

2.- UTILIZACIÓN DE LOS OBJETOS PREDEFINIDOS DE JAVASCRIPT.

OBJETIVOS

- Conocer cuáles son los principales objetos predefinidos de JavaScript.
- Comprender las propiedades y métodos de los principales objetos de JavaScript.
- Aprender a generar código HTML desde sentencias JavaScript.
- Manipular y gestionar la creación y apariencia de las ventanas del navegador, además de la comunicación entre ellas.

2.1.- Introducción

Un objeto es una colección de:

- Propiedades (valores).
- Métodos (funciones).
- Eventos (acciones)

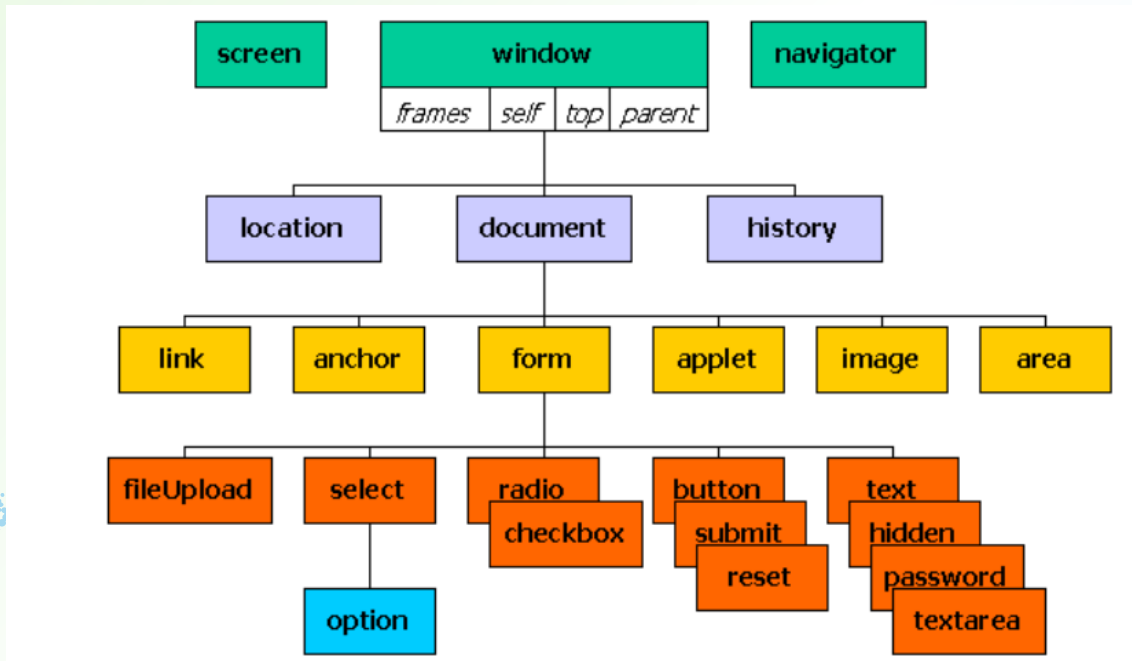
Para acceder a una propiedad o método de un objeto hay que utilizar el punto (.)

```
Objeto.propiedad;  
Objeto.método([argumentos]);
```

2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

En la jerarquía de objetos, tenemos en la parte superior el objeto windows.



2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

Acceso a propiedad y métodos

```
window.nombrePropiedad  
window.nombreMétodo( [parámetros] )
```

Un objeto **window** también se podrá referenciar mediante la palabra **self**

```
self.nombrePropiedad  
self.nombreMétodo( [parámetros] )
```

Como contiene el resto de objeto podemos omitir su nombre:

```
nombrePropiedad  
nombreMétodo( [parámetros] )
```

2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

Propiedades:

Propiedad	Descripción
closed	Devuelve un valor Boolean indicando cuando una ventana ha sido cerrada o no.
defaultStatus	Ajusta o devuelve el valor por defecto de la barra de estado de una ventana.
document	Devuelve el objeto document para la ventana.
frames	Devuelve un array de todos los marcos (incluidos iframes) de la ventana actual.
history	Devuelve el objeto history de la ventana.
length	Devuelve el número de frames (incluyendo iframes) que hay en dentro de una ventana.
location	Devuelve la Localización del objeto ventana (URL del fichero).

