

DHTML

JAVA  
SCRIPT

LẬP TRÌNH  
WEB

PHP

# Chương 5

## JavaScript nâng cao

# Mục tiêu bài học



- ❖ Kiểm tra tính hợp lệ của Form
- ❖ Sử dụng thời gian

❖ Đối tượng Form chứa 3 thuộc tính:

- Accept
- Action
- Method

❖ Ví dụ:

```
<Form ACTION="Simple.htm" Accept="TEXT/HTML"  
Method="POST">
```

# Đối tượng Textfield (1)



- ❖ Textfield nhận biết các sự kiện onBlur, onFocus và onChange
  - onFocus xuất hiện khi nhấp chuột vào bên trong trường văn bản.
  - onBlur xảy ra khi người dùng di chuyển ra khỏi trường văn bản
  - onChange xảy ra khi người dùng có sự thay đổi trong trường văn bản và sau đó di chuyển ra khỏi trường văn bản

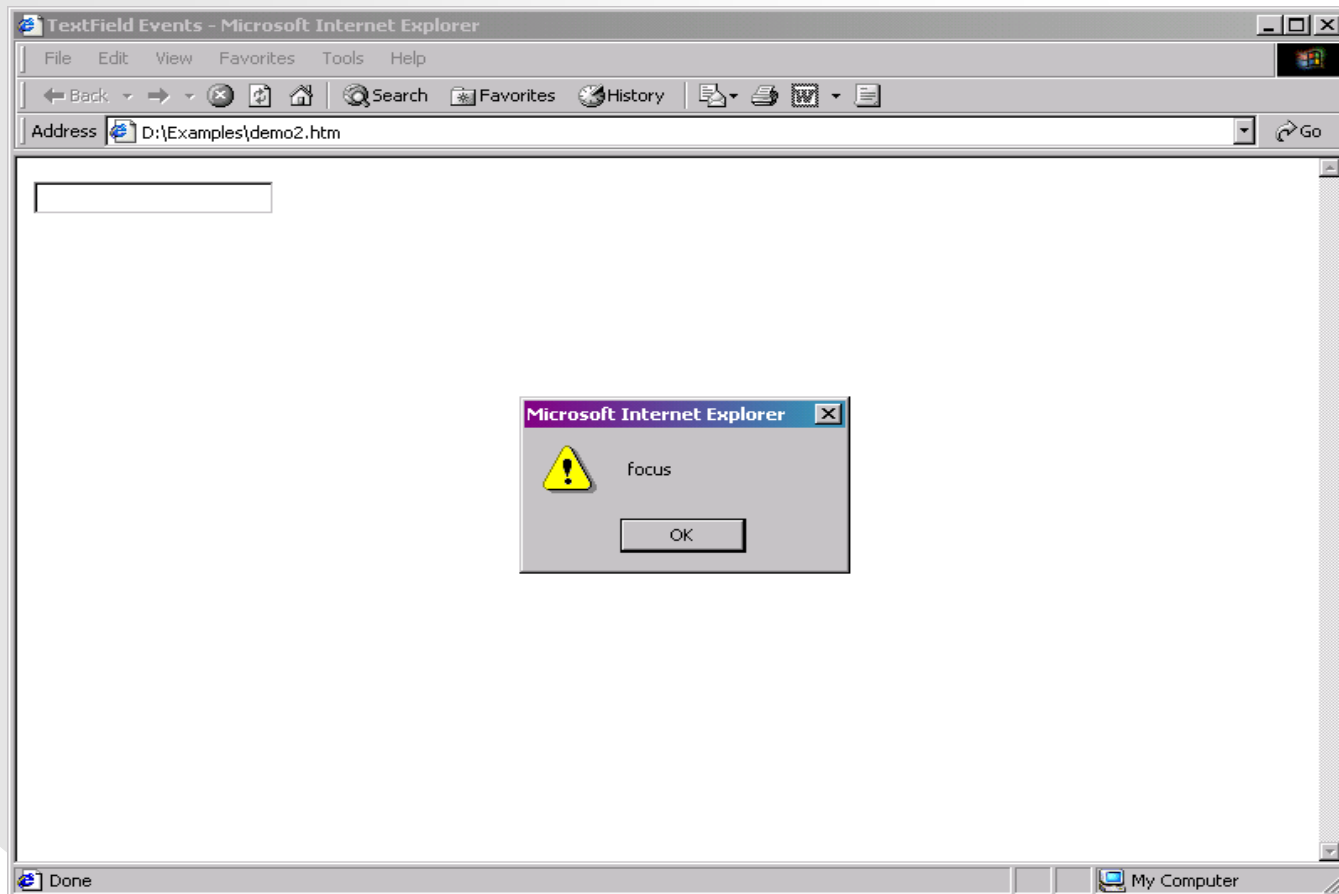
# Đối tượng Textfield (2)



