

Bài 4

JS – JavaScript

Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Giới thiệu về Javascript

- Script?
 - Ngôn ngữ kịch bản
 - Là một ngôn ngữ lập trình cho phép tạo ra sự điều khiển trong một ứng dụng khác.
 - Các ngôn ngữ kịch bản thường được thông dịch
- JS là ngôn ngữ Client-side script hoạt động trên trình duyệt của người dùng (client)
- Chia sẻ xử lý trong ứng dụng web. Giảm các xử lý không cần thiết trên server.
- Giúp tạo các hiệu ứng, tương tác cho trang web.

Giới thiệu về Javascript

- Client-Side Script:

- Script được thực thi tại Client-Side (trình duyệt): Thực hiện các tương tác với người dùng (tạo menu chuyển động, ...) , kiểm tra dữ liệu nhập, ...

- Server-Side Script:

- Script được xử lý tại Server-Side, nhằm tạo các trang web có khả năng phát sinh nội dung động. Một số xử lý chính: kết nối CSDL, truy cập hệ thống file trên server, phát sinh nội dung html trả về người dùng...

Giới thiệu về Javascript

- Cung cấp sự tương tác với người dùng
 - Sự kiện do người dùng tạo ra: di chuột, click chuột...
 - Sự kiện do hệ thống tạo ra: thay đổi kích thước trang, tải trang...
- Thay đổi động nội dung
 - Thay đổi nội dung và vị trí các thành phần trên trang Web theo sự tương tác của người dùng
- Hợp lệ hóa dữ liệu
 - Kiểm tra xem dữ liệu có hợp lệ không trước khi nó được gửi (submit) đến Web Server để xử lý

Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Nhúng Javascript vào trang web

Định nghĩa script trực tiếp trong trang html:

```
<script type="text/javascript">  
  <!--  
  // Lệnh Javascript  
  -->  
</script>
```

Nhúng sử dụng script cài đặt từ 1 file .js khác:

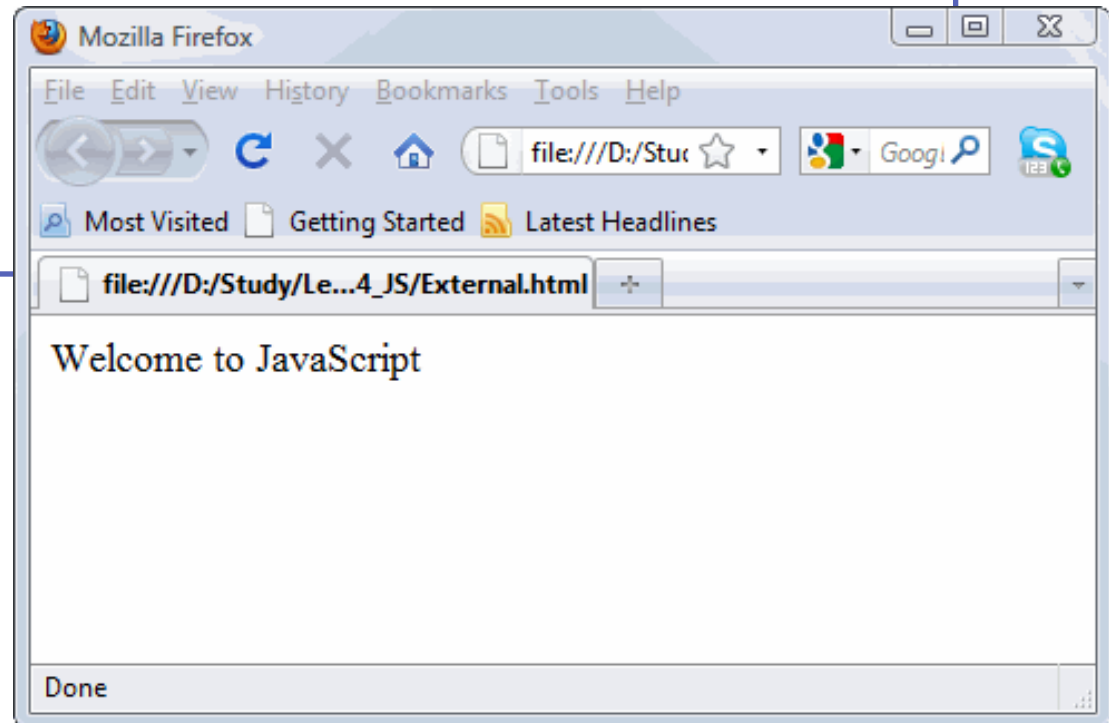
```
<script src="xxx.js"></script>
```

Nhúng Javascript vào trang web

- Đặt giữa tag `<head>` và `</head>`: script sẽ thực thi ngay khi trang web được mở.
- Đặt giữa tag `<body>` và `</body>`: script trong phần body được thực thi khi trang web đang mở (sau khi thực thi các đoạn script có trong phần `<head>`).
- Số lượng đoạn client-script chèn vào trang không hạn chế.
- File JavaScript bên ngoài cho phép sử dụng lại nhiều lần và chỉnh sửa dễ dàng

Nhúng Javascript vào trang web

```
<html>
  <head>
    <script language="javascript">
      document.write("Welcome to JavaScript");
    </script>
  </head>
  <body>
  </body>
</html>
```



Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Các quy tắc chung

- Khối lệnh được bao trong dấu {}
- Mỗi lệnh nên kết thúc bằng dấu ;
- Cách ghi chú thích:
 - // Chú thích 1 dòng
 - /* Chú thích
nhiều dòng */

Biến số trong Javascript

- Cách đặt tên biến

- Bắt đầu bằng một chữ cái hoặc dấu _
- A..Z,a..z,0..9,_ : phân biệt HOA, Thường

- Khai báo biến

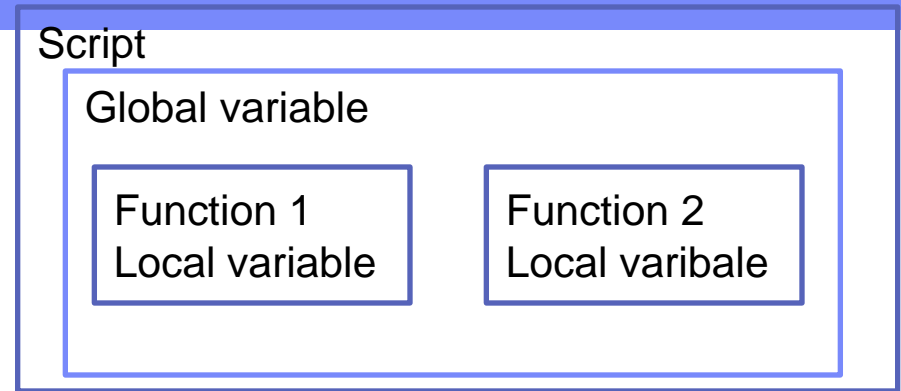
- Sử dụng từ khóa **var**

Ví dụ: **var count=10,amount;**

- Không cần khai báo biến trước khi sử dụng, biến thật sự tồn tại khi bắt đầu sử dụng lần đầu tiên

Biến số trong Javascript

- Phạm vi của biến
 - Được xác định bởi nơi biến được khai báo
- Biến toàn cục
 - Khai báo bên ngoài hàm
 - Được truy cập từ mọi nơi trong kịch bản
 - Khai báo không cần từ khóa var
- Biến cục bộ
 - Khai báo bên trong hàm
 - Phạm vi trong hàm khai báo
 - Bắt buộc phải có từ khóa var



Kiểu dữ liệu trong Javascript

Kiểu dữ liệu	Ví dụ	Mô tả
Object	<code>var listBooks = new Array(10) ;</code>	Trước khi sử dụng, phải cấp phát bằng từ khóa <code>new</code>
String	<code>"The cow jumped over the moon."</code> <code>"40"</code>	Chứa được chuỗi unicode Chuỗi rỗng <code>""</code>
Number	0.066218 12	Theo chuẩn IEEE 754
boolean	true / false	
undefined	<code>var myVariable ;</code>	<code>myVariable = undefined</code>
null	<code>connection.Close();</code>	<code>connection = null</code>

1 Biến trong javascript có thể lưu bất kỳ kiểu dữ liệu nào.

Đổi kiểu dữ liệu

- Biến tự đổi kiểu dữ liệu khi giá trị mà nó lưu trữ thay đổi

Ví dụ:

```
var x = 10;           // x kiểu Number
x = "hello world !"; // x kiểu String
```

- Có thể cộng 2 biến khác kiểu dữ liệu

Ví dụ:

```
var x;
x = "12" + 34.5; // KQ: x = "1234.5"
```

- Hàm `parseInt(...)`, `parseFloat(...)` : Đổi KDL từ chuỗi sang số.
- Chuyển không thành công: "NaN"

Đổi kiểu dữ liệu

```
<script language="javascript">
    var x = 10;
    document.write(x);
    x=true;
    x = "Hello World";
    document.write(x);
    x = "12"+4.3;
    document.write(x);
    x = parseInt("-32");
    document.write(x);
</script>
```


Phép toán số học

Phép toán	Mô tả	Ví dụ
+	Cộng	$A = 5 + 8$
-	Trừ	$A = 8 - 5$
/	Chia	$A = 20 / 5$
%	Số dư	$10 \% 3 = 1$
++	Tăng 1	++ x và x ++
--	Giảm 1	-- x và x --
-	Đảo dấu	nếu $A = 5$ thì $-A = -5$

Phép toán so sánh

Phép toán	Mô tả so sánh	Ví dụ (x=5)	Kết quả
<code>==</code>	Bằng giá trị	<code>x == 5</code>	true
<code>===</code>	Bằng giá trị và kiểu	<code>x === "5"</code>	false
<code>!=</code>	Không bằng	<code>x != 4</code>	true
<code>></code>	Lớn hơn	<code>x > 5</code>	false
<code>>=</code>	Lớn hơn hoặc bằng	<code>x >= 5</code>	true
<code><</code>	Nhỏ hơn	<code>x < 4</code>	false
<code><=</code>	Nhỏ hơn hoặc bằng	<code>x <= 5</code>	true

Phép toán logic

Phép toán	Mô tả	Ví dụ ($x=6, y=3$)	Kết quả
<code>&&</code>	And	<code>(x<10) && (y>1)</code>	true
<code> </code>	Or	<code>(x<1) (y>5)</code>	false
<code>!</code>	Not	<code>!(x==y)</code>	true

Phép toán định lượng

- Phép toán điều kiện

Cú pháp

```
tênbiến = điềukiện ? giátrị1:giátrị2;
```

Ví dụ:

```
status = (diem >= 5) ? “đạt”: “thi lại”;
```

Phép toán định lượng

- `typeof`
 - Trả về chuỗi xác định kiểu dữ liệu của biến

Ví dụ:

```
var x = 5;  
document.write(typeof(x));
```

=> number

Hàm trong Javascript

- Dạng thức khai báo chung:

```
function Tên_hàm(thamso1, thamso2,...)
{
    .....
}
```

- Hàm có giá trị trả về:

```
function Tên_hàm(thamso1, thamso2,...)
{
    .....
    return (value);
}
```

Hàm trong Javascript

- Ví dụ:

```
function Sum(x, y)  
{  
    tong = x + y;  
    return tong;  
}
```

- Gọi hàm:

```
var x = Sum(10, 20);
```

Câu lệnh if

```
if (condition)  
{  
    statement[s] if true  
}  
else  
{  
    statement[s] if false  
}
```

Ví dụ:

```
var x = 5, y = 6, z;
```

```
- if (x == 5) { if (y == 6) z = 17; } else z = 20;
```


Câu lệnh switch

```
switch (expression)
```

```
{  
    case label :  
        statementlist  
    case label :  
        statementlist  
    ...  
    default :  
        statement list  
}
```

Ví dụ :

```
var diem = "G";  
switch (diem) {  
    case "Y":  
        document.write("Yếu");  
        break;  
    case "TB":  
        document.write("Trung bình");  
        break;  
    case "K":  
        document.write("Khá");  
        break;  
    case "G" :  
        document.write("Giỏi");  
        break;  
    default:  
        document.write("Xuất sắc")  
}
```

Vòng lặp for

```
for ([initial expression]; [condition];  
      [update expression]) {  
    statement[s] inside loop  
}
```

Ví dụ:

```
var myarray = new Array();  
for (i = 0; i < 10; i++)  
{  
    myarray[i] = i;  
}
```

Vòng lặp while, do.. while

while (*expression*)

```
{  
    statements  
}
```

Ví dụ:

```
var i = 9, total = 0;  
while (i < 10)  
{  
    total += i * 3 + 5;  
    i = i + 5;  
}
```

do

```
{  
    statement  
} while (expression);
```

Ví dụ:

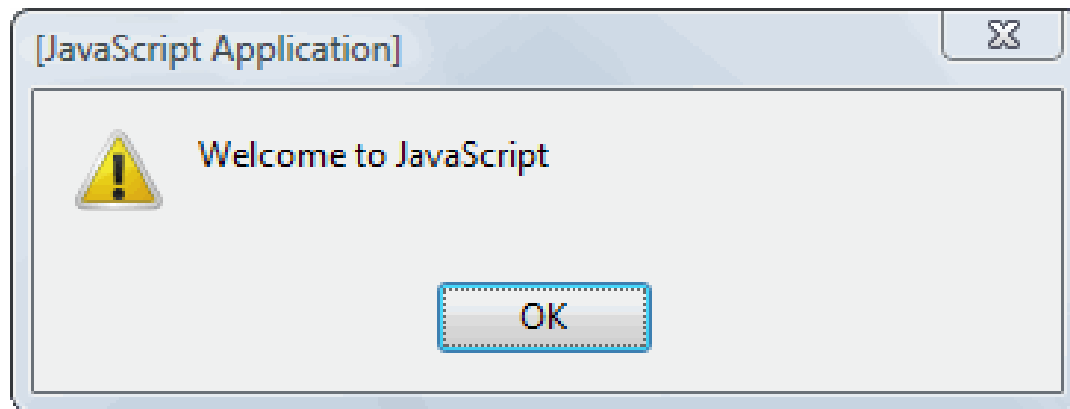
```
var i = 9, total = 0;  
do  
{  
    total += i * 3 + 5;  
    i = i + 5;  
} while (i > 10);
```

Hộp thông báo

- Sử dụng các phương thức: confirm, alert, prompt
- Phương thức alert()

Cú pháp

```
alert ("thông_báo");
```

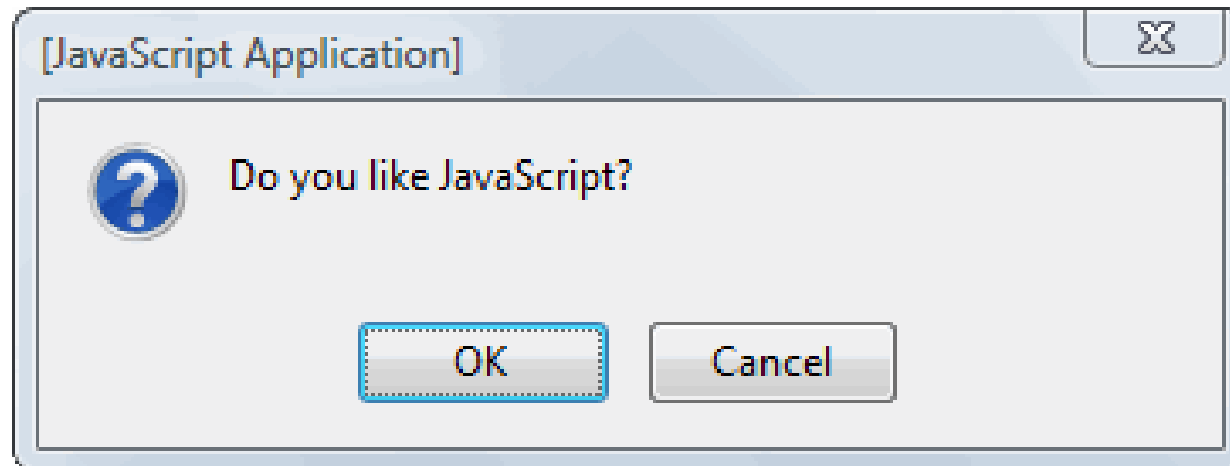


Hộp thông báo

- Phương thức confirm()

