - Termina el programa\n');
stdin.write('Opción: ')
```

.... sigue siguiente página.

```
//Input del usuario
stdin.on('data', function(data){
  if (data == 1){
    console.log(`Mayores de 50\n${PatternTest.mayores()}\n`);
    stdin.write('Opción: ');
  }
  else if (data == 2){
    stdin.write('Introduzca un número: ');
    stdin.once('data', function(user_num){
      stdin.write(`Menores que ${user_num}\n${PatternTest.menoresQue(user_num)}\n\n`);
      stdin.write('Opción: ');
    });
  }
  else if (data == 0){
    console.log('Programa terminado');
    process.exit();
  }
});
```

Arrays 9) Uso de información relacionada, en dos arrays diferentes.

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

<script>
var alum = ["Andres", "Angela", "Bernat", "Carol", "Maria", "Joan"];
var notas = [];

//Bucle que añade seis elementos a "notas", todos con el valor: 5.
for (let i = 0; i < 6; i++) {
    notas.push(5);
}

document.write(`<h2>Notas de la asignatura</h2>`);

numero = window.prompt(`Introduzca el número del alumno (0-5)`, `0`);
numero = parseInt(numero);

document.write(`<br>Nota de ${alum[numero]} inicial ${notas[numero]}`);

datoNota = window.prompt(`Introduce su nueva nota`, `0`);
valorNota = parseInt(datoNota);
//Elemento de "notas" cuyo índice es "lista" toma el valor "valorNota".
notas[numero] = valorNota;

document.write(`<br><h2>Notas finales:</h2>`);

//Bucle donde se imprime el número del alumno, su nombre y su nota.
for (let i = 0; i < alum.length; i++) {
    document.write(`alum num: ${i} nombre:${alum[i]} nota:${notas[i]} <br>`);
}
</script>
</head>
<body></body>
</html>
```

Arrays 10) Propiedades disponibles: PUSH

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

    var dias = new Array("lunes", "martes", "miércoles", "jueves", "viernes");
    dias.push("sabado");
    dias.push(100);
    document.writeln(dias);

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

Arrays 11) Funciones para mostrar y modificar contenido de Arrays.

```

<!DOCTYPE html>
<html>
<head><meta charset="utf-8">
<script>
    //Variables Globales
    var alum = ["Ana", "Bernat", "Carol", "David", "Elena", "Francesc"];
    var notas = [];
    for (let i = 0; i < 6; i++){
        notas.push(5);
    }

    function mostrar()
    {
        document.write(`<h2>Notas actuales: </h2>`);
        for (let i = 0; i < 5; i++){
            document.write(` ${i} - ${alum[i]}: ${notas[i]} <br>`);
        }
    }

    function modificar(numAlumno){
        nuevaNota = window.prompt(`Introduzca su nueva nota`, `0`);
        nuevaNota = parseInt(nuevaNota);
        notas[numAlumno] = nuevaNota;
    }

</script>
</head>

<body>
<script>
    document.write(`<h3>Inicio del Programa Notas</h3>`);
    mostrar();

    lista = window.prompt(`Introduzca el número del alumno (0-5)`, `0`);
    lista = parseInt(lista);
    modificar(lista);
    mostrar();

    document.write(`<h3>Fin del Programa Notas</h3>`);

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

```

Arrays 12) Crear programa con dos Arrays, el primero llamado DatosIniciales con 10 números entre el 0 y el 50 elegidos al azar. El segundo Array se llama DatosFinales, y se declara con 10 posibles valores.

A través de una función, se rellena el Array DatosFinales con los números mayores de 25 del Array DatosIniciales.

Arrays 13) Realizar programa en base al código que se da a continuación:

- a) Añadir informaciones: hora salida y hora llegada al array de datos
- b) Añadir datos hasta llegar a tener al menos 10 “ vuelos”
- c) Completar programa, con las funciones necesarias, para que el usuario vea en pantalla un menú operativo con las siguientes opciones (botones de acción):
 - 1) Ver todos los vuelos
 - 2) Ver vuelos de un destino determinado.
 - 3) Ver vuelos que llegan más tarde que una hora determinada (números enteros)

```
<!DOCTYPE html>
<html>
<head><title>JSON Aviones</title></head>
<body>
    <div id="salidaDatos"></div>
</body>

<!-- El código se ejecuta cuando ya tenemos disponible la <div id=salidaDatos
      Otra alternativa es disparar el código con botones -->

<script type="text/javascript" >
var arrayVuelos = [
{
"origen":"Palma",
"destino": "Oslo",
"url": "http://www.ibserveis.com/vuelo1.html"
},
{
"origen":"Lugo",
"destino": "Madrid",
"url": "http://www.ibserveis.com/vuelo2.html"
},
{
"origen":"Barcelona",
"destino": "Paris",
"url": "http://www.ibserveis.com/vuelo3.html"
},
{
"origen":"Palma",
"destino": "Madrid",
"url": "http://www.ibserveis.com/vuelo4.html"
},
{
"origen":"Palma",
"destino": "Paris",
"url": "http://www.ibserveis.com/vuelo5.html"
}
]
```

