



# **LẬP TRÌNH JAVASCRIPT**

## **BÀI 5: MÔ HÌNH DOM**

- ❑ DOM là một chuẩn được định nghĩa bởi **W3C** (World Wide Web Consortium) để có thể **truy cập** và **thao tác** với các tài liệu như **html** hay **xml** bằng các **ngôn ngữ lập trình** như Javascript, VB...

*The Document Object Model is a platform- and language-neutral interface that will allow programs and scripts to dynamically access and update the content, structure and style of documents (Định nghĩa bởi W3C)*

## ❑ DOM được chia thành 3 mức

- ❖ Core DOM: Tiêu chuẩn cho bất kỳ tài liệu có cấu trúc nào
- ❖ XML DOM: Tiêu chuẩn cho tài liệu XML
- ❖ HTML DOM: Tiêu chuẩn cho tài liệu HTML

- ❑ HTML DOM định nghĩa các **đối tượng** và **thuộc tính** của tất cả các thành phần HTML và **phương thức** để truy cập đến chúng
- ❑ Hay: **HTML DOM là chuẩn để lấy, thay đổi, thêm, xóa các thành phần HTML**

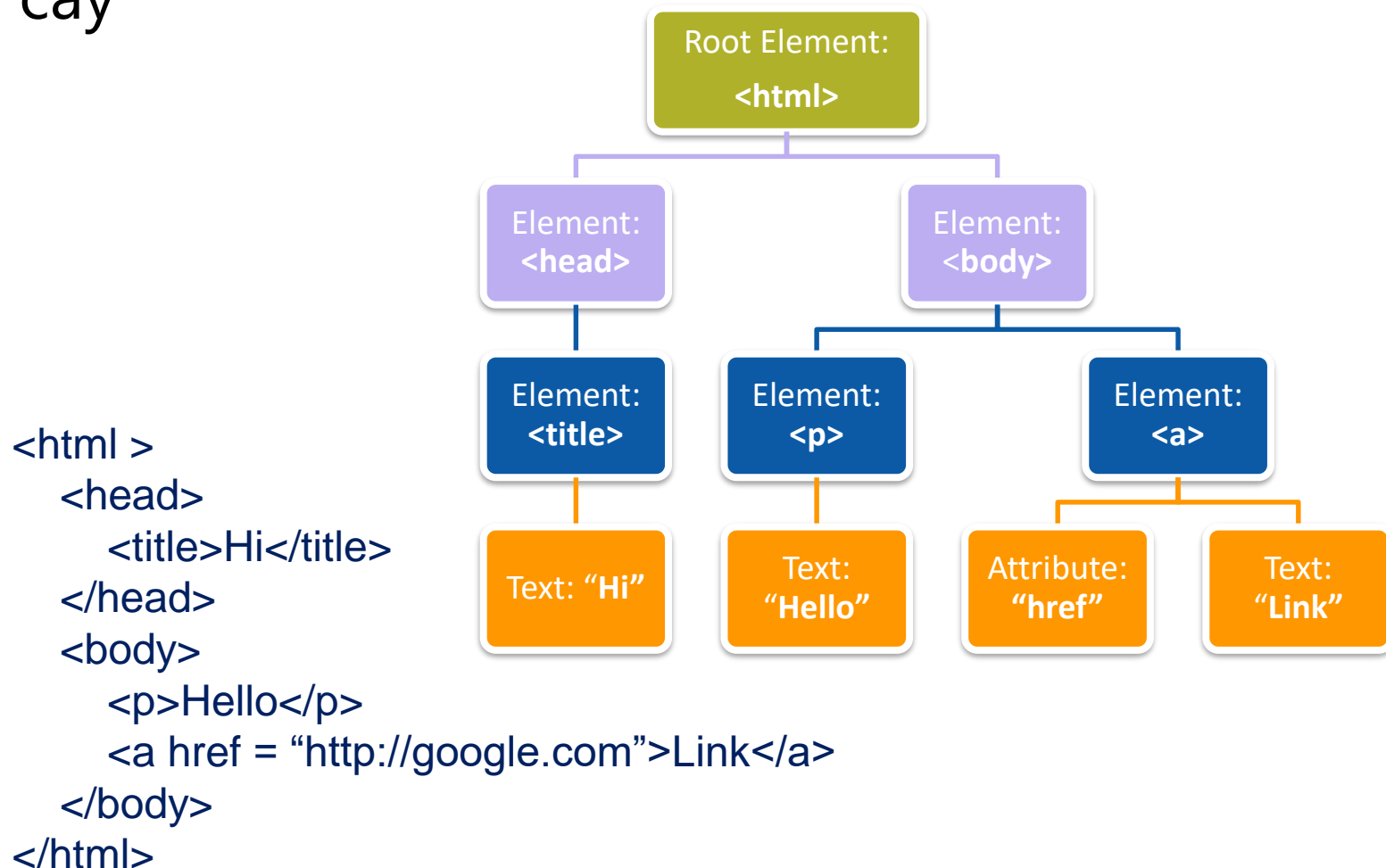
*Truy cập trang web <http://www.w3schools.com/html/dom/default.asp> để tìm hiểu thêm về HTML DOM*