Cú pháp

```
ret = confirm ("thông_báo");
```

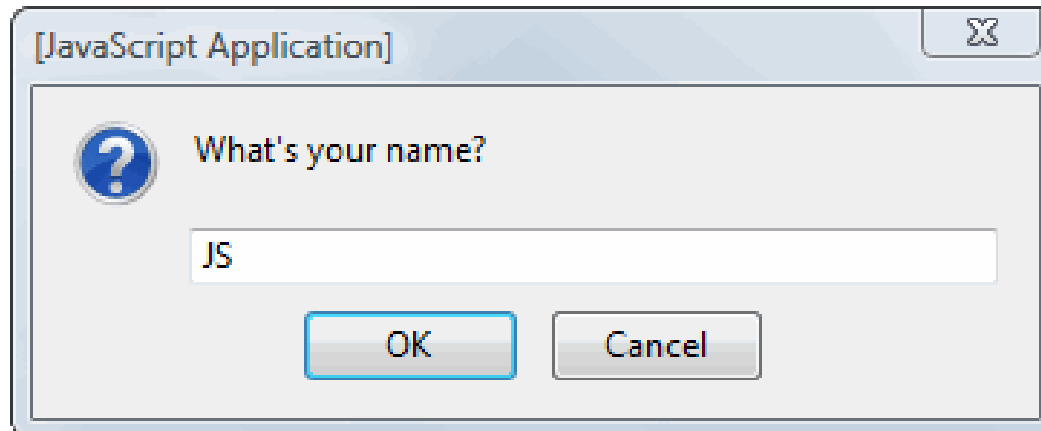


Hộp thông báo

- Phương thức `prompt()`

Cú pháp

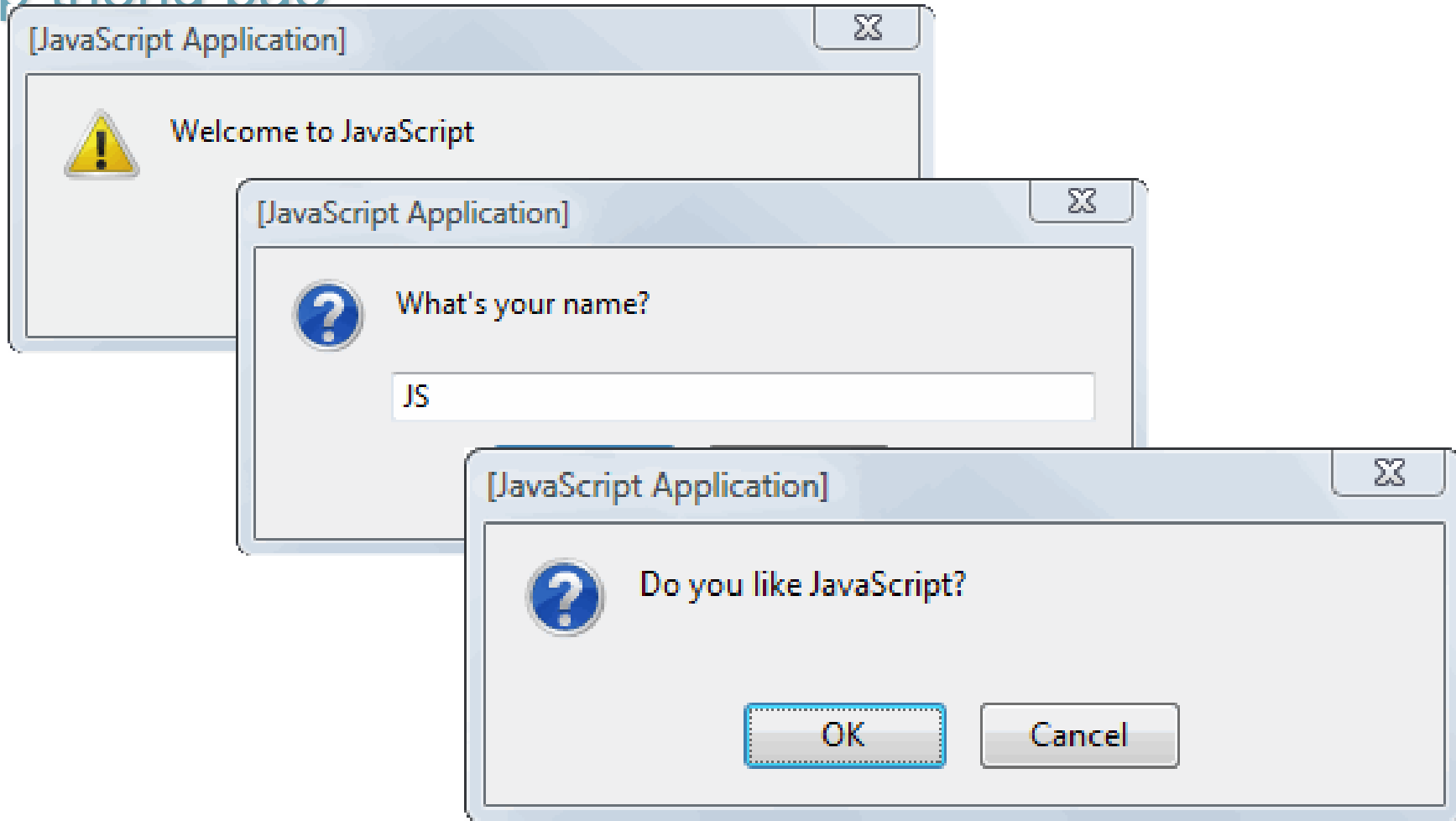
```
value = prompt("thông_báo",  
giá_trị_mặc_định);
```



Hộp thông báo

```
<script language="javascript">
  alert("Welcome to JavaScript");
  var value = prompt("What's your
                    name?", "JS");
  alert("Hello "+value);
  var ret = confirm("Do you like
                   JavaScript?");
  alert("You pressed "+ret);
</script>
```

Hộp thông báo



Viết chú thích

- Viết chú thích trên một dòng: //
- Viết chú thích trên nhiều dòng: /* và */

```
/*Todo: Example in MessageBox  
include: alert(), confirm(),  
prompt()  
*/  
var value = prompt("What's your  
name?", "JS"); //get your name
```

Các kí tự đặc biệt

- Vấn đề:

```
var txt="He is very "intelligence";
```

- Các kí tự đặc biệt trong JavaScript:

```
" ' & \ n r t b f
```

- Để sử dụng: thêm dấu \ phía trước

```
var txt=
```

```
"He is very \"intelligence\"";
```

Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Giới thiệu về sự kiện

- Sự kiện (event)
 - Là các hành động có thể được thể hiện bằng JavaScript
 - Mỗi thành phần trên trang Web đều có những sự kiện xác định, có thể kích hoạt một script
- Ví dụ
 - Sự kiện click chuột
 - Trang web hoặc ảnh được tải
 - Nhập dữ liệu vào một trường trên form

Giới thiệu về sự kiện

- Chú ý:
 - Sự kiện thường được kết hợp với hàm. Hàm sẽ không thực thi chừng nào sự kiện chưa xuất hiện
 - Cú pháp

```
sự_kiện = "tên_hàm ();"
```

hoặc

```
sự_kiện = "tên_hàm (ds_tham_số);"
```

Các sự kiện thông dụng

- Các sự kiện được hỗ trợ bởi hầu hết các đối tượng
 - **onClick**
 - **onFocus**
 - **onChange**
 - **onBlur**
 - **onMouseOver**
 - **onMouseOut**
 - **onMouseDown**
 - **onMouseUp**
 - **onLoad**
 - **onSubmit**
 - **onResize**
 -

Xử lý sự kiện cho các thẻ HTML

- Cú pháp 1:

```
<TAG eventHandler = "JavaScript Code">
```

- Ví dụ:

```
<body>
```

```
<INPUT TYPE="button" NAME="Button1"
```

```
VALUE="OpenSesame!"
```

```
onClick="window.open('mydoc.html');">
```

```
</body>
```

- Lưu ý: Dấu "..." và '...'

Xử lý sự kiện bằng function

```
<head>
```

```
  <script language="Javascript">
```

```
    function GreetingMessage ()
```

```
    {
```

```
        window.alert("Welcome to my world");
```

```
    }
```

```
  </script>
```

```
</head>
```

```
<body onload="GreetingMessage () ">
```

```
</body>
```


Một số sự kiện thông dụng

Sự kiện	Ý nghĩa
onMouseUp	nút chuột được nhả
onMouseDown	ấn chuột
onMouseOver	chuột di chuyển qua vùng
onMouseOut	chuột di chuyển khỏi vùng
onMouseMove	chuột được di chuyển
onclick	click chuột
ondblclick	click đúp chuột
onFocus	Đối tượng được sử dụng (đặt con trỏ)
onBlur	Đối tượng không còn được sử dụng
onChange	Đối tượng được thay đổi (nhập dữ liệu)
onSubmit	Xác nhận tất cả dữ liệu trên form

Ví dụ: onclick Event

```
<SCRIPT LANGUAGE="JavaScript">
```

```
function compute(frm)
```

```
{
```

```
var x = frm.expr.value;
```

```
result.innerHTML = x*x;
```

```
}
```

```
</SCRIPT>
```

```
<FORM name="frm">
```

```
X = <INPUT TYPE="text" NAME="expr" SIZE=15>
```

```
<BR><BR>
```

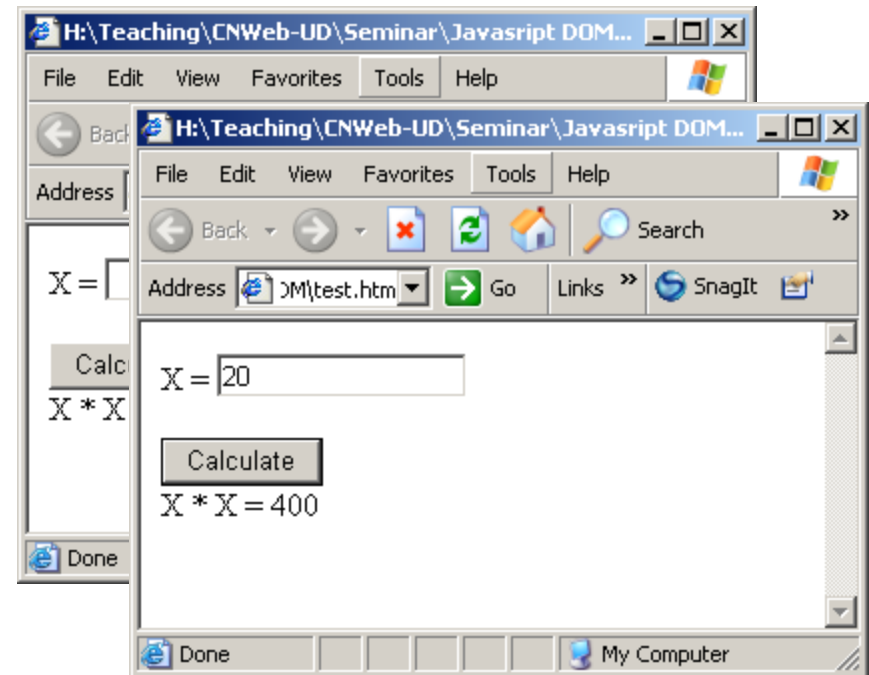
```
<INPUT TYPE="button" VALUE="Calculate"
```

```
ONCLICK="compute(this.form)">
```

```
<BR>
```

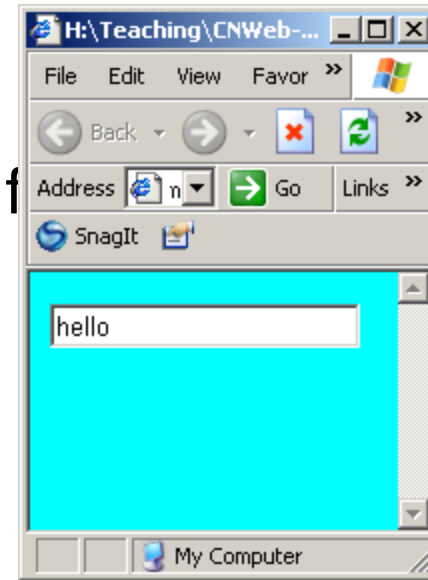
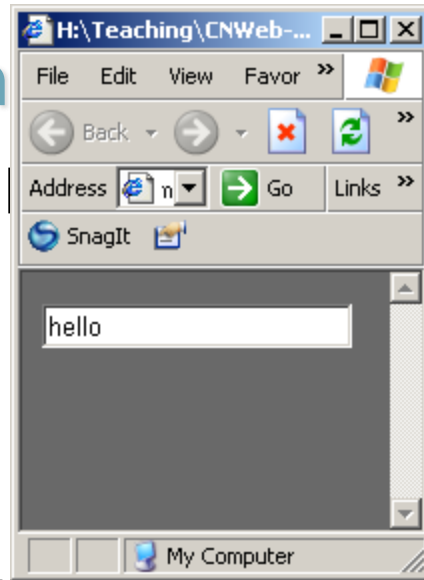
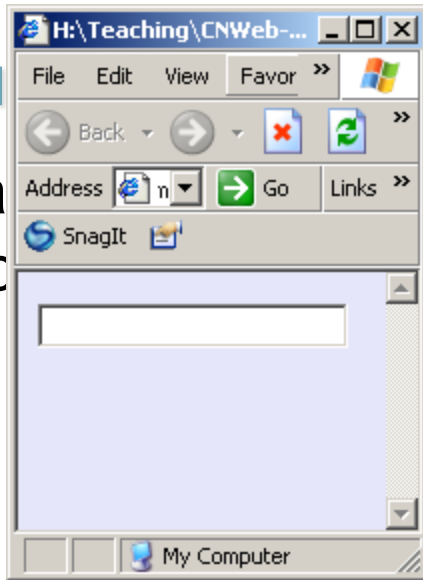
```
X * X = <SPAN ID="result"></SPAN>
```

```
</FORM>
```



Ví dụ

- Xử lý focus
- Ví dụ



) và mất

```
<BODY BGCOLOR="lavender">
```

```
<FORM>
```

```
<INPUT type="text" name="myTextbox"
```

```
onblur="(document.bgColor='aqua')"
```

```
onfocus="(document.bgColor='dimgray')">
```

```
</FORM>
```

```
</BODY>
```

Bài tập 1

- Tạo 2 phần tử: một nút bấm và một ô textbox. Hãy viết đoạn mã JavaScript khi người dùng click chuột vào nút Welcome thì hiển thị nội dung chứa trong textbox.
 - *Hướng dẫn:* Để lấy giá trị của một phần tử HTML: <Tên phần tử>.value
 - Ví dụ: **msg.value** cho ta giá trị của text tên là msg.

Bài tập 1

```
<html>  
  <head> </head>  
  <body>  
    <form name="frm">  
      <input type="button" name="welcome"  
        value="Welcome"  
        onclick="alert(document.frm.msg.value);" />  
      <input type="text" name="msg" value="Welcome to  
        JavaScript" size="30" />  
    </form>  
  </body>  
</html>
```

Bài tập 2

- Tạo 2 nút, nút thứ nhất có value = "**Xanh**", nút thứ hai có value = "**Đỏ**".
- Yêu cầu: Khi người dùng click vào nút xanh thì màu nền của tài liệu là: xanh (blue), còn khi người dùng click vào nút đỏ thì màu nền của tài liệu là: đỏ (red).
 - Hướng dẫn: Thuộc tính màu nền của tài liệu được lưu trong thuộc tính bgColor của đối tượng document. Thuộc tính này có thể thay đổi được.

Bài tập 2

```
<html>  
  <head> </head>  
  <body>  
    <form name="frm">  
      <input type="button" value="Xanh"  
        onclick="window.document.backgroundColor = 'blue'; " />  
      <input type="button" value="Đỏ"  
        onclick="window.document.backgroundColor = 'red'; " />  
    </form>  
  </body>  
</html>
```

Bài tập 3

- Tạo một danh sách lựa chọn gồm có 4 màu: red, blue, brown và lavender.
- Khi người dùng chọn một màu thì màu nền của tài liệu sẽ thay đổi tương ứng.


```
<html>
```

```
  <head> </head><body>
```

```
    <script language="JavaScript">
```

```
    function DoiMau() {
```

```
      document.bgColor = frm.Mau.value;
```

```
      // Hoặc viết: window.document.bgColor = frm.Mau.value;
```

```
    }
```

```
  </script>
```

Bạn hãy chọn màu nền:

```
  <form name="frm">
```

```
    <select name="Mau" onchange="DoiMau();" >
```

```
      <option value="red">Màu đỏ</option>
```

```
      <option value="blue">Màu xanh</option>
```

```
      <option value="brown">Màu nâu</option>
```

```
      <option value="lavender">Màu xanh nhạt</option>
```

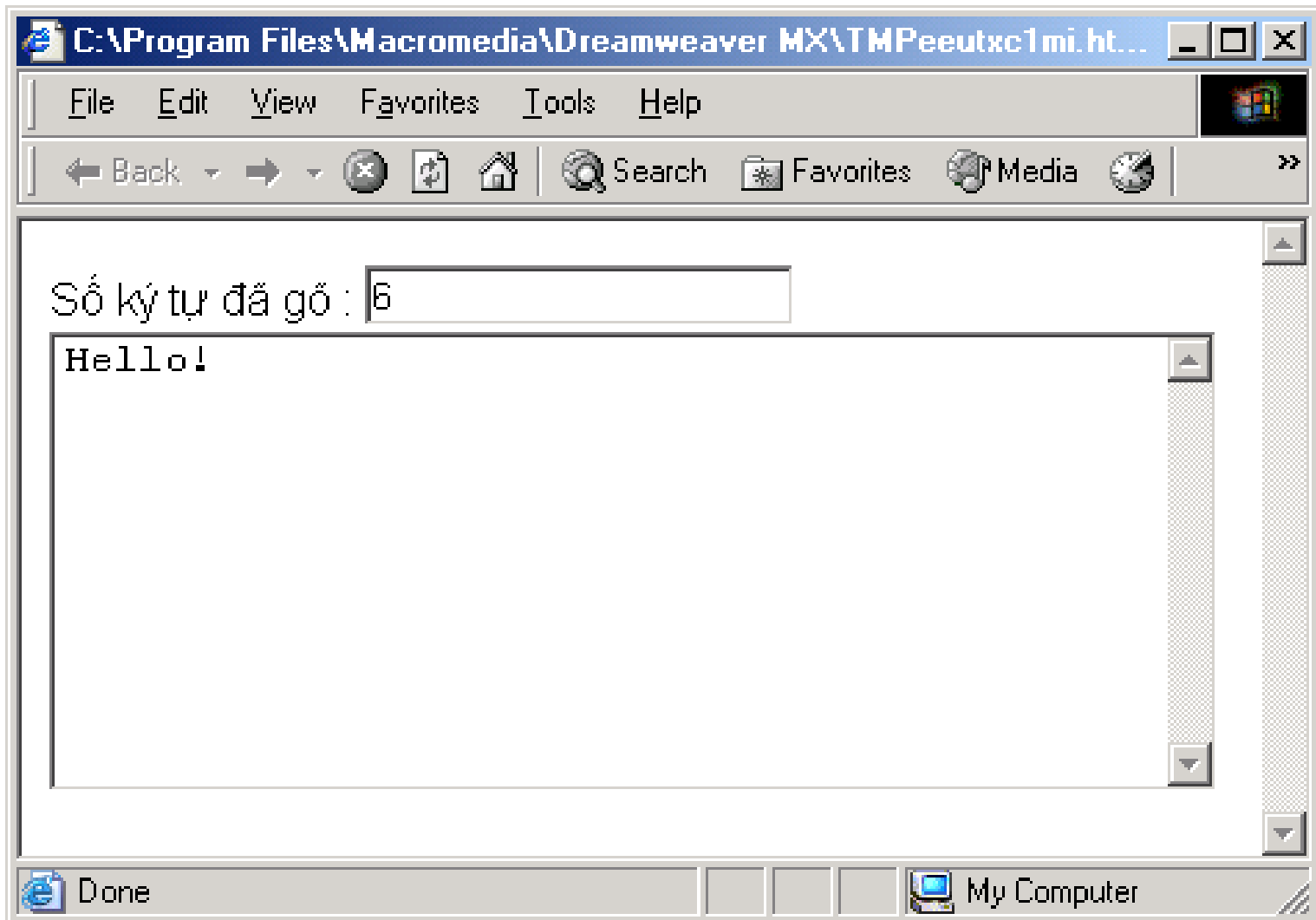
```
    </select> </form>
```

```
  </body></html>
```

Bài tập 4

- Tạo một textarea có tên là NoiDung, một Textbox có tên là : SoKyTu.
- Với yêu cầu như sau:
 - Khi người sử dụng gõ các phím vào trong textarea thì số lượng ký tự (Độ dài xâu) chứa trong textarea đó sẽ được hiển thị trong textbox.
 - Nếu số lượng ký tự trong textarea gõ vào vượt quá 200 ký tự thì thông báo : "Bạn đã gõ quá số ký tự cho phép !".