Ví dụ:

```
<input type="text" name="first_text"  
      onFocus="writelt('focus');"  
      onBlur="writelt('blur');"  
      onChange="writelt('change');">
```

# Đối tượng Textfield(3)



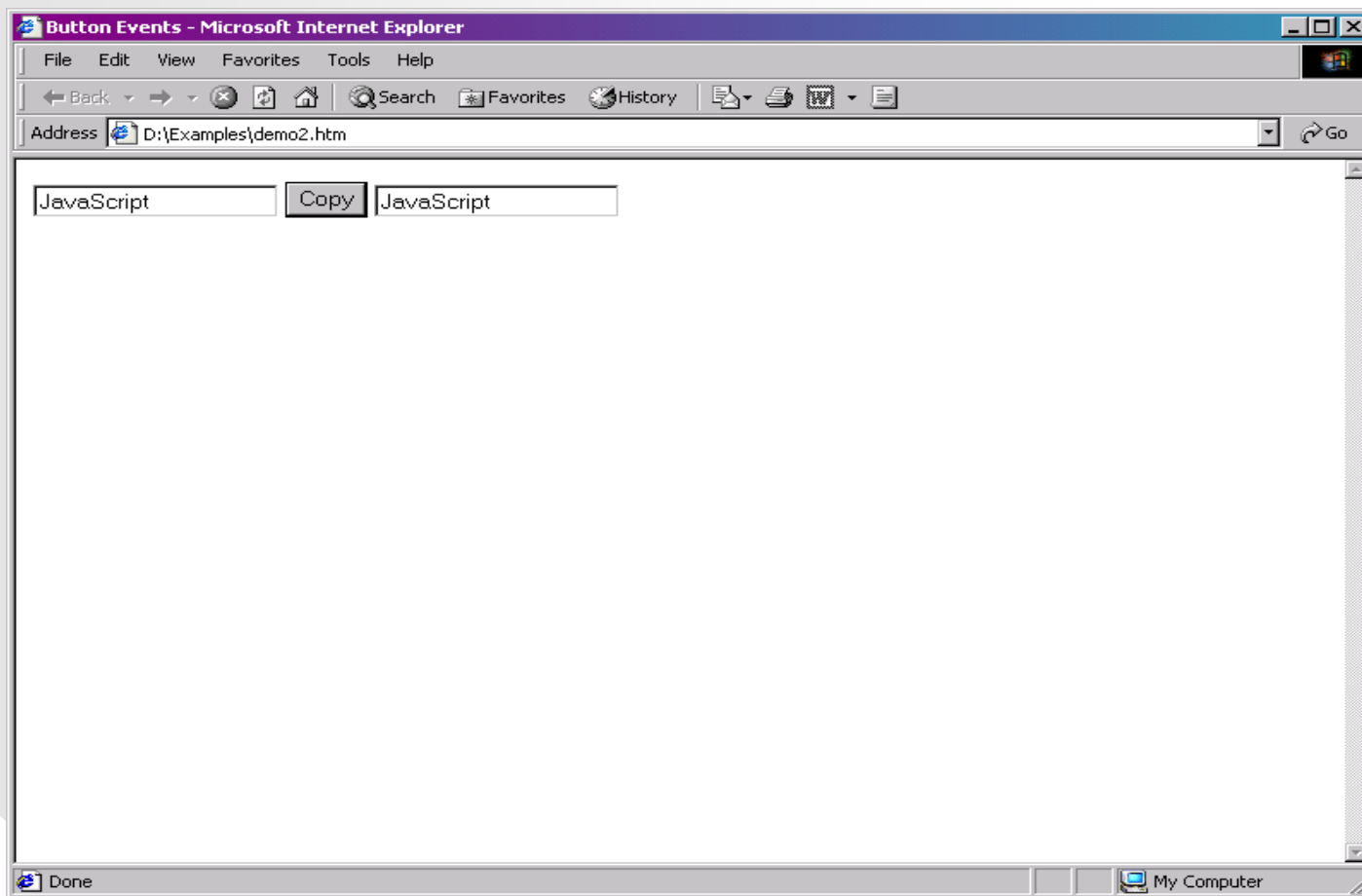
# Đối tượng Command Button(1)



