



BÀI 1: JAVASCRIPT CƠ BẢN (PHẦN 1)

- ☐ Giới thiệu về Javascript
- ☐ Đưa Javascript vào trong HTML
- ☐ Kiểu dữ liệu và biến
- ☐ Các toán tử
- ☐ Các cấu trúc điều khiển



□ Javascript:

- ❖ Là một ngôn ngữ kịch bản
- ❖ Là một ngôn ngữ thông dịch
- ❖ Được thiết kế để tăng khả năng tương tác với người dùng của HTML
- ❖ Được nhúng trực tiếp vào trong trang HTML
- ❖ Chỉ là một đoạn mã lệnh máy tính có thể thực hiện được
- ❖ Không phân biệt kiểu dữ liệu như C++ hay Java. Nó không phân biệt rõ ràng giữa các kiểu xâu kí tự hay mảng.

- ❑ Javascript cho phép:
 - ❖ Đưa các đoạn mã chương trình động vào trong HTML. Ví dụ hiển thị ngày tháng hiện tại
 - ❖ Trang web có thể phản ứng lại với các sự kiện. Ví dụ khi người dùng nhấn vào một nút bấm trong một form thì dữ liệu trong form sẽ được kiểm tra
 - ❖ Đọc và viết các thành phần HTML

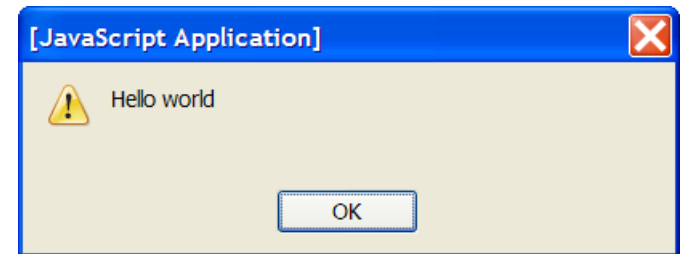
❑ Lịch sử

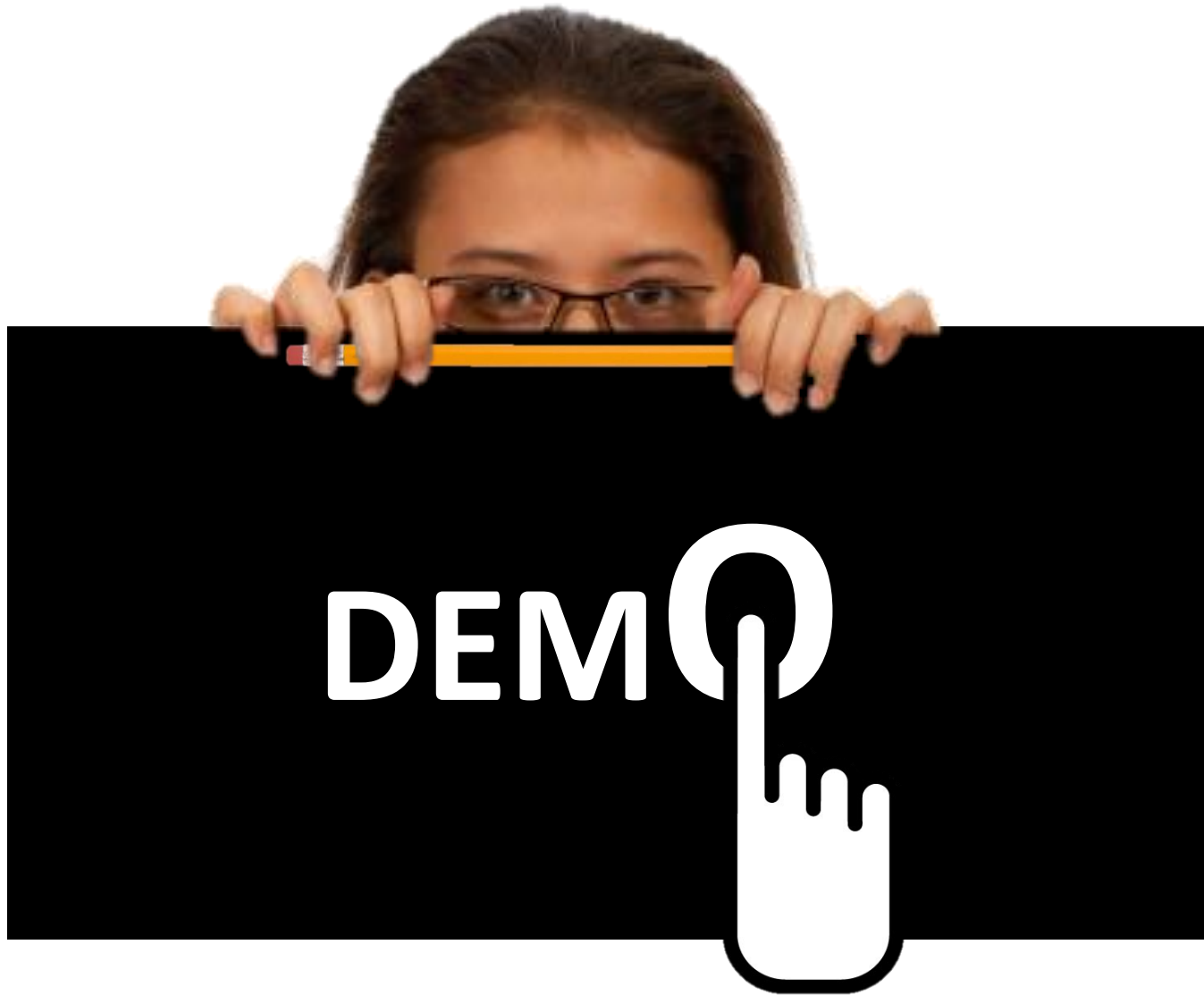
- ❖ **Java** được đưa ra bởi Sun nhằm nâng cao và hỗ trợ khái niệm lập trình hướng đối tượng
- ❖ **Livescript** được tạo ra bởi Netscape nhằm đưa lập trình hướng đối tượng vào Web. Về sau, nó được đổi tên là **Javascript** vì nó giống Java
- ❖ Microsoft cũng không muốn đứng ngoài cuộc và đưa ra **JScrip**t, một phiên bản Javascript sở hữu riêng của Microsoft

