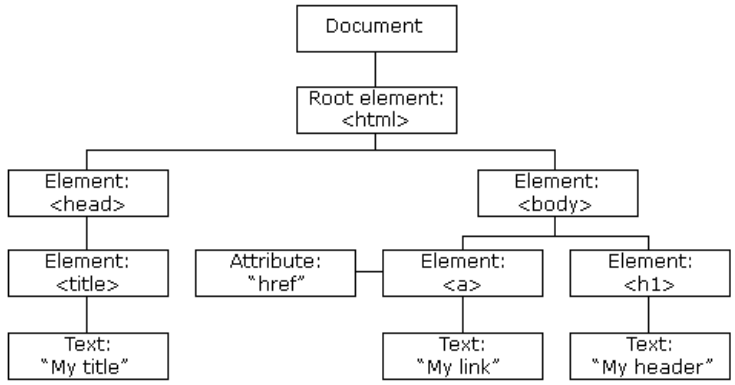
Javascript + HTML

- **DOM**: Es un acrónimo de Document Object Model que en español sería modelo de objetos del documento (HTML), el DOM nos permitirá represetar nuestro documento html en un modelo de objetos de la siguiente forma:



Ese modelo es equivalente a este código HTML:

```
<!DOCTYPE html>
<html>
  <head>
      <title>My title</title>
  </head>
  <body>
      <h1>My header</h1>
      <a href="URL">My link</a>
  </body>
</html>
```

JavaScript y DOM

Para poder manipular nuestro documento **HTML** con el uso de un script en **JS** tendremos que usar el objeto **document** de la siguiente manera:

```
<script>
     document;
</script>
```

Una vez escribamos **document** podremos comenzar a invocar los métodos que el objeto tiene y así efectuar manipulaciones en nuestro documento **HTML**, los métodos son:

- Escribir en el documento: Para escribir en el body de nuestra página usaremos el método **write** del objeto ***document***.

```
 <script>
     document.write('Hola mundo');
 </script>
// en HTML tendremos
<body>
    Hola Mundo
</body>
```

- Crear un elemento: Para crear un nuevo elemento HTML usaremos el método ***createElement***.

```
document.createElement('p');
// Esto creará un elemento html <p>
```

- Obtener un elemento: Para obtener elementos html podremos hacerlo de 3 formas:
  - Obtener elemento por nombre de tag **(getElementsByTagName)**:

```
<body>
  <p>Pedro</p>
  <p>Juan</p>
  <p>Diego</p>
</body>
<script>
  var paragraphs = document.getElementsByTagName('p');
  // paragraphs será un arreglo de 3 elementos en el orden de los nodos p
</script>
```

- Obtener elemento por nombre de id **(getElementById)**:

```
<body>
    <p id="name_paragraph">Pedro</p>
    <p>Juan</p>
    <p>Diego</p>
</body>
<script>
    var paragraph = document.getElementById('name_paragraph');
    // paragraph el nodo html con id 'name_paragraph'
</script>
```

- Obtener elemento por nombre de clase **(getElementsByClassName)**:

```
<body>
    <p class="paragraph">Pedro</p>
    <p class="paragraph">Juan</p>
    <p>Diego</p>
</body>
<script>
    var paragraphs = document.getElementsByClassName('paragraph');
    // paragraphs será un arreglo de 2 elementos en el orden de los nodos p con la clase 'paragraph'
</script>
```

- Añadir elementos hijos: Para añadir un elemento html a uno existente primero deberemos seleccionarlo mediante algún **getElement** y luego usar el método **appendChild**.

```
<body>
    <div id="title"></div>
</body>
<script>
    var titleSection = document.getElementById('title');
    // titleSection tendrá el nodo div con id 'title'
    // Ahora creamos un elemento h1
    var h1 = document.createElement('h1');
    // Creamos nodo de texto
    var text = document.createTextNode('Este es el título de la página');
    h1.appendChild(text);
    // ahora nuestro h1 es así <h1>Este es el...</h1>
    titleSection.appendChild(h1);
</script>
```

- Cambiar el texto del elemento seleccionado: Para ello deberemos seleccionar el elemento y luego cambiar su **NodeText**, esto cambiando la propiedad **innerHTML**:

```
<body>
    <h1 id="title">title</h1>
</body>
<script>
    var title = document.getElementById('title');
    // title tendrá el nodo con id 'title'
    // ahora cambiamos el texto con innerHTML
    title.innerHTML = 'Nuevo title';
</script>
```

- Eliminar elementos: Para eliminar un elemento usaremos el método **removeChild** después de seleccionar el elemento a borrar:

```
<body>
    <h1 id="title">title</h1>
</body>
<script>
    var title = document.getElementById('title');
    // titleSection tendrá el nodo con id 'title'
    // ahora eliminaremos el nodo
    document.removeChild(title);
</script>
```

- Cambiar atributos de un elemento seleccionado: Primero debemos seleccionar el elemento y luego cambiar su propiedad **asignando** el nuevo valor, para ello utilizaremos el método **setAttribute**.

```
<body>
    <h1 id="title">title</h1>
</body>
<script>
    var title = document.getElementById('title');
    // titleSection tendrá el nodo con id 'title'
```

```
    // ahora agregaremos una clase
    title.setAttribue('class', 'title_class');
    // ahora nuestro nodo también tendrá una clase
</script>
```

- Agregar eventos: Javascript podrá agregar escuchadores de eventos que ocurran en nuestro **documento HTML** los cuales son funciones que son invocadas cuando ocurre un evento, por ejemplo podemos agregar escuchadores directamente en nuestro documento html:
  - **click** : Podremos escuchar el click de un boton por ejemplo de la siguiente manera.

    ```
    <button onclick="mostrarAlerta()">Clic aquí!</button>
    <script>
     function mostrarAlerta() {
        window.alert('Hola esto es un mensaje');
     }
    </script>
    ```

  - **change** : Podremos escuchar el cambio de un input por ejemplo de la siguiente manera.

    ```
    <input onchange="mostrarAlerta()"/>
    <script>
     function mostrarAlerta() {
        window.alert('Hola esto es un mensaje');
     }
    </script>
    ```

  - **submit** : Podremos escuchar el submit de un formulario por ejemplo de la siguiente manera.

    ```
    <form onsubmit="mostrarAlerta()">...</form>
    <script>
     function mostrarAlerta() {
        window.alert('Hola esto es un mensaje');
     }
    </script>
    ```

- Cambiar estilos **CSS**: Primero debemos seleccionar el elemento y luego accederemos a su propiedad **style** y para modificar la propiedad con su nuevo valor.
  - Cambio de **color**:

    ```
    <body>
      <h1 id="title">title</h1>
    </body>
    <script>
      var title = document.getElementById('title');
      // title tendrá el nodo con id 'title'
      // ahora cambiamos el texto con el atributo style y la propiedad a cambiar
      // el nuevo valor siempre va entre comillas
      title.style.color = 'blue';
    </script>
    ```

  - Cambio del **tamaño de letra**:

    ```
    <body>
      <h1 id="title">title</h1>
    </body>
    <script>
      var title = document.getElementById('title');
      // title tendrá el nodo con id 'title'
      // ahora cambiamos el texto con el atributo style y la propiedad a cambiar
      // el nuevo valor siempre va entre comillas
      title.style.fontSize = '45px';
    </script>
    ```