- ❖ Command button nhận biết sự kiện onClick
- ❖ onClick xuất hiện khi người sử dụng nhấp chuột vào command button

```
<INPUT TYPE="button" value="Copy"  
onClick="writelt(myfm.first_text.value);">
```

# Đối tượng Command Button(2)



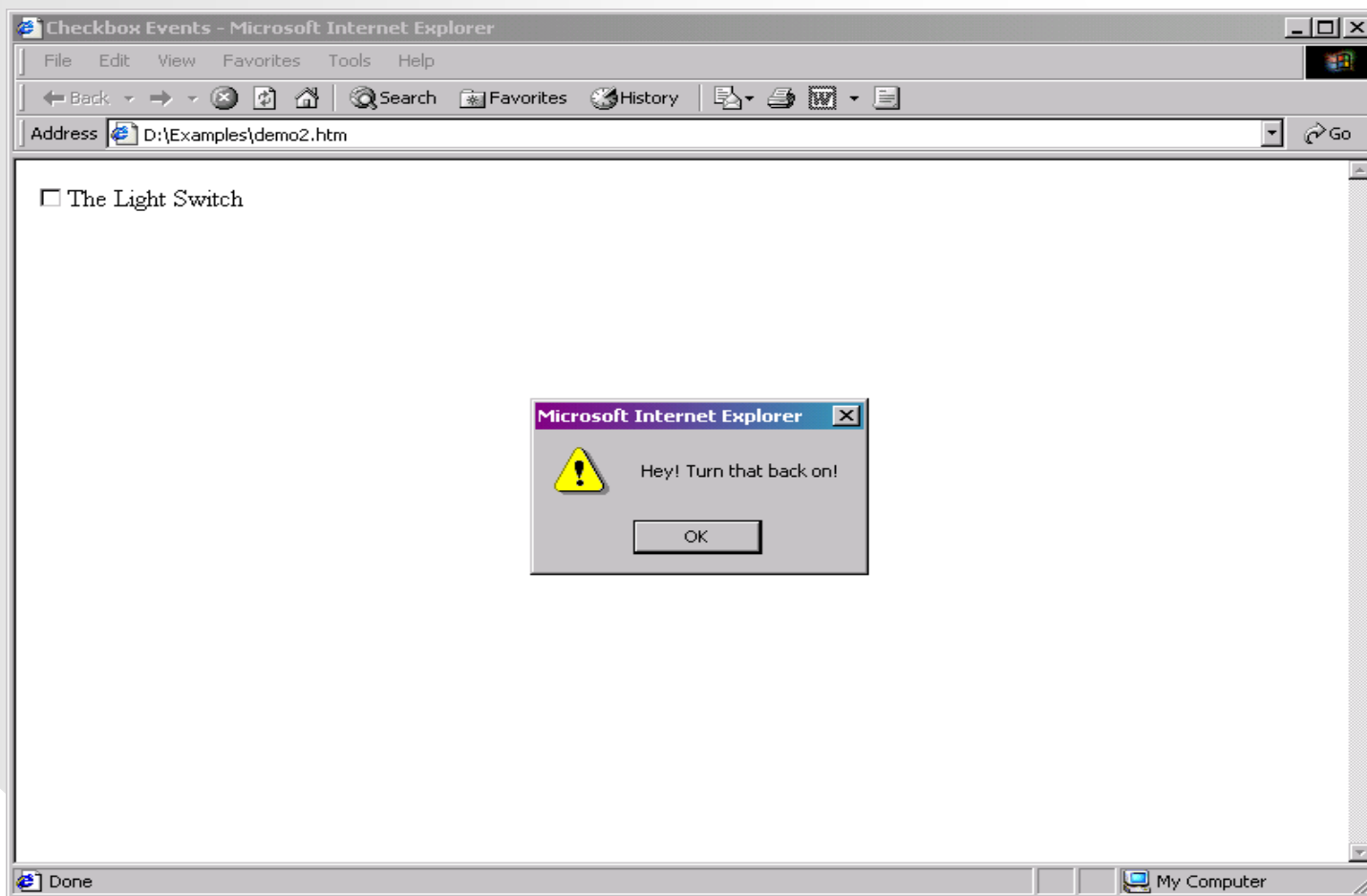


# Đối tượng Checkbox(1)



- ❖ Checkbox là một đối tượng form HTML, hoạt động theo cơ chế bật tắt
- ❖ Checkbox có thể được Check hoặc không
- ❖ Giống như nút lệnh checkbox cũng hiểu được sự kiện onClick