Bài tập 4



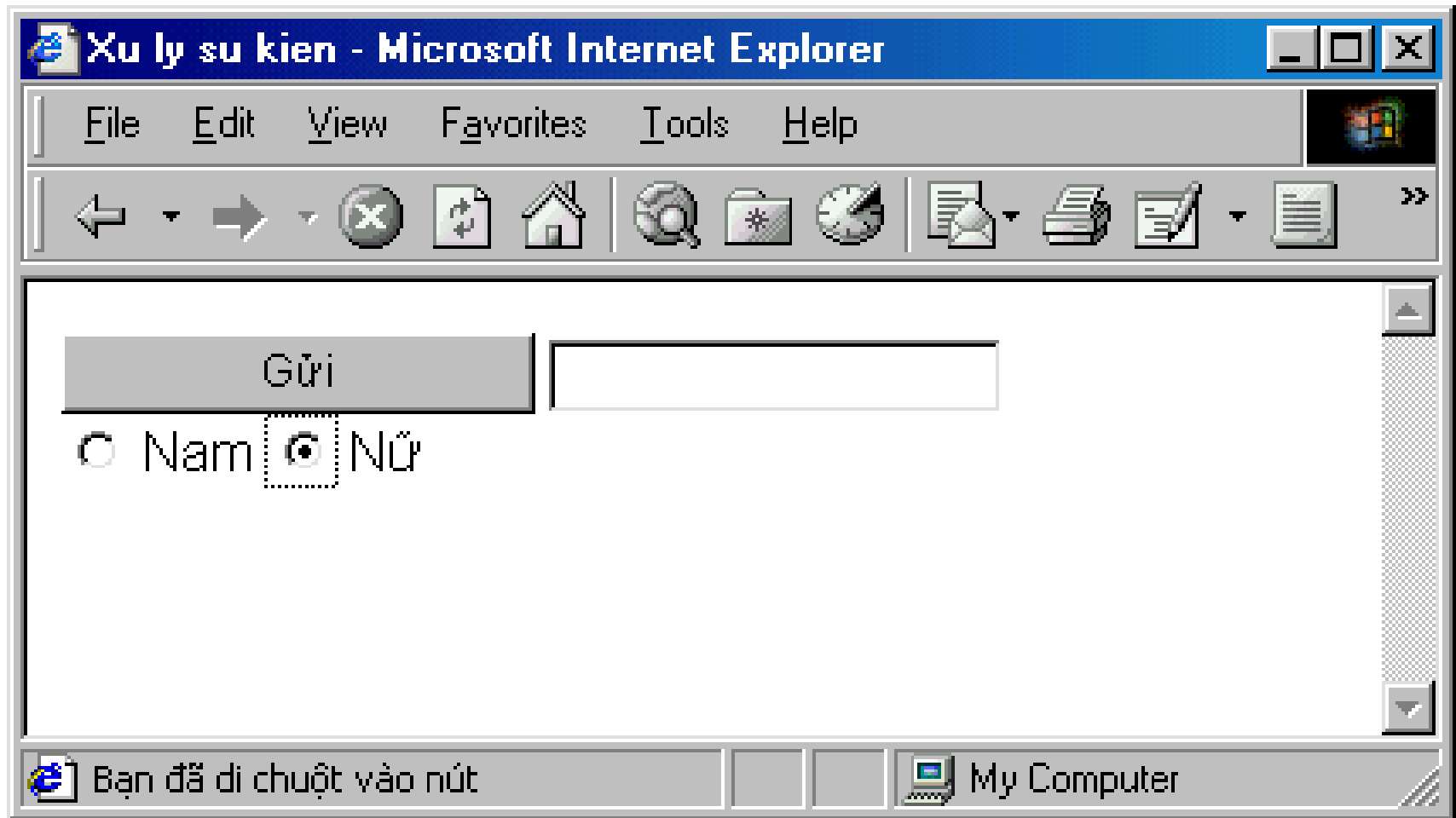
Bài tập 4

```
<html><head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>
<script language="JavaScript">
    function KiemTra() {
        if (frm.NoiDung.value.length > 200)
            alert("Bạn đã gõ quá số ký tự cho phép !");
        frm.SoKyTu.value = frm.NoiDung.value.length;
    }
</script>
<body style="font-family:arial"><form name="frm">
    Số ký tự đã gõ : <input type="text" name="SoKyTu"> <BR>
    <textarea name="NoiDung" cols="50" rows="10"
    onKeyUp="KiemTra();"> </textarea></form>
</body>
</html>
```

Bài tập 5

- Tạo một nút có value = "Gửi", textbox có name = "HoTen", 2 nút radio có tên là GioiTinh và nhãn tương ứng là Nam, nữ.
- Yêu cầu: Khi người dùng di chuyển chuột vào phần tử nào thì hiển thị thông báo tương ứng dưới thanh trạng thái.
 - Ví dụ nếu người sử dụng di chuyển chuột qua nút nhấn "Gửi" thì thanh trạng thái sẽ là "Bạn đang di chuyển chuột vào nút"...

Bài tập 5



Bài tập 5

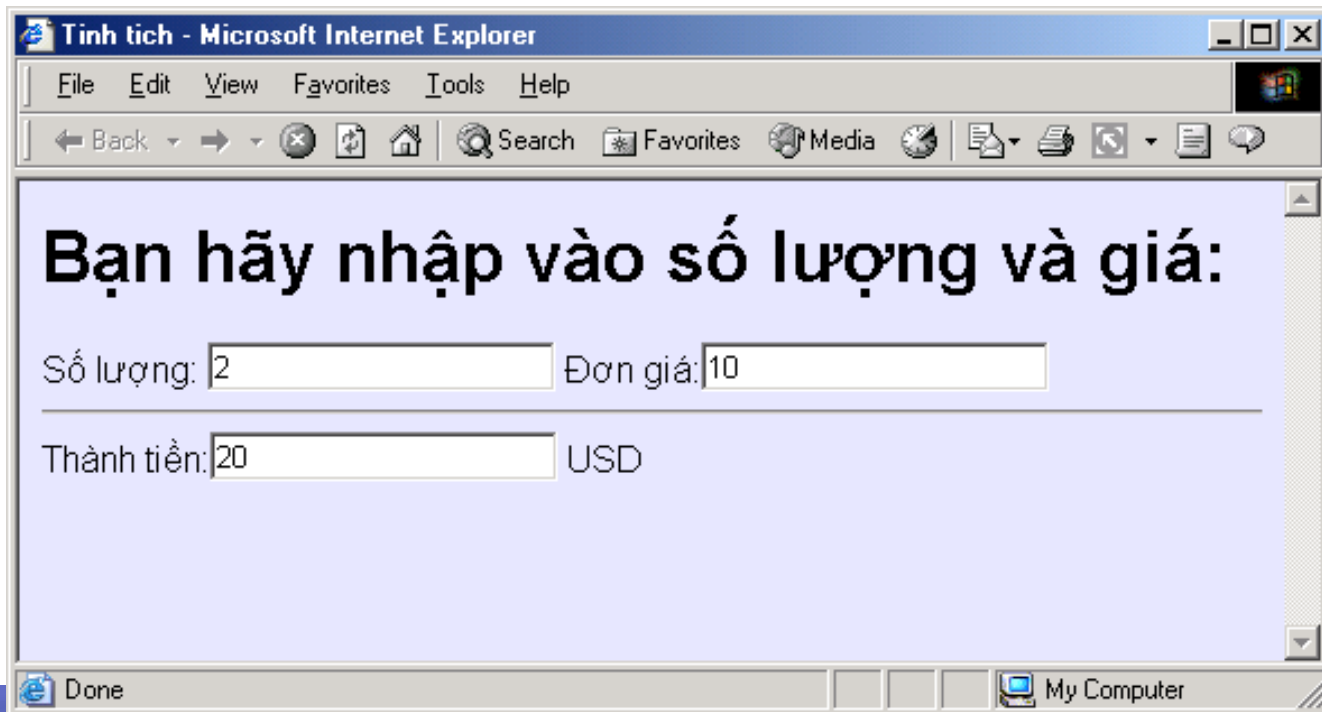
```

<html>
<head>
    <title>Xu ly su kien</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>
<body style="font-family:arial">
    <input type="button" value="Gửi"
onmousemove="window.status = 'Chuột trong nút'; ">
    <input onMouseMove ="window.status='Chuột trong
textbox';">    <br />
    <input type="radio" name="GioiTinh"
onMouseMove="window.status='Trong nam';">Nam </option>
    <input type="radio" name="GioiTinh"
onMouseMove="window.status='Trong nữ';"> Nữ </option>
</body>
</html>

```

Bài tập 6

- Tạo 3 textbox, có tên lần lượt là: SoLuong (Số lượng), DonGia (Đơn giá) và ThanhTien (Thành tiền);
- Yêu cầu: Khi người dùng nhập giá trị trong DonGia thì kết quả sẽ được cập nhật ngay trong ThanhTien.



The screenshot shows a Microsoft Internet Explorer browser window titled "Tinh tich - Microsoft Internet Explorer". The address bar is empty. The main content area has a light blue background and contains the text "Bạn hãy nhập vào số lượng và giá:" in large black font. Below this text are three textboxes. The first row contains two textboxes: "Số lượng:" followed by a text box containing the value "2", and "Đơn giá:" followed by a text box containing the value "10". A horizontal line separates this row from the second row, which contains "Thành tiền:" followed by a text box containing the value "20" and the text "USD" to its right. The browser's status bar at the bottom shows "Done" on the left and "My Computer" on the right.

Bài tập 6

```
<html>
<head>
  <title>Tinh tich</title>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
</head>
<body style="font-family:arial;background-color:lavender">
  <h1>Bạn hãy nhập vào số lượng và giá:</h1>
  Số lượng: <input name="SoLuong">
  Đơn giá:<input name="DonGia"
    <b>onKeyUp="ThanhTien.value=SoLuong.value*DonGia.value">
  <hr />
  Thành tiền:<input name="ThanhTien"> USD
</body>
</html>
```

Bài tập 7

- Cho người dùng nhập vào tên và tuổi. Hãy viết lại tên và tuổi của người đó ra màn hình bằng hàm document.write, trong đó tên có màu đậm, tuổi được gạch chân.

```
<script language = "JavaScript">  
    var Ten, Tuoi; // Khai báo 2 biến để lưu tên và tuổi  
    Ten = prompt("Bạn hãy nhập vào tên ", "");  
    Tuoi = prompt("Bạn hãy nhập vào Tuổi : ", 20);  
    document.write("Chào bạn : <B> " + Ten + "</B>");  
    document.write("<BR>"); // Xuống dòng  
    document.write("Tuổi của bạn là : <U> " + Tuoi + "</U>");  
</script>
```

Bài tập 8

- Viết đoạn Script cho người dùng nhập vào một số nguyên. Nếu người dùng nhập số 1 thì mở trang <http://www.facebook.com>, nếu nhập số 2 thì mở trang <http://www.hust.edu.vn>, còn nếu nhập một số khác với 1, 2 thì mở trang <http://www.google.com>

```
<script language = "JavaScript">
```

```
var LuaChon;
```

```
LuaChon = prompt("Bạn hãy nhập vào một số để mở trang web : ", 1);
```

```
switch (LuaChon) {
```

```
    case "1" : window.open("http://www.facebook.com"); break;
```

```
    case "2" : window.open("http://www.ou.edu.vn"); break;
```

```
    default : window.open("http://www.google.com");
```

```
}
```

```
</script>
```

Bài tập 9

- Hiển thị Giờ và phút trong thanh tiêu đề của cửa sổ khi trang Web được tải

```
<script language = "JavaScript">
```

```
    var D = new Date();
```

```
    document.title = "Bây giờ là: " + D.getHours()+" giờ "+ D.getMinutes()+  
    " phút.";
```

```
</script>
```

Bài tập 10

- Cho người dùng nhập vào năm sinh của họ, sau đó hiển thị tuổi tương ứng.

```
<script language = "JavaScript">  
    var D = new Date();  
    var NamSinh, NamHienTai;  
    NamHienTai = D.getFullYear(); // Lưu năm hiện tại vào biến  
    NamSinh = prompt("Bạn sinh năm bao nhiêu ? : ","");  
    alert("Tuổi của bạn bây giờ là : " + (NamHienTai-NamSinh));  
</script>
```

Bài tập 11

- Xây dựng trang Web thực hiện các phép tính số học cơ bản (cộng, trừ, nhân, chia) như sau:

$$\boxed{12} \quad \boxed{+} \quad \boxed{-} \quad \boxed{\times} \quad \boxed{/} \quad \boxed{\wedge} \quad \boxed{14} = \boxed{-2}$$

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8" />
  <title>Tinh toan so hoc</title>
</head>
<body>
  <form name="formX"><p>
    <input type="text" size="4" value="12" name="a">
    <input type="button" value="+"
      onClick="a_cong_b(document.formX)"/>
    <input type="button" value="-" onClick="a_tru_b(document.formX)"/>
    <input type="button" value="x"
      onClick="a_nhan_b(document.formX)"/>
    <input type="button" value="/" onClick="a_chia_b(document.formX)"/>
    <input type="button" value="^" onClick="a_mu_b(document.formX)"/>
    <input type="number" size="4" value="3" name="b"/> =
    <input type="number" value="0" name="ans" size="9"/>
  </p></form>
</body>
</html>
```

```
<script language="JavaScript">
function a_cong_b(form) {
    a=eval(form.a.value);
    b=eval(form.b.value);
    c=a+b;
    form.ans.value = c;
}
function a_tru_b(form) {
    a=eval(form.a.value);
    b=eval(form.b.value);
    c=a-b;
    form.ans.value=c;
}
function a_nhan_b(form) {
    a=eval(form.a.value);
    b=eval(form.b.value);
    c=a*b;
```

```
    form.ans.value=c;
}
function a_chia_b(form) {
    a=eval(form.a.value);
    b=eval(form.b.value);
    c=a/b;
    form.ans.value = c;
}
function a_mu_b(form) {
    a=eval(form.a.value);
    b=eval(form.b.value);
    c=Math.pow(a, b);
    form.ans.value = c;
}
</script>
```


Bài tập 12

- Kiểm tra tính hợp lệ của thông tin nhập vào. Xây dựng một trang Web cho người dùng nhập vào tên và tuổi, kiểm tra thông tin nhập vào không được để trống và phần tuổi chỉ chấp nhận giá trị số.

Name: Age:

```
<html>
  <head> <title>Kiem tra du lieu vao</title> </head>
<body>
  <form name="testform" onSubmit="return validate();">
    Name: <input type="text" size=30 name="name">
    Age:   <input type="text" size=3 name="age">
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Bài tập 12

```
<script language="JavaScript">
function validate(){
    var digits="0123456789";
    var temp;
    if (document.testform.Name.value=="") {
        alert("No Name !");
        return false;
    }
    if (document.testform.age.value=="") {
        alert("Invalid Age !");
        return false;
    }
    for (var i=0; i<document.testform.age.value.length; i++){
        temp=document.testform.age.value.substring(i,i+1);
        if (digits.indexOf(temp)==-1){
            alert("Invalid Age !");
            return false;
        }
    }
    return true;
}
</script>
```

Nội dung

- Giới thiệu về Javascript
- Nhúng Javascript vào trang web
- Kiểu dữ liệu & Các cú pháp Javascript
- Sử dụng các đối tượng trong Javascript
- Xử lý sự kiện
- **DOM HTML với Javascript**
- Ví dụ

Một số đối tượng trong JavaScript

- 1. Giới thiệu về đối tượng
- 2. Đối tượng của Browser
- 3. Đối tượng của JavaScript
- 4. Đối tượng của HTML

