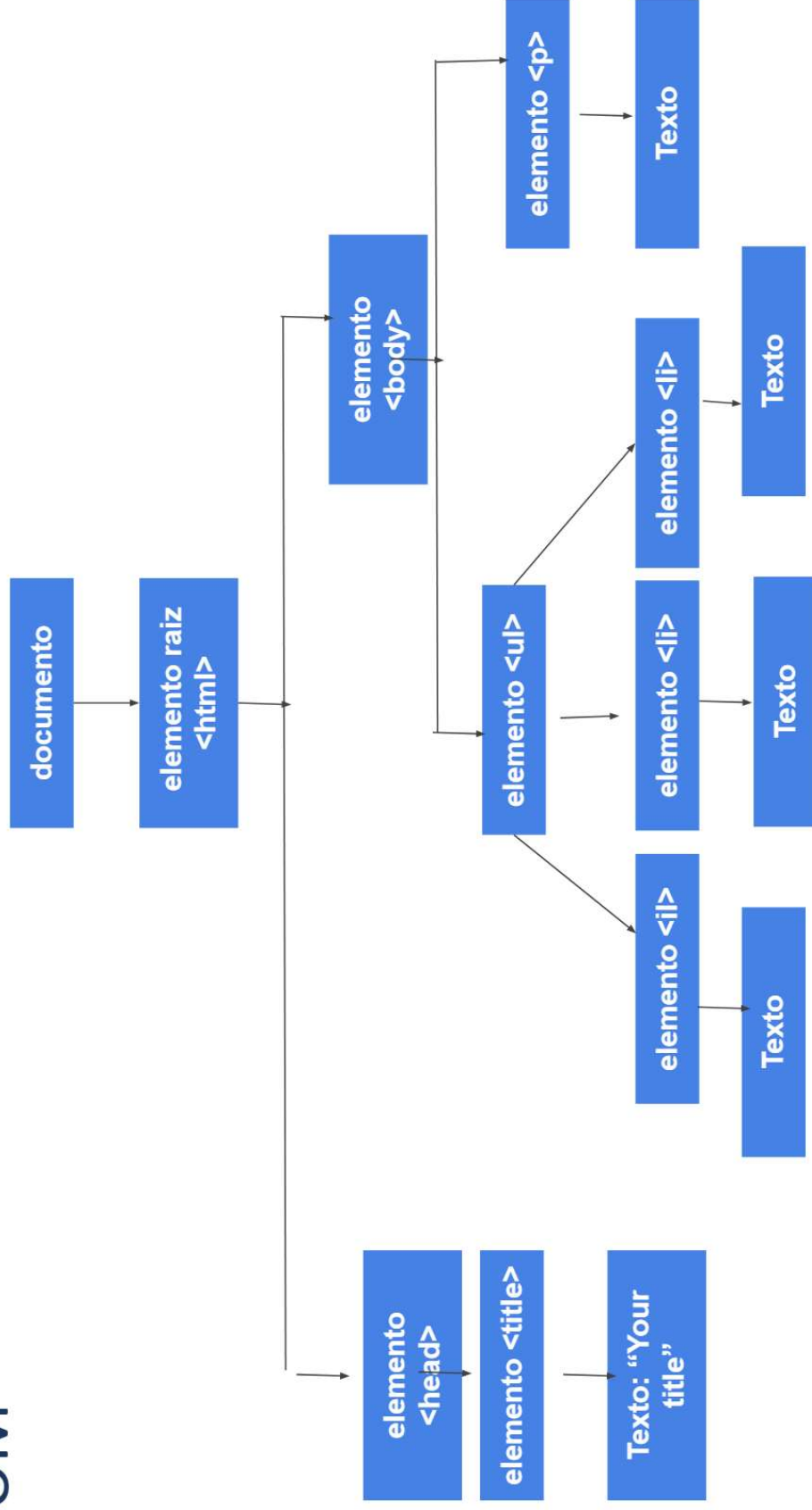


DOM

# DOM



# DOM

DOM significa Document Object Model (Modelo de objeto de documento).

