

**UNIVERSIDAD LAICA ELOY ALFARO DE MANABI  
FACULTAD DE CIENCIAS INFORMATICAS  
CARRERA TECNOLOGIA DE LA INFORMACION**

**TALLER DE GRUPO 7**

**TEMA:**  
DOM

**NOMBRE:**  
MACIAS PICO JOSSELYN STEFANY

**CURSO:**  
QUINTO "A"

**ASIGNATURA:**  
APLICACIONES WEB I

**DOCENTE:**  
ING. PATRICIA ALEXANDRA QUIROZ PALMA

**PERIODO:**  
2020(2)

**MANTA-MANABI-ECUADOR**

**CODIGO HTML**

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet" href="Nodos.css">
  <title>Ejercicio del grupo 7</title>
  <script src="Nodos.js"></script>
</head>
<body>
  <h1 align="center" >¿Qué es Dom?</h1>
  <p>Para poder utilizar las utilidades de DOM, es necesario "transformar" la página original. Una página HTML normal no es más que una sucesión de caracteres, por lo que es un formato muy difícil de manipular. Por ello, los navegadores web transforman automáticamente todas las páginas web en una estructura más eficiente de manipular.Árbol de Nodos.</p>
  <h3>Árbol de Nodos</h3>
  <div>La creación del Document Object Model o DOM es una de las innovaciones que más ha influido en el desarrollo de las páginas web dinámicas y de las aplicaciones web más complejas.</div>
  <p>DE CLIC EN EL BOTON PARA MOSTAR LSO NODOS  </p>
  <center>
    <button onclick="myFunction();">CANTIDAD DE NODOS</button>
  </center>
</body>
</html>
```

**CODIGO CSS**

```
h1{
  font-family: fantasy;
  color: rgb(0, 128, 0);
  font-size: 250%
}
body{
  background-image: url(imagen/imagen1.jpg);
  background-position:center;
  text-align: justify;
}

h3{
  font-variant: small-caps;
  color: #FD532E;
  font-size: 150%
}
```

```
div{
    font-size: 150%
}

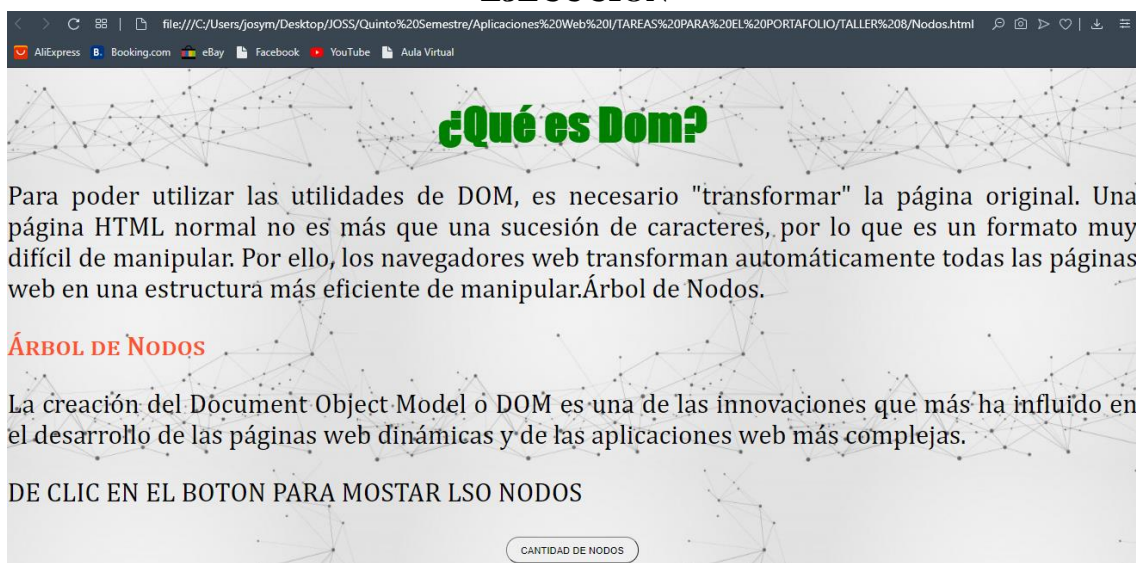
p{
    font-size: 150%
}

button{
    top: 5px;
    left: 5px;
    width:193px;
    height:38px;
    margin:0px auto;
    -webkit-border-radius:20px;
    -moz-border-radius:20px;
    border-radius:20px;
    cursor:pointer;
}
```

### CODIGO CSS