1. Giới thiệu về đối tượng

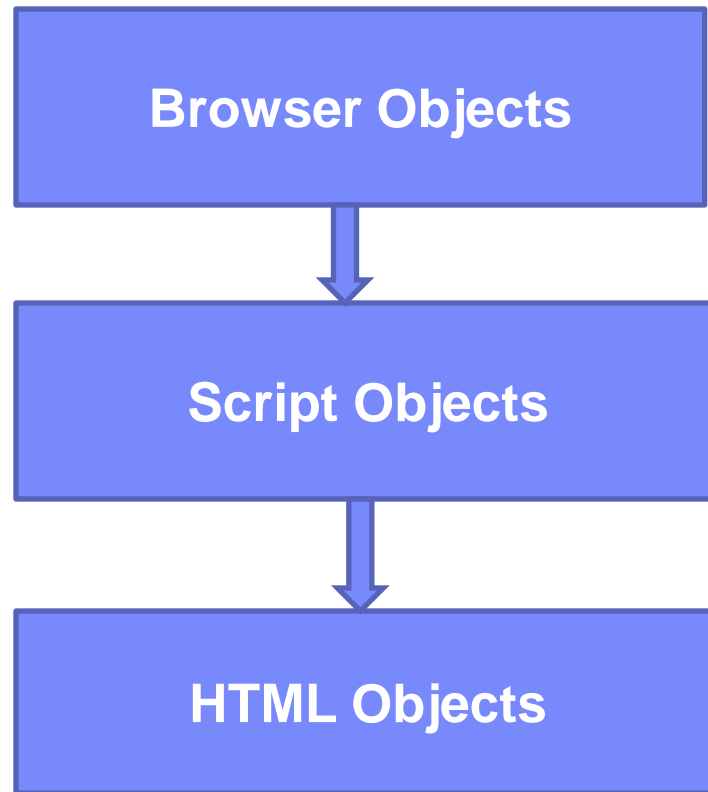
- Đối tượng là một thực thể trong thế giới thực được mô hình hóa
- Đối tượng = Dữ liệu + Phương thức



1. Giới thiệu về đối tượng

- JavaScript là ngôn ngữ lập trình hướng đối tượng
- Khi tạo trang Web, chúng ta có thể sử dụng các đối tượng được cung cấp bởi browser, JavaScript, HTML
- Truy nhập
 - Thuộc tính: *tên_đối_tượng . tên_thuộc_tính*
 - Phương thức: *tên_đối_tượng . tên_phương_thức()*

1. Giới thiệu về đối tượng



Phân tầng các đối tượng

1. Giới thiệu về đối tượng

- Khởi tạo đối tượng

```
tên_đối_tượng = new  
    đối_tượng(danh_sách_tham_số)
```

- Cấu trúc for..in và with

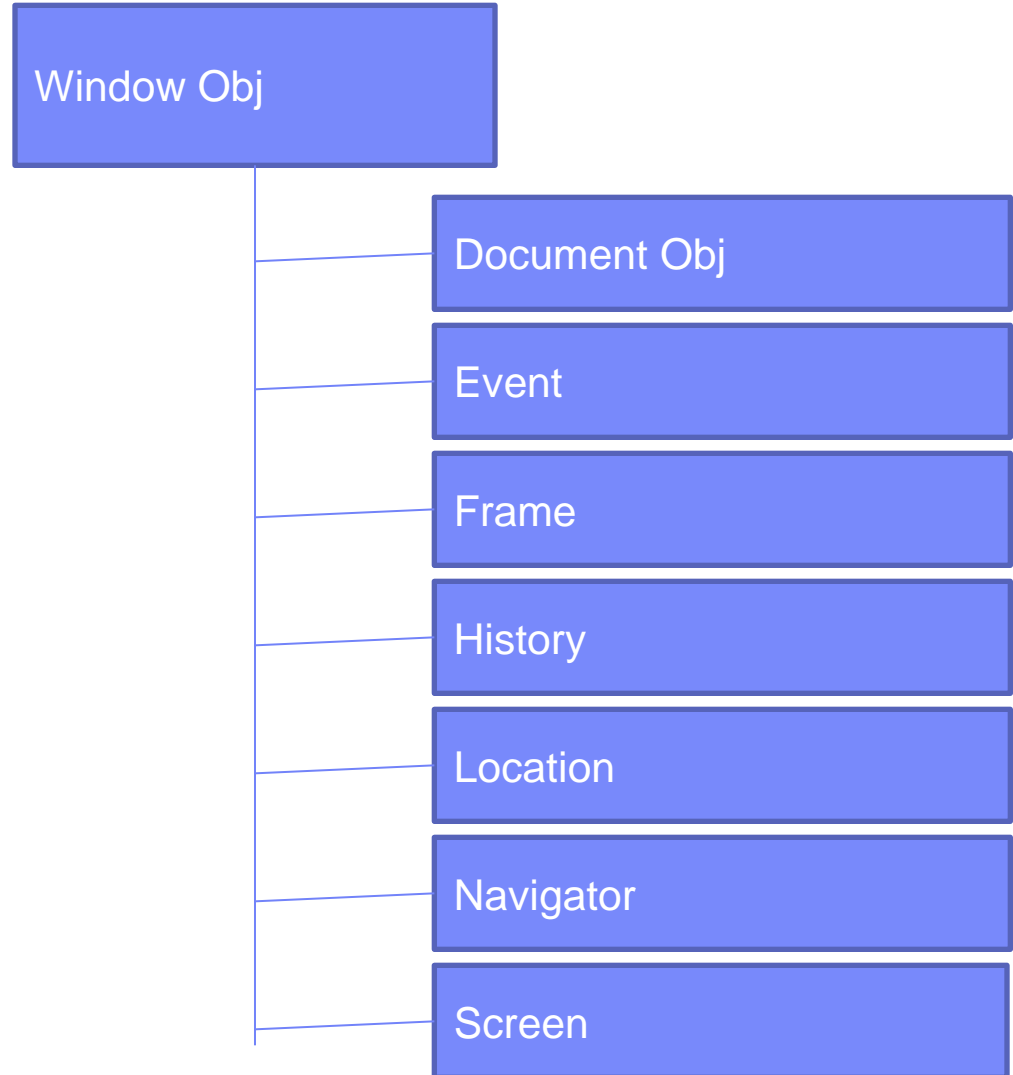
- Cú pháp

```
for (variable in object)  
    lệnh;
```

và

```
with (object)  
    lệnh;
```


Đối tượng của browser



- IE Objects

Đối tượng Window

- Mỗi thành phần trên trang web được xem như một đối tượng,
- Tất cả các đối tượng trong Javascript được tổ chức phân cấp dạng hình cây gọi là mô hình DOM (Document Object Model),
- Ở mức trên cùng là đối tượng **window** biểu thị cho khung hay cửa sổ của trình duyệt, các phần tử còn lại là đối tượng con của window
- Để truy xuất đến một đối tượng trong IE và thay đổi dữ liệu cho chính phần tử đó dùng thuộc tính ID , tất cả các đối tượng trên trang đều có một ID duy nhất.

Đối tượng Window

- Đối tượng Window
 - Thể hiện cửa sổ của trình duyệt
 - Được sử dụng để
 - nhận các thông tin về trạng thái của cửa sổ
 - hiển thị các đối tượng khác
 - truy cập các sự kiện

Đối tượng Window

- document Obj: biểu diễn tài liệu HTML
- event: biểu diễn trạng thái của sự kiện
- frame: biểu diễn các frame (child windows)
- history: thông tin về URLs đã mở
- location: thông tin về URL hiện tại
- navigator: thông tin về Web browser
- screen: thông tin về màn hình client, khả năng render

Đối tượng Window

Ví dụ:

```
window.defaultStatus="String";
```

```
window.status="String";
```

```
window.location="URL";
```

Ta cũng có thể mở một trang web mới bằng lệnh:

```
window.location.href= "URL";
```

Đối tượng Window

Phương Thức	Mô tả
<code>alert("msg")</code>	Hiển Thị hộp thoại thông báo
<code>blur()</code>	Di chuyển con trỏ đến cửa sổ hiện hành
<code>clearInterval(ID)</code>	Hủy thời gian đã thiết lập bằng <code>setInterval()</code>
<code>clearTimeout(ID)</code>	Hủy thời gian đã thiết lập bằng <code>setTimeout()</code>
<code>close()</code>	Đóng cửa sổ hiện hành
<code>confirm("msg")</code>	Hiển thị hộp thoại xác nhận với hai nút Cancel và OK
<code>focus()</code>	Đưa con trỏ về cửa sổ hiện hành
<code>MoveBy(x,y)</code>	Di chuyển cửa sổ đến một vị trí mới một đoạn pixel so với cửa sổ hiện hành
<code>MoveTo(x,y)</code>	Di chuyển cửa sổ qua trái và lên trên một đoạn pixel cụ thể so với cửa sổ hiện hành.

Đối tượng Window

<code>open(URL,"windowname", "FeatureList")</code>	URL : đc trang web nạp vào cửa sổ. WindowName: là tên cửa sổ . FeatureList : danh sách các thuộc tính của cửa sổ: toolbars, menu, status
<code>print()</code>	in nội dung trong cửa sổ hiện hành.
<code>prompt("msg","reply")</code>	Hiện thị hộp thoại nhập liệu
<code>setTimeout(func,millisec)</code>	Thiết lập thời gian mili giây để gọi 1 hàm
<code>stop()</code>	Hủy việc download một cửa sổ. Tương tự như việc đóng một cửa sổ trình duyệt.
<code>resizeBy(dx,dy)</code>	Thay đổi kích thước cửa sổ sang phải dx, dưới dy pixel
<code>resizeTo(x,y)</code>	Thay đổi kích thước x, y pixel
<code>scrollBy(dx,dy)</code>	cuộn nội dung sang phải dx, xuống dưới dy pixel
<code>scrollTo(x,y)</code>	cuộn nội dung trên trang đến vị trí x,y

Đối tượng Window

Ví dụ: `Objwindow.close()`

❖ **Từ khoá self**: trong trường hợp muốn thao tác trên cửa sổ hiện hành ta dùng từ khoá `self` thay thế cho đối tượng `window`

Ví dụ : đóng cửa sổ hiện hành:

`Self.close()` hoặc `window.close()`

Đối tượng Document

- Đối tượng document cung cấp các thuộc tính và phương thức để làm việc với toàn bộ tài liệu hiện hành gồm: form, liên kết, hình ảnh, tiêu đề, vị trí hiện hành, màu hiện hành...
- Đối tượng document được định nghĩa khi tag body được xử lý trong trang HTML và nó vẫn tồn tại nếu trang được nạp.
- Các thuộc tính của document phản ánh thuộc tính của tag body.
 - Trong body có 2 sự kiện OnLoad và Unload

Đối tượng Document

Thuộc tính

Thuộc tính	Mô tả
alinkcolor	Thiết lập hoặc trả về giá trị màu của liên kết đang xem của tài liệu
bgcolor	Thiết lập hoặc trả về giá trị màu nền của tài liệu
cookie	Chứa giá trị các cookies dành cho tài liệu hiện hành
domain	Trả về giá trị tên miền máy chủ chứa document
fgcolor	Thiết lập hoặc trả về giá trị màu chữ của tài liệu
lastmodified	Trả về giá trị ngày giờ cuối cùng mà tài liệu được cập nhật

Đối tượng Document

- Thuộc tính

linkcolor	Thiết lập hoặc trả về giá trị màu của liên kết trong tài liệu
location	mở một trang web mới
referrer	Returns the URL of the document that loaded the current document
title	Trả về giá trị của tựa đề của tài liệu
url	Trả về đường dẫn củ tài liệu hiện hành
vlinkcolor	Thiết lập hoặc trả về giá trị màu của liên kết đã xem của tài liệu

Đối tượng Document

▪ Phương thức

Phương thức	Mô tả
<code>clear()</code>	Xóa tài liệu
<code>close()</code>	Đóng một tài liệu
<code>focus()</code>	Đưa trở về một đối tượng trong trang
<code>open(["mimetype"])</code>	Mở một stream để thu thập dữ liệu của các phương thức <code>write</code> và <code>writeln</code> .
<code>write("str")</code>	viết một chuỗi vào một tài liệu
<code>writeln("str")</code>	viết một chuỗi vào một tài liệu và xuống dòng

Đối tượng History

- Cung cấp các URLs được mở gần nhất
- Phương thức
 - back: quay lại URL ngay trước
 - go(number): quay lại vị trí xác định trong tài liệu
 - forward: mở URL tiếp theo

Đối tượng HTML DOM

- DOM = Document Object Model
- Là tập hợp các đối tượng HTML chuẩn được dùng để **truy xuất** và **thay đổi thành phần HTML** trong trang web (thay đổi nội dung tài liệu của trang)
- Một số đối tượng của DOM: window, document, history, link, form, frame, location, event, ...

Đối tượng Window - DOM

■ Properties

- document
- event
- history
- location
- name
- navigator
- screen
- status

● Methods

- alert
- confirm
- prompt
- blur
- close
- focus
- open

Đối tượng Window - DOM

- Ví dụ:

```
<html>
```

```
  <body>
```

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

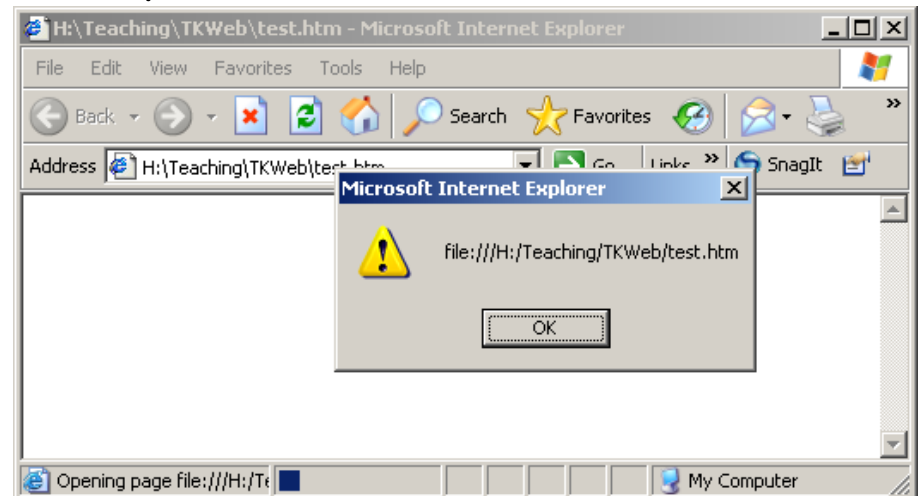
```
      var curURL = window.location;
```

```
      window.alert (curURL) ;
```

```
    </script>
```

```
  </body>
```

```
</html>
```



Đối tượng Document - DOM

- Biểu diễn cho **nội dung trang HTML** đang được hiển thị trên trình duyệt
- Dùng để lấy thông tin về tài liệu, các thành phần HTML và xử lý sự kiện



Đối tượng Document - DOM

- **Properties**

- **aLinkColor**
- **bgColor**
- **body**
- **fgColor**
- **linkColor**
- **title**
- **URL**
- **vlinkColor**
- **forms[]**
- **images[]**
- **childNodes[]**

- **documentElement**
- **cookie**
- **.....**

- **Methods**

- **close**
- **open**
- **createTextNode(" text ")**
- **createElement("HTMLtag")**
- **getElementById("id")**
- **...**

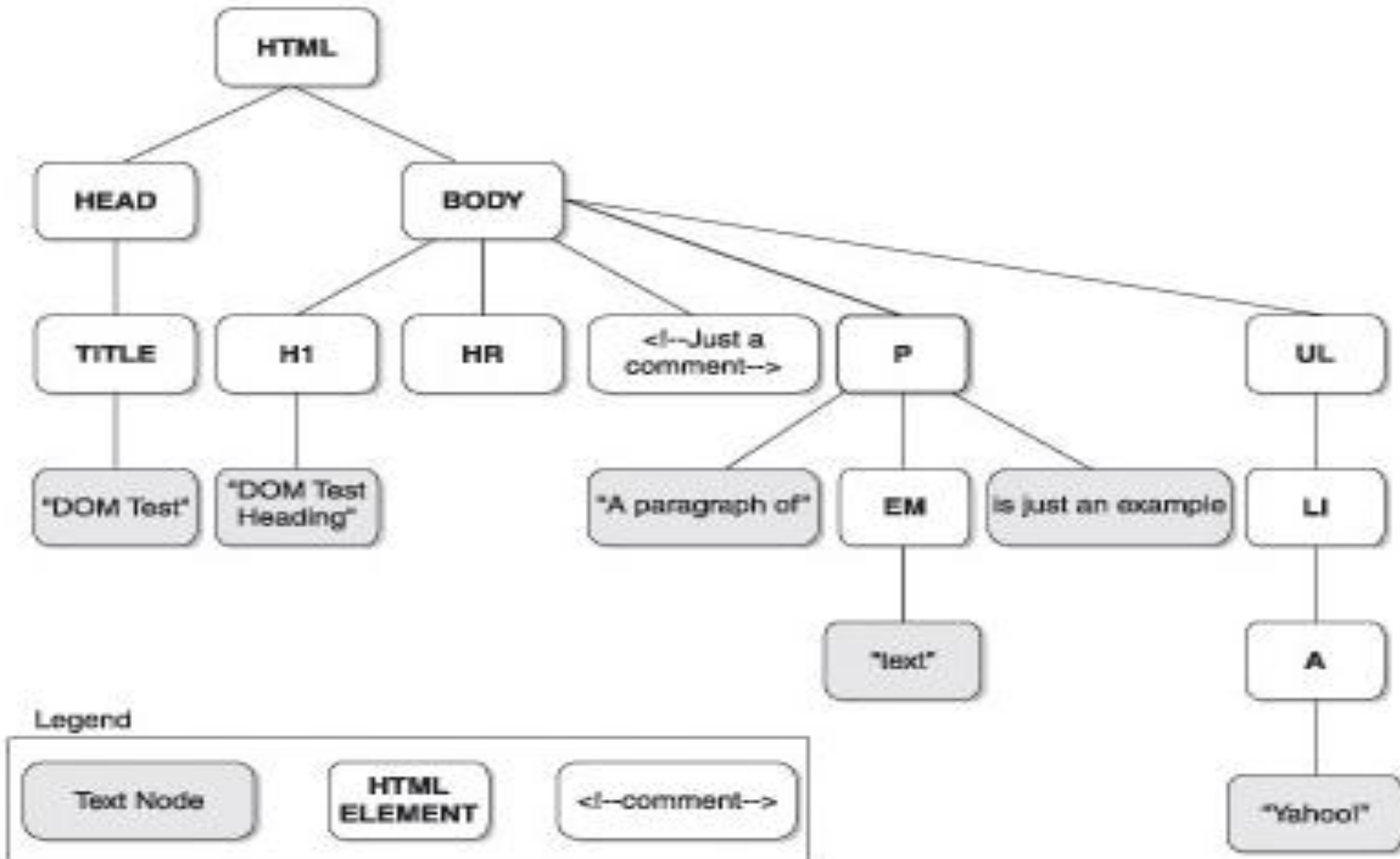
Đối tượng Document - DOM

- Biểu diễn nội dung của tài liệu theo cấu trúc cây

```
<html>
  <head>
    <title>DOM Test</title>
  </head>
  <body>
    <h1>DOM Test Heading</h1>
    <hr />
    <!-- Just a comment -->
    <p id="p1" >A paragraph of <em>text</em>
      is just an example</p>
    <ul>
      <li>
        <a href="http://www.yahoo.com" > Yahoo!
        </a>
      </li>
    </ul>
  </body>
</html>
```