... sigue en página siguiente

```
function mostrarArray(datos) {  
    var texto = "";  
    for( let i = 0; i < datos.length; i++) {  
        texto += `<p> origen: ${datos[i].origen} <a href= ${datos[i].url}>  
                destino: ${datos[i].destino}</a></p><br>`;  
    }  
    document.getElementById("salidaDatos").innerHTML = texto ;  
}  
  
mostrarArray(arrayVuelos);  
  
</script>  
</html>
```

Arrays 14) En base al siguiente Archivo BOOKS.JSON

```
[  
  {  
    "id" : "978-0641723445",  
    "cat" : ["book","hardcover"],  
    "name" : "The Lightning Thief",  
    "author" : "Rick Riordan",  
    "series_t" : "Percy Jackson and the Olympians",  
    "genre_s" : "fantasy",  
    "inStock" : true,  
    "price" : 12,  
    "pages_i" : 384  
  },  
  {  
    "id" : "978-1857995879",  
    "cat" : ["book","paperback"],  
    "name" : "Sophie's World : The Greek Philosophers",  
    "author" : "Jostein Gaarder",  
    "genre_s" : "fantasy",  
    "inStock" : true,  
    "price" : 3,  
    "pages_i" : 64  
  },  
  {  
    "id" : "978-1933988177",  
    "cat" : ["book","paperback"],  
    "name" : "Lucene in Action, Second Edition",  
    "author" : "Michael McCandless",  
    "genre_s" : "IT",  
    "inStock" : true,  
    "price" : 30,  
    "pages_i" : 475  
  }  
]
```

- a) Añadir 3 elementos más al archivo de datos.
- b) Crear programa que muestra todos los libros a petición del usuario.
- c) Crear otra función en el programa que muestra los libros de precio mayor que 20€
- d) Añadir función al programa donde el usuario elige el autor y ve los libros correspondientes.



Arrays 15) En base al siguiente código: realizar MENÚ, con botones de acción, para que el usuario pueda decidir cuando quiere ver el listado de alumnos y cuando quiere modificar sus notas.

b) Añadir función de "Añadir alumno" y su botón correspondiente en el menú.

```
<!DOCTYPE html>
<html><head><meta charset="utf-8">
<script>
    var alumnos = [
        {
            "nombre": "Marcos",
            "nota": 5
        },
        {
            "nombre": "Isabel",
            "nota": 7
        },
        {
            "nombre": "Paula",
            "nota": 9
        }
    ]
    function Basis(){
        while (confirm(`Aceptar para modificar la nota de alumno`)){
            Modificar();
        }
    }

    function Intro(){
        num = window.prompt(`Introduzca el número del alumno`, `0`);
        num = parseInt(num);
        nota = window.prompt(`Introduzca la nueva nota`, `0`);
        nota = parseInt(nota);
    }

    function Modificar(){
        Intro();
        alumnos[num].nota = nota;
    }

    function Ver(){
        //Muestra el nombre y la nota para cada objeto de la array
        alumnos.forEach(function (item, index, array) {
            document.write(`Nombre: ${array[index].nombre} - Nota: ${array[index].nota}<br>`);
        });
    }
</script>
</head>

<body>
    <script>
        document.write(`<h3>Programa Notas</h3><hr>`);
        Basis();
        Ver();
    </script>
</body>
</html>
```



Arrays 16) En base al siguiente código, realizar programa para gestionar notas de clase dese un menú con botones en pantalla.

```
<!DOCTYPE html>
<html><head><meta charset="utf-8">
<script>
var alumnos = [];

function Inicio(){
var n_alum = window.prompt(`¿Cuántos alumnos desea dar de alta?`, `0`);
for (var i = 0; i < n_alum; i++){
    //Rellena el array con espacios y les asigna 2 propiedades, "nombre" y "nota".
    alumnos.push({}); 
    nom = window.prompt(`Introduzca el nombre del alumno (nº ${i})`, ` `);
    alumnos[i].nombre = nom;
    alumnos[i].nota = " ";
}
}

function Basis(){
while (confirm(`Desea modificar la nota de algún alumno?`)){
    Modificar();
}
}

function Intro(){
num = window.prompt(`Introduzca el número del alumno`, `0`);
num = parseInt(num);
nota = window.prompt(`Introduzca la nueva nota`, `0`);
nota = parseInt(nota);
}

function Modificar(){
Intro();
alumnos[num].nota = nota;
}

function Ver(){
//Muestra el nombre y la nota para cada objeto de la array
alumnos.forEach(function (item, index, array) {
document.write(`Nombre: ${array[index].nombre} | Nota: ${array[index].nota}<hr>`);
});
}

</script>
</head>
<body>
<script>
document.write(`<h3>Programa Notas</h3><hr>`);
Inicio();
Basis();
Ver();
</script>
</body>
</html>
```

