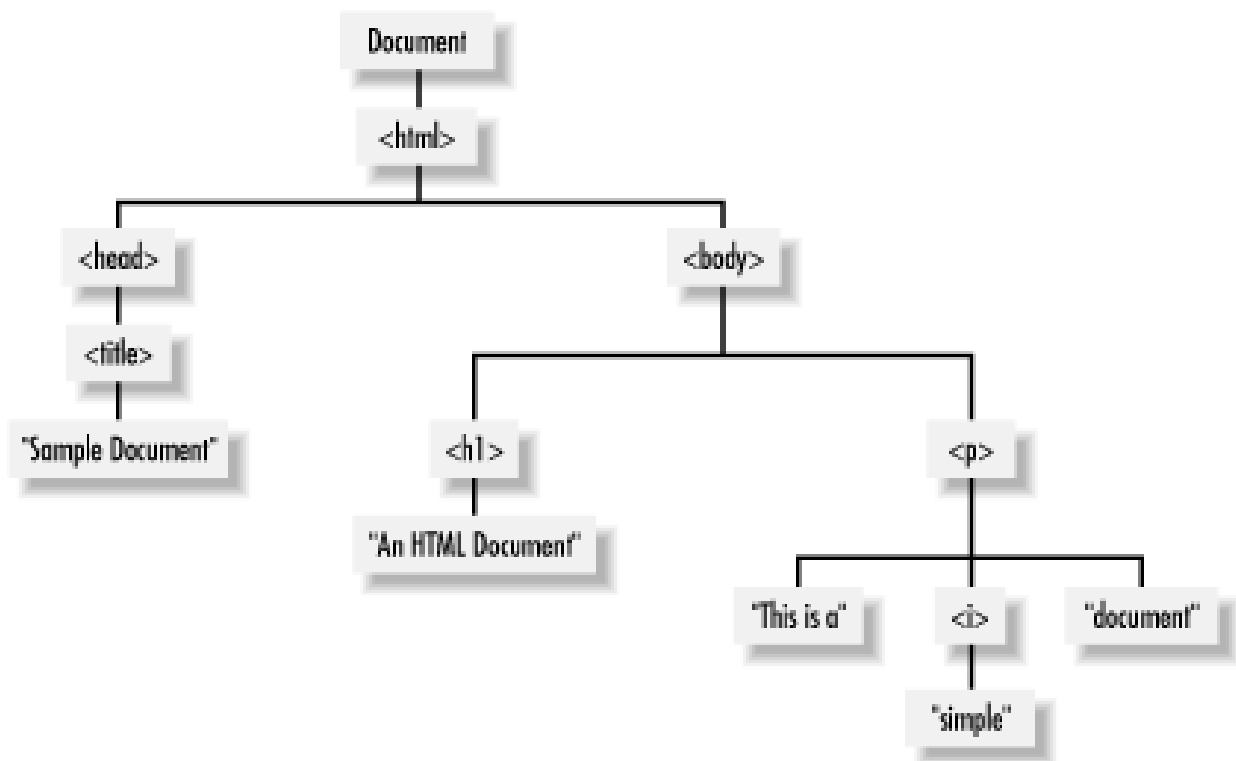


# DOM: Document Object Model

Siempre que entramos al mundo de la interacción con el usuario y el dinamismo web necesariamente tocamos el lenguaje de programación JavaScript que es el que corre por excelencia en todos los navegadores de manera casi estándar, pero para poder crear contenido en JavaScript e integrarlo a nuestra web debemos saber algo denominado el DOM, Document Object Model.

Básicamente el DOM es el documento html denominado por objetos a los cuales podemos acceder por su id o por su clase, cuando hablamos de JavaScript puro hablamos de que podemos acceder de una forma casi estándar en cada navegador pero no es de preocupación el tema pues lo más necesario como el acceder por id y por clase es demasiado estándar.

Para entender mejor qué es el DOM el siguiente gráfico:



Tenemos a document que hará referencia a nuestro documento html y después de este tenemos nuestro documento estructurado como lo vemos en la imagen.

## Acceso al DOM

Podemos acceder a un id casa de la siguiente forma:

```
> document.getElementById('main')
<div id="main">...</div>
```

Por su clase:

```
> document.getElementsByClassName('chapter')
[<div class="chapter">...</div>,<a class="chapter" href=
"dom_obj_document.asp">
    « Previous</a>,<a class="chapter" href="dom_obj_all.asp">Next
Reference »</a>]
```

En este caso nos devuelve un arreglo de accesos al DOM.

## Eventos

También tenemos eventos del mouse y el teclado:

### onkeydown:

```
<script>
    function myFunction() {
        alert("You pressed a key inside the input field");
    }
</script>
<input type="text" onkeydown="myFunction()">
```

En este caso cuando presione una tecla me saldrá el alert.

## onclick:

```
<script>
  function myFunction() {
    document.getElementById("demo").innerHTML="Hello World";
  }
</script>
<button onclick="myFunction()">Click me</button>
<p id="demo"></p>
```

En este caso al darle click al botón desplegará hello world en el p con id demo.

## Otras Alternativas

A muchos les gusta ser hardcode y utilizar JavaScript puro para hacer cosas como estas, pero tenemos frameworks que nos hacen la vida más fácil como los siguientes:

1. [jQuery](#)
2. [MooTools](#)
3. [Prototype](#)

En este curso trabajaremos con el framework que está en pleno auge y que personalmente me gusta mucho para algunas cosas (no para todo).