Đối tượng Document - DOM

- Cấu trúc cây nội dung tài liệu



Đối tượng Document - DOM

■ Các loại DOM Node chính

Node Type Number	Loại	Mô tả	Vi dụ
1	Element	(X)HTML or XML element	<code><p>... </p></code>
2	Attribute	Thuộc tính của HTML hay XML element	<code>align="center"</code>
3	Text	Nội dung chứa trong HTML or XML element	This is a text fragment!
8	Comment	HTML comment	<code><!-- This is a comment --></code>
9	Document	Đối tượng tài liệu gốc, thường là element nằm ở cấp cao nhất trong cây cấu trúc của tài liệu	<code><html></code>
10	DocumentType	Định nghĩa loại tài liệu	<code><!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"></code>

Đối tượng Document - DOM

- `getElementById (id1)`

Trả về node có giá trị thuộc tính `id = id1`

Ví dụ:

```
//<p id="id1" >  
//   some text  
//</p>
```



Text Node

```
var node = document.getElementById("id1");  
var nodeName = node.nodeName; // p  
var nodeType = node.nodeType; // 1  
var nodeValue = node.nodeValue; // null  
var text      = node.innerText ; // some text
```

Đối tượng Document - DOM

- `createElement (nodeName)`

Cho phép tạo ra 1 node HTML mới tùy theo đối số `nodeName` đầu vào

Ví dụ:

```
var imgNode = document.createElement("img");  
imgNode.src = "images/test.gif";
```

```
// 
```

Đối tượng Document - DOM

■ createTextNode (content)

Ví dụ:

```
var textNode = document.createTextNode("New  
text");  
var pNode = document.createElement("p");  
pNode.appendChild(textNode);  
  
// <p>New text</p>
```


Đối tượng Document - DOM

- `appendChild (newNode)`

Chèn node mới **newNode** vào cuối danh sách các node con của một node.

Ví dụ:

```
//<p id="id1" >  
//     some text  
//</p>
```

```
var pNode = document.getElementById("id1");  
var imgNode = document.createElement("img");  
imgNode.src = "images/test.gif";  
pNode.appendChild(imgNode);
```

```
//<p id="id1" >  
//     some text  
//</p>
```

Đối tượng Document - DOM

- innerHTML

Chỉ định nội dung HTML bên trong một node.

Ví dụ:

```
//<p id="para1" >  
// some text  
//</p>
```

```
var theElement = document.getElementById("para1");  
theElement.innerHTML = "Some <b> new </b> text";
```

```
// Kết quả :  
// <p id="para1" >  
// Some <b> new <b/> text  
// </p>
```

Đối tượng Document - DOM

- innerText

Tương tự innerHTML, tuy nhiên bất kỳ nội dung nào đưa vào cũng được xem như là text hơn là các thẻ HTML.

Ví dụ:

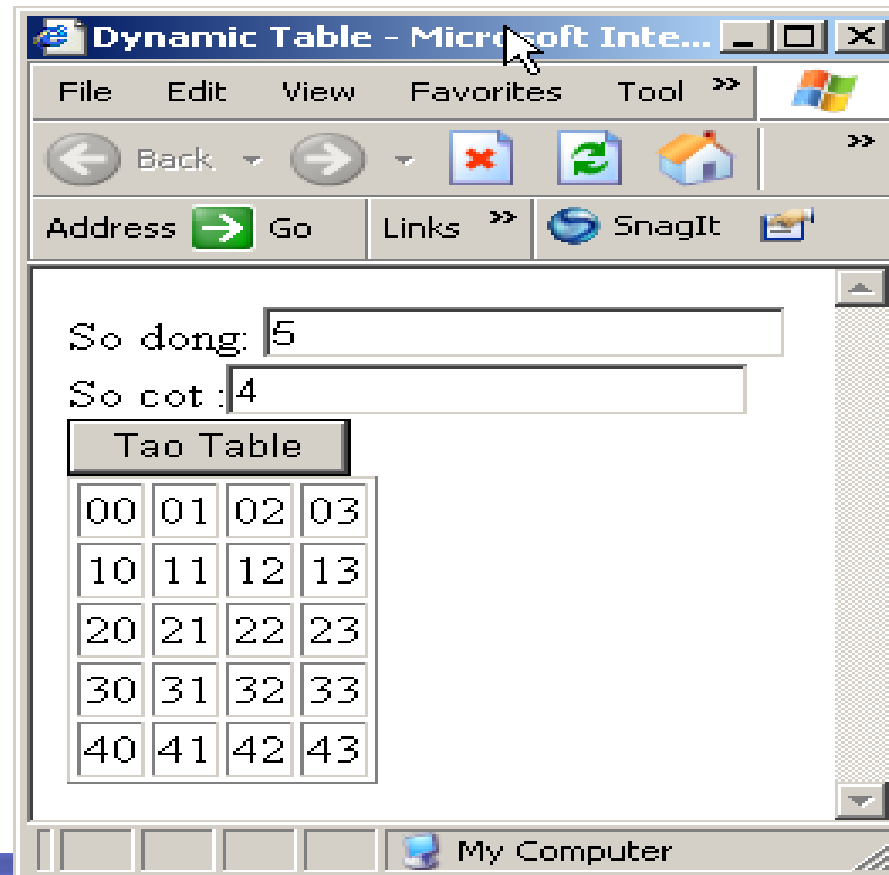
```
var theElement = document.getElementById("para1");  
theElement.innerText = "Some <b> new </b> text";  
// Kết quả hiển thị trên trình duyệt  
// bên trong thẻ p: "Some <b> new </b> text"
```

Nội dung

- Giới thiệu về Javascript
- Kiểu dữ liệu & Các cú pháp Javascript
- Nhúng Javascript vào trang web
- Sử dụng các đối tượng trong Javascript
- Xử lý sự kiện
- DOM HTML với Javascript
- Ví dụ

Ví dụ: Dynamic Table

- Viết trang web cho phép tạo table có số dòng, số cột do người dùng nhập vào.



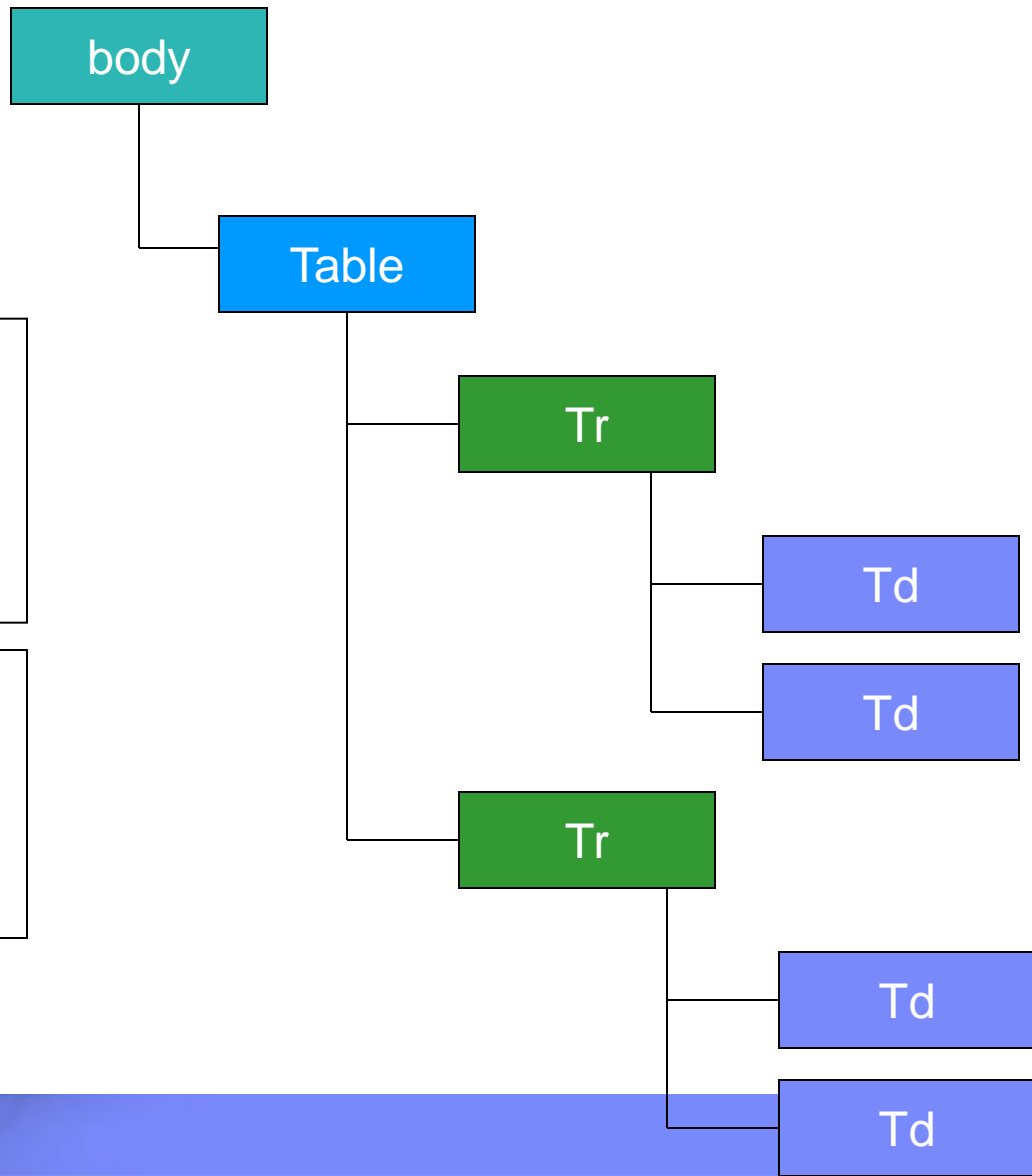
Ví dụ: Dynamic Table

<Table>

```
<Tr>  
    <td> 12 </td>  
    <td> 13 </td>  
</Tr>
```

```
<Tr>  
    <td> 21 </td>  
    <td> 22 </td>  
</Tr>
```

</Table>



Ví dụ:

`<Table>`

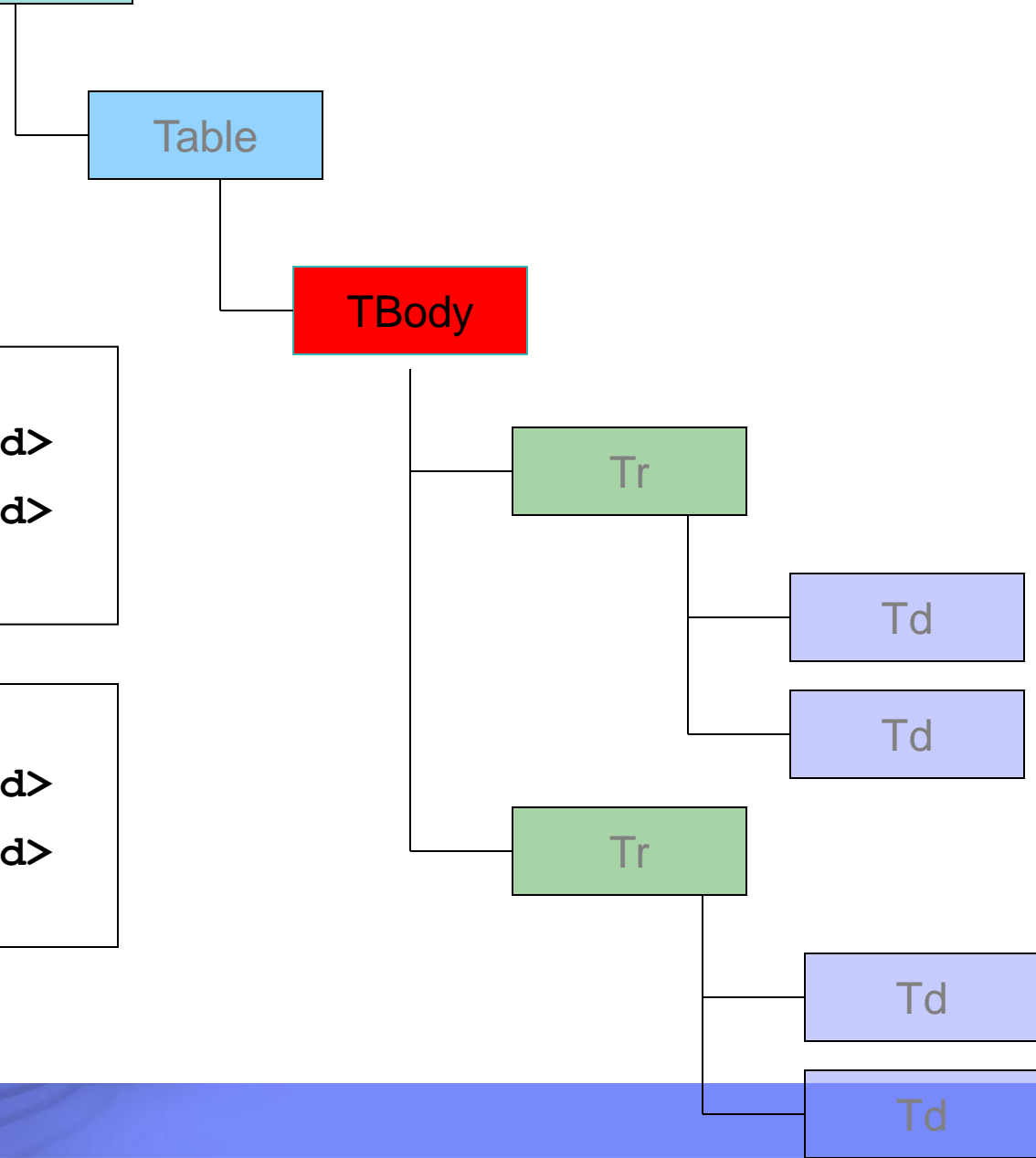
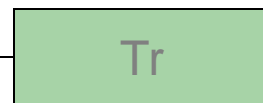
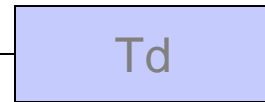
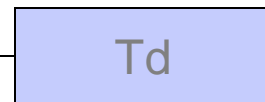
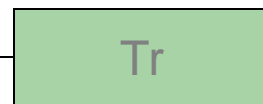
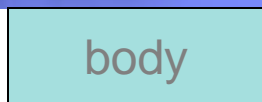
`<Tbody>`

```
<Tr>  
  <td> 12 </td>  
  <td> 13 </td>  
</Tr>
```

```
<Tr>  
  <td> 21 </td>  
  <td> 22 </td>  
</Tr>
```

`</Tbody>`

`</Table>`



Ví dụ: Dynamic Table

- `Document.createElement(...)`: Tạo một đối tượng thẻ DOM HTML
- `Object.appendChild(...)`: Thêm một đối tượng thẻ DOM HTML như là nút con.


```
function CreateTable(divTable)
{
    var tagTable = document.createElement("table");
    tagTable.border = 1;
    var tagTBody = document.createElement("tbody");
    tagTable.appendChild(tagTBody);

    var nDong = txtSoDong.value;
    var nCot = txtSoCot.value;

    for (i=0; i<nDong; i++)
    {
        var tagTR = document.createElement("tr");
        for (j=0; j<nCot; j++)
        {
            var tagTD = document.createElement("td");
            var textNode = document.createTextNode(i+" "+j);
            tagTD.appendChild(textNode);

            tagTR.appendChild(tagTD);
        }

        tagTBody.appendChild(tagTR);
    }

    divTable.appendChild(tagTable);
}
```

Đối tượng JavaScript

- 1. Đối tượng String
- 2. Đối tượng Date
- 3. Đối tượng Math
- 4. Đối tượng Array

1. Đối tượng String

- Được sử dụng để làm việc với chuỗi văn bản
- Khởi tạo

```
var tên_xâu = "nội_dung_xâu";
```

```
tên_xâu="nội_dung_xâu";
```

```
var tên_xâu = String("nội_dung_xâu");
```

1. Đối tượng String

- Thuộc tính
 - length: độ dài chuỗi
- Phương thức
 - big: tăng kích thước chuỗi
 - small: giảm kích thước chuỗi
 - toLowerCase: chuyển thành chữ thường
 - toUpperCase: chuyển thành chữ hoa
 - fontcolor: xác định màu chữ
 - charAt: vị trí kí tự trong chuỗi

2. Đối tượng Date

- Chứa các thông tin về ngày, giờ
- Lưu trữ thời điểm thông qua số lượng ms từ ngày 1/1/1970 tại 00:00:00