# Đối tượng Checkbox(2)

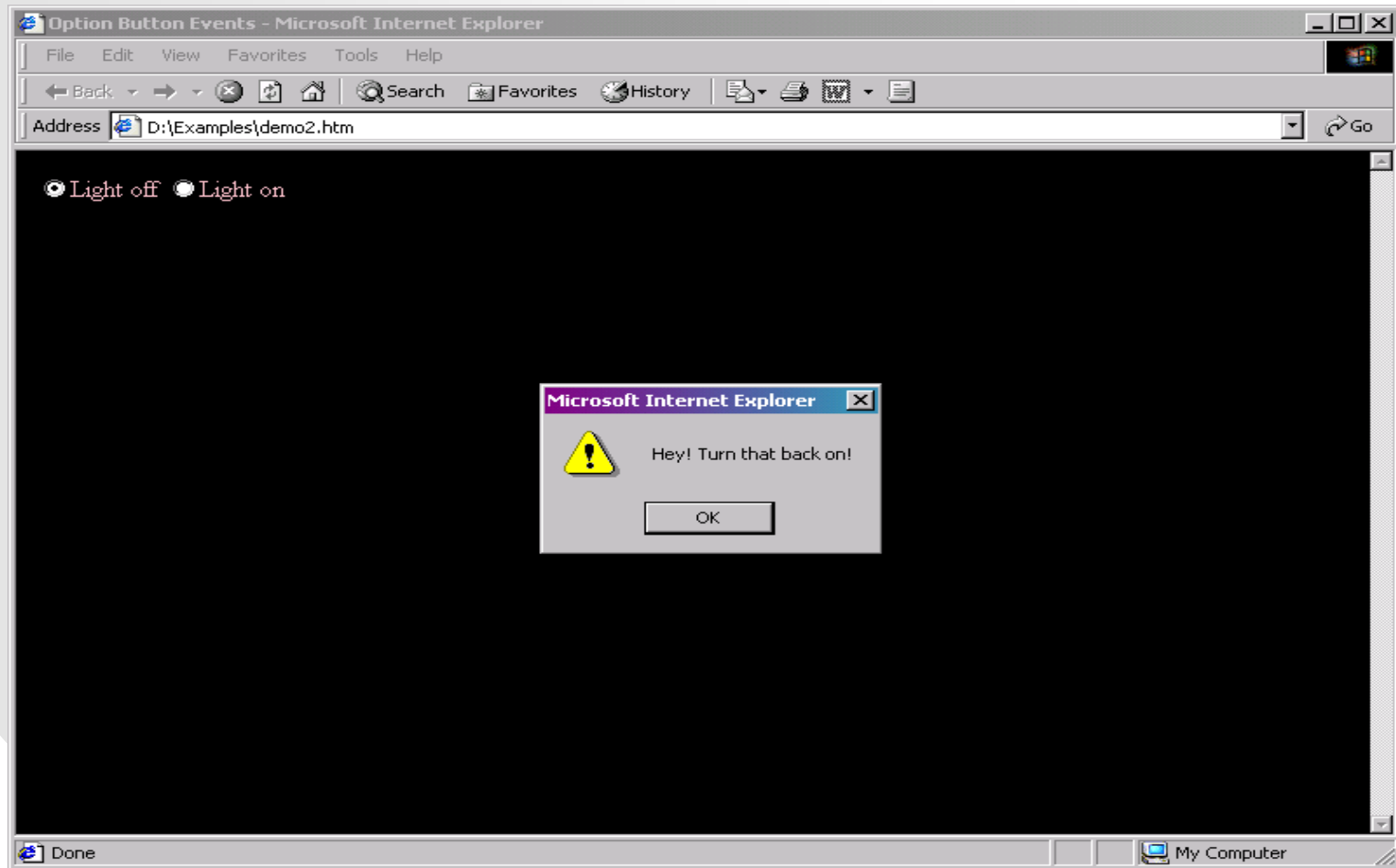


# Radio Button (1)



- ❖ Radio buttons gần giống như checkboxes
- ❖ Sự khác biệt ở đây là chỉ có một Radio được chọn. Khi một Radio được chọn, nó sẽ giữ nguyên sự lựa chọn đó cho đến khi nút khác được chọn.
- ❖ Radio button hiểu được sự kiện onClick.