2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

Propiedades:

Propiedad	Descripción
name	Ajusta o devuelve el nombre de una ventana.
navigator	Devuelve el objeto navigator de una ventana.
opener	Devuelve la referencia a la ventana que abrió la ventana actual.
parent	Devuelve la ventana padre de la ventana actual.
self / window	Devuelve la ventana actual.
status	Ajusta el texto de la barra de estado de una ventana.
top	Nombre alternativo de la ventana del nivel superior

2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

Métodos:

Método	Descripción
alert(mensaje)	Muestra una ventana emergente de alerta y un botón de aceptar.
blur()	Elimina el foco de la ventana actual.
clearInterval(id)	Resetea el cronómetro ajustado con setInterval().
setInterval(expresion, tiempo)	Llama a una función o evalúa una expresión en un intervalo especificado (en milisegundos).
clearTimeout(nombre)	Cancela el intervalo referenciado por 'nombre'.
setTimeout(expresion, tiempo)	Evalúa la expresión después de que hayan pasado un intervalo de tiempo (en milisegundos).
close()	Cierra la ventana actual.
confirm(mensaje)	Muestra una ventana emergente con un mensaje, un botón de aceptar y un botón de cancelar.

2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

Métodos:

Método	Descripción
focus()	Coloca el foco en la ventana actual.
moveBy(x,y)	Mueve la ventana actual el número de píxeles especificados.
moveTo(x,y)	Mueve la ventana a la posición especificada.
open(url,nombre,características)	Abre una nueva ventana de navegación.
prompt(mensaje,respuesta_defecto)	Muestra una ventana de diálogo para introducir datos.
scrollBy(x,y)	Mueve el scroll de la ventana el número de píxeles indicados.
scrollTo(x,y)	Mueve el scroll de la ventana a la posición indicada.

2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

Características de una ventana:

Método	Descripción
toolbar = [yes no 1 0]	Indica si la ventana tendrá o no barra de herramientas (yes=1; no=0).
location = [yes no 1 0]	Indica si la ventana tendrá campo de localización o no (yes=1; no=0).
directories = [yes no 1 0]	Indica si la ventana tendrá botones de dirección o no (yes=1; no=0).
status = [yes no 1 0]	Indica si la ventana tendrá barra de estado o no (yes=1; no=0).
menubar = [yes no 1 0]	Indica si la ventana tendrá barra de menús o no (yes=1; no=0).
scrollbars = [yes no 1]	Indica si la ventana tendrá barras de desplazamiento o no (yes=1; no=0).
resizable = [yes no 1 0].	Indica si la ventana se podrá redimensionar o no (yes=1; no=0).
width = px / height = px	Ancho y alto de la ventana en píxeles
outerWidth = px / outerHeight = px	Ancho y alto total de la ventana en píxeles.
top = px / left = px	Distancia de la ventana desde la parte superior e izquierda en píxeles.

2.2.- Objetos de más alto nivel.

2.2.1.- Objeto Window

Crear ventanas:

Un script no creará nunca la ventana principal de un navegador pero un script que se está ejecutando en una ventana principal, sí que podrá crear o abrir nuevas sub-ventanas mediante el método `open()`.

```
var subVentana=window.open("nueva.html","nueva","height=800,width=600");
```

La variable `subVentana`, hace referencia a la nueva ventana. Para cerrar esa ventana:

✓ `subVentana.close()`

2.2.- Objetos de más alto nivel.