¿Qué es el documento? La página misma:

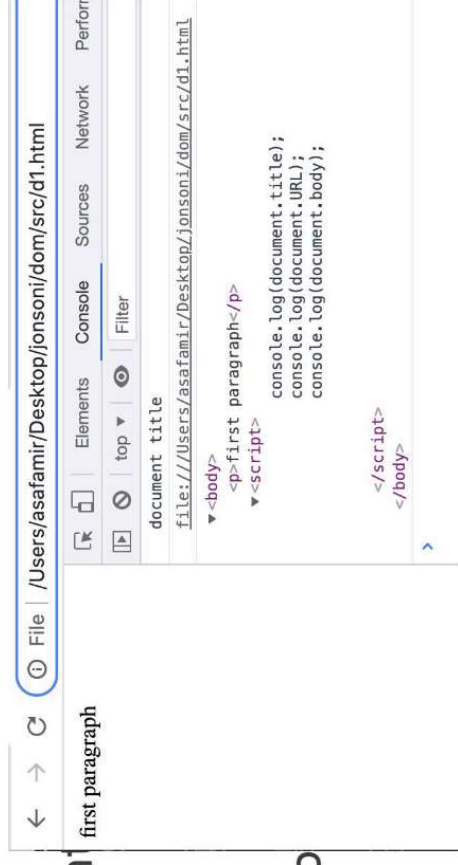
Algunos ejemplos de objetos:

h1 es un objeto, page es un objeto, p es un objeto.

# Documento

Algunos ejemplos de propiedades del documento:

- 1.) `document.title;` // título del documento
- 2.) `document.body;` // El cuerpo del documento
- 3.) `document.URL;` // La url del documento



# Funciones

Las funciones que un objeto tiene:

- 1.) getTitle
- 2.) getLastParagraph
- 3.) getAllElements

Algunas otras funciones importantes:

- 4.) `getElementById("id");`
- 5.) `getElementsByClassName("css-class");`
- 6.) `getElementsByTagName("html-tag");`



# Funciones

## Nombre de la función

`getElementById`

`querySelector`

`querySelectorAll`

`Node.removeChild`

`addEventListener`

`getElementsByTagName`

`parentNode`

## Descripción

El método del [Document](#) `getElementById()` retorna un [Element](#) objeto representado el elemento del cual el la propiedad `id` coincide con el string específico.

El método del [Document](#) `querySelector()` retorna el primer [Elemento](#) en todo el documento que coincide con el selector especificado, o el grupo de selectores. Si no hay coincidencias, retorna null:

The meto del [Document](#) `querySelectorAll()` retorna una [NodeList \(Lista de nodos\)](#) estática (no en tiempo real) representando una lista de los elementos del documento que coinciden con el grupo de selectores:

El método `Node.removeChild()` elimina el child (hijo) del nodo del DOM y retorna el nodo eliminado.

El método [EventTarget](#) `addEventListener()` configura una función que va a ser llamada cada vez que este evento específico llega a su destino.

El método `getElementsByTagName` de la interfaz [Document](#) retorna un [HTMLCollection](#) de elementos con esa tag name (nombre de etiqueta).

`parentNode` es el padre del nodo actual. El padre de un elemento es un nodo de elemento, un nodo del Documento, o un nodo de DocumentFragment (Fragmento de documento).

# Documento

```
<html>
  <head>
    <title>document title</title>
  </head>
  <body>
    <p>first paragraph</p>
  </body>
  <script>
    console.log(document.title);
    console.log(document.URL);
    console.log(document.body);
  </script>
</html>
```

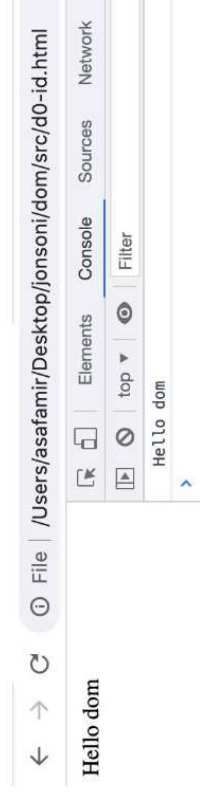


# getElementById

```
var element = document.getElementById("p1");
```

Obtenemos una referencia del elemento y podemos cambiarlo. O acceder a sus “hijos”.

```
<html>
<head>
  <title>document title</title>
</head>
<body>
  <p id="p1">Hello dom</p>
</body>
<script>
  console.log(document.getElementById("p1").innerText);
</script>
</html>
```