- Ví dụ đơn giản

```
<html>
<body>
  <script type = "text/javascript">
    alert("Hello world");
  </script>
</body>
</html>
```

- Mã javascript được viết bên trong thẻ `<script>... </script>`





❑ Mã javascript có thể được đặt trong phần HEAD

❖ <head>

 <script type="text/javascript">

 </script>

</head>

❖ Đảm bảo script được nạp trước khi người sử dụng có thể dùng nó

❖ Script được thực hiện khi một hàm của nó được gọi trong phần BODY

➤ Đặt lời gọi hàm bên trong thẻ <script> ... </script>

❑ Mã javascript có thể được đặt trong phần BODY

- ❖ Script được thực hiện khi trang web được nạp.
- ❖ Script tạo ra nội dung của trang web tại vị trí mà nó được viết

❖ `<html>`

`<head> ... </head>`

`<body>`

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

`....`

`</script>`

`</body>`

`</html>`

❑ Đặt javascript trong một file ngoài

- ❖ Mã lệnh javascript đặt trong file .js
- ❖ Để sử dụng các mã lệnh này, cần chỉ ra đường dẫn đến file trong trường src:

```
<head>
```

```
<script src = "url" >
```

```
</head>
```

- ❖ File .js có thể sử dụng bởi nhiều file HTML
- ❖ Khi gọi 1 hàm hay thủ tục trong file .js, lời gọi phải được viết trong thẻ <script> ... </script>