- ❑ Tất cả các thành phần trong tài liệu HTML đều được biểu diễn bằng đối tượng node
  - ❖ Toàn bộ tài liệu là **document node**
  - ❖ Tất cả các thành phần của HTML đều là **element node**
  - ❖ Văn bản trong thành phần HTML là **text node**
  - ❖ Tất cả các thuộc tính là **attribute node**
  - ❖ Chú thích là **comment node**

- ❑ Node gốc là `<html>`
- ❑ `<html>` có hai node con là `<head>` và `<body>`
- ❑ `<head>` có một node con là `<title>`
- ❑ `<body>` chứa hai node con là `<p>` và `<a>`

```
<html >
  <head>
    <title> Hi</title>
  </head>
  <body>
    <p> Hello</p>
    <a href = "http://google.com">Link</a>
  </body>
</html>
```

- DOM trình bày tài liệu HTML theo cấu trúc hình cây

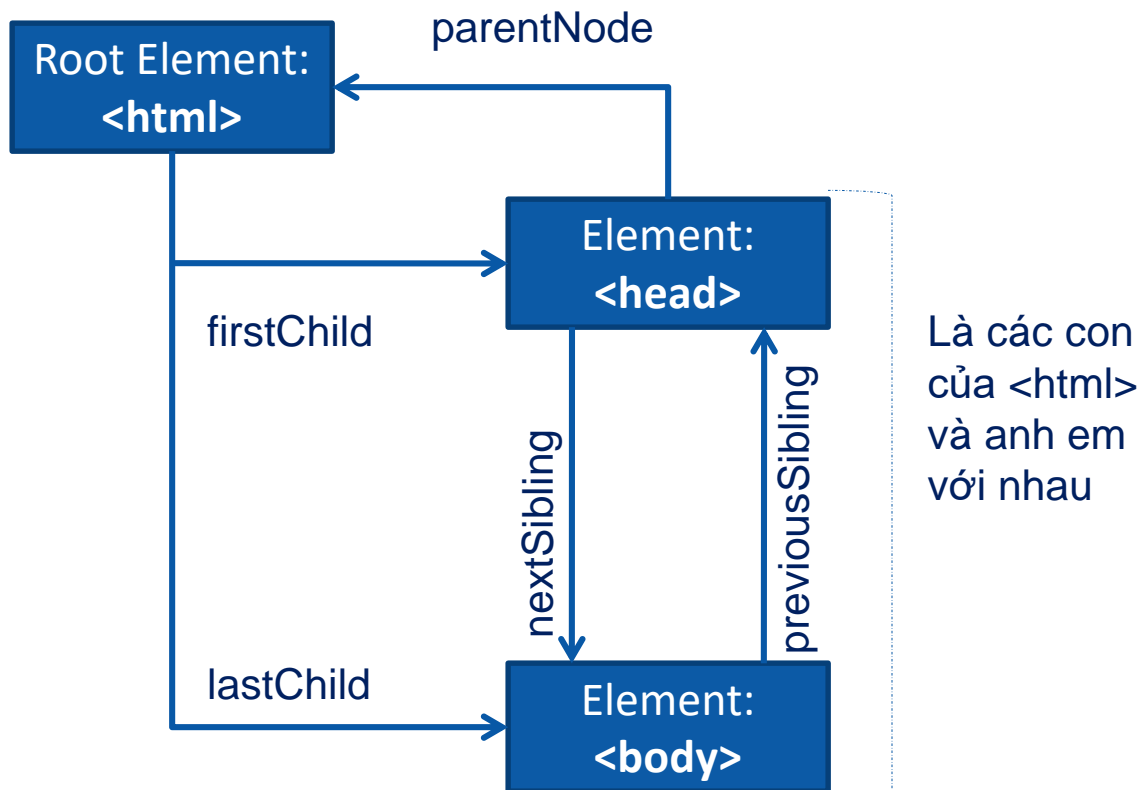


- ❑ Giữa các node có mối quan hệ **cha, con** (parent, children) và **anh em** (siblings)
  - ❖ Node cha có các node con, các node con cùng cấp bậc gọi là anh em
- ❑ Trên cây, node đầu tiên được gọi là **root** (gốc)
- ❑ Tất cả các node, ngoại trừ root **chỉ có một node cha**
- ❑ Một node có thể có **nhiều node con** hoặc **không có** node con nào
- ❑ **Node lá** (leaf) là node **không có node con**
