

# Esempi di Loris Tissino – Javascript

[Loris Tissino home](#)

Quella che segue è una raccolta di esempi di pagine web contenenti codice Javascript che ho utilizzato in un corso di aggiornamento per i programmatori di una software house, intenzionati a sviluppare applicazioni web da affiancare a quelle tradizionali.

Gli esempi fanno riferimento esclusivamente al DOM (Document Object Model) di Internet Explorer e tralasciano alcuni aspetti importanti: browser-sniffing, programmazione cross-browser, supporto di browser che non interpretano javascript, ecc.

Alcuni degli esempi hanno nel codice sorgente dei commenti, altri no. Durante il corso li integravo a voce. Un giorno forse troverò il tempo di scrivere una documentazione un po' più adeguata. Per adesso li pubblico nella speranza che per qualcuno possano comunque tornare utili.

Non mostrare i codici sorgente.

È disponibile anche una versione PDF.

## Esempi di base

- document.write

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - document.write</title>
</head>
<body>
<p>Testo HTML normale</p>
<p><script language="Javascript1.3">
document.write("Hello, World (scritto dallo script JS)");
</script>
</p>
<!--
notare:
- versione js
- virgolette parametro
- punto e virgola finale
- commenti script
- differenza tra document.write e document.writeln;
- caratteri di escape \" \' \n ecc.
-->

</body>
</html>
```

- window.alert

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - window.alert</title>
</head>
```

```

<body>
<p>Testo HTML normale</p>
<p><script language="Javascript1.3">
window.alert("Hello, World (scritto dallo script JS)");
</script>
</p>
<!--
notare:
- funzionerebbe anche senza window.
-->

</body>
</html>

```

- window.alert onLoad

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - window.alert onLoad</title>
</head>
<body onLoad="window.alert('Hello, World (scritto dallo script
JS)');">
<p>Testo HTML normale</p>
<!--
notare:
- cambiamento virgolette parametro
-->

</body>
</html>

```

- window.status

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - window.status</title>
</head>
<body onLoad="window.status='Hello, World (scritto dallo script
JS)';">
<p>Testo HTML normale. Il messaggio è sulla riga di
stato...</p>
<!--
notare:
- cambiamento virgolette parametro
- eventi su link nella pagina
<p>un <a href="altro.html"
      onMouseOver="window.status='Altra scritta impostata con JS...';
return true;"
      onMouseOut="window.status='...quello che vogliamo';"
      >link qui sotto</a>...</p>
-->
</body>
</html>

```

- funzioni ed eventi

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

```

```

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - funzioni ed eventi</title>
<script language="Javascript1.3">
function salutaIlMondo() {
    window.alert("Hello world (da uno script dentro la
funzione)");
}
</script>
</head>
<body>
<p>Testo HTML normale.</p>
<form>
<input type="button" value="Sii educato, saluta!"
onClick="salutaIlMondo();">
</form>
<!--
notare:
- parentesi per il richiamo della funzione
- cosa succede se uso document.writeln?
- provare con eventi diversi, quali onLoad, onMouseOver e onMouseOut
(esercizio)
-->
</body>
</html>

```

- **cambiamento valore pulsante**

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - cambiamento valore pulsante</title>
<script language="Javascript1.3">
function salutaIlMondo() {
    window.alert("Hello world (da uno script dentro la
funzione)");
    return "Ho salutato";
}
</script>
</head>
<body>
<p>Testo HTML normale.</p>
<form>
<input type="button" value="Sii educato, saluta!"
onClick="this.value=salutaIlMondo();">
</form>
<!--
notare:
-->
</body>
</html>

```

- **file javascript esterno**

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">

```

```

    <title>Javascript - file javascript esterno</title>
<script language="Javascript1.3" src="codice.js"
type="text/javascript"></script>
</head>
<body>
<script language="Javascript1.3">
saluta("Hello world");
</script>

<!--
notare:
- inserimento del codice in un file esterno
- MIME type
-->

</body>
</html>

```

## Strutture di controllo

- selezione

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - selezione</title>
<script language="Javascript1.3">

a = 1; // dichiarazione insieme all'assegnazione di un valore

var b; // dichiarazione
b = 2; // assegnazione di un valore

if (a == b) {
    testol = "Sono uguali";
}
else {
    testol = "Sono diversi";
};
// classico if-then-else

testo2 = (a == b) ? "Sono uguali" : "Sono diversi";
// assegnazione mediante operatore ternario
</script>
</head>
<body>
<script language="Javascript1.3">
document.write("<p>Il primo testo è: <strong>" +
testol + "</strong></p>");
document.write("<p>Il secondo testo è: <strong>" +
testo2 + "</strong></p>");
</script>

<!--
notare:
- operatore ==, !=
- sintassi oper. ternario
- le parentesi graffe non servirebbero per una sola istruzione
- separazione fra head e body del codice
- codice HTML dentro codice JS
- testo.length
-->
</body>
</html>