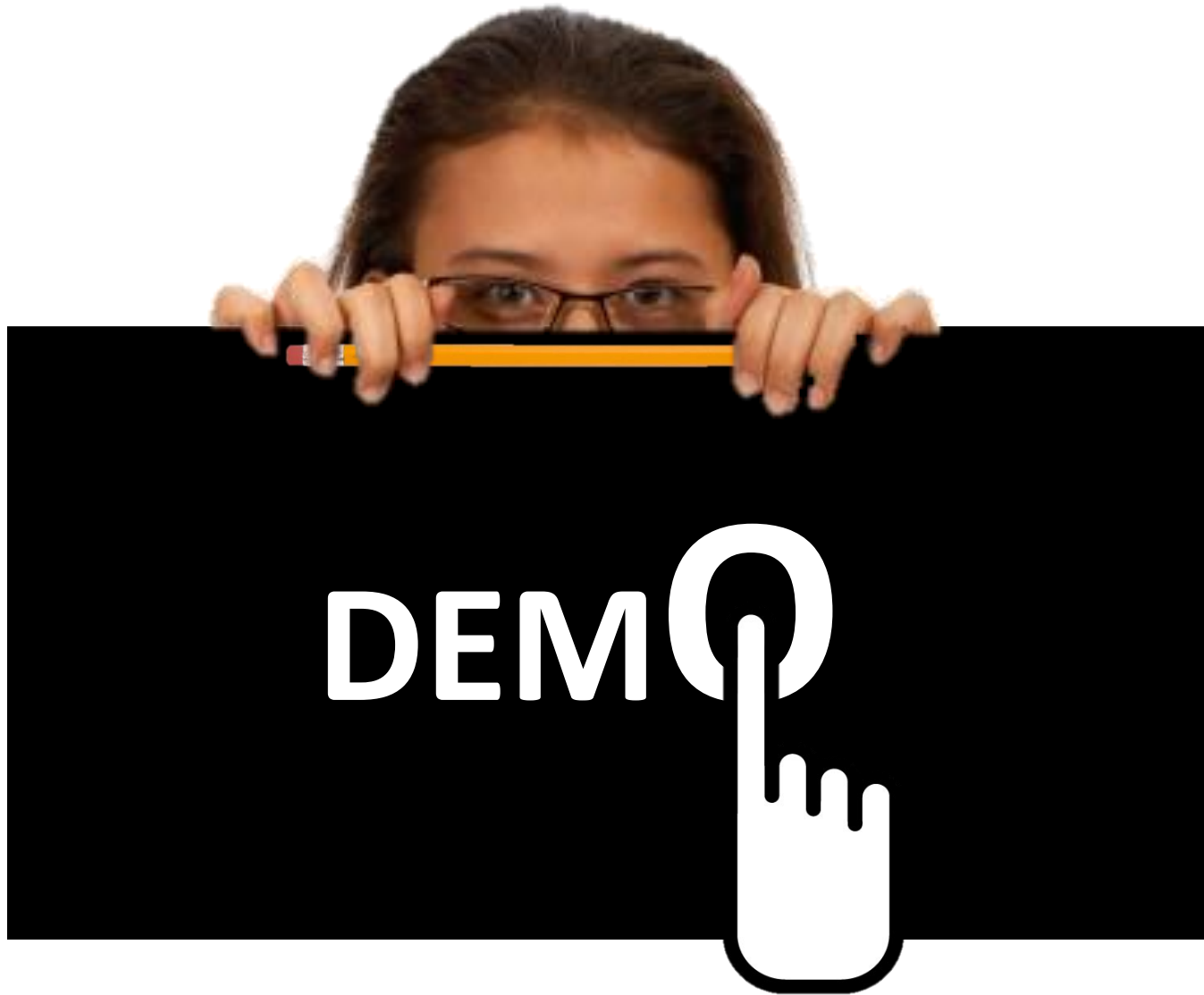
❑ Ví dụ:

❖ HTML:

```
<html>
<head>
  <script src = "hello.js">
</head>
<body>
<script> hello() </script>
</body>
</html>
```

❖ Javascript:

```
function hello()
{  alert("Hello world"); }
```



- ❑ Tương tự như trong Java
 - ❖ Mỗi câu lệnh kết thúc bằng 1 dấu ;
 - ❖ Các câu lệnh có thể được nhóm với nhau bởi dấu { và dấu }
- ❑ `<script type="text/javascript">`
{
document.write("<h1>This is a header</h1>");
document.write("<p>This is a paragraph</p>");
document.write("<p>This is another paragraph</p>");
}
</script>
- ❑ Ghi chú:
 - ❖ `//` đây là một ghi chú chỉ gồm một dòng
 - ❖ `/*` đây là một ghi chú có thể kéo dài trên nhiều dòng `*/`

□ Các kiểu dữ liệu cơ bản

❖ Kiểu boolean

- 2 giá trị: true, false

❖ Kiểu số

- Số nguyên:

- ✓ 2002 (cơ số 10)
- ✓ 0xFF (cơ số 16)

- Số thực:

- ✓ 3.1415926535
- ✓ 3.0E+8

❖ Kiểu xâu kí tự

- Chứa một chuỗi các kí tự tùy ý

➤ Các kí tự đặc biệt

- ✓ \b: xoá 1 kí tự trước đó
- ✓ \t: tab
- ✓ \n: xuống dòng
- ✓ \f: nhảy sang trang tiếp theo
- ✓ \": "
- ✓ \': '
- ✓ \\: \

❖ null: thiếu giá trị

❖ undefined: một giá trị không được định kiểu

❑ Một số tính chất

❖ infinity: giá trị vô cùng lớn

❖ NaN: biểu diễn kết quả của một phép toán không hợp lệ. Ví dụ chia cho 0.