- ❑ Những **node anh em** (siblings) với nhau là node có cùng một node cha



- ❖ `<html>` không có node cha, `<html>` là node gốc
- ❖ `<html>` có hai node con là `<head>` và `<body>`, `<head>` và `<body>` là anh em
- ❖ `<head>` có một node con là `<title>`
- ❖ `<title>` có một node con là text node "hi"
- ❖ `<body>` chứa hai node con là `<p>` và `<a>`, `<p>` và `<a>` là anh em
- ❖ Node cha của node `<head>` và `<body>` là node `<html>`
- ❖ Node cha của text node "Hello" là node `<p>`
- ❖ Text node "Hi", "Hello" , "Link" và attribute node "href" đều là node lá

- ❖ `<head>` và `<body>` là anh em, trong đó `<head>` là con cả của `<html>` còn `<body>` là con út của `<html>`
- ❖ `<p>` và `<a>` là anh em, trong đó `<p>` là con cả của `<body>` còn `<a>` là con út của `<body>`

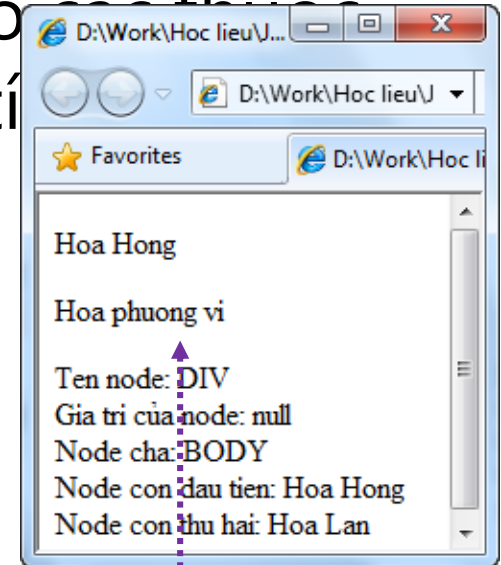


- ❑ **DOM** định nghĩa các **thuộc tính** và các **phương thức** cho các node để hỗ trợ cho việc lập trình
- ❑ **Thuộc tính** định nghĩa các đặc tính cho node
- ❑ **Phương thức** để thực hiện các thao tác với node
  - ❖ Truy cập đến node
  - ❖ Thêm node con cho node
  - ❖ Xóa node con