2.2.2.- Objeto Location

- ✓ El objeto **location** contiene información referente a la URL actual.
- ✓ Es parte del objeto **window**

Propiedad	Descripción
hash	Cadena que contiene el nombre del enlace, dentro de la URL.
host	Cadena que contiene el nombre del servidor y el número del puerto, dentro de la URL.
hostname	Cadena que contiene el nombre de dominio del servidor (o la dirección IP), dentro de la URL.
href	Cadena que contiene la URL completa.
pathname	Cadena que contiene el camino al recurso, dentro de la URL.
port	Cadena que contiene el número de puerto del servidor, dentro de la URL.
protocol	Cadena que contiene el protocolo utilizado (incluyendo los dos puntos), dentro de la URL.
search	Cadena que contiene la información pasada en una llamada a un script, dentro de la URL.

2.2.- Objetos de más alto nivel.

2.2.2.- Objeto Location

Métodos del objeto Location

<code>assign()</code>	Carga un nuevo documento.
<code>reload()</code>	Vuelve a cargar la URL especificada en la propiedad <code>href</code> del objeto <code>location</code> .
<code>replace()</code>	Reemplaza el historial actual mientras carga la URL especificada en <code>cadenaURL</code> .

2.2.- Objetos de más alto nivel.

2.2.2.- Objeto Navigator

✓ El objeto **Navigator** contiene información sobre el navegador.

Propiedad	Descripción
appName	Cadena que contiene el nombre en código del navegador.
appVersion	Cadena que contiene el nombre del cliente.
cookieEnabled	Cadena que contiene información sobre la versión del cliente.
language	Determina si las cookies están o no habilitadas en el navegador.
onLine	Devuelve el idioma por defecto del navegador. "es" si es español o "en" si es inglés.
platform	Indica si el navegador está offline u online. O sea que se ha desconectado de Internet usando la opción del navegador.
userAgent	Cadena con la plataforma sobre la que se está ejecutando el programa cliente.
	Cadena que contiene la cabecera completa del agente enviada en una petición <u>HTTP</u> . Contiene la información de las propiedades <code>appName</code> y <code>appVersion</code> .

2.2.- Objetos de más alto nivel.

2.2.2.- Objeto Navigator

Métodos del objeto Navigator

Método	Descripción
<code>javaEnabled()</code>	Devuelve true si el cliente permite la utilización de Java, en caso contrario, devuelve false.

2.2.- Objetos de más alto nivel.

2.2.3.- Objeto document

- ✓ El objeto **document** proporciona a los scripts, el acceso a todos los elementos HTML.
- ✓ Cada documento cargado en una ventana del navegador, será un objeto de tipo **document**.

Colecciones del objeto Document

Colección	Descripción
<code>anchors[]</code>	Es un array que contiene todos los hipervínculos del documento.
<code>forms[]</code>	Es un array que contiene todos los formularios del documento.
<code>images[]</code>	Es un array que contiene todas las imágenes del documento.
<code>links[]</code>	Es un array que contiene todos los enlaces del documento.

2.2.- Objetos de más alto nivel.

2.2.3.- Objeto document

Propiedades del objeto Document

Propiedad	Descripción
cookie	Devuelve todos los nombres/valores de las cookies en el documento.
domain	Cadena que contiene el nombre de dominio del servidor que cargó el documento.
referrer	Cadena que contiene la URL del documento desde el cuál llegamos al documento actual.
title	Devuelve o ajusta el título del documento.
URL	Devuelve la URL completa del documento.

2.2.- Objetos de más alto nivel.

2.2.3.- Objeto document

Métodos del objeto document

Método	Descripción
<code>close()</code>	Cierra el flujo abierto previamente con <code>document.open()</code> .
<code>getElementById()</code>	Para acceder a un elemento identificado por el id escrito entre paréntesis.
<code>getElementsByName()</code>	Para acceder a los elementos identificados por el atributo <code>name</code> escrito entre paréntesis.
<code>getElementsByTagName()</code>	Para acceder a los elementos identificados por el tag o la etiqueta escrita entre paréntesis.
<code>open()</code>	Abre el flujo de escritura para poder utilizar <code>document.write()</code> o <code>document.writeln</code> en el documento.
<code>write()</code>	Para poder escribir expresiones HTML o código de JavaScript dentro de un documento.
<code>writeln()</code>	Lo mismo que <code>write()</code> pero añade un salto de línea al final de cada instrucción.