□ Các biến đơn

- ❖ Là một đơn vị bộ nhớ lưu trữ dữ liệu và được truy cập qua định danh
- ❖ Quy tắc tên biến
 - phân biệt giữa chữ hoa và chữ thường
 - Bắt đầu bằng 1 chữ cái hoặc “_”
 - Khác với các từ khoá của Javascript
- ❖ Cách khai báo
 - `var ten_bien = gia_tri;`
 - `ten_bien = gia_tri;`
- ❖ Kiểu dữ liệu không được đặc tả khi khai báo
- ❖ Thời gian sống của biến –Life time

□ Ví dụ:

❖ Các biến nhận giá trị kiểu boolean

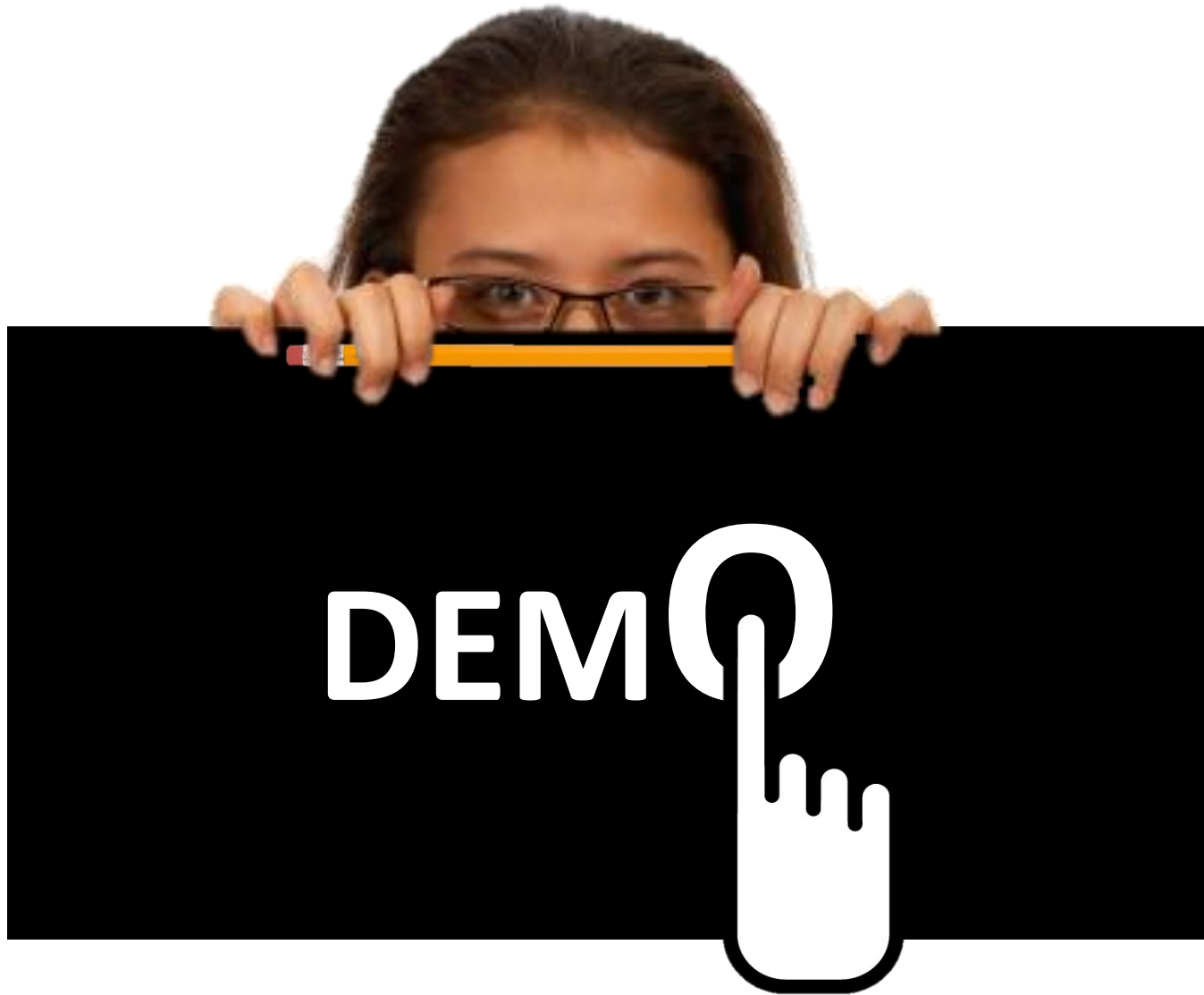
- `var fini = false;`
- `termine = true;`
- `non_non = (0 == 1);`

