

## 9. Xử lý chuỗi

### a. Các phương thức kiểm tra xử lý chuỗi

- + **indexOf(s)**: Trả về vị trí của chuỗi s trong chuỗi mẹ tính từ đầu chuỗi
- + **lastIndexOf(s)**: Trả về vị trí của chuỗi s trong chuỗi mẹ tính từ cuối chuỗi trở về đầu chuỗi.
- + **charAt(i)**: Trả về kí tự tại vị trí thứ i của chuỗi.
- + **substring(m,n)**: Trả về một chuỗi con lấy từ chuỗi mẹ tại vị trí m, lấy n kí tự.
- + **split(s)**: Cắt chuỗi mẹ thành nhiều chuỗi con với ký tự ngăn cách các chuỗi là s, trị trả về của hàm này là một mảng các chuỗi con.

Ví dụ:

```
<html><head>
<script language="JavaScript">
<!--
function validdate(s){
if (s.split("/").length==3)
{
    var d= s.split("/")[0];
    var m= s.split("/")[1];
    var y= s.split("/")[2];
    alert("Hom nay la ngay "+d+" thang "+m+" nam "+y);
}
else alert("invalid date");
}
-->
</script></head><body>
<form name="form">
<input type="text" name="t1" value="0" size="20">
<input type="button" value="Kiem tra" size="20"
onClick="validdate(window.document.form.t1.value);">
</form></body></html>
```

***Bạn cần chú ý thuộc tính maxlength, disabled, checked của đối các tượng text, button, radio, checkbox trong form.***

Ví dụ:

```
<input type="text" name="t1" maxlength="10" size="20">
<input type="button" value="See Hidden" disabled>
<input type="radio" name="Hidden" checked>
<input type="text" name="t2" disabled size="20">
```

## b. Chương trình ví dụ:

*Sau đây là một số hàm mẫu để xử lý dữ liệu khi nhập vào từ một số đối tượng:*

+ Hàm kiểm tra dữ liệu nhập vào từ một TextField có đúng theo qui định hay không.

```

/*****
// hàm: validString
// form    -- Tên form
// object   -- Tên đối tượng
// lenMax   -- Độ dài lớn nhất
// lenMin   -- Độ dài nhỏ nhất
// space    -- Cho phép chuỗi chứa khoảng trống hay không (0- cho phép; 1- không )
// require  -- Cho phép hay không cho phép bỏ qua (0- cho phép; 1- không)
/*****
function validString(form, object, lenMax, lenMin, space, require ){
    var tmp, tmp2, msg;
    tmp2 = " ";
    msg = "";
    tmp = document.forms(form).all(object).value;
    len1 = tmp.length;
    if (require == 1){
        if (len1 < 1){    msg = 'Invalid ';
            alert(msg);
            document.forms(form).all(object).focus();
            return false;
        }
    }
    if (lenMax != lenMin){
        if (len1 > lenMax){
            msg = 'Invalid ' + lenMax;
            alert(msg);
            document.forms(form).all(object).focus();
            return false;
        }
        if (len1 < lenMin){
            msg = 'Invalid ' + lenMin;
            alert(msg);
            document.forms(form).all(object).focus();
            return false;
        }
    }
    if (space == 1){
        for(i=0; i<len1; i++){
            if (tmp.charAt(i) == tmp2){
                msg = 'Invalid ';
                alert(msg);
            }
        }
    }
}

```

```

document.forms(form).all(object).focus();
return false;
}
}
}
return true;
}

```

## 10. Cookie và HTML động

### a. Cookie

Là một mẫu thông tin mà khi bạn truy cập vào một website nó sẽ tạo ra và được lưu lại dưới đĩa cứng của người sử dụng.

#### + Thiết lập một cookie

Ta thiết lập cookie cho document bằng cách:

**document.cookie= <chuỗi cookie>**

Sau đó là sử dụng hàm escape() để mã hoá cookie cho bạn:

**Ví dụ:**

```

function setcookie()
{
    var name= prompt("Input your name ","");
    var cookiename="username="+escape(name);
    document.cookie=cookiename;
}

```

#### + Đọc một cookie

Ta sử dụng hàm unescape() giải mã cookie của bạn:

**Ví dụ:**

```

function readcookie()
{
    var thecookie= document.cookie;
    var cookiename= thecookie.split("=");
    var name= cookiename[1];
    name=unescape(name);
    alert("your name: "+name);
}

```

### b. HTML động và khái niệm cơ bản về CSS.

#### + Thẻ DIV

Thẻ này cho phép người dùng có thể định vị một thẻ HTML bất kỳ vị trí nào trên trang của bạn. Đây là cách làm cho trang web của bạn động, có thể thấy hình ảnh di chuyển được trên màn hình.