Thuộc tính	Giải thích
<i><b>x là đối tượng node</b></i>	
x.innerHTML	Giá trị văn bản của x
x.nodeName	Tên của x
x.nodeValue	Giá trị của x
x.nodeType	Kiểu của Node
x.parentNode	Node cha của x
x.childNodes	Các node con của x
x.attributes	Các node thuộc tính của x

- ❑ Có thể xem hoặc thiết lập giá trị cho thuộc tính bằng cách truy cập đến thuộc tính

```
<body>
  <div id = "dHoa">
    <p>Hoa Hong</p>
    <p>Hoa Lan</p>
  </div>
  <script type = "text/javascript">
    var pHoa = document.getElementById("dHoa")
    document.write("Ten node: " + pHoa.nodeName + "<br>");
    document.write("Gia tri của node: " + pHoa.nodeValue + "<br>");
    document.write("Node cha: " + pHoa.parentNode.nodeName + "<br>");
    document.write("Node con dau tien: " + pHoa.firstChild.innerHTML + "<br>");
    document.write("Node con thu hai: " +
pHoa.childNodes[1].childNodes[0].nodeValue + "<br>");
    pHoa.childNodes[1].innerHTML = "Hoa phuong vi";
  </script>
</body>
```



Thay đổi nội dung  
trong thẻ <p>

# GIÁ TRỊ THUỘC TÍNH QUAN TRỌNG

Thuộc tính	Của node	Có giá trị là
nodeName	element	Tên thẻ HTML tương ứng
	attribute	Tên thuộc tính
	text	#text
	document	#document
nodeValue	element	null
	attribute	Giá trị của thuộc tính
	text	Văn bản
nodeType	element	1
	attribute	2
	text	3
	comment	8
	document	9

```
<head>
```

```
  <script type="text/javascript">
```

```
    function hienThi() {
```

```
      var node = document.getElementById("anhDiv");
```

```
      node.innerHTML = "<img src='mu.jpg'>";
```

```
    }
```

```
  </script>
```

```
</head>
```

```
<body>
```

Nhấn vào đây để hiển thị ảnh:

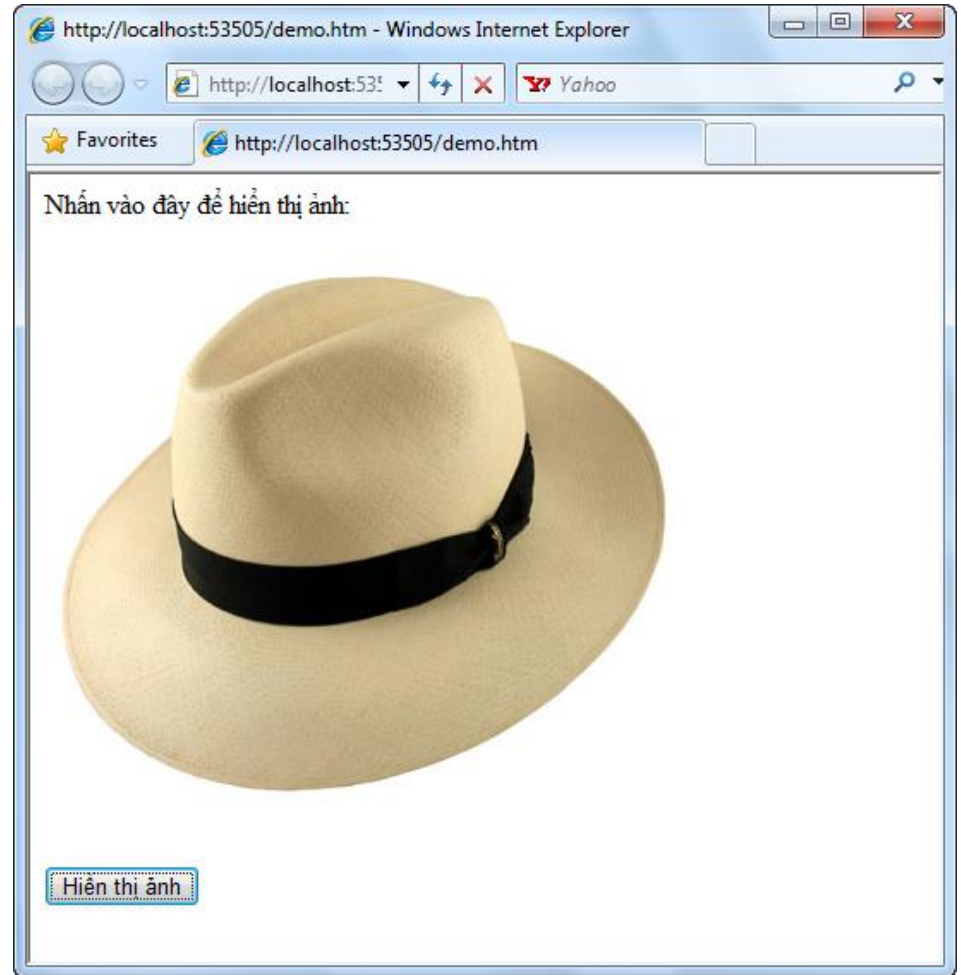
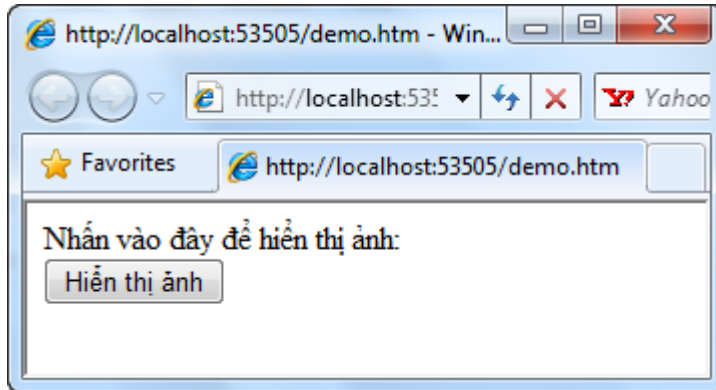
```
<div id="anhDiv"></div>
```

```
<input type="button" value="Hiển thị ảnh"  
onclick="hienThi()"/>
```

```
</body>
```

```
</html>
```

# DEMO VỀ THUỘC TÍNH INNERHTML



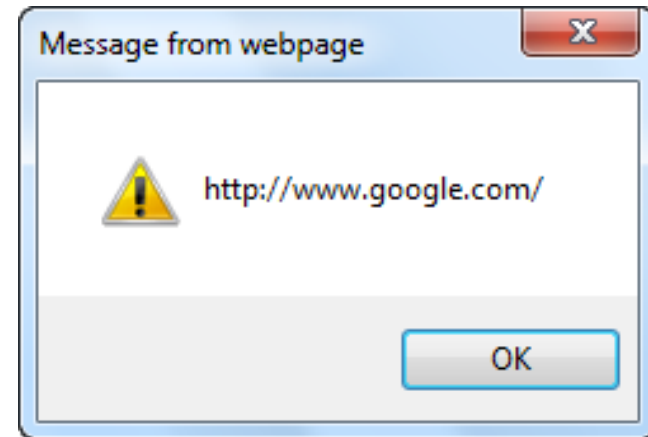


Thuộc tính	Giải thích
<i><b>x là đối tượng node</b></i>	
<code>x.getElementById(<i>id</i>)</code>	Trả về thành phần có id xác định
<code>x.getElementsByTagName(<i>name</i>)</code>	Trả về tất cả các thành phần với tên thẻ xác định
<code>x.appendChild(<i>node</i>)</code>	Thêm node con vào node x
<code>x.removeChild(<i>node</i>)</code>	Xóa node con của node x