❖ Các biến kiểu số:

- `var nam = 2002, thang = 3, ngay = 30;`
- `n = 27;`

❖ Các biến kiểu chuỗi:

- `var ten = "Jean-Claude";`
- `ho = "Nadeau\n";`
- `var titre = new String("Day la mot xau ki tu");`



JavaScript 1.0	JavaScript 1.1	JavaScript 1.2	JavaScript 1.4
break	boolean	delete	catch
continue	typeof	do	instanceof
else	void	export	throw
for		import	try
function		switch	
if			
in			
new			
return			
this			
var			
while			
with			

abstract	enum	interface	static
byte	extends	long	super
case	false	native	synchronized
char	final	null	throws
class	finally	package	transient
const	float	private	true
debugger	goto	protected	volatile
default	implements	public	
double	int	short	

☐ Toán tử số học

Toán tử	Giải thích
+	Cộng
-	Trừ
*	Nhân
/	Chia
%	Chia lấy dư

☐ Toán tử logic

Toán tử	Giải thích
&	Và
	Hoặc
^	XOR
!	NOT

❑ Toán tử quan hệ và toán tử bằng

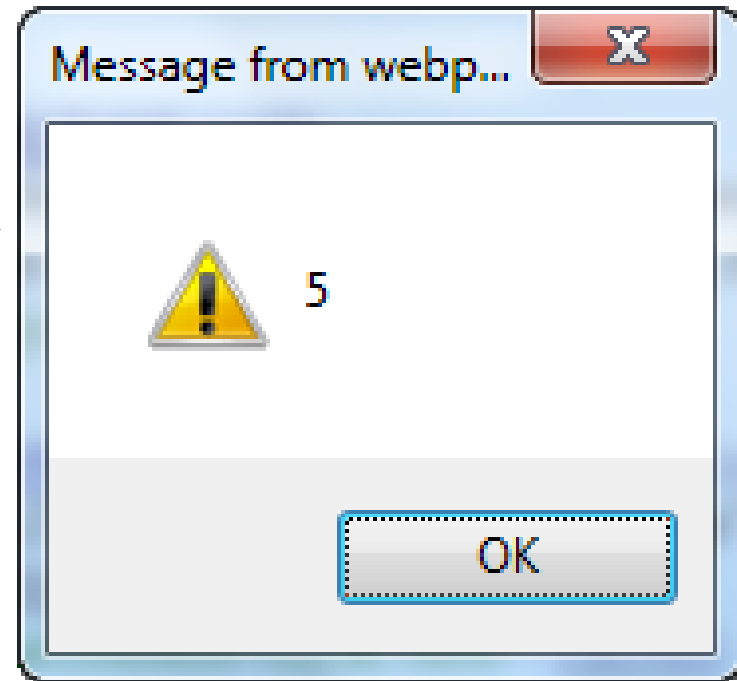
Toán tử	Giải thích
>	Lớn hơn
<	Bé hơn
>=	Lớn hơn hoặc bằng
<=	Bé hơn hoặc bằng
==	Bằng
!=	Khác

Toán tử một ngôi

Toán tử	Giải thích
+	Chuyển toán hạng sang số dương
-	Chuyển toán hạng sang số âm
++	Tăng
--	Trừ

```
var x = 4;  
x++;  
alert (x);
```