2.3.- Objetos nativos en Javascript.

Los objetos nativos no dependen del navegador

- Date
- Math
- Number
- Boolean
- String

Sintaxis para crear un objeto:

var / let / const miObjeto = new Clase();

2.3.- Objetos Nativos en JavaScript.

2.3.1.- Objeto Date

El objeto **Date** se utiliza para trabajar con fechas y horas.

Instanciar:

- `var d = new Date();`
- `var d = new Date(milisegundos);`
- `var d = new Date(cadena de Fecha);`
- `var d = new Date(año, mes, día, horas, minutos, segundos, milisegundos);`
`// (el mes comienza en 0, Enero sería 0, Febrero 1, etc.)`

2.3.- Objetos Nativos en JavaScript.

2.3.1.- Objeto Date

Propiedades:

Propiedad	Descripción
constructor	Devuelve la función que creó el objeto Date.
prototype	Te permitirá añadir propiedades y métodos a un objeto.

2.3.- Objetos Nativos en JavaScript.

2.3.1.- Objeto Date

Métodos:

Método	Descripción
getDate()	Devuelve el día del mes (de 1-31).
getDay()	Devuelve el día de la semana (de 0-6).
getFullYear()	Devuelve el año (4 dígitos).
getHours()	Devuelve la hora (de 0-23).
getMilliseconds()	Devuelve los milisegundos (de 0-999).
getMinutes()	Devuelve los minutos (de 0-59).
getMonth()	Devuelve el mes (de 0-11).
getSeconds()	Devuelve los segundos (de 0-59).

2.3.- Objetos Nativos en JavaScript.

2.3.1.- Objeto Date

Métodos:

Método	Descripción
getTime()	Devuelve los milisegundos desde media noche del 1 de Enero de 1970.
getTimezoneOffset()	Devuelve la diferencia de tiempo entre GMT y la hora local, en minutos.
getUTCDate()	Devuelve el día del mes en base a la hora UTC (de 1-31).
getUTCDay()	Devuelve el día de la semana en base a la hora UTC (de 0-6).
getUTCFullYear()	Devuelve el año en base a la hora UTC (4 dígitos).
setDate()	Ajusta el día del mes del objeto (de 1-31).
setFullYear()	Ajusta el año del objeto (4 dígitos).
setHours()	Ajusta la hora del objeto (de 0-23).

2.3.- Objetos Nativos en JavaScript.

2.3.2.- Objeto Math

El objeto **Math** permite realizar operaciones matemáticas. **Math** no es un constructor, por tanto, no permite instanciar un objeto.

Crear:

```
Let numPi = Math.PI; // devuelve el número Pi
```

```
Let raizC= Math.sqrt(16); // devuelve la raíz cuadrada de 16
```


2.3.- Objetos Nativos en JavaScript.

2.3.2.- Objeto Math

Propiedades:

Propiedad	Descripción
E	Devuelve el número Euler (aproximadamente 2.718).
LN2	Devuelve el logaritmo neperiano de 2 (aproximadamente 0.693).
LN10	Devuelve el logaritmo neperiano de 10 (aproximadamente 2.302).
LOG2E	Devuelve el logaritmo base 2 de E (aproximadamente 1.442).
LOG10E	Devuelve el logaritmo base 10 de E (aproximadamente 0.434).
PI	Devuelve el número PI (aproximadamente 3.14159).
SQRT2	Devuelve la raíz cuadrada de 2 (aproximadamente 1.414).

2.3.- Objetos Nativos en JavaScript.