- Khởi tạo

```
tên_đối_tượng = new Date(tham_số);
```

- Ví dụ

```
today = new Date();
```

```
dateobj = new Date("Octorber  
13,2009,10:10:10");
```

2. Đối tượng Date

- Phương thức chia làm:

get: lấy giá trị ngày và giờ

set: thiết lập giá trị ngày và giờ

Bao gồm: Hours, Minutes, Seconds, Time, Year, Month, Date

- Giá trị

- seconds và minutes: 0-59

- hours: 0-23

- day: 0-6

- date: 0-31

- months: 0-11

3. Đối tượng Math

- Thuộc tính
 - PI: số Pi, xấp xỉ 3.1415
 - E: số e, xấp xỉ 2.718
- Phương thức
 - abs: lấy giá trị tuyệt đối
 - sin, cos, tan
 - min (number1, number2)
 - max (number1, number2)
 - sqrt
 - pow

4. Đối tượng Array

■ Khởi tạo

```
tên_mảng = new Array(kích_thước_mảng);  
hoặc  
tên_mảng = new Array(ds_phần_tử);
```

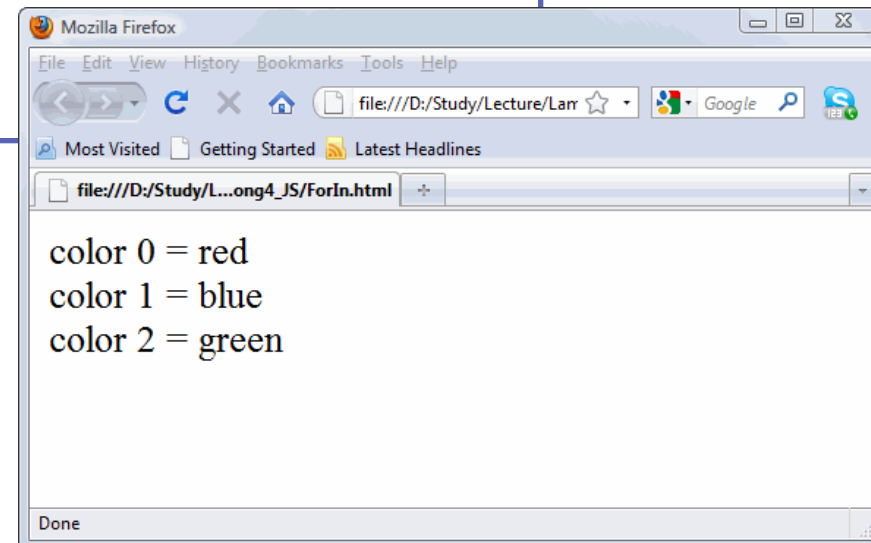
Ví dụ: mybook = new Array(2);
 mybook[0] = "Access 2000";
 mybook[1] = "Programming Techniques";

■ Phương thức

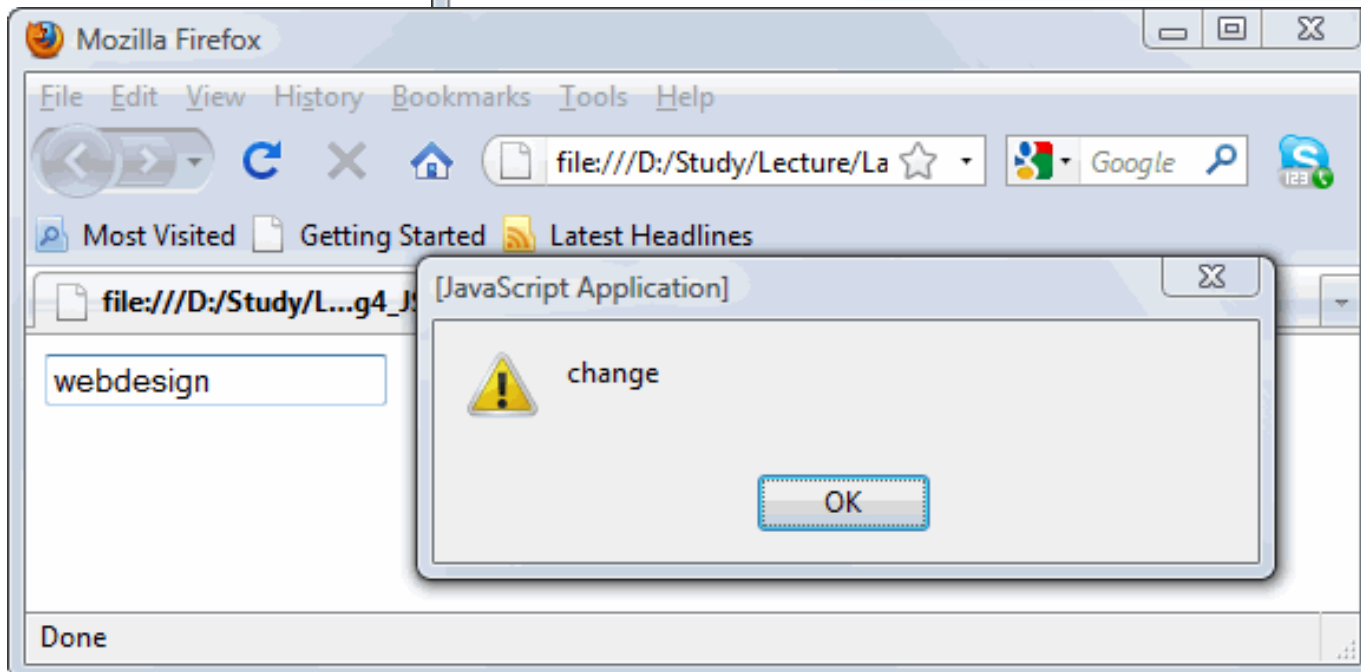
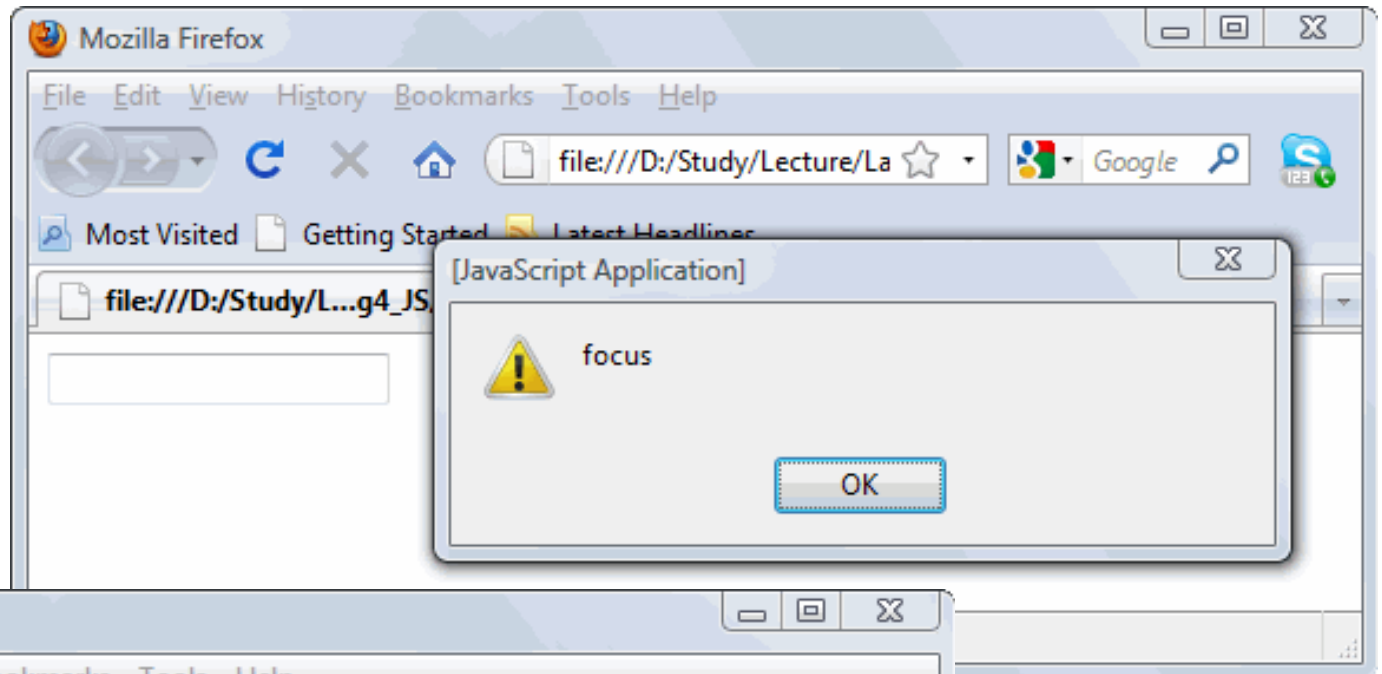
- reverse: đảo ngược các phần tử trong mảng
- pop: trả về phần tử cuối cùng của mảng, rồi xóa nó
- push: thêm một phần tử vào cuối mảng

4. Đối tượng Array

```
<head>
  <script language="javascript">
    color = new Array("red", "blue", "green");
    var scolor;
    for(var elem in color){
      scolor = "color " + elem + " = " +
              color[elem] + "<br>";
      document.write(scolor);
    }
  </script>
</head>
```



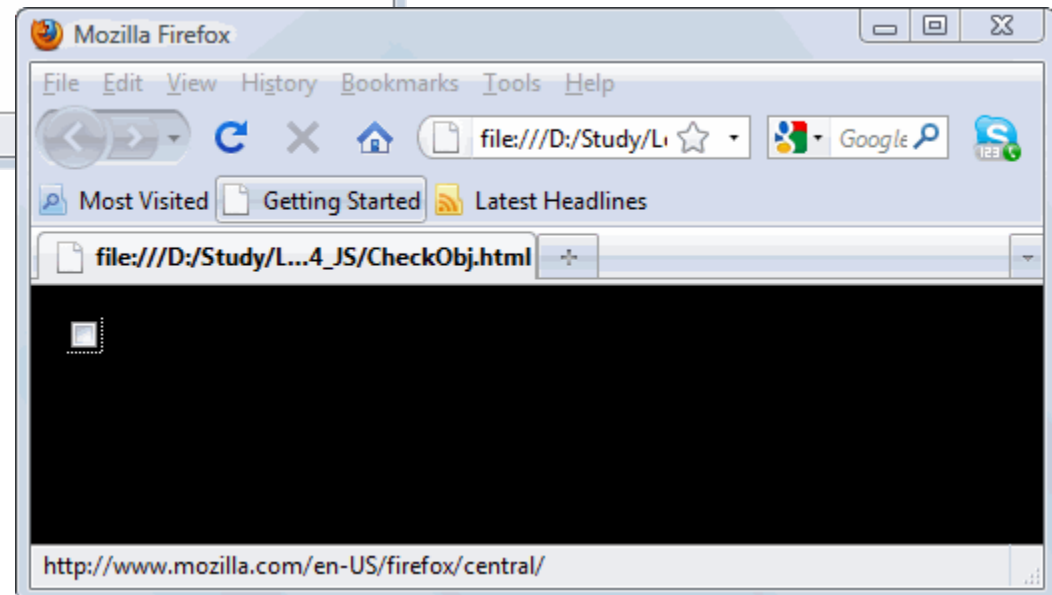
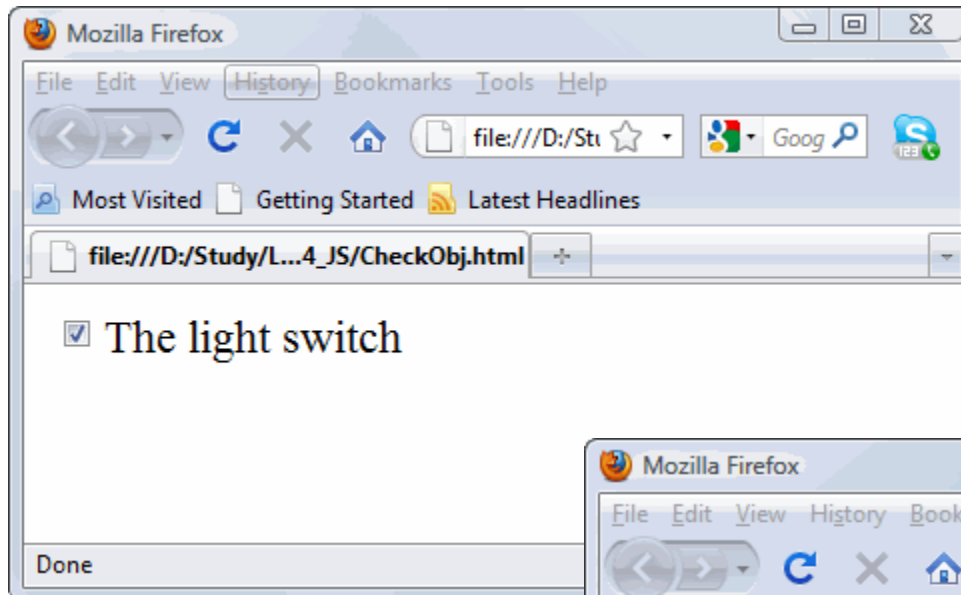
Bài tập 1



Bài tập 1

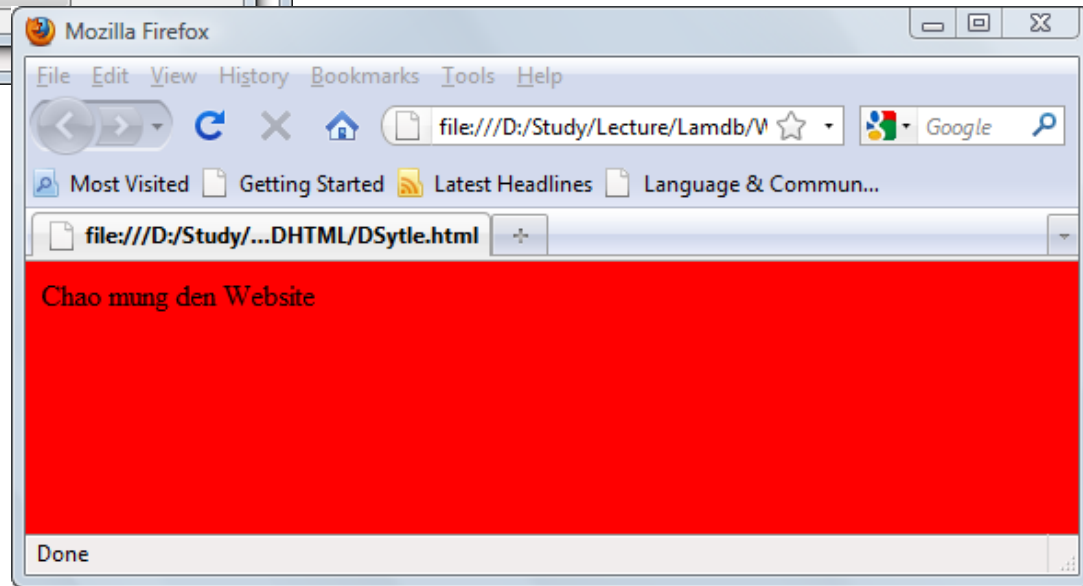
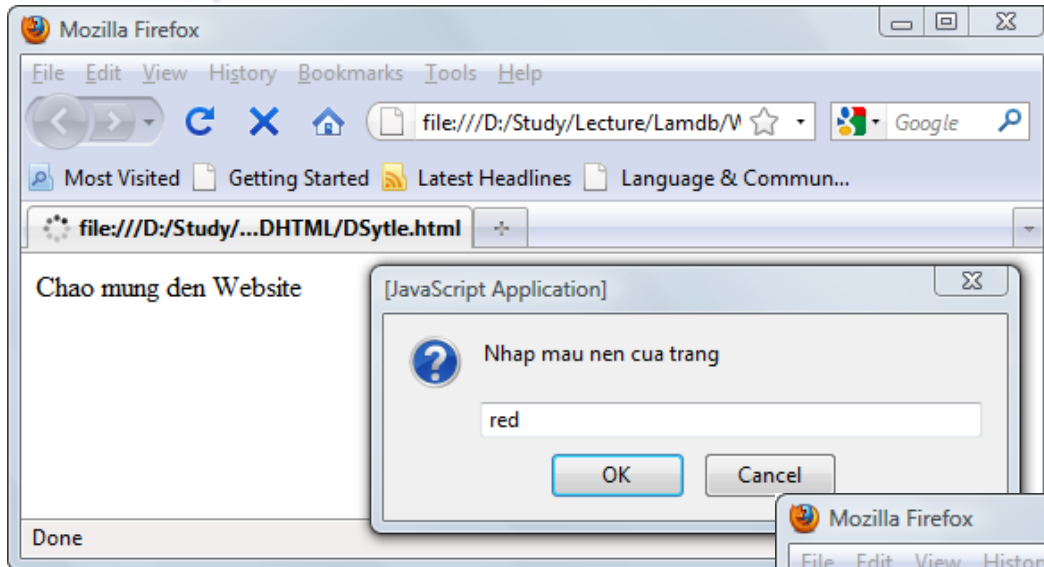
```
<html>
  <head>
    <script language="javascript">
      function show(content) {
        alert (content);
      }
    </script>
  </head>
  <body>
    <form name="myfrm">
      <input type="text onFocus="show (' focus ' );"
        onChange="show ( ' change ' );">
    </form>
  </body>
</html>
```

Bài tập 2 – Sự kiện trên checkbox



```
<html>
  <head>
    <script language="javascript">
      function switchlight() {
        var check = window.document.frmlight.chklight;
        if (check.checked)
          document.bgColor = 'white';
        else
          document.bgColor='black';
      }
    </script>
  </head>
  <body bgcolor="white">
    <form name = frmlight>
      <input type="checkbox" name = "chklight"
        onClick="switchlight();">The light switch
    </form>
  </body>
</html>
```