KẾT QUẢ

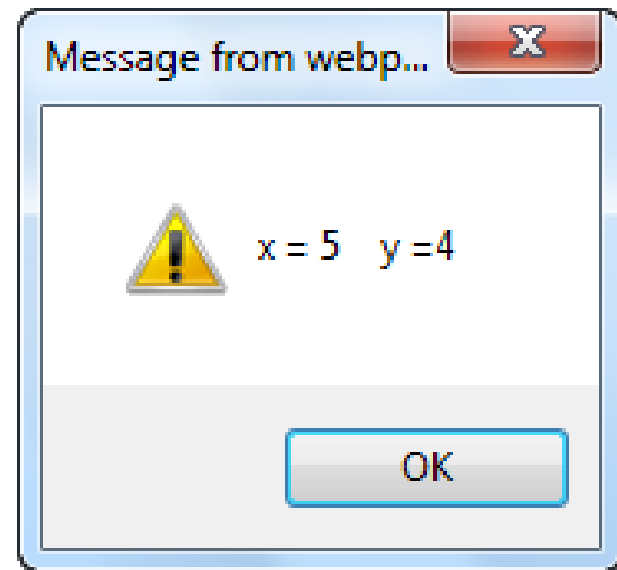


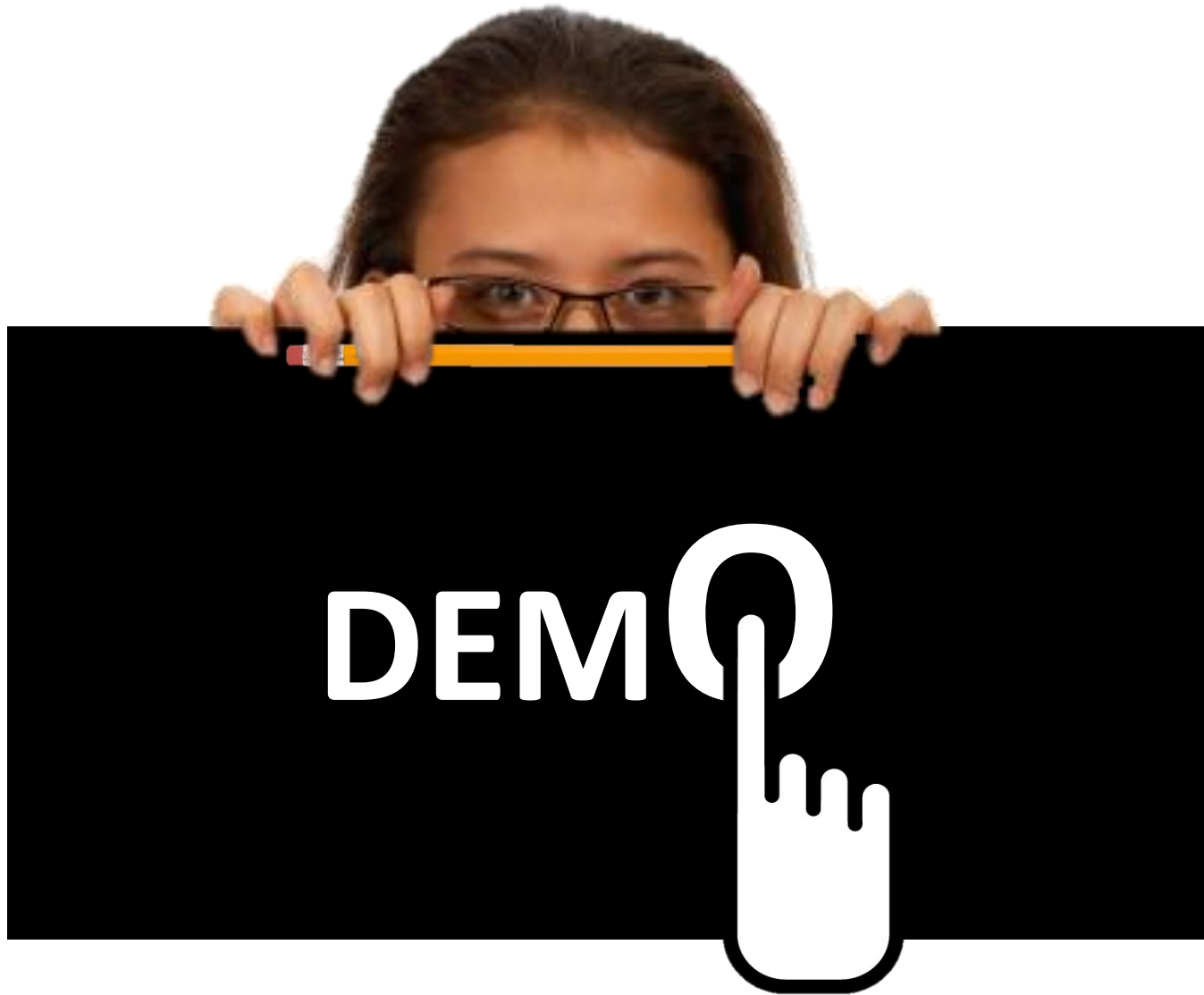
❑ Sự khác nhau giữa ++x và x++

```
var x = 4;  
var y = ++x;  
alert ("x = "+x+"    y =" +y);
```



```
var x = 4;  
var y = x++;  
alert ("x = "+x+"    y =" +y);
```







BÀI 1: JAVASCRIPT CƠ BẢN (PHẦN 2)