```
function myFunction() {
    var etiquetas = document.querySelectorAll(" title, h1,h3,
div, p");
    var a;
    for (a = 1; a < etiquetas.length; a++) {
        alert(a);
    }
}
```

### EJECUCION



file:///C:/Users/josym/Desktop/IOSS/Quinto%20Semestre/Aplicaciones%20Web%20I/TAREAS%20PARA%20EL%20PORTAFOLIO/TALLER%208/Nodos.html

AliExpress Booking.com eBay Facebook YouTube Aula Virtual

## ¿Qué es Dom?

Para poder utilizar las utilidades de DOM, es necesario "transformar" la página original. Una página HTML normal no es más que una sucesión de caracteres, por lo que es un formato muy difícil de manipular. Por ello, los navegadores web transforman automáticamente todas las páginas web en una estructura más eficiente de manipular.Árbol de Nodos.

### ÁRBOL DE NODOS

La creación del Document Object Model o DOM es una de las innovaciones que más ha influido en el desarrollo de las páginas web dinámicas y de las aplicaciones web más complejas.

DE CLIC EN EL BOTON PARA MOSTAR LSO NODOS

CANTIDAD DE NODOS

Quinto%20Semestre/Aplicaciones%20Web%20I/TAREAS%20PARA%20EL%20PO

**Esta página dice**

1

Aceptar

Quinto%20Semestre/Aplicaciones%20Web%20I/TAREAS%20PARA%20EL%20PO

**Esta página dice**

2

Aceptar

Quinto%20Semestre/Aplicaciones%20Web%20I/TAREAS%20PARA%20EL%20PO

**Esta página dice**

3

Aceptar

**Esta página dice**

4

Aceptar

Quinto%20Semestre/Aplicaciones%20Web%20I/TAREAS%20PARA%20EL%20PO

**Esta página dice**

5

Aceptar

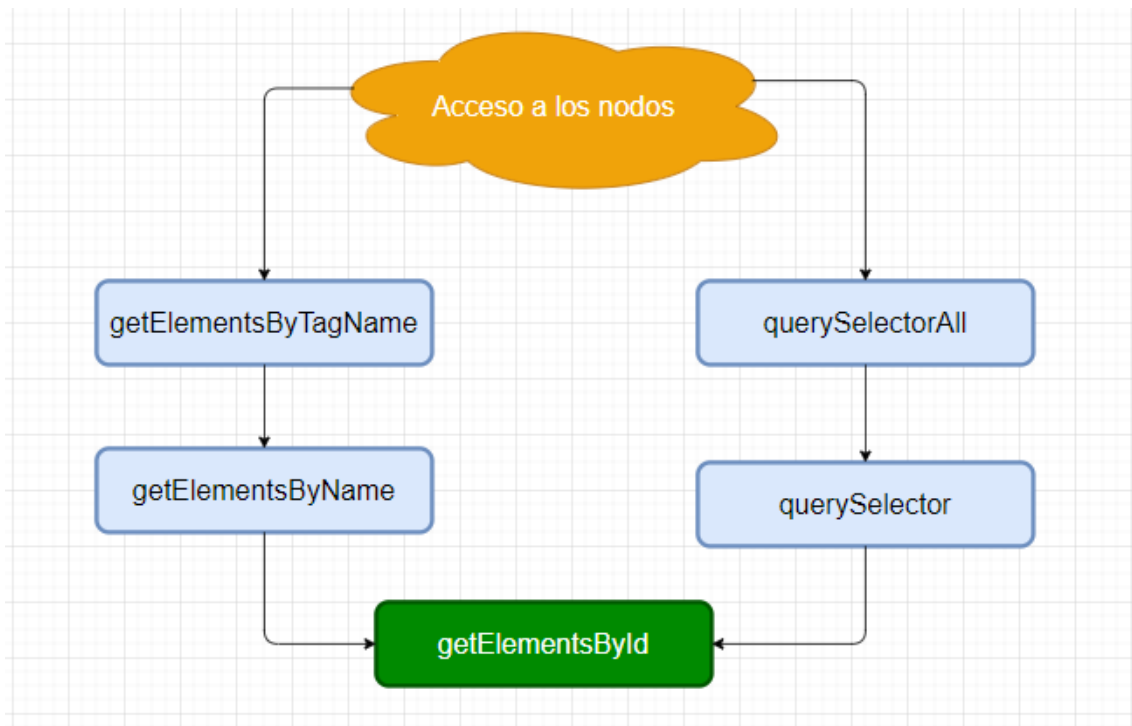
**Realice dos tablas, una con las constantes de la interfaz node y otra con los métodos, indicando los valores que reciben y los que devuelven.**

CONSTANTES	VALOR
Node.ELEMENT_NODE	1
Node.ATTRIBUTE_NODE	2
Node.TEXT_NODE	3
Node.CDATA_SECTION_NODE	4
Node.ENTITY_REFERENCE_NODE	5
Node.ENTITY_NODE	6
Node.PROCESSING_INSTRUCTION_NODE	7
Node.COMMENT_NODE	8
Node.DOCUMENT_NODE	9
Node.DOCUMENT_TYPE_NODE	10
Node.DOCUMENT_FRAGMENT_NODE	11
Node.NOTATION_NODE	12

METODOS	RECIBE	DEVUELVE
<b>addNode</b>	rol de tipo Serie. nodo de tipo Node.	Ninguno.
<b>getNode</b>	rol de tipo Serie	nodo de tipo Node
<b>getAllNodes</b>	Ninguno.	nodes.values de tipo Collection<Node>.
<b>getProperties</b>	Ninguno.	Propiedades de tipo HashMap<String, Object>
<b>convert</b>	converterContext de tipo ConverterContext.	Valor de tipo Nodes
<b>getConvertedTo</b>	Ninguno.	Valor de tipo serie.
<b>getType</b>	Ninguno	Valor de tipo serie
<b>convertWire</b>	targetFlow de tipo MessageFlow. sourceFlow de tipo Flow. inputNodeInSourceFlow de tipo Node. sourcePrimitive de tipo Node. primitiveToNodeCollection de tipo HashMap<Node, Nodes>.	Ninguno.