**Ví dụ:**

```

function movediv(which) {
    var thediv;
    if (window.document.all)
        { thediv=window.document.all.mydiv.style;
        }
}

```

```

else if(document.layers)
    {
        thediv=window.document.mydiv;
    }else
    {
        alert("xin chao");return;
    }
thediv.left=parseInt(thediv.left)+10*(Math.random()*10-5);
thediv.top=parseInt(thediv.top)+10*(Math.random()*10-5);
}

```

Sau đó trong phần <body> bạn viết như sau:

**<body onClick="movediv()">**

## 11. Một số phương thức thường dùng:

### a. applet

Tất cả mọi applet trên trang được lưu vào mảng thông qua cách truy cập như sau:

```
window.document.applets[];
```

Để kích hoạt nó ta sử dụng phương thức start() và để tắt nó ta sử dụng phương thức stop().

**Ví dụ:** window.document.applets[0].start();

### b. Area

Các thẻ area được lưu trong mảng sau:

```
window.document.links[];
```

Mỗi Area được lưu như một đối tượng và ta có thể truy cập nó như những đối tượng khác.

### Một số phương thức thông dụng:

+ **confirm()**: Sẽ đưa ra một câu thông báo và Trả về true hay false nếu người dùng chọn YES hay NO.

+ **toUpperCase()**: Chuyển một chuỗi thành chuỗi chữ hoa.

+ **toLowerCase()**: Chuyển một chuỗi thành chuỗi chữ thường.

+ **focus()**: Sẽ di chuyển con trỏ đến đối tượng mà ta thiết đặt focus(), thường sử dụng trong việc kiểm tra lỗi và yêu cầu người dùng nhập lại.

+ **history.back()**: Trở lại trang trước đó.

+ **eval(s)**: Định giá trị của một biểu thức.

+ **bgColor()**: Thiết đặt màu nền.

+ **concat()**: ghép nối chuỗi.

Ví dụ:

```

function setcolorbg ()
{
    window.document.bgColor="#008800";
    return false;
}

```

### c. Các hàm toán học:

+ **abs()**

+ *acos()*  
 + *asin()*  
 + *atan()*  
 + *cos()*  
 + *exp()*  
 + *log()*  
 + *max(a,b)*  
 + *min(a,b)*  
 + *pow(e,x)*  
 + *random()*  
 + *round()*  
 + *sin()*  
 + *sqrt()*  
 + *tan()*

## Bài tập củng cố

1. Để cắt một chuỗi tự thành các chuỗi con thông qua ký tự ngăn cách nào đó ta sử dụng phương thức				
<input type="radio"/>	<input type="radio"/> SubString()	<input type="radio"/> Split()	<input type="radio"/> Cả hai	
2. Để trả về vị trí của chuỗi con trong chuỗi mẹ tính từ đầu chuỗi ta sử dụng				
<input type="radio"/>	<input type="radio"/> IndexOf()	<input type="radio"/> LastIndexOf()	<input type="radio"/> SubString()	
<input type="radio"/>				
3. Để giới hạn độ dài tối đa của một đối tượng Textbox khi nhập dữ liệu từ bàn phím ta sử dụng				
<input type="radio"/>	<input type="radio"/> maxlength	<input type="radio"/> Disabled	<input type="radio"/> Hidden	
<input type="radio"/>				
4. Để thiết lập chế độ bảo mật của một trang web nào đó ta sử dụng				
<input type="radio"/>	<input type="radio"/> Session	<input type="radio"/> Cookie	<input type="radio"/> Cả hai có thể	
<input type="radio"/>				
5. Để mã hoá một đối tượng cookie ta dùng				
<input type="radio"/>	<input type="radio"/> Unescape()	<input type="radio"/> escape()	<input type="radio"/> Cả hai	

Xem kết quả