- ❑ Có thể truy cập đến các node bằng 3 cách sau:
  - ❖ Sử dụng phương thức **getElementById(id)**
  - ❖ Sử dụng phương thức **getElementsByName(name)**
  - ❖ Sử dụng phương thức **getElementsByTagName(name)**
  - ❖ Sử dụng mối quan hệ giữa các node để **điều hướng**

# DEMO SỬ DỤNG GETELEMENTBYID(ID)

```
<html >
  <head>
    <title>Hi</title>
  </head>
  <body>
    <p>Hello</p>
    <a id = "link" href = "http://www.google.com">Link</a>
    <script type = "text/javascript" >
      var linkNode = document.getElementById("link");
      alert(linkNode.href);
    </script>
  </body>
</html>
```



❑ Giả sử đoạn mã được viết lại như sau

```
<html >
  <head>
    <title>Hi</title>
  </head>
  <body >
    <p>Hello</p>
    <script type = "text/javascript" >
      var linkNode = document.getElementById("link");
      alert(linkNode.href);
    </script>
    <a id = "link" href = "http://www.google.com">Link</a>
  </body>
</html>
```

❑ **Tại sao không hiển thị hộp thoại????**

**Trả lời:** *Tại vì Browser làm việc theo cơ chế thông dịch. Tức là dịch từng dòng một, khi đến lệnh JavaScript `document.getElementById("link")` thì chưa có Id nào tên là "Link" nên không có node nào trả về cho biến `linkNode`*

□ Hãy dự đoán kết quả đoạn mã sau 😊

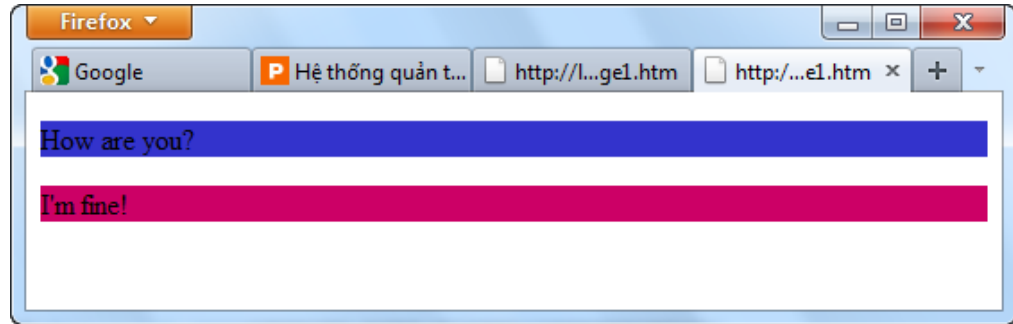
```
<html >
  <head>
    <title>Hi</title>
    <script type ="text/javascript" >
      function checkHref() {
        var linkNode = document.getElementById("link");
        alert(linkNode.href);
      }
    </script>
  </head>
  <body onload = "checkHref()">
    <p>Hello</p>
    <a id = "link" href = "http://www.google.com">Link</a>
  </body>
</html>
```



# **LẬP TRÌNH JAVAScript**

## **BÀI 5: MÔ HÌNH DOM**

```
<html>
<head>
  <title></title>
</head>
<body>
  <p>How are you?</p>
  <p>I'm fine!</p>
  <script type = "text/javascript">
    var p = document.getElementsByTagName("p");
    p[0].style.background = "#3333CC";
    p[1].style.background = "#CC0066";
  </script>
</body>
</html>
```