CẤU TRÚC ĐIỀU KHIỂN

- ☐ Cấu trúc **tuần tự**
- ☐ Cấu trúc **lựa chọn**
- ☐ Cấu trúc **lặp**

- ☐ Lệnh lựa chọn **đơn**
- ☐ Lệnh lựa chọn **kép**
- ☐ Lệnh **đa** lựa chọn

❑ Cú pháp

```
if (điều_kiện) {  
    //Thực hiện  
}
```

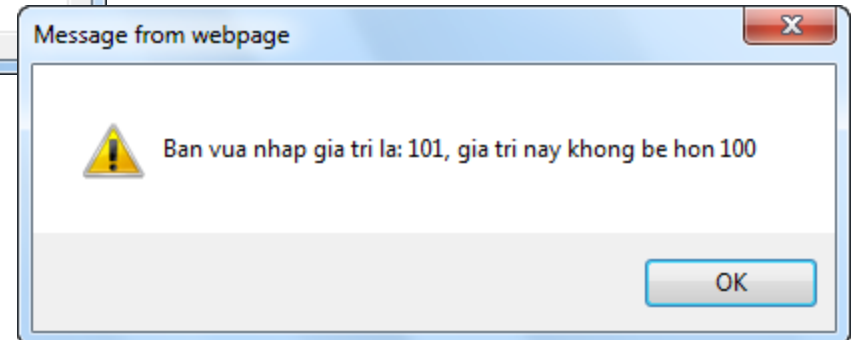
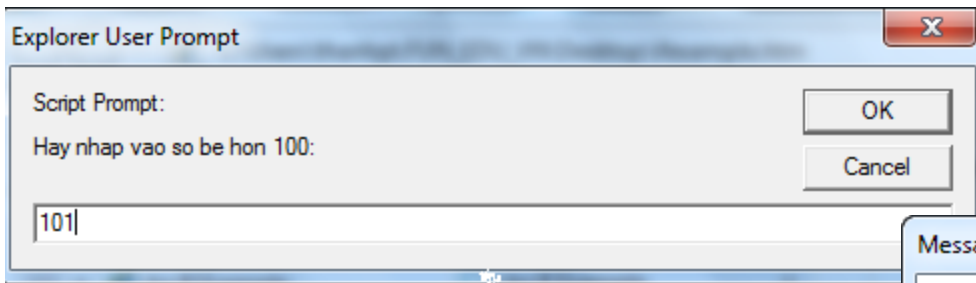
❑ Các trường hợp

```
if (! điều_kiện) {  
    //Thực hiện  
}
```

```
if (x < 350) {  
    //Thực hiện  
}
```

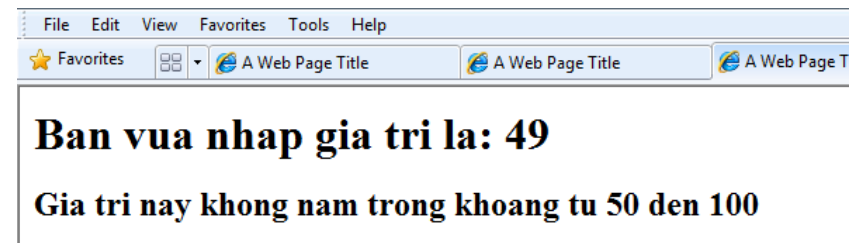
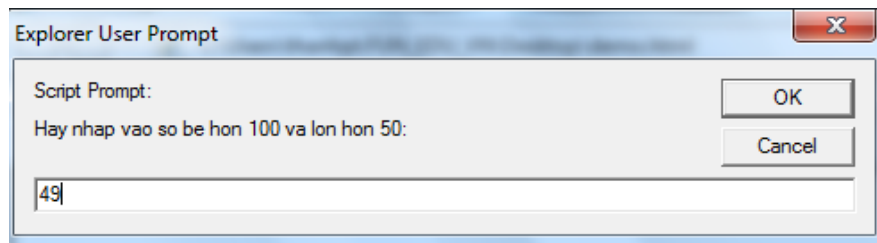
```
var x = 3;  
var y = 4;  
if (x == y) {  
    //Thực hiện  
}
```

