

MIND HUB.

DOM



DOCUMENT OBJECT MODEL

El **Modelo de Objetos del Documento (DOM)** es una interfaz de programación de aplicaciones (API) que proporciona un conjunto estándar de objetos para representar documentos **HTML** y **XML**, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos.

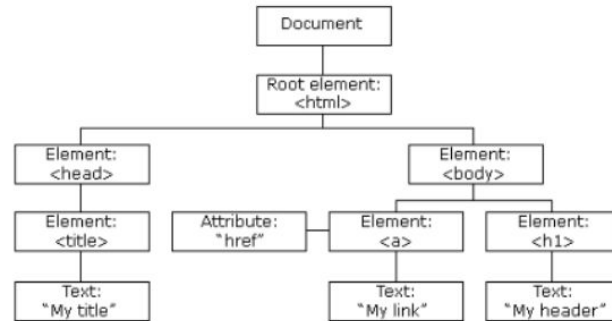
A través del DOM, los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML

DOCUMENT OBJECT MODEL

Cuando una página web está cargada, el navegador crea un Modelo de Objetos del Documento de la página.

EL **DOM** del HTML es construido como un árbol de Objetos:

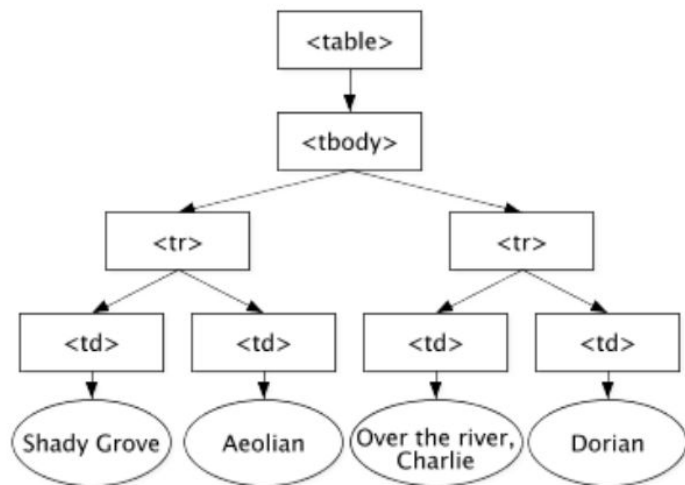
The HTML DOM Tree of Objects



DOCUMENT OBJECT MODEL

Donde cada elemento del HTML es representado por un nodo, mientras que las flechas representan las relaciones entre cada uno de estos.

```
<TABLE>
<TBODY>
<TR>
<TD>Shady Grove</TD>
<TD>Aeolian</TD>
</TR>
<TR>
<TD>Over the River, Charlie</TD>
<TD>Dorian</TD>
</TR>
</TBODY>
</TABLE>
```



DOM & JS

Con el **modelo de objetos**, **JavaScript** puede crear un **HTML** dinámico:

- modificando los elementos **HTML** de la página
- modificando los atributos **HTML** de la página
- modificando los estilos **CSS** de la página
- removiendo los elementos y atributos existentes
- creando nuevos elementos y atributos
- reaccionando a los eventos **HTML** existentes en la página
- creando nuevos eventos **HTML** en la página

DOM & JS

	Método	Descripción
Encontrando elementos HTML	<code>document.getElementById("myId")</code>	busca un elemento por su Id
	<code>document.getElementsByTagName("tagName")</code>	busca elementos por tag
	<code>document.getElementsByClassName("myClass")</code>	busca elementos por clase
Modificando elementos HTML	<code>element.innerHTML = nuevo contenido html</code>	cambia el contenido de un elemento HTML
	<code>element.attribute = nuevo valor</code>	cambia el valor del atributo de un elemento HTML
	<code>element.style.property = nuevo estilo</code>	cambia el estilo de un elemento HTML
Agregando y quitando elementos HTML	<code>document.createElement(element)</code>	crea un elemento HTML
	<code>document.removeChild(element)</code>	quita un elemento HTML
	<code>document.appendChild(element)</code>	agrega un elemento HTML
	<code>document.replaceChild(new, old)</code>	reemplaza un elemento HTML
Añadiendo controladores de Eventos	<code>document.getElementById(id).onclick = function(){code}</code>	añade un evento click a un elemento HTML

¡Muchas gracias!

MIND
HUB.