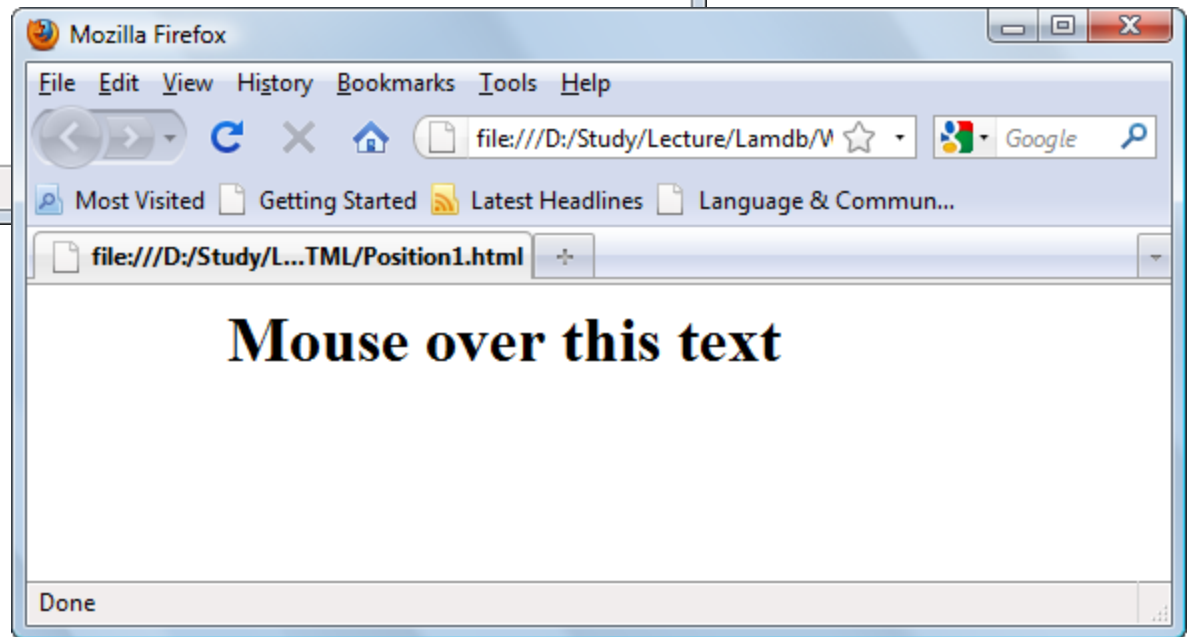
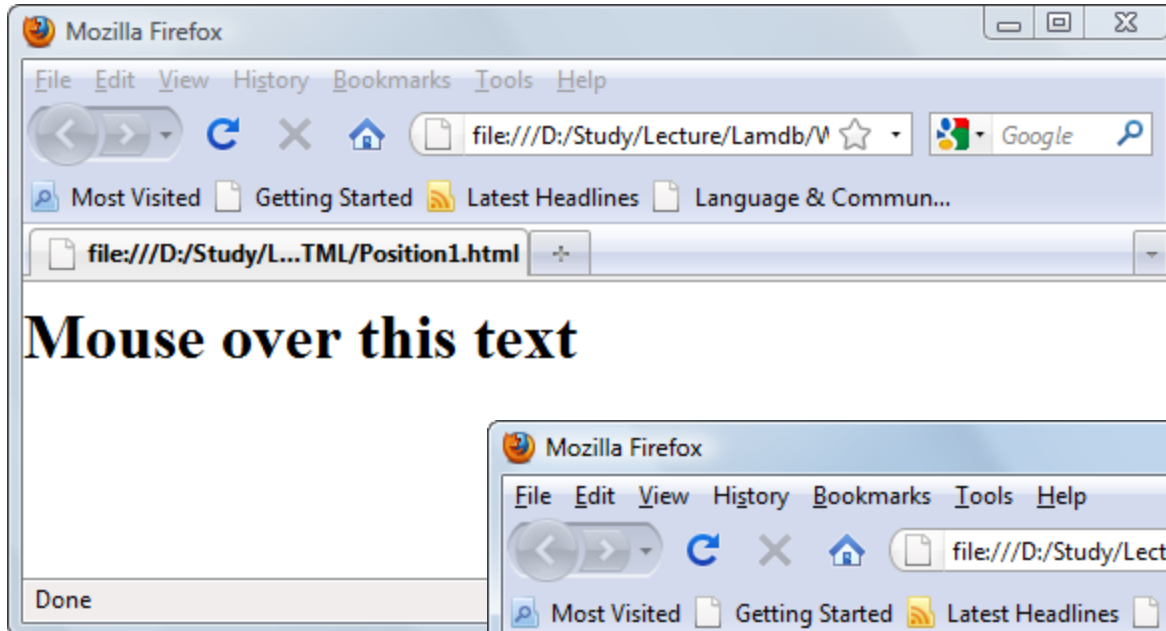
Bài tập 3 – Đổi màu nền



Bài tập 3

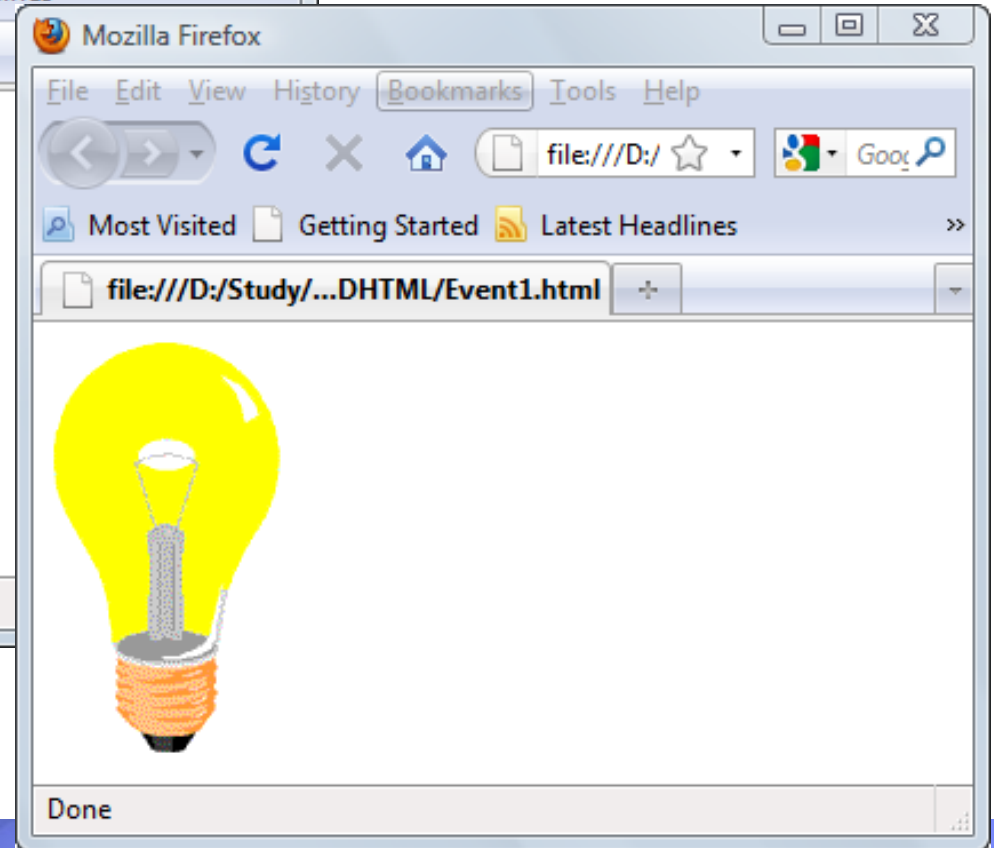
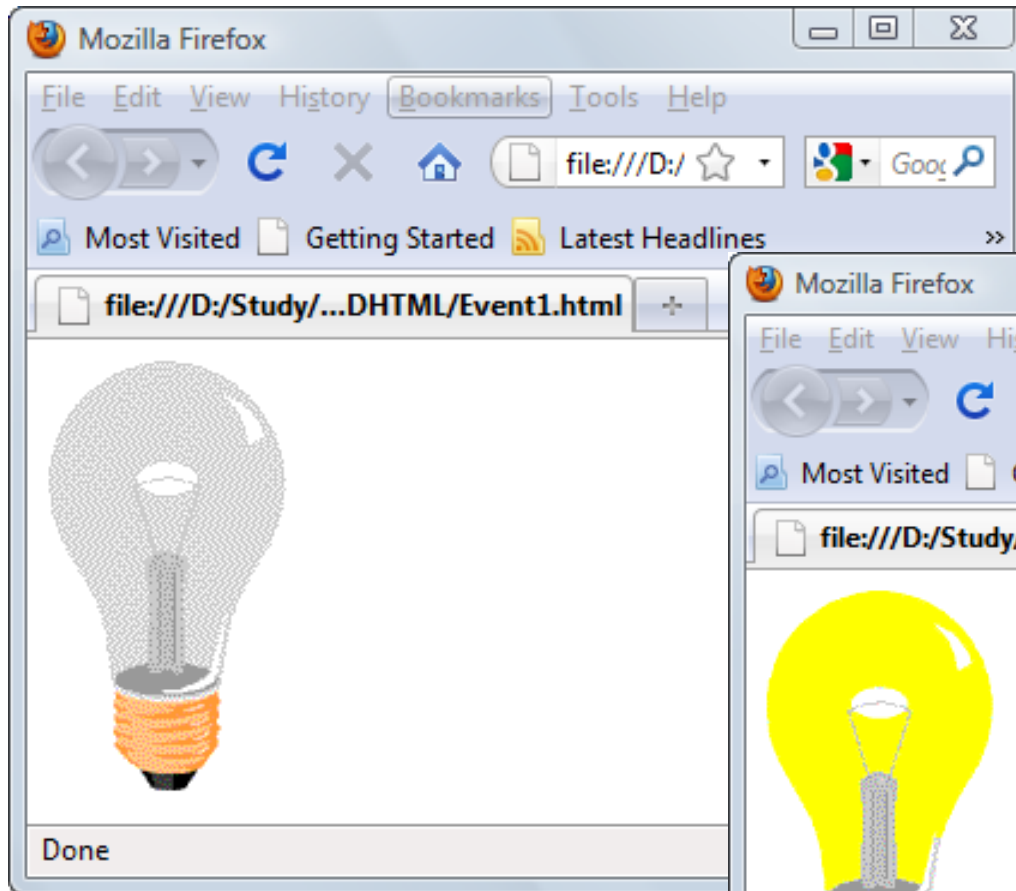
```
<html>
  <head>
    <script language="javascript">
      function bat_dau() {
        var mau=prompt("Nhap mau nen cua trang","");
        document.body.style.backgroundColor=mau;
      }
    </script>
  </head>
  <body onload="bat_dau();">
    <p>Chao mung den Website</p>
  </body>
</html>
```

Bài tập 4 – Văn bản di chuyển




```
<html>
<head>
<script language="javascript">
  function moveleft() {
    document.getElementById('header').style.position=
      "absolute";
    document.getElementById('header').style.left="0";
  }
  function moveback() {
    document.getElementById('header').style.left="100";
  }
</script>
</head>
<body>
  <h1 id="header" onmouseover="moveleft()"
    onmouseout="moveback()"> Mouse over this text</h1>
</body>
</html>
```

Bài tập 5 – Đèn tắt, bật

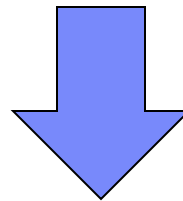


```
<html>
  <head>
    <script language="javascript">
      bLight=0;
      function changeimage() {
        if (bLight==0) {
          bLight=1; document.myimage.src="bulbon.gif";
        }
        else{
          bLight=0; document.myimage.src="bulboff.gif";
        }
      }
    </script>
  </head>
  <body>
    
  </body>
</html>
```

Bài tập 6 – Bổ sung các thành phần động

- Dynamic

Enter Your Mail Address



Enter Your Mail Address	<input type="text"/>	<input type="button" value="Add More"/>
Enter Your Mail Address	<input type="text"/>	<input type="button" value="Remove"/>
Enter Your Mail Address	<input type="text"/>	<input type="button" value="Remove"/>
Enter Your Mail Address	<input type="text"/>	<input type="button" value="Remove"/>
Enter Your Mail Address	<input type="text"/>	<input type="button" value="Remove"/>
Enter Your Mail Address	<input type="text"/>	<input type="button" value="Remove"/>

Bài tập 6

Enter Your Mail Address

```
<body>
```

```
  <div id="MainDiv">
```

```
    Enter Your Mail Address <input id="txtAddr1" type="text" />
```

```
    <input id="Button1" type="button" value="Add More"  
      onclick="Button1_onclick()" /></div>
```

```
</body>
```

```
<script language="javascript" type="text/javascript">
var NumOfRow=1;
function Button1_onclick()
{
NumOfRow++;
// lấy tham chiếu đến main Div
var mainDiv=document.getElementById('MainDiv');
// tạo một div để chứa các điều khiển khác
var newDiv=document.createElement('div');
newDiv.setAttribute('id','innerDiv'+NumOfRow);
//tạo span để chứa đoạn text
var newSpan=document.createElement('span');
newSpan.innerHTML="Enter Your Mail Address ";
// tạo một textbox mới
var newTextBox=document.createElement('input');
newTextBox.type='text';
newTextBox.setAttribute('id','txtAddr'+NumOfRow);
```

```
// tạo một nút button Remove mới
var newButton=document.createElement('input');
newButton.type='button';
newButton.value='Remove';
newButton.id='btn'+NumOfRow;
// gắn sự kiện click cho button mới
newButton.onclick=function RemoveEntry() { var
mainDiv=document.getElementById('MainDiv');
mainDiv.removeChild(this.parentNode);
}
// append span, textbox và the button vào div
newDiv.appendChild(newSpan);
newDiv.appendChild(newTextBox);
newDiv.appendChild(newButton);
// cuối cùng gắn div mới vào main div
mainDiv.appendChild(newDiv);
}
```

Bài tập 7

- Xây dựng một trang Web liệt kê danh sách các hóa đơn “**Orders.html**”:

Your orders

<input type="checkbox"/>	Order No	Customer Name	Order Date	Value
<input type="checkbox"/>	SHD 11	Nguyen Van A	12/2/2009	<input type="text" value="20000"/>
<input type="checkbox"/>	SHD 12	Nguyen Bich Thao	12/11/2009	<input type="text" value="55000"/>
Total:				<input type="text"/>
				<input type="button" value="Update"/> <input type="button" value="Save"/> <input type="button" value="Back"/> <input type="button" value="Reset"/>

- Nút “Save” cho phép các gửi các hóa đơn được đánh dấu đến file “**Process.php**” trên server để xử lý, nút “Back” cho phép quay lại trang trước đó và nút “Reset” dùng để xóa các nút chọn (checkbox).
- Nút chọn trên cùng cho phép có thể chọn tất cả hoặc hủy chọn tất cả các dòng hóa đơn bên dưới.
- Nếu tất cả các dòng bên dưới đều được chọn thì nút trên cùng cũng được đánh dấu.

<input type="checkbox"/>	Order No	Customer Name	Order Date	Value
<input type="checkbox"/>	SHD 11	Nguyen Van A	12/2/2009	20000
<input type="checkbox"/>	SHD 12	Nguyen Bich Thao	12/11/2009	55000
Total:				
				<input type="button" value="Update"/> <input type="button" value="Save"/> <input type="button" value="Back"/> <input type="button" value="Reset"/>

```
<form name="frmList">
```

```
<input type="checkbox" name="chkall"
onClick="docheck(document.frmList.chkall.checked);"/>
```

```
<input type="checkbox" name="chkid" value="SHD11"
onClick="docheckone();"/>
```

```
<input type="checkbox" name="chkid" value="SHD12"
onClick="docheckone();"/>
```

```
</form>
```