2.3.2.- Objeto Math

Métodos:

Método	Descripción
abs(x)	Devuelve el valor absoluto de x.
acos(x)	Devuelve el arcocoseno de x, en radianes.
asin(x)	Devuelve el arcoseno de x, en radianes.
atan(x)	Devuelve el arcotangente de x, en radianes con un valor entre $-PI/2$ y $PI/2$.
atan2(y,x)	Devuelve el arcotangente del cociente de sus argumentos.
ceil(x)	Devuelve el número x redondeado al alta hacia el siguiente entero.
cos(x)	Devuelve el coseno de x (x está en radianes).
floor(x)	Devuelve el número x redondeado a la baja hacia el anterior entero.
log(x)	Devuelve el logaritmo neperiano (base E) de x.

2.3.2.- Objeto Math

Métodos:

Método	Descripción
<code>max(x,y,z,...,n)</code>	Devuelve el número más alto de los que se pasan como parámetros.
<code>min(x,y,z,...,n)</code>	Devuelve el número más bajo de los que se pasan como parámetros.
<code>pow(x,y)</code>	Devuelve el resultado de x elevado a y.
<code>random()</code>	Devuelve un número al azar entre 0 y 1.
<code>round(x)</code>	Redondea x al entero más próximo.
<code>sin(x)</code>	Devuelve el seno de x (x está en radianes).
<code>sqrt(x)</code>	Devuelve la raíz cuadrada de x.
<code>tan(x)</code>	Devuelve la tangente de un ángulo.

2.3.- Objetos Nativos en JavaScript.

2.3.3.- Objeto Number

El objeto **Number** no se suele instanciar; normalmente asignamos valores numéricos a una variable.

Crear:

```
let num=new Number(23);
```

```
let num2=23;
```

Si el parámetro que se pasa al constructor no se puede convertir devuelve el valor NaN (Not and Numeric).

```
let num=new Number("a23"); // devuelve NaN
```

2.3.- Objetos Nativos en JavaScript.

2.3.3.- Objeto Number

Propiedades:

Propiedad	Descripción
constructor	Devuelve la función que creó el objeto Number.
MAX_VALUE	Devuelve el número más alto disponible en JavaScript.
MIN_VALUE	Devuelve el número más pequeño disponible en JavaScript.
NEGATIVE_INFINITY	Representa a infinito negativo (se devuelve en caso de overflow).
POSITIVE_INFINITY	Representa a infinito positivo (se devuelve en caso de overflow).
prototype	Permite añadir nuestras propias propiedades y métodos a un objeto.

2.3.- Objetos Nativos en JavaScript.

2.3.3.- Objeto Number

Métodos:

Método	Descripción
toExponential(x))	Convierte un número a su notación exponencial.
toFixed(x)	Formatea un número con x dígitos decimales después del punto decimal.
toPrecision(x)	Formatea un número a la longitud x.
toString()	Convierte un objeto Number en una cadena. <ul style="list-style-type: none">• Si se pone 2 como parámetro se mostrará el número en binario.• Si se pone 8 como parámetro se mostrará el número en octal.• Si se pone 16 como parámetro se mostrará el número en hexadecimal.
valueOf()	Devuelve el valor primitivo de un objeto Number.

2.3.- Objetos Nativos en JavaScript.

2.3.4.- Objeto Boolean

El objeto **Boolean** se utiliza para convertir un valor no booleano, en un valor booleano (true o false).

```
let bool=new Boolean(1);
```

Devuelve un objeto

```
let semaforo =true;
```

Devuelve un valor primitivo

```
console.log(bool.toString());
```

//devuelve "true"

```
console.log(semaforo.valueOf());
```

//devuelve true

2.3.- Objetos Nativos en JavaScript.

2.3.4.- Objeto Boolean

Propiedades:

Propiedad	Descripción
constructor	Devuelve la función que creó el objeto Boolean.
prototype	Te permitirá añadir propiedades y métodos a un objeto.

Métodos:

Método	Descripción
toString()	Convierte un valor Boolean a una cadena y devuelve el resultado.
valueOf()	Devuelve el valor primitivo de un objeto Boolean.

2.3.- Objetos Nativos en JavaScript.