Arrays 17) Clase Strings (tipo particular de Array de caracteres)

```
<!DOCTYPE html>
<html><head>
<script type="text/javascript">

    var str = "abcdefghijklmnñopqrstuvwxyz";
    document.write("Número de caracteres de " + str + ": " + str.length);

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

Arrays 18) Strings - Substring

```
<!DOCTYPE html>
<html><head>
<script type="text/javascript">
    var str = "abcdefghijklmnñopqrstuvwxyz";

    document.write(str.substring(8));
    document.write("<br>");
    document.write(str.substring(2, 5));

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

Arrays 19) Formulario con array de radio botones.

```
<html>
<head><meta charset="utf-8">
<script>
    function GetData(radioName){
        var radi = document.getElementsByName(radioName);
        for (var i = 0; i < radi.length; i++){
            if (radi[i].checked) return radi[i].value;
        }
    }
    function Discount(){
        var desc = 0, preu = 100, rest;
        let curs = GetData('curs');
        let menj = GetData('menjador');
        let germ = GetData('germans');
        if (curs == 'FP'){
            if (germ == 'true') desc = 0.1;
            else if (menj == 'true') desc = 0.05;
        }
        else {
            if (menj == 'true') desc = 0.08;
            else desc = 0.06;
        }
        rest = preu * desc;
        preu = preu - rest;
        alert("El preu final és " + preu);
    }
</script>
</head>
<body>
    <form>
        <h4>Curs</h4>
        <input type="radio" name="curs" value="FP">FP<br>
        <input type="radio" name="curs" value="BATX">BATX<br>
        <h4>Menjadør</h4>
        <input type="radio" name="menjador" value="true">Sí<br>
        <input type="radio" name="menjador" value="false">No<br>
        <h4>Germans</h4>
        <input type="radio" name="germans" value="true">Sí<br>
        <input type="radio" name="germans" value="false">No<br>
        <input type="button" onclick="Discount()">
    </form>
</body>
</html>
```

Arrays 20) Uso de prototype para añadir funcionalidad al objeto Array.

```

<!DOCTYPE html>
<html>
<head><meta charset="utf-8">
<script>
var Alumnos = ["Joan", "Ana", "Juana", "Mario"];

//Creamos una nueva propiedad para el objeto Array
Array.prototype.pasarMayusculas = function(){
    for (let i = 0; i < this.length; i++){
        this[i] = this[i].toUpperCase();
    }
}

function Transformar(){
//Utilizamos la nueva propiedad "pasarMayusculas"
Alumnos.pasarMayusculas();
}

function Mostrar(){
var objVer = document.getElementById("iniciar");
objVer.innerHTML = `${Alumnos}`;
}
</script>
</head>

<body>
<p id="iniciar">Primero puedes ver, luego pasar a mayúsculas y volver a ver</p>
<br>
<button onclick="Mostrar()">Ver Alumnos</button>
<button onclick="Transformar()">Pasar a mayúsculas</button>
</body>
</html>

```

Arrays 21) Realizar programa donde el usuario introduce una palabra y el ordenador responde el número total de vocales contenidas. Realizar código con una FUNCIÓN ,que recibe como parámetro esa palabra.**Arrays 22)**

Realizar programa donde el usuario introduce una palabra. El programa consta, de al menos, DOS FUNCIONES: la función f_vocales(variable) recibe la palabra y saca por pantalla el número total de vocales de esa palabra.
 La segunda función se llama f_total(variable), esta función recibe la palabra y saca por pantalla el número total de letras.

Arrays 23) Función “rollover” (añadir imágenes prova1.gif , prova2.gif en la misma carpeta)

```
<!DOCTYPE html>
<html><head>
<script type="text/javascript">
    imagen01 = new Image()
    imagen01.src = "prueba1.jpg"
    imagen02 = new Image()
    imagen02.src = "prueba2.jpg"

    function rollover(nombreImagen, nova) {
        document.images[nombreImagen].src = nova.src
    }
</script>
</head>

<body>
<a href="#" onmouseover="rollover('flecha', imagen02)"
onmouseout="rollover('flecha', imagen01)">
 </a>
</body>
</html>
```

Arrays 24) Realizar juego SIMPLE, tipo "Ahorcado": el usuario introduce letras tratando de adivinar una palabra que el ordenador tiene almacenada.