❑ Mã cho 

```
<html >
```

```
<body>
```

```
  <div id = "hoa">
```

```
    <p>Hoa Hong</p>
```

```
    <p>Hoa Lan</p>
```

```
    <p>Hoa Dao</p>
```

```
  </div>
```

```
  <script type ="text/javascript">
```

```
    var divHoa = document.getElementById("hoa");
```

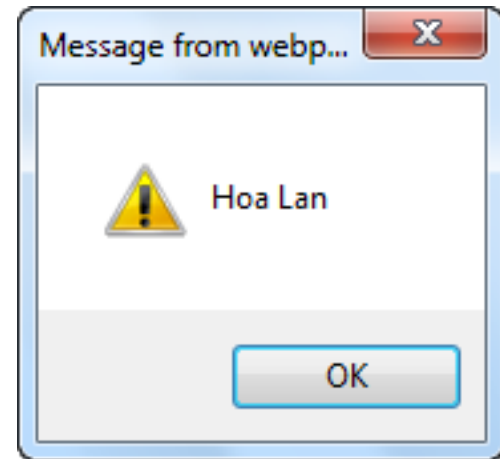
```
    var pHoaHong = divHoa.firstChild;
```

```
    alert(pHoaHong.nextSibling.childNodes[0].nodeValue);
```

```
  </script>
```

```
</body>
```

```
</html>
```





```
<html>
<body>
  <div id = "hoa">
    <p>Hoa Hong</p>
    <p>Hoa Lan</p>
    <p>Hoa Dao</p>
  </div>
  <script type = "text/javascript">
    var divHoa = document.getElementById("hoa");
    var pHoaHong = divHoa.firstElementChild;
    alert(pHoaHong.nextSibling.nextSibling.childNodes[0].nodeValue);
  </script>
</body>
</html>
```

Mỗi Browser định nghĩa  
phương thức riêng để  
truy cập đến các phần tử

Cấu trúc DOM của mỗi  
Browser là khác nhau

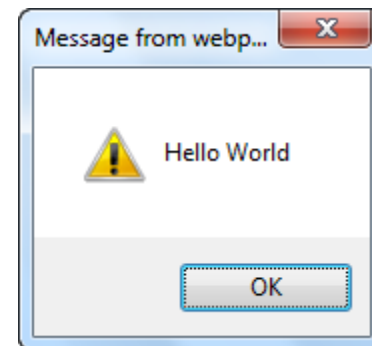
- ❑ Sử dụng phương thức createElement và appendChild để thêm các node element vào tài liệu


```
<html >
<head></head>
<body>
<script type="text/javascript">
    var newElement = document.createElement("p");
    document.body.appendChild(newElement);
    var text = document.createTextNode("Hello World");
    newElement.appendChild(text );
</script>
</body>
</html>
```

- ❑ Dùng phương thức `setAttribute` của node để thiết lập thuộc tính cho chính node đó

```
<head></head>
<body id="bd">
<script type="text/javascript">
    var newElement = document.createElement("p");
    newElement.setAttribute("id", "newElement");
    document.body.appendChild(newElement);
    var text = document.createTextNode("Hello World");
    newElement.appendChild(text);

    var ne = document.getElementById("newElement");
    alert(ne.innerText);
</script></body></html>
```



**Chú ý: Đoạn mã này dành cho** 

- ❑ Sử dụng phương thức `removeChild(nodeId)` của node để xóa các node element của node

```
<html >
<head>
</head>
<body >
    <p id="pHello">Hello</p>
    <p id="pHi">Hi</p>
<script type="text/javascript">

document.body.removeChild(pHi);
</script></body></html>
```

**Đoạn mã này dành cho**



```
<html >
<head>
</head>
<body >
    <p id="pHello">Hello</p>
    <p id="pHi">Hi</p>
<script type="text/javascript">
var pHi = document.getElementById("pHi");
document.body.removeChild(pHi);
</script></body></html>
```

**Đoạn mã này dành cho**



# FPT POLYTECHNIC



KẾT THÚC