# Radio Button (1)

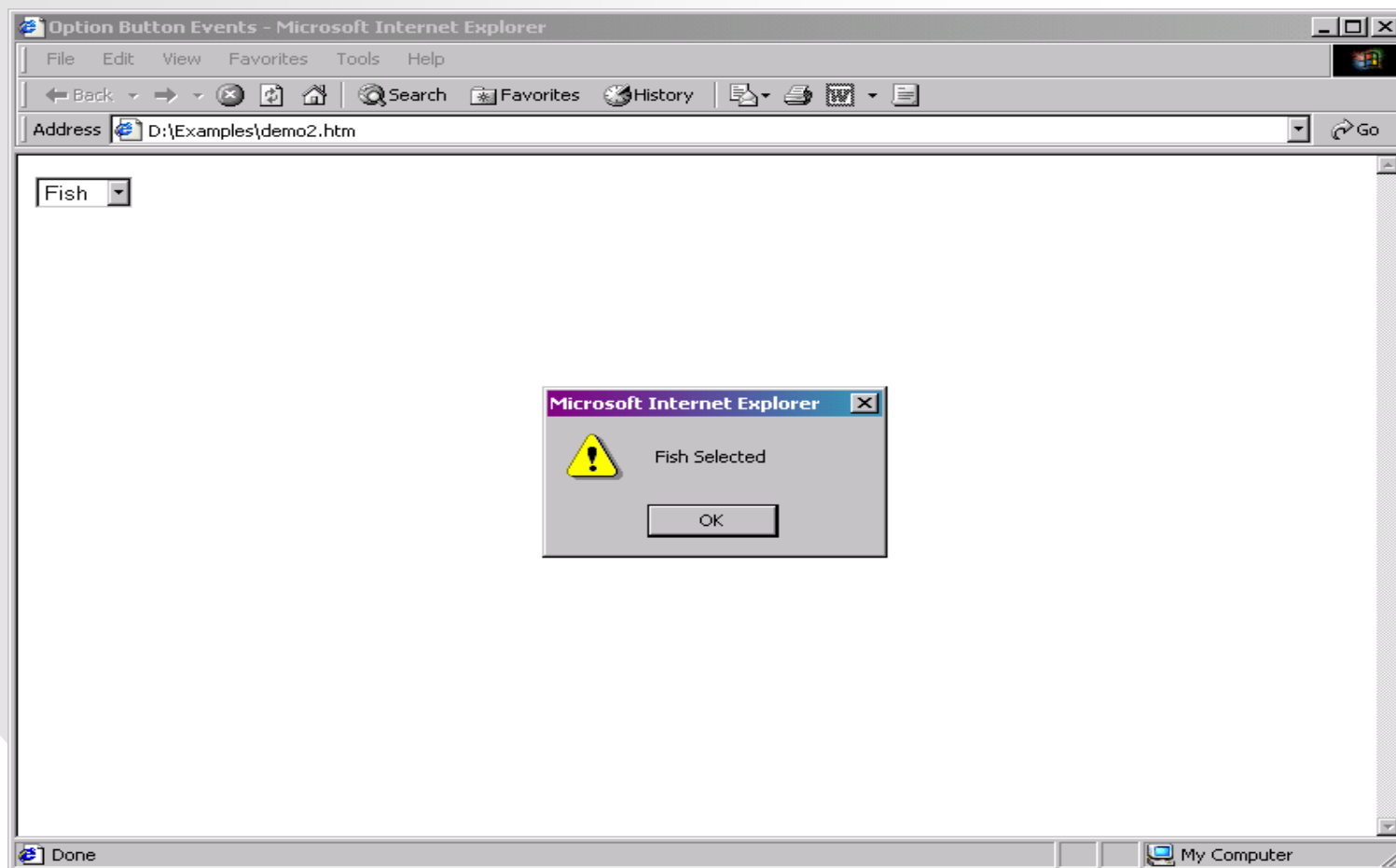


# ComboBox/Đối tượng Select(1)



- ❖ Đối tượng ComboBox trong form HTML xuất hiện giống như một danh mục sổ xuống hoặc danh mục cuộn của các tùy chọn
- ❖ Có thể sử dụng thanh cuộn để thay đổi sự hiển thị danh sách các lựa chọn
- ❖ ComboBox hỗ trợ các sự kiện onBlur, onFocus, and onChange

# ComboBox/Đối tượng Select (2)



- ❖ Việc kiểm tra rất quan trọng, vì có thể có trường không chứa dữ liệu.
- ❖ Cũng có thể có trường chứa dữ liệu không hợp lệ.
- ❖ Xem ví dụ sau:

```
<HTML>  
<HEAD>  
  <TITLE> Form Events </TITLE>  
  <SCRIPT LANGUAGE="JavaScript">  
  <!--
```

# Kiểm tra tính hợp lệ của Form(2)



```
function validateFirstName()  
{  
    var str= form1.fname.value;  
    if(str.length==0)  
    {  
        alert(" The first name cannot be empty");  
        return false;  
    }  
    return true  
}  
function validateLastName()  
{  
    var str= form1.lname.value;  
    if(str.length==0)  
    {  
        alert(" The last name cannot be empty");  
        return false;  
    }  
    return true;  
}
```



# Kiểm tra tính hợp lệ của Form(3)



```
function validateEmail()
{
    var str= form1.email.value;
    if(str.length==0)
    {
        alert(" The Email field cannot be empty");
        return false;
    }
}

function processForm()
{
    disp=open("", "result")
    disp.document.write("<TITLE> Result Page </TITLE>"+ "<PRE>")
        disp.document.write("<H2 ALIGN='CENTER'>"+
            "Thanks for signing in"+ "</H2>"+ "<HR>"+ "<BR><BR>")
        disp.document.write("First name \t\t: "+form1.fname.value+"<BR>")
    disp.document.write("Last name \t\t: "+form1.lname.value+"<BR>")
    disp.document.write("Email \t\t\t: "+form1.email.value+"<BR>")
    disp.document.write("Your Comments \t\t: "+form1.comment.value+"<BR>")
    disp.document.write("<PRE>")
}
```