Arrays 25) Crear programa con dos Arrays, utilizando '.map' conseguimos un segundo array con los valores duplicados del primero.

```
<!DOCTYPE html>
<html>
<head><meta charset="utf-8">
<script>
    var inicial = [10,20,30,40,60,70,80];
    var final = [];

    function ver(){
        document.getElementById("salida").innerHTML=`${inicial}`;
        if (final.length != 0)
            document.getElementById("salida2").innerHTML=`${final}`;
    }

    function efectuar(iniciales){
        return iniciales*2
    }

    function modificar(valor){
        final = inicial.map(efectuar)
        ver();
    }

</script>
</head>
<body>
<div id="salida"></div>
<div id="salida2"></div>
<input type="button" onclick="ver()" value="Ver lista">
<input type="button" onclick="modificar(2)" value="Duplicar">
</body>
</html>
```

Arrays 26) Array bidimensional: 3 asignaturas, 5 alumnos (15 notas en total)

```
<!DOCTYPE html>
<html><head>
<script type="text/javascript">
var notas = new Array (6);
for(i=1;i<6;i++)
    notas[i]=new Array(4); // 3 asignaturas cada alumno.

function inicializar()
{
    for (var i = 1; i < 6; i++) // Cada Alumno
        for (var j = 1; j < 4; j++) // 3 notas
            notas [i][j] = 0;
}
function mostrar()
{
    document.write("<h3>Los datos ahora son: </h3>");

    for (var i = 1; i < 6; i++) // Cada Alumno
        for (var j = 1; j < 4; j++)
            document.write("Alumno " + i + " , nota : " + notas[i][j] + "<br>");
}
function modificar(numalumno)
{
    dato= window.prompt("Introduce asignatura para cambiar nota", "0");
    asignatura = parseInt(dato);

    nuevanota = window.prompt("Introduce su nueva nota", "0");
    valornota = parseInt(nuevanota);

    notas[numalumno][asignatura] = nuevanota;
}
</script>
</head>
<body>
<script type="text/javascript">
    document.write("<h2> Inicio Programa Notas </h2>");

    inicializar();
    mostrar();

    dato = window.prompt("Dime el número del alumno (1-5)", "0");
    x = parseInt(dato);

    modificar(x);
    mostrar();
</script>
<p><br>Final Programa</p>
</body>
</html>
```

JAVASCRIPT - jQuery

Inicio con jQuery - variables

Como nos indican en el curso de jQuery de [W3School](#)

- jQuery es una librería (archivo añadido al proyecto web) que facilita mucho la programación.
Permite manejar los siguientes elementos con una programación directa y sencilla:

- HTML/DOM , su manipulación y modificación.
- CSS, modificaciones y asignaciones.
- HTML eventos y métodos.
- Efectos y animaciones.
- AJAX
- Otras utilidades

Para que pueda funcionar la librería de jQuery, necesitamos que, en la misma carpeta donde tenemos el archivo HTML donde vamos a trabajar, hagamos lo siguiente:

- Añadir el archivo "jquery-1.12.0.min.js" (o la versión más actualizada) .

El archivo se obtiene en jquery.com: [Download the compressed, production jQuery](#)

jQuery Inicio 1) Crear archivo nuevo 'jQuery1.htm' y copiar el siguiente código.

En la misma carpeta, añadir el archivo "jquery-1.12.0.min.js"

Una vez guardado, ejecutar el programa abriendo el navegador.

```
<!DOCTYPE html >
<html>
<head>
<meta charset="utf-8">
<script src="jquery-1.12.0.min.js"></script>

<script type="text/javascript">
$(document).ready(function () {
    var saludo = "Hola mundo!";

    $("#boton").click(function () {
        $("#salida").append(saludo);
    });
}); // fin document.ready
</script>

</head>
<body>

<div id="boton"> Pulsa aquí </div>
<div id="salida"></div>

</body>
</html>
```

Cuando se hace click sobre la <div> botón,
se imprime el contenido de la variable
saludo en la <div> salida.

jQuery Inicio 2)

2a) Cambiar el id de la <div> "salida" por otro nombre y hacer que funcione el programa (cambiar también otra instrucción).

2b) Añadir otra <div> y hacer que salga también en ella el saludo.



jQuery Inicio 3) Crear archivo nuevo 'jQuery_Ej3.htm' y copiar el siguiente código.
Una vez guardado, ejecutar el programa abriendo el navegador.

```
<!DOCTYPE html >
<html><head>
<script src="jquery-1.12.0.min.js"></script>

<script type="text/javascript">
$(document).ready(function () {
    var saludo = "Hola mundo!";
    $("div p").click(function () {
        $("#salida").append(saludo);
    });
});           // fin document.ready
</script>
</head>
<body>
<div><p> Pulsa aquí </p></div>
<div id="salida"></div>
</body>
</html>
```

Al hacer click sobre elemento <p> DENTRO de un elemento <div>, se ejecuta la función .append.

jQuery Inicio 4) Ejecutar el siguiente programa

```
<!DOCTYPE html >
<html><head>
<script src="jquery-1.12.0.min.js"></script>

<script type="text/javascript">
$(document).ready(function () {
    var a, b;
    a = 10; b = 5;
    var resultado = a + b;

    $("#salida").append("La suma de a+b es " + resultado );
});

</script>
</head>

<body>
<div id="salida"></div>
</body>
</html>
```