# querySelector

```
var element=document.querySelector("#p1");
```

Obtenemos una referencia del elemento y podemos cambiarlo. O acceder a sus "hijos".

```
<html>
<head>
  <title>document title</title>
</head>
<body>
  <p id ="p1">Hello dom - querySelector</p>
</body>
<script>
  console.log(document.querySelector("#p1").innerText);
</script>
</html>
```



# querySelector **class**

```
var element = document.querySelector(".p1");
```

Obtenemos una referencia del elemento y podemos cambiarlo. O acceder a sus "hijos".

```
<html>
<head>
  <title>document title</title>
</head>
<body>
  <p class="p1">
    Hello dom - querySelector class
  </p>
</body>
<script>
  console.log(document.querySelector(".p1").innerText);
</script>
</html>
```



# Primer Elemento / Todos los Elementos

```
<html>
<head>
  <title>document title</title>
</head>
<body>
  <div class="main-nav">
    <a href="#a1">link a1</a>
    <a href="#a2">link a2</a>
    <a href="#a3">link a3</a>
  </div>
  <div class="post-content">
    <p>first paragraph</p>
    <p>second paragraph</p>
    <p>third paragraph</p>
  </div>
</body>
</script>
console.log(document.querySelector(".main-nav a")); // solo el primer elemento
var arr = document.querySelectorAll(".post-content p"); // arreglo de todos los elementos
console.log(arr[0]); // primer párrafo
</script>
</html>
```



# Primer elemento que coincide con el selector especificado

Algunos ejemplos de obtener el primer elemento que coincide con el selector.

- 1.) `document.querySelector(".main-nav a");` // obtiene el primer elemento
- 2.) `document.querySelectorAll(".post-content p");` // obtiene todos los párrafos

# getElementById / getElementsByTagName

```
<html>
<body>
  <ul id="navlinks">
    <li>first</li>
    <li>second</li>
  </ul>
</body>
<script>
  var element = document.getElementById("navlinks");
  var myLiItems = document.getElementsByTagName("li"); // retorna un arreglo
  console.log("node type:", element.nodeType); // 1
  console.log("innerHTML is", myLiItems.innerHTML); // undefined
  console.log("we have:", myLiItems.length); // 2
  console.log("we have:", myLiItems[0].innerText); // primer
  console.log("we have:", myLiItems[1].innerText); // segundo
</script>
</html>
```

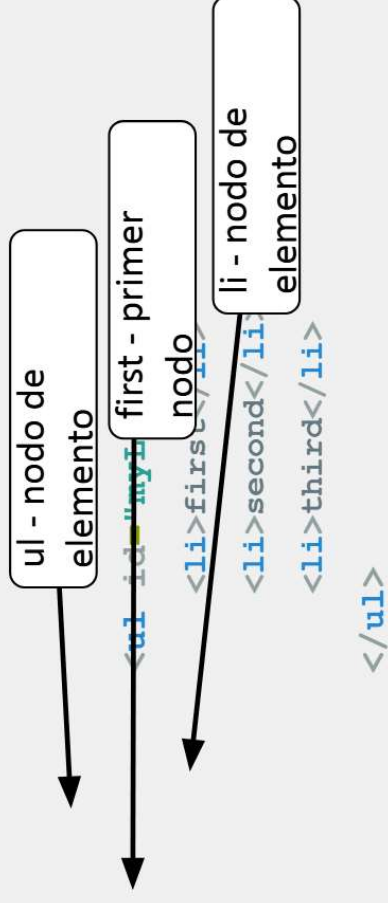




## getElementByTagName

```
var myLiItems = document.getElementsByTagName("li"); // retorna un
arreglo
console.log("node type:", element.nodeType); // 1
console.log("innerHTML is", myLiItems.innerHTML); // undefined
console.log("we have:", myLiItems.length); // 2
console.log("we have:", myLiItems[0].innerText); // primer
console.log("we have:", myLiItems[1].innerText); // segundo
```

# Nodo y Elementos



# Crea un elemento DOM

```
<html>
  <body>
    <ul id="my-list"></ul>
  </body>
</html>

<script>
  let newLi = document.createElement("li");
  newLi.innerText = "This is my new text";
  let element = document.getElementById("my-list");
  element.appendChild(newLi);
</script>
</html>
```



