

# 3.A. Modelo de objetos predefinidos en JavaScript



## 2.- Marcos.

### 2.2.- Comunicación entre marcos.

#### Referencias Padre-Hijos.

Desde el momento en el que el documento padre contiene uno o más marcos, ese documento padre mantiene un array con sus marcos hijo. Podemos acceder a un marco a través de la sintaxis de array o por el nombre que le hemos dado a ese marco, por el id o con el atributo `name` que hemos puesto en la marca `<frame>`.

Ejemplos de referencias a los marcos hijo:

(Recordar que todo lo que va entre corchetes `[]` es opcional).

```
[window.]frames[n].objeto-función-variable-nombre
```

```
[window.]frames["nombreDelMarco"].objeto-función-variable-nombre
```

```
[window.]nombreDelMarco.objeto-función-variable-nombre
```



El índice numérico `n`, que indica el número de frame, está basado en el orden en el que aparecen en el documento `frameset`. Se recomienda que pongamos un nombre a cada frame en dicho documento, ya que así la referencia a utilizar será mucho más fácil.

#### Referencias Hijo-Padre.

Es bastante más común enlazar scripts al documento padre (`frameset`), ya que éste se carga una vez y permanecerá cargado con los mismos datos, aunque hagamos modificaciones dentro de los marcos.

Desde el punto de vista de un documento hijo (aquel que está en un frame), su antecesor en la jerarquía será denominado el padre (`parent`). Por lo tanto para hacer referencia a elementos del padre se hará:

```
parent.objeto-función-variable-nombre
```

Si el elemento al que accedemos en el padre es una función que devuelve un valor, el valor devuelto será enviado al hijo sin ningún tipo de problemas. Por ejemplo:

```
var valor=parent.NombreFuncion();
```

Además como la ventana padre está en el top de la jerarquía de ventanas, opcionalmente podríamos escribir:

```
var valor=top.NombreFuncion();
```

#### Referencias Hijos-Hijos.

El navegador necesita un poco más de asistencia cuando queremos que una ventana hija se comunique con una hermana. Una de las propiedades de cualquier ventana o marco es su padre (`parent` - el cuál será `null` cuando estamos hablando de una ventana sin hijos). Por lo tanto, la forma de comunicar dos ventanas o marcos hermanos va a ser siempre referenciándolos a través de su padre, ya que es el único nexo de unión entre ambos (los dos tienen el mismo padre).

Podemos utilizar alguno de los siguientes formatos:

```
parent.frames[n].objeto-función-variable-nombre
```

```
parent.frames["nombreDelMarco"].objeto-función-variable-nombre  
parent.nombreDelMarco.objeto-función-variable-nombre
```

Créditos de la imagen

Autoría: toxickore.

Licencia: CC BY 2.0.