**jQuery DHTML 1) Copiar y ejecutar el siguiente código:**

```
<!doctype html>
<html lang="es"><head>
  <meta charset="utf-8">
  <title>id demo</title>
  <style>
    div {
      width: 90px; height: 90px;
      float: left; padding: 5px; margin: 5px;
      background-color: #eee;
    }
  </style>
  <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
</head>
<body>
  <div id="primera"><p>id="primera"</p></div>
  <div id="myDiv">id="myDiv"</div>
  <script>
    $("#myDiv").css( "border", "3px solid red" );
  </script>
</body>
</html>
```

jQuery DHTML 2) Copiar y ejecutar el siguiente código:

```
<!DOCTYPE html >
<html><head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
<script type="text/javascript">

$(document).ready(function () {

  $("#boton1").click(function(){
    $("#titulo").css("color", "#00ff00");
  });

  $("#boton2").click(function(){
    $("#titulo").css("font-size", "28px")
  });
});

</script>
</head>
<body>
<p id="titulo">Pulsa para cambiar este título dinámico</p>
<input id="boton1" type="button" value="Cambiar Color">
<input id="boton2" type="button" value="Cambiar Tamaño">
</body>
</html>
```

jQuery DHTML 3)

Evento 'mouseenter' sobre <div> provoca disparo de function.

```
<!DOCTYPE html >
<html><head>
<script src="https://code.jquery.com/jquery-1.10.2.js"></script>

<script type="text/javascript">
$(document).ready(function () {

    $("#pulsador1").mouseenter(function () {
        alert("Has entrado en el pulsador1!");
    });
});

</script>
</head>
<body>

<div id="pulsador1" style="border-style: solid; background-color: #AA80BB; width: 100px;">Entra en este espacio</div>

</body>
</html>
```

jQuery DHTML 4) Copiar y ejecutar el siguiente código:

```
<!DOCTYPE html >
<html><head>
<script src="jQuery.js"></script>

<style type="text/css">
#central
{
    background-color: #AA0080;
    width: 100px;
    height: 100px;
}
</style>

<script type="text/javascript">
    var cuenta =1;

    $(document).ready(function () {
        $("p").click(function () {
            alert('se presionó el párrafo');
        });

        $("#central").click(function () {
            var cadena = "se presionó <div> " + cuenta + " veces";
            alert(cadena);
            cuenta = cuenta +1;
        });
    });

    </script>
</head>
<body>
<p>Esto es un párrafo</p><br /><br />
<div id="central"> Pulsa aqui </div>
</body>
</html>
```

jQuery DHTML 5)

Realizar programa para que pulsando una de las 3 <div> inferiores, cambie el color de la <div> superior.

**ROJO****VERDE****AZUL**

jQuery DHTML 6) Copiar y ejecutar el siguiente código:

```
<!DOCTYPE html >
<html>
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>

<style type="text/css">
#contenedor
{
    position: relative;
    background-color: #C0C0C0;
}
#cuartado1
{
    background-color: #00FF00;
    width: 100px; height: 100px;
    position: relative;
    left: 10px
}
#cuartado2
{
    background-color: #AA00FF;
    width: 100px; height: 100px;
    position: relative;
    left: 10px;
}
</style>

<script type="text/javascript">
$(document).ready(function () {
    $("#cuadrado2").click(function () {
        posx= $(this).css("left");
        if (posx == "") posx = "20";
        num = parseInt(posx)
        num = num + 10;
        $(this).css("left",num);
    });
});
</script>
</head>

<body>
<div id="contenedor">
    <div id="cuadrado1"> No me toques </div>
    <div id="cuadrado2" > A mi si </div>
</div>
</body>
</html>
```

jQuery DHTML 7) Atributo Id dispara función attr('id')

```
<!DOCTYPE html>
<html><head>
<script src="jQuery.js"></script>
<style type="text/css">
#contenedor
{
    position: relative;
    background-color: #C0C0C0;
}

#cuartado1
{
    background-color: #00FF00;
    width: 100px; height: 100px;
    position: relative;
    left: 10px
}
#cuartado2
{
    background-color: #AA00FF;
    width: 100px; height: 100px;
    position: relative;
    left: 10px;
}
</style>

<script type="text/javascript">
$(document).ready(function () {
    $("div").click(function () {

        posx= $(this).css("left");
        if (posx == "") posx = "20";
        num = parseInt(posx)

        num = num + 10;
        $(this).css("left",num);

        if ($(this).attr('id') == "cuadrado1")
            mueve("#cuadrado1",num)
    });
    function mueve (objeto, pos) {
        pos = pos + 30;
        $(objeto).css("left",pos)
    }
});
</script>
</head>

<body>
<div id="contenedor">
<div id="cuadrado1"> No me toques </div>
<div id="cuadrado2" > A mi si </div>
</div>
</body>
</html>
```

jQuery DHTML 8)

Realizar una aplicación “Apuestas en las carreras”

El usuario escoje un participante de 5 posibles (imágenes diferentes dentro de <divs>) y realiza una apuesta.

Empieza una carrera aleatoria entre los participantes para llegar al final de la pantalla.