Crea el Element

Agrega el elemento al documento

## Agrega una Clase/ Remueve una clase

```
<html>
  <head>
    <title>add class</title>
    <style>
      .hidden { display: none; }
    </style>
  </head>
  <body>
    <button onclick="hide()">hide
    <button onclick="show()">show
    <p class="p1">Hi, 🐼 </p>
  </script>
  <script>
    function show() {
      document.querySelector(".p
    }

    function hide() {
      document.querySelector(".p
    }
  </script>
</body>
</html>
```



# Hazlo tú mismo 1

Dada la siguiente página de html. Agrega a la página una función llamada **changeSecondItemToRed** que cambie el color del segundo ítem en la lista a rojo.

```
<html>
<body>
<h1>Change second to red 🤖</p>
<button onclick="changeSecondItemToRed()">Change second to red</button>
<ul id="friends-list">
  <li>hi 🤗</li>
  <li>by 🤖</li>
  <li>dy 🤖</li>
</ul>
<script>
  function changeSecondItemToRed() {
    // your implementation
  }
</script>
</body>
</html>
```

(Pista: lee acerca `.style` y resuelvelo)






# Hazlo tú mismo 2

Dada la siguiente página de html. Agrega a la página una función llamada `addFriend` que agregue a la lista (`ul`) cuando el usuario cuando el usuario lo introduzca

```
<html>
```

```
<body>
```

```
<h1>Please add friends 
```

```
<input type="text" id="friend" placeholder="add friend">
```

```
<button onclick="addFriend()">Add friend</button>
```

```
<ul id="myFriends"></ul>
```

```
</body>
```

```
<script>
```

```
function addFriend() {
```

```
}
```

```
</script>
```

```
</html>
```

Please add friends 

- Asaf
- Mike
- David

# Remueve ítems del DOM

```

<html>
<body>
  <h1>Remove All Items 🤖</h1>
  <button onclick="removeAllItems()">Remove All Items </button>
  <ul id="items">
    <li>100 😊</li>
    <li>30 🤡</li>
    <li>400 🤖</li>
  </ul>
</body>
</html>
<script>
  function removeAllItems () {
    var list = document.querySelector('#items li');

    for (var i = 0; li = list[i]; i++) { // puedes usar length
      li.parentNode.removeChild(li);
    }
  }
</script>
</html>

```

Remove All Items 🤖

Remove All Items


- 100 😊
- 30 🤡
- 400 🤖

Remove All Items 🤖

Remove All Items

## Hazlo tú mismo 3

Dada la siguiente página HTML. Agrega a la página una función llamada `removeFriend`. La función va a remover los ítems de la lista de acuerdo con lo que introduzca.

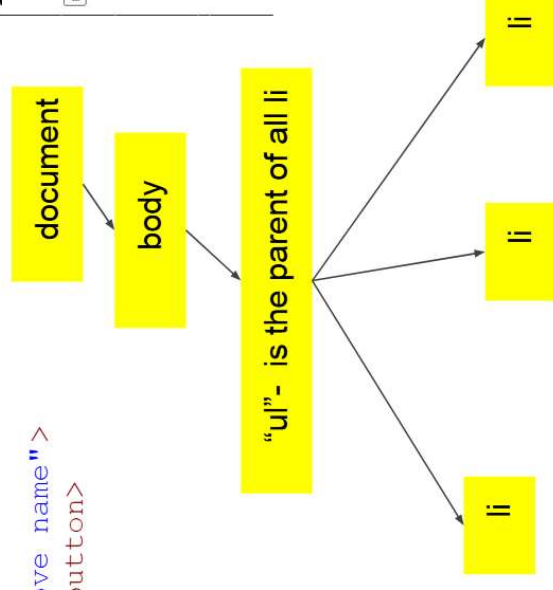
```
<html>
  <body>
    <h1>Remove Name 

```

**Remove Name** 



- Mike
- David
- Suzan



A nighttime photograph of a city skyline. In the foreground, a large, curved skyscraper with a grid-like facade is illuminated with blue and white lights. Behind it, another tall building with a similar grid pattern is visible. To the right, a street with traffic lights and other buildings is shown. The sky is dark, and the city lights create a vibrant scene.

**wawiwa**

¿Preguntas?