<b>setConversionContext</b>	contexto de tipo ConversionContext	Ninguno
<b>getInputTerminal</b>	sourceTerminalName de tipo serie. nodos de tipo Nodes.	Terminal de tipo com.ibm.broker.config.app dev.Terminal.
<b>getOutputTerminal</b>	sourceTerminalName de tipo serie. nodos de tipo Nodes.	Terminal de tipo com.ibm.broker.config.app dev.Terminal.
<b>convert</b>	converterContext de tipo ConverterContext.	Valor de tipo Nodes.

<b>getConvertedTo</b>	Ninguno.	Valor de tipo serie.
<b>getDisplayName</b>	Ninguno.	Valor de tipo serie.
<b>getInputTerminal</b>	sourceTerminalName de tipo serie. nodos de tipo Nodes.	Terminal de tipo com.ibm.broker.config.app dev.Terminal
<b>getOutputTerminal</b>	sourceTerminalName de tipo serie. nodos de tipo Nodes	Terminal de tipo com.ibm.broker.config.app dev.Terminal.

Haga un pequeño esquema sobre cuáles son las distintas formas de acceder a un nodo en la estructura de árbol.



Describa en forma de puntos, cuáles son los pasos para crear y eliminar un nodo tipo element, que tiene a su vez un valor de texto.

#### CREACION

- A la etiqueta hay que pasarla como parámetro
- Utilizamos una variable
- Le damos el nombre a la variable
- Ponemos el método a utilizar que es `document.createElement()`;
- Por último dentro de los paréntesis ponemos el nombre del elemento

#### ELIMINAR

- Para eliminar un nodo de árbol utilizamos una función llamada `removeChild()`;
- Luego declaramos una variable.
- Y utilizamos el método `document.getElementById()`;

- Dentro de los paréntesis vamos a ubicar el id que tenemos en la etiqueta que hemos identificado.
- Y luego vamos a escribir el nombre de la variable utilizada
- Seguido de parentNode
- Y por ultimo el método utilizado removeChild()
- Y entre los paréntesis el nombre de nuestra variable.

**Enumere los pasos para realizar la inserción de un nuevo nodo en el árbol. Para ello hay que tener en cuenta que la página está cargada completamente.**

1. Crear un nodo flotante utilizando una variable
2. Luego utilizamos el método `document.createElement()`;
3. Dentro de esta vamos a poner la etiqueta que estamos usando
4. Vamos a dotar de contenido o atributos al nodo flotante
5. `Nodo.innerHTML = "ddjfdfh";`
6. Insertar el nodo dentro del DOM utilizando
7. El código `document.getElementById()`;
8. Dentro de los paréntesis vamos a ubicar el nombre del id que le hemos dado a nuestra etiqueta en el html
9. `appendChild()`
10. Dentro del paréntesis ubica el nombre de tu variable utilizada.