# Kiểm tra tính hợp lệ của Form(4)



```
if(dispatch.confirm("Is this information correct"))
    dispatch.close()
}
//-->
</SCRIPT>
</HEAD>
<BODY BGCOLOR="#FFFFFF">
<H2 ALIGN="CENTER"> Handling Form Events</H2><HR>
<FORM name="form1"><P> First Name : <INPUT TYPE="text"
    NAME="fname" size=10 onBlur="validateFirstName()">
    Last Name : <INPUT TYPE="text" NAME="lname" size=15
    onBlur="validateLastName()"></p>
<P> Email : <INPUT TYPE="text" NAME="email" size=10 onBlur="validateEmail()">
    Comments : <TEXTAREA NAME="comment" rows=4 cols=30 > Enter your comments
</TEXTAREA></p>
<P ALIGN="CENTER"><INPUT TYPE="button" value="Submit this form"
    onClick="processForm()">
<INPUT TYPE="reset"></P>
</FORM>
</BODY>
</HTML>
```

# Kiểm tra tính hợp lệ của Form(5)



Form Events - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Home Search Favorites History Print View Source

Address <D:\Examples\validatefm.htm> Go

---

## Handling Form Events

---

First Name :  Last Name :

Email :  Comments : 

This is a very good example of form validation

Done My Computer

# Sử dụng time (1)



- ❖ Có thể sử dụng kịch bản để thay đổi nội dung nhằm trả lời lại những những tương tác của người sử dụng tại những khoảng thời gian đều đặn được xác định bởi hệ thống.
- ❖ Các phương thức mà bạn có thể sử dụng là:
  - Phương thức setTimeout () của đối tượng window được sử dụng để thực thi một biểu thức cụ thể tại những khoảng thời gian đều đặn. Cú pháp:

setTimeout(expression, msec)

## Sử dụng time (2)



- ❖ Phương thức `clearTimeout()` được sử dụng để huỷ việc thực hiện biểu thức được thiết lập bởi phương thức `setTimeout()`.

```
var timerID = setTimeout(expression, msec);  
clearTimeout(timerID);
```

- ❖ Phương thức `setInterval()` cũng có thể được sử dụng để thực thi một tập hợp các câu lệnh tại các khoảng thời gian đều đặn.

```
var timerID = setInterval(expression, msec);  
clearInterval(timerID);
```

- ❖ Sự khác biệt cơ bản giữa phương thức `setTimeout ()` và `setInterval ()` là
  - `setTimeout ()` được thực hiện một lần sau một khoảng thời gian xác định trôi qua.
  - `setInterval()` được thực hiện mỗi lần khi một khoảng thời gian xác định trôi qua.

# Khác biệt setTimeout() và setInterval ()



## setTimeout ()

```
❖ moreSnow();  
function moreSnow() {  
  // content of moreSnow function  
  setTimeout("moreSnow()",  
    , speed);  
}
```

## setInterval ()

```
❖ moreSnow();  
setInterval("moreSnow()",  
  speed);  
function moreSnow() {  
  // content of moreSnow function  
}
```

# Thí dụ: 1



```
❖ <html>
  <head>
    <script type="text/javascript">
      function timedMsg()
      {
        var t=setTimeout("alert('5 seconds!')",5000);
      }
    </script>
  </head>

  <body>
    <form>
      <input type="button" value="Display timed alertbox!"
      onClick="timedMsg()" />
    </form>
  </body>
</html>
```



## Thí dụ: 2



```
❖ < <html>
  <head>
    <script type="text/javascript">
      var c=0;
      var t;
      var timer_is_on=0;

      function timedCount()
      {
        document.getElementById('txt').value=c;
        c=c+1;
        t=setTimeout("timedCount()",1000);
      }
```

# Thí dụ: 2(tt)



```
❖ function doTimer()  
{  
  if (!timer_is_on)  
  {  
    timer_is_on=1;  
    timedCount();  
  }  
}  
</script>  
</head>  
  
<body>  
<form>  
<input type="button" value="Start count!" onClick="doTimer()">  
<input type="text" id="txt" />  
</form>  
</body></html>
```