2.3.5.- Objeto String

Instanciación:

- **Podemos utilizar dos maneras:**

- `let texto = new String ("JavaScript");` El objeto no es inmutable, ya que podemos manipularlos
- `let texto= "JavaScript";` Los valores primitivos son inmutables. Pueden ser recorridos pero no se pueden modificar

- **Podemos utilizar comillas dobles o simples:**

- `let texto = '<input type="checkbox" name="coche" />Audi A6';`
- `let texto= "<input type='checkbox' name='coche' />Audi A6';`

- **Podemos utilizar caracteres de escape:**

- `let texto = '<input type=\\"checkbox\\" name=\\"coche\\" />Audi A6';`

2.3.- Objetos Nativos en JavaScript.

2.3.5.- Objeto String

- **Podemos concatenar cadenas muy largas con +=:**
 - `let texto = ""`
 - `texto+= "JavaScript";`
 - `texto+= " , lenguaje interpretado;`
- **Podemos concatenar cadenas con variables con el operador +:**
 - `let nombreEquipo = prompt("Introduce el nombre de tu equipo favorito");`
 - `let mensaje = "El " + nombreEquipo + " ha sido el campeón de la liga";`
- **Interpolación de variables en cadenas mediante los Template Strings:**
 - `let mensaje=`El ${nombreEquipo} ha sido el campeón de la liga`;`

2.3.- Objetos Nativos en JavaScript.

2.3.5.- Objeto String

Propiedades:

Propiedad	Descripción
length	Contiene la longitud de una cadena.

Métodos:

Método	Descripción
charAt()	Devuelve el carácter especificado por la posición que se indica entre paréntesis.
charCodeAt()	Devuelve el Unicode del carácter especificado por la posición que se indica entre paréntesis.
concat()	Une una o más cadenas y devuelve el resultado de esa unión.
fromCharCode()	Convierte valores Unicode a caracteres.
indexOf()	Devuelve la posición de la primera ocurrencia del carácter buscado en la cadena.
lastIndexOf()	Devuelve la posición de la última ocurrencia del carácter buscado en la cadena.

2.3.- Objetos Nativos en JavaScript.

2.3.5.- Objeto String

Métodos:

Método	Descripción
match()	Busca una coincidencia entre una expresión regular y una cadena y devuelve las coincidencias o null si no ha encontrado nada.
replace()	Busca una subcadena en la cadena y la reemplaza por la nueva cadena especificada. <i>sustitución de cadena1 por la cadena2</i>
search()	Busca una subcadena en la cadena y devuelve la posición dónde se encontró.
slice()	Extrae una parte de la cadena y devuelve una nueva cadena. <i>slice(inicio, [longitud])</i>
split()	Divide una cadena en un array de subcadenas. <i>Si no se pasa ningún carácter, se devuelve un array carácter a carácter</i>
substr()	Extrae los caracteres de una cadena, comenzando en una determinada posición y con el número de caracteres indicado.
substring()	Extrae los caracteres de una cadena entre dos índices especificados.
toLowerCase()	Convierte una cadena en minúsculas.
toUpperCase()	Convierte una cadena en mayúsculas.

2.3.- Objetos Nativos en JavaScript.

2.3.5.- Objeto String

Métodos:

Método	Descripción
<u>includes()</u>	Devuelve true <u>is</u> una cadena contiene una <u>subcadena</u>
<u>localeCompare()</u>	Devuelve un número que indica si la cadena de referencia viene antes, después o es equivalente a la cadena dada
<u>padEnd()</u>	Rellena la cadena actual desde el final, con otra cadena
<u>padStart()</u>	Rellena la cadena actual desde el principio con otra cadena
<u>replaceAll()</u>	Busca una <u>subcadena</u> en la cadena y reemplaza todas <u>subcadenas</u>
<u>repeat()</u>	Devuelve una cadena con una serie de copias de esa cadena
<u>startsWith()</u>	Devuelve true si es una cadena comienza con la <u>subcadena</u> especificada
<u>trimEnd()</u>	Elimina los espacios que hay al final de una cadena
<u>trimStart()</u>	Elimina los espacios que hay al comienzo de una cadena

Trim()

Elimina los espacios al principio y al final de la cadena