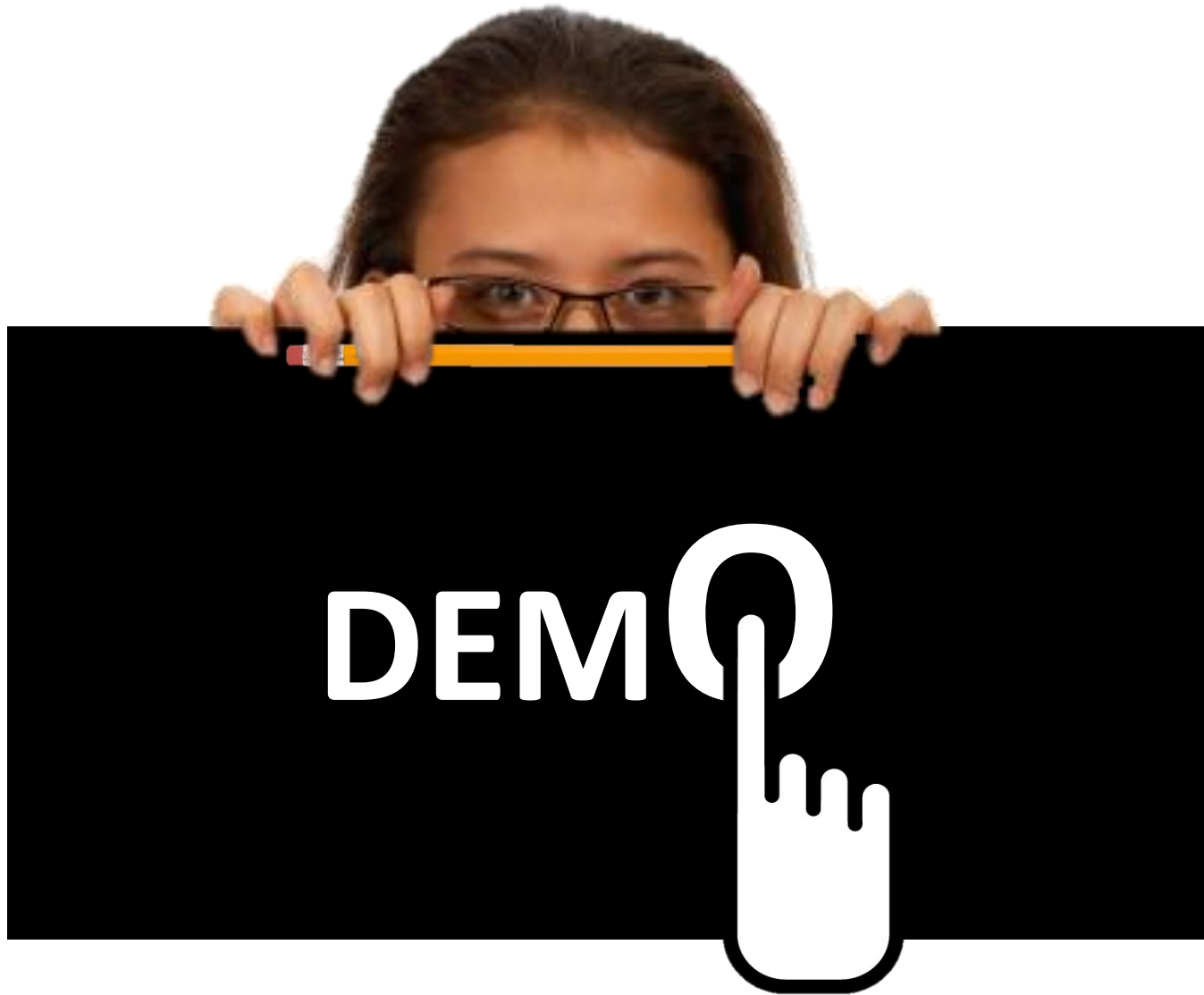
```
var x = prompt("Hay nhap vao so be hon 100:");  
if (x >= 100)  
{  
    alert("Ban vua nhap gia tri la: " + x + ", gia tri nay khong  
    be hon 100");  
}
```



- ❑ Dùng toán tử logic để **kết hợp** nhiều **mệnh đề đơn** thành một **mệnh đề phức tạp** hơn

```
var x = prompt("Hay nhap vao so be hon 100 va lon hon 50:");  
  
if ((x < 51) || (x > 99)) {  
    document.write("<h1>Ban vua nhap gia tri la: " + x + "</h1>");  
    document.write("<h2>Gia tri nay khong nam trong khoang tu 50  
den 100</h2>");  
}
```





- Giới thiệu về Javascript
- Đưa Javascript vào trong HTML
- Kiểu dữ liệu và biến
- Các toán tử
- Các cấu trúc điều khiển





Cảm ơn