Si el corredor del usuario ha ganado, se le suma al total de su dinero, el doble de lo apostado.

jQuery DHTML 9)

Evento ‘hover’ sobre párrafos <p> provoca cambio de propiedad CSS del mismo objeto.

```
<!DOCTYPE html>
<html>
<head>
<script src="scripts/jQuery.js"></script>
<script type="text/javascript">

$(document).ready(function () {
    var obj;
    obj = $("p");
    obj.hover(entradaMouse, saleMouse);
});

function entradaMouse() {
    $(this).css("background-color", "#ff0");
    $(this).css("font-size", "28");
}

function saleMouse() {
    $(this).css("background-color", "#fff");
    $(this).css("font-size", "24");
}

</script>
</head>
<body>
<p> Ya no pienso que soy un artista, lo soy </p>
<br>
<p> El problema de este mundo, es que los sabios dudan </p>
<br>
<p> La solución más sencilla es la más cierta</p>
<br>
</body>
</html>
```

jQuery DHTML 10)

Evento 'click' sobre Id provoca cambio en las propiedades CSS del mismo objeto.

```
<!DOCTYPE html >
<html><head>
<script src="jQuery.js"></script>
<script type="text/javascript">

$(document).ready(function () {

    $("#titulo1").click(presionTitulo1)
});

function presionTitulo1()
{
    var obj;
    obj = $("#titulo1");
    obj.css("color", "#ff0000");
    obj.css("background-color", "#ffff00");
    obj.css("font-family", "Courier");
}

</script>
</head>

<body>
<h1 id="titulo1">Primer título</h1>
<h1 id="titulo2">Segundo título</h1>
</body>
</html>
```



jQuery DHTML 11)

Evento 'mouseenter' sobre Id provoca cambio de propiedad CSS del mismo objeto.

```
<!DOCTYPE html>
<html>
<head>
<script src="jQuery.js"></script>
<script type="text/javascript">

$(document).ready(function () {

    $("#zonaX").click(function () {
        $("#zonaX").css("background-color", "#bbbaa00");
    });

    $("#zonaX").dblclick(function () {
        $("#zonaX").css("background-color", "#aaffaa");
    });

});

</script>
</head>

<body>

<div id="zonaX" style="border-style: solid; background-color: #008080; width: 300px;">
    Pulsa 1 o 2 veces en este espacio
</div>

</body>
</html>
```

jQuery Ejercicio Forms 1)

```
<!DOCTYPE html >
<html>
<head><meta charset="utf-8">
<script src="scripts/jQuery.js"></script>

<script type="text/javascript">
$(document).ready(function () {
    var dato, resultado;
    $("#entrada").keypress(function(e){
        if (e.keyCode == 13) {
            nombre = $("#entrada").val();
            $("#salida").append("Hola, buenos dias " + nombre);
        }
    });
    </script>

</head>

<body>
    Escribe tu nombre: <input id="entrada" type="text"/>
    <br>
    <div id="salida"></div>
</body>
</html>
```

Al presionar una tecla sobre #entrada, se comprueba su código. Si es 13, es la tecla ENTER.

Clases

Javascript obj 1) El objeto "Persona" tiene dos propiedades y una función propia (método)

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

function Persona(nom, ape)
{
    this.nombre = nom;
    this.apellido = ape;
    this.cadena = function () {
        return this.nombre + " " + this.apellido;
    }
}
</script>
</head>
<body>
<div id="divPersona">&nbsp;</div>
<script>
    var p1 = new Persona("Mat", "Salave");
    document.getElementById("divPersona").innerHTML = p1.cadena();
</script>
</body>
</html>
```

**Javascript obj 2) Método: Funciones propias de los objetos.**

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

function evaluar(n) ← Función propia del objeto : Método.
{                                         'n' forma parte del Global del objeto, desde aquí
    if (n >= 5)                         puede acceder a su valor.
        this.nota ="Aprobado"
    else
        this.nota="Suspenido"
}

function CreaObjeto(nom,n) ← Función "constructora", asigna valores iniciales.
{
    this.nombre = nom;
    this.edad = n;
    this.asignarnota = evaluar;

}
</script>
</head>

<body>
<script type="text/javascript">

var alumno = new CreaObjeto("Carlos",18);

alumno.nota = 6;
document.write("La nota de " + alumno.nombre + ", ahora es: " + alumno.nota);

document.write("<br>");

alumno.asignarnota(alumno.nota);
document.write("La nota ahora es: " + alumno.nota + "<br>");

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

Javascript obj 3) Completo ejercicio con todas las posibilidades de los "objetos"

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

function FichaDatos()
{
    var nombre;
    var edad;
    var peso;
    this.inicio = function Cargar()
    {
        this.nombre = "Pedro";
        this.edad = 29;
        this.peso = 75.5;
    }

    this.ver = function Mostrar()
    {
        var mensaje1 = "El nombre es " + this.nombre;
        document.writeln(mensaje1);

        var mensaje2 = "su peso es " + this.peso + " y su edad " + this.edad;
        document.writeln(mensaje2);
    }
}

</script>
</head>
<body>
<script type="text/javascript">

    var Paciente = new FichaDatos();
    Paciente.inicio();
    Paciente.ver();

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

Javascript obj 4)

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

var paciente = {
    "nombre": "Norat",
    "apellido": "Puig",
    "edad": 25,
    "direccion": [{{
        "calle": "Avenida Rossello",
        "ciudad": "Palma",
        "codigo": "08011"
    }}]
}

document.write("El paciente es : " + paciente.apellido + "<br><br>");
document.write("Su dirección : " + paciente.direccion[0].calle);
document.write(", la ciudad es " + paciente.direccion[0].ciudad);

</script>
</head>

<body>
<br><br><p>Las variables en formato JSON pueden ser tratadas como objetos.</p>
</body>
</html>
```