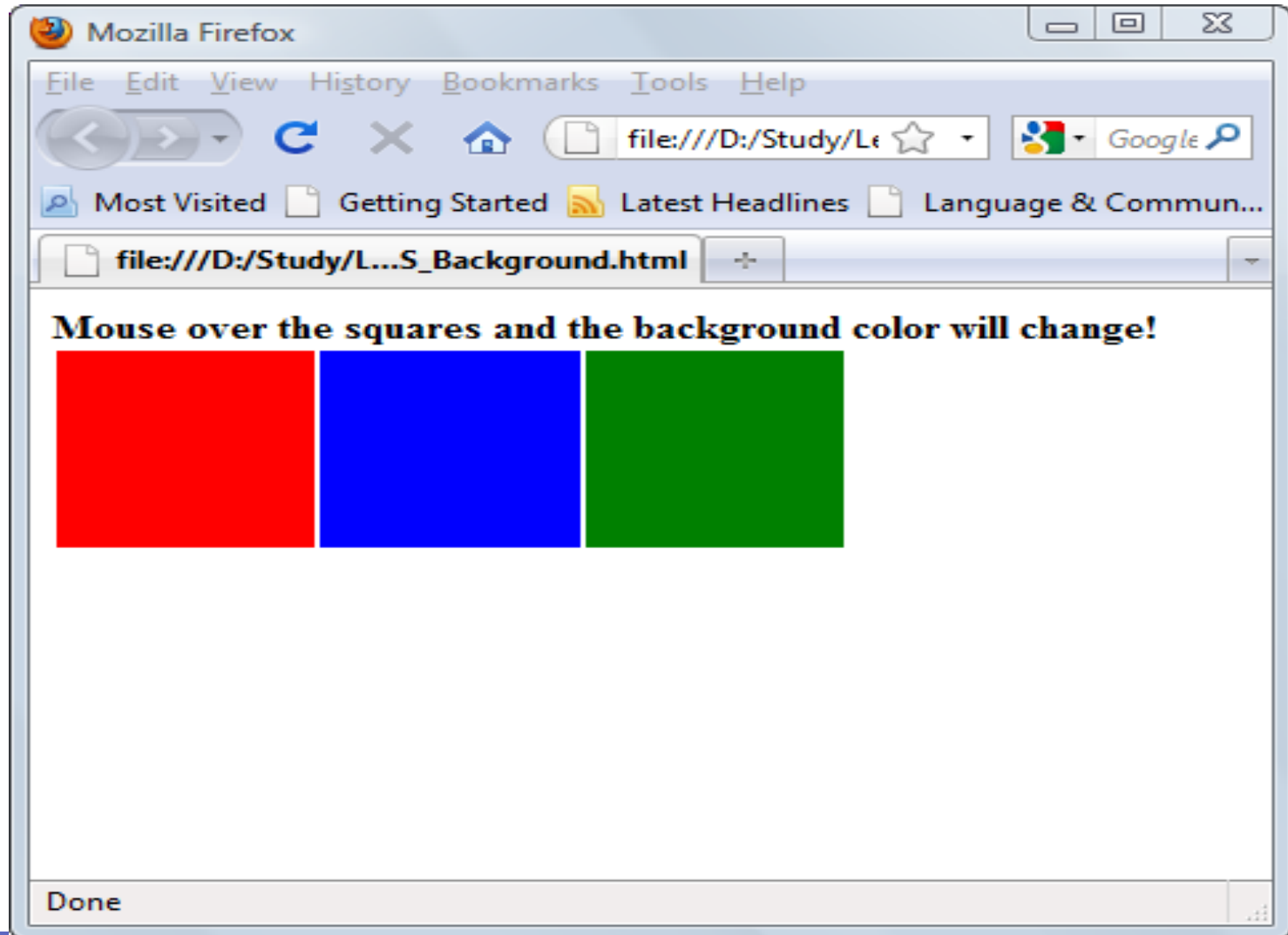

Bài tập 7

```
<script language="javascript">  
function docheck(status){  
    var num = document.frmList.chkid.length;  
    for(var i=0; i<num; i++)  
        document.frmList.chkid[i].checked = status;  
}  
</script>
```

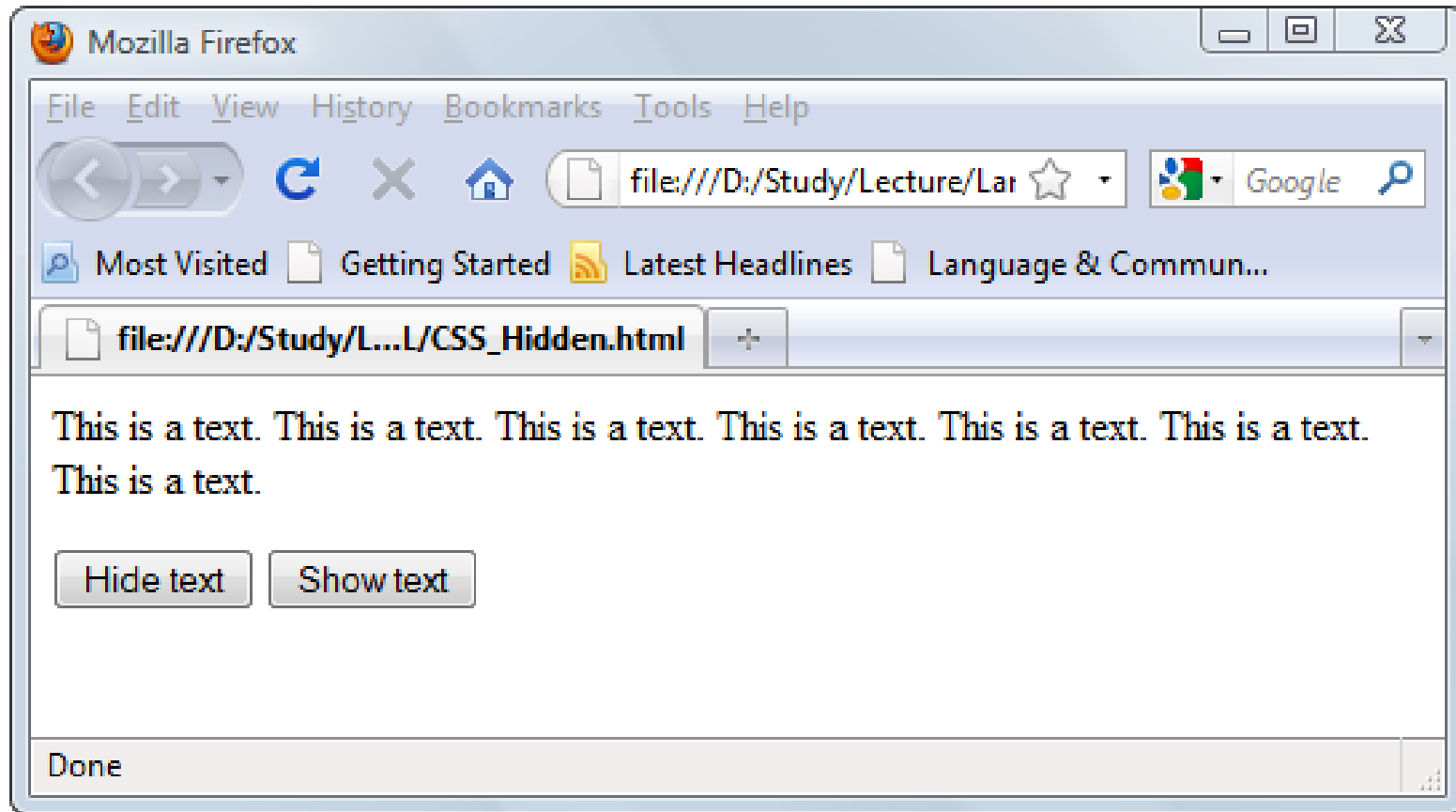
Bài tập 7

```
<script language="javascript">
function docheckone(){
    var checked = true;
    var num = document.frmList.chkid.length;
    for(var i=0; i<num; i++)
        if(document.frmList.chkid[i].checked==false){
            checked = false;
            break;
        }
    document.frmList.chkall.checked = checked;
}
</script>
```

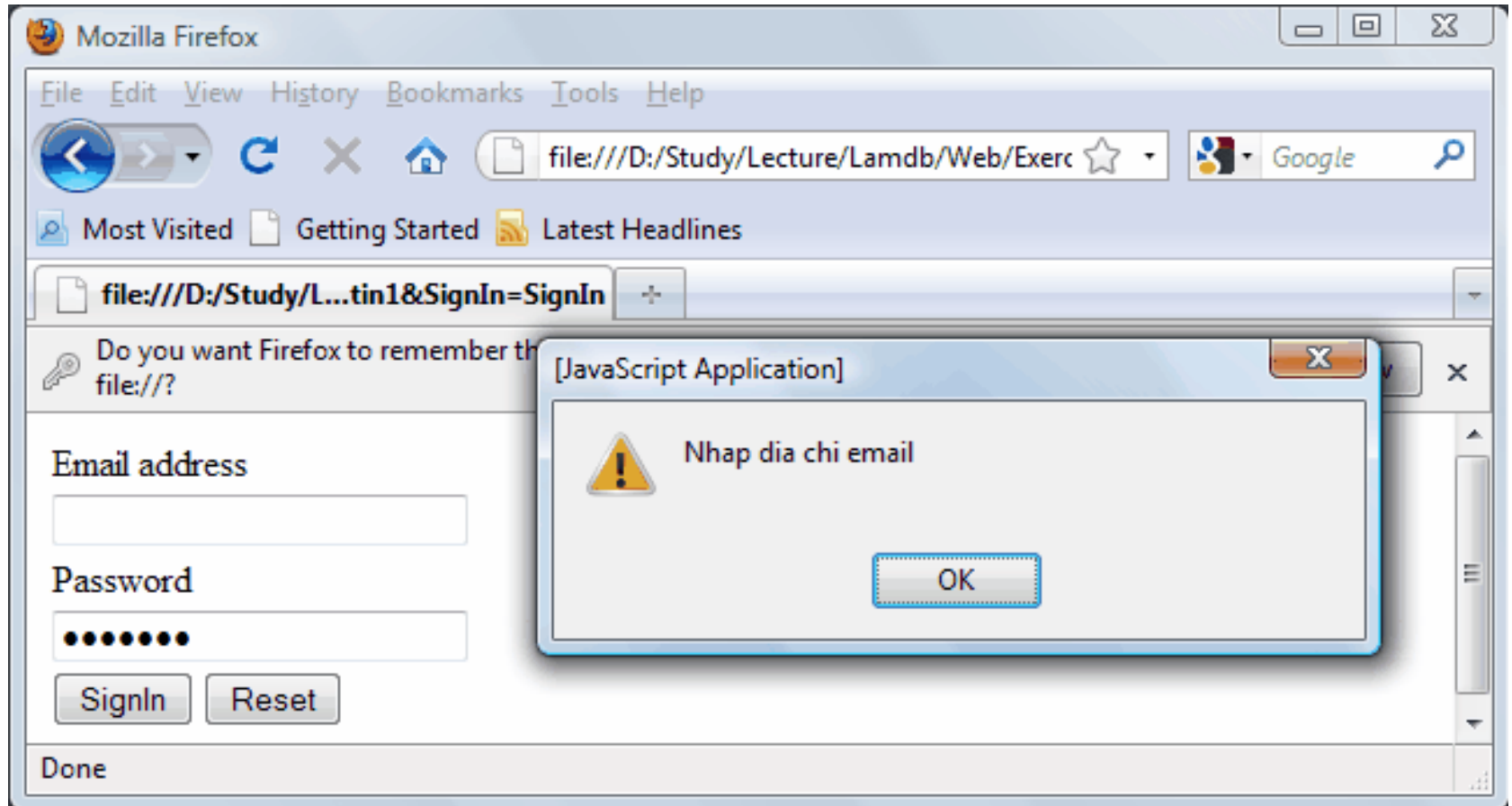
Bài tập 8 – Đổi màu nền thành màu của ô



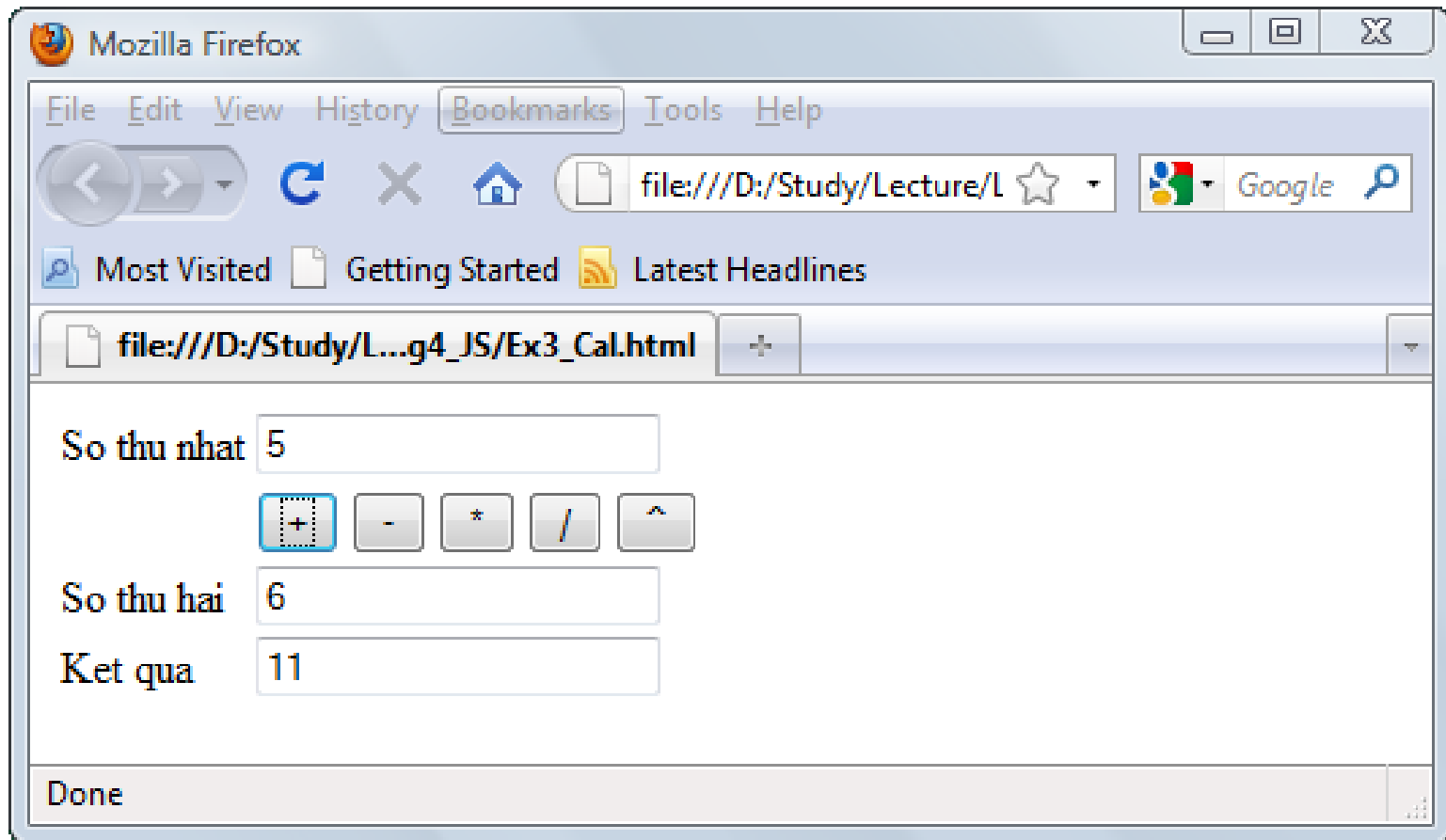
Bài tập 9 - Ẩn hiện văn bản



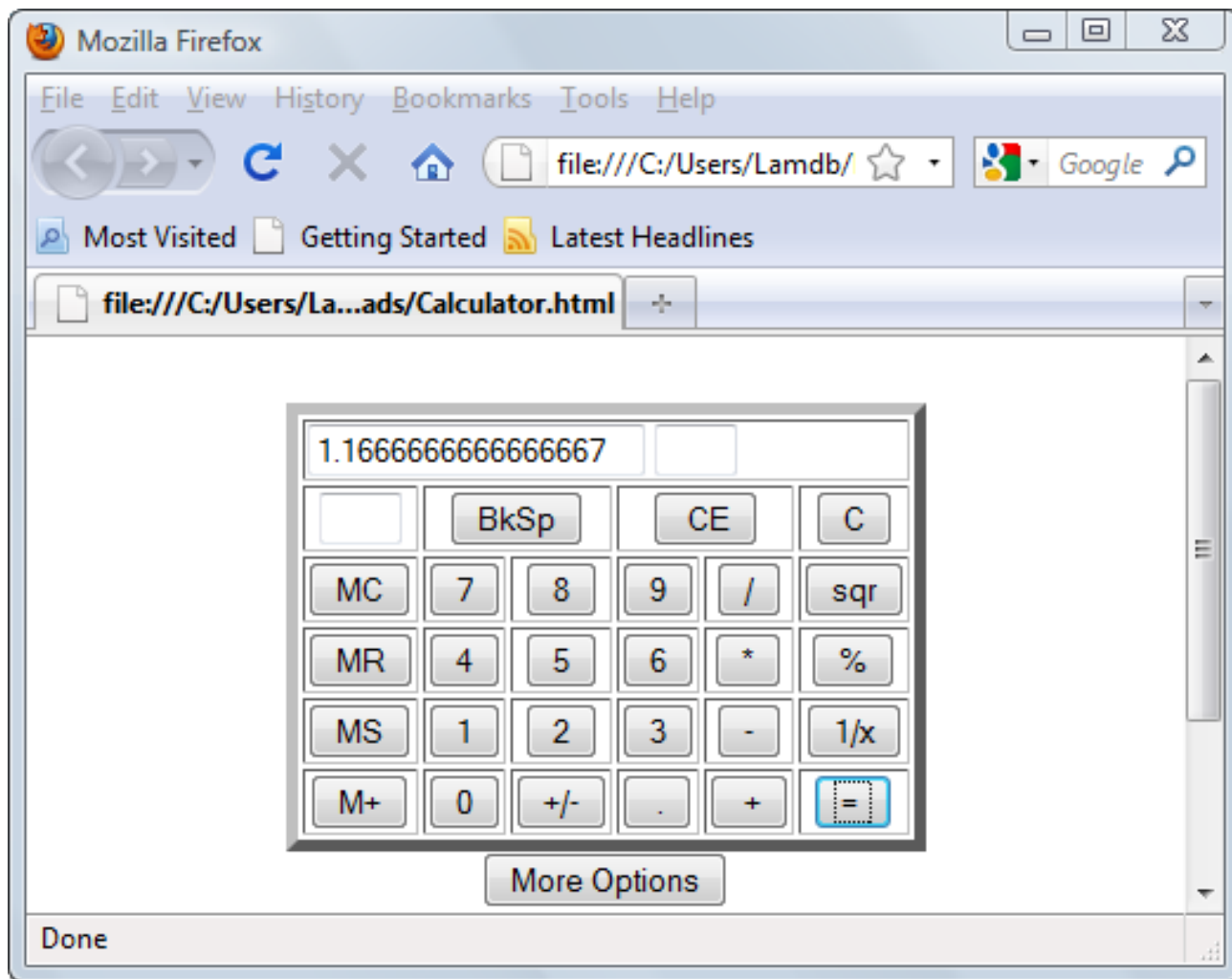
Bài tập 10 – Kiểm tra các ô nhập liệu



Bài tập 11 – Tính toán đơn giản



Bài tập 12 – Mô phỏng máy tính



Bài tập 13 – Đồng hồ

