

Document Object Model

DOM

heredado o legado



Francisco Arce
www.pacoarce.com

Document Object Model

- document.anchors
- document.body
- document.documentElement
- document.embeds
- document.forms
- document.head
- document.images
- document.links

Document Object Model

- `document.url`
- `document.title`
- `document.cookie`
- `document.write()`

Document Object Model

document.embeds



Francisco Arce
www.pacoarce.com

Document Object Model

`document.embeds`

- Regresa las referencias de las etiquetas `<embed>`, por ejemplo, los elementos Flash-swf.

Document Object Model

`document.anchors`



Francisco Arce
www.pacoarce.com

Document Object Model

`document.anchors`

- Regresa un arreglo con todos los hipervínculos o “anchors” de las etiquetas `<a>`

```
document.anchors[0];  
document.anchors["name"];
```

Document Object Model

- Desactualizado en HTML5, aunque algunos navegadores lo pueden seguir soportando por compatibilidad.

Document Object Model

`document.body`



Francisco Arce
www.pacoarce.com

Document Object Model

`document.body`

- Regresa la referencia del objeto `<body>`
- Puedes sobrescribir los elemento dentro de `<body>`

`document.documentElement`

- Regresa el elemento `<html>`
- Es soportado por todos los navegadores.

Francisco Arce
www.pacoarce.com

Document Object Model

`document.documentElement`



Francisco Arce
www.pacoarce.com

Document Object Model

`document.documentElement`

- Regresa el elemento `<html>`
- Este elemento no se puede modificar
- Si el documento no tiene la etiqueta `<html>` regresa un valor nulo.

Document Object Model

document.forms



Francisco Arce
www.pacoarce.com

Document Object Model

`document.forms`

- Regresa las referencias de las etiquetas `<form>` en un arreglo.

Acepta:

```
document.form[0];  
document.form["name"];  
documento.forma.elementoForma;
```

Francisco Arce
www.pacoarce.com

Document Object Model

`document.head`



Francisco Arce
www.pacoarce.com

Document Object Model

`document.head`

- Regresa el elemento `<head>`
- Si existe más de una etiqueta, regresará la primera.

Document Object Model

`document.images`



Francisco Arce
www.pacoarce.com

Document Object Model

`document.images`

- Regresa un arreglo con las referencias a las etiquetas ``.
- El orden del arreglo es como aparecen las imágenes en el documento HTML.
- El arreglo no registra a las etiquetas `<input>` con atributo `type="image"`.
- Tiene la propiedad `"length"` y el método `item()`

Document Object Model

document.links



Francisco Arce
www.pacoarce.com

Document Object Model

`document.links`

- Se obtiene las direcciones de las etiquetas `<a>` y `<area>` de los atributos href.
- Si las etiquetas no tienen el atributo href, no se obtiene nada.
- Las referencias aparecen en el arreglo según aparezcan las etiquetas en el código fuente.
- Posee la propiedad `length` y el método `item()`

Francisco Arce
www.pacoarce.com

Document Object Model

document.url



Francisco Arce
www.pacoarce.com

Document Object Model

`document.URL`

- Regresa la dirección URL de la página.
- Es similar a la propiedad `location.href`

Document Object Model

document.title



Francisco Arce
www.pacoarce.com

Document Object Model

`document.title`

- Obtiene el contenido de la etiqueta `<title>`
- También podemos modificar el título.

Document Object Model

document.cookie



Francisco Arce
www.pacoarce.com

Document Object Model

`document.cookie`

- Permite grabar información en el disco duro del cliente.
- La información se almacena como *nombre=valor*.
- Los valores son persistentes.
- Podemos crear, modificar y borrar las cookies con JS.

Document Object Model

- Podemos crear una cookie por medio de:

```
document.cookie = "usuario=Pedro Infante; expires=Thu, 18  
Dec 2013 12:00:00 UTC";
```

Document Object Model

- Podemos leer una cookie:

```
var x = document.cookie;
```

- El formato sería una cadena similar a:

```
cookie1=valor; cookie2=valor; cookie3=valor;
```

Document Object Model

- Podemos modificar una **cookie** de la misma forma en que las creamos:

```
document.cookie = "nombre=Pedro Infante; expires=Thu, 18  
Dec 2016 12:00:00 UTC; path=/";
```

Document Object Model

- Para borrar una **cookie**, se le da una fecha de vigencia pasada:

```
document.cookie = "usuario=; expires=Thu, 01 Jan 1970  
00:00:00 UTC";
```

Document Object Model

`document.write()`



Francisco Arce
www.pacoarce.com

Document Object Model

```
document.write();
```

- El método write() escribe desde el código JavaScript al documento HTML.
- Si el documento ha terminado de ser escrito y utiliza el método write(), el contenido de la página será borrado.
- Puede aceptar varios parámetros, separados por comas.