Javascript Obj 5) Programa que crea un objeto "Rectángulo" con ls funciones propias para calcular area y su perímetro.

```
<!DOCTYPE html>
<html>
<head><meta charset="utf-8"/>
<title>Calculo ecuaciones segundo grado</title>
<script type="text/javascript">

function Rectangulo() {
    var resultado;
    this.calculaArea = function (A,B)
    {
        resultado = A*B;
    }
    this.calculaPerimetro = function (A,B)
    {
        resultado = 2*(A+B);
    }
    this.resultadofinal = function ()
    {
        return resultado;
    }
}

function programa() {

    var num1 = parseFloat(document.fdatos.entradaA.value);
    var num2 = parseFloat(document.fdatos.entradaB.value);

    var obj = new Rectangulo();
    obj.calculaArea (num1,num2);
    document.writeln("El area es "+ obj.resultadofinal() + "<br>");

    obj.calculaPerimetro (num1,num2);
    document.writeln("El perimetro es "+ obj.resultadofinal());

}

</script>
</head>
<body>
<h2>Calculo Area y perimetro rectangulo</h2>
<form name="fdatos">
    Escribe Base: <input type="text" size="2" name="entradaA"/>
    Escribe Altura: <input type="text" size="2" name="entradaB"/>
    <input type="button" value="Pulsa aquí" onclick="programa()" />
</form>
</body>
</html>
```

Javascript Obj 6) Ejecutar el siguiente programa: ARRAYS DE OBJETOS

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

function Person (name,age) {
this.name = name;
this.age = age;
}

var family = new Array ();
family[0] = new Person("alice", 40);
family[1] = new Person("bob" , 42);
family[2] = new Person("michelle", 8);
family[3] = new Person("timmy", 6);

document.writeln("My family:");

for (i=0;i<4;i++){
document.writeln(family[i].name);
}

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

Javascript Obj 7)

- a) Añadir función al ejercicio anterior que añade x años a cada elemento del array.
- b) Añadir función al ejercicio anterior que muestre los objetos del array.

Javascript Obj 8) Añadir función al siguiente programa para agregar objetos al array

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

var family = new Array ();
family[0] = new Person("alice", 40);
family[1] = new Person("bob" , 42);
family[2] = new Person("michelle", 8);
family[3] = new Person("timmy", 6);

function Person (name,age) {
this.name = name;
this.age = age;
}

function PasoTiempo(n){
for (i=0;i<4;i++){
family[i].age = family[i].age + n ;
}
}

function Mostrar(inicio,final)
{
for (i=inicio;i<final;i++){
document.writeln(family[i].name + "-" + family[i].age);
}
}

document.writeln("My family:");
Mostrar (0,4)
PasoTiempo(5);
document.writeln("<br><br>En 5 años:");
Mostrar (0,4);

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

JAVASCRIPT - complementario

Javascript complementario 1)

Crear programa, en una página web, donde el usuario introduce su altura (cm), edad, sexo y peso actual. El programa responde con el peso ideal.

Por ejemplo : usuario introduce 176, 22, H, 79.

El programa ejecuta la fórmula: $50 + ((\text{Altura}-150) / 4)^3 + (\text{Edad} - 20) / 4$ * (0.9 si es mujer)

El resultado es: peso ideal = 70,5.

```
<!DOCTYPE html >
<html>
<head>
<title>Peso ideal</title>
<script type="text/javascript">

function pesoideal() {

    var dato, peso, altura, sexo, edad;
    dato = window.prompt("Dígame su altura en cm");
    altura = parseInt(dato);

    dato = window.prompt("Dígame su edad en años")
    edad = parseInt(dato);

    sexo = window.prompt("Dígame su sexo, hombre = h o mujer = m");

    if (sexo == "h")
    {
        peso = 50 + ((altura - 150) / 4)*3 + (edad - 20)/4;
    }
    else peso = peso * 0.9;

    alert("Su peso ideal es " + peso + " Kg")
}

</script>
</head>

<body>
<input id="pesoideal" type="button" value="peso ideal" onclick="pesoideal()"/>
</body>
</html>
```

jQuery Ejercicio complementario 1)

Crear programa, en una pàgina web, donde el usuario introduce su altura (cm), edad, sexo y peso actual. El programa responde con el peso ideal.

Por ejemplo : usuario introduce 176, 22, H, 79.

El programa ejecuta la fórmula: $50 + ((\text{Altura}-150) / 4) * 3 + (\text{Edad} - 20) / 4$ * (0.9 si es mujer)

```
<!DOCTYPE html >
<html>
<head><meta charset="utf-8">
<script src="scripts/jquery.js"></script>
<title>Peso ideal</title>
<script type="text/javascript">

$(document).ready(function () {

    $("button").click(function () {
        var dato1 = parseInt($("#entrada1").val());
        var dato2 = parseInt($("#entrada2").val());
        var dato3 = $("#entrada1").val();
        Calcula (dato1, dato2, dato3);
    });

    function Calcula (altura,edad,sexo){
        var resultado =0;
        resultado = 50 + ((altura - 150) / 4)*3 + (edad - 20)/4;

        if (sexo == "m")  resultado = resultado * 0.9;

        Pantalla(resultado);
    }

    function Pantalla (verfinal) {
        $("#salida").append("Su peso ideal es: " + verfinal);
    }
});

</script>
</head>
<body>
<br>
    Introduce tu altura en cm: <input id="entrada1" type="text"/>
    <br>
    Edad en años: <input id="entrada2" type="text"/>
    <br>
    Hombre o Mujer (h/m): <input id="entrada3" type="text"/>
    <br>
    <button>Pulsa para calcular peso ideal</button>

    <div id="salida"></div>
</body>
</html>
```



Javascript Ejercicio complementario 2)

Añadir código al programa de peso ideal para que la edad introducida sólo esté permitida entre 1 y 120 años.

Javascript Ejercicio complementario 3)

Realizar "Seguimiento de Variables" gracias a los puntos de Interrupción.

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">

function IntroducirDatos()
{
    var dato1, n1;
    dato1 = window.prompt("Primer entero", "0");
    n1 = parseInt(dato1);
    Cuadrado(n1);
}

function Cuadrado(numero)
{
    var calculo = numero * numero;
    document.write("<p/> Resultado = " + calculo + "</p>");
}
</script>
</head>

<body>
<p><input type="button" value=" PULSA " onclick="IntroducirDatos()" /></p>
</body>

</html>
```

Javascript - JSON

JSON 1) Programación "AEROPUERTO" con JSON

Archivo: InicioAviones.htm

```
<html>
<head>
<title>JSON Aviones</title>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript" >

    var xmlhttp = new XMLHttpRequest();
    var url = "aviones.txt";

    xmlhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            var arrayVuelos = JSON.parse(this.responseText);
            mostrarArray(arrayVuelos);
        }
    };

    xmlhttp.open("GET", url, true);
    xmlhttp.send();

    function mostrarArray(datos) {
        let out = "";
        for( let i = 0; i < datos.length; i++) {
            out += `<p> origen: ${datos[i].origen} <a href= ${datos[i].url}>
                    destino: ${datos[i].destino}</a></p><br>`;
        }
        document.getElementById("salidaDatos").innerHTML = out;
    }
}

</script>
</head>

<body>
    <div id="salidaDatos">
        <div id="content">
        </div>
    </div>
</body>
</html>
```

**Archivo: "Aviones.txt"**

```
[  
{  
  "origen": "Palma",  
  "destino": "Oslo",  
  "url": "http://www.ibserveis.com/vuelo1.html"  
},  
{  
  "origen": "Lugo",  
  "destino": "Madrid",  
  "url": "http://www.ibserveis.com/vuelo2.html"  
},  
{  
  "origen": "Barna",  
  "destino": "Paris",  
  "url": "http://www.ibserveis.com/vuelo3.html"  
},  
{  
  "origen": "Palma",  
  "destino": "Madrid",  
  "url": "http://www.ibserveis.com/vuelo4.html"  
},  
{  
  "origen": "Palma",  
  "destino": "Paris",  
  "url": "http://www.ibserveis.com/vuelo5.html"  
}  
]
```

2.1 Añadir informaciones: hora salida y hora llegada al XML anterior.

2.2 Añadir datos hasta llegar atener al menos 10 “ vuelos”

2.3 Completar programa donde el usuario ve en pantalla un menú:

- 1) Ver todos los vuelos
- 2) Ver vuelos de un destino determinado.
- 3) Ver vuelos que llegan más tarde que una hora determinada (números enteros)

2.4 ¿Se podrian evitar las variables globales? ¿cómo?



JSON2) Realizar programa para gestionar notas de clase desde un menú , para el usuario, con botones en pantalla.

**JSON 3) Realizar página web donde se pide Usuari /contraseña para entrar a ver información.
Los datos usuario/contraseña estan en un archivo JSON**