```

## • ciclo while

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - ciclo while</title>
<script language="Javascript1.3">

function scriviLettera() {
    i = 1;
    while (i <= 10) {
        document.write("Lettera A scritta la " + i + "^
volta<br>");
        i++;
    };
}
</script>
</head>
<body>
<script language="Javascript1.3">
scriviLettera();
</script>

<!--
notare:
- incremento/decremento unitario con forma abbreviata
- forma sintetica: dentro l'istruzione
- funzione con passaggio di parametro
-->
</body>
</html>
```

## • ciclo for

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - ciclo for</title>
<script language="Javascript1.3">

function scriviLettera() {
    for (i=1; i<=10; i++) {
        document.write("Lettera A scritta la " + i + "^
volta<br>");
    };
}
</script>
</head>
<body>
<script language="Javascript1.3">
scriviLettera();
</script>

<!--
notare:
- anche qui si potrebbe passare un parametro...
-->
</body>
</html>
```

## • ciclo do... while

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - ciclo do... while</title>
<script language="Javascript1.3">

function scriviLettera() {
    i = 1;
    do {
        document.write("Lettera A scritta la " + i + "^
volta<br>");
        i++;
    }
    while (i<=10);
}
</script>
</head>
<body>
<script language="Javascript1.3">
scriviLettera();
</script>

<!--
notare:
-->
</body>
</html>
```

## • switch

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - switch</title>
<script language="Javascript1.3">

function scriviLettera(lettera) {
    switch (lettera) {
        case 'a': document.write("prima opzione"); break;
        case 'b' : document.write("seconda opzione"); break;
        default: document.write("altra opzione");
    }
}
</script>
</head>
<body>
<p><strong>a: </strong><script
language="Javascript1.3">
scriviLettera('a');
</script>

<!--
notare:
- assenza di istruzione break
- e' possibile mettere due casi: es. case 'a': case 'b':
document.write...
-->
</body>
```

```
</html>
```

## Input con finestre popup

- prompt

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - prompt</title>
<script language="Javascript1.3">

function chiediescrivi() {
    a = prompt("Come ti chiami?", "Eventuale nome
proposto");
    document.writeln("Hai detto di chiamarti " + a);
}
</script>
</head>
<body>
<script language="Javascript1.3">
chiediescrivi();
</script>

<!--
notare:
- cosa succede se si preme "annulla" (== null)
-->
</body>
</html>
```

- confirm

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - confirm</title>
<script language="Javascript1.3">

function chiediescrivi() {
    a = confirm("Questa operazione cancellerà gli articoli selezionati
dal tuo carrello della spesa. Sei sicuro di voler procedere?");
    if (a) {
        document.writeln("Gli articoli sono stati cancellati!!");
    }
    else {
        document.writeln("L'operazione è stata annullata");
    }
}
</script>
</head>
<body>
<script language="Javascript1.3">
chiediescrivi();
</script>

<!--
notare:
- ovviamente cose di questo tipo non si fanno sull'evento onLoad
```

```
normalmente...
-->
</body>
</html>
```

## Array e oggetti

- array

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - array</title>
  <title>Javascript</title>
<script language="Javascript1.3">

function mostraArray() {
  var Nomi = Array();
  Nomi[0] = "Tizio";
  Nomi[1] = "Caio";
  Nomi[2] = "Sempronio";
  document.writeln("<p>Il mio array di nomi contiene " +
Nomi.length + " nomi.</p>");

  /*
   var s = new String("");
   for (i=0; i<Nomi.length; i++) {
     s += "<li>" + Nomi[i];
   };
   s = "<ul>" + s + "</ul>";
   document.writeln(s);
  */

  /*
   var s = new String("");
   for (n in Nomi) {
     s += "<li>" + Nomi[n];
   };
   s = "<ul>" + s + "</ul>";
   document.writeln(s);
  */

  };

  /*
function altraFunzione() {
  document.write(Nomi[0]);
};
  */

</script>
</head>
<body>
<script language="Javascript1.3">
mostraArray();
//altraFunzione();
</script>

<!--
notare:
- le variabili hanno una visibilit  locale a meno che non siano definite
globalmente
- gli array vanno dichiarati
- la propriet  length dice quanti elementi sono presenti
- (per aggiungere un elemento, si potrebbe utilizzare [nomearray.length])
- notazione sintetica +=, ecc.
```



```
- indice degli array potrebbe essere una stringa (array associativi),  
scansione  
  con for (elemento in vettore)  
-->  
</body>  
</html>
```

- oggetti

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0  
Transitional//EN">  
  
<html>  
<head>  
  <meta http-equiv="content-type" CONTENT="text/html;  
charset=ISO-8859-1">  
  <title>Javascript - oggetti</title>  
</head>  
<body>  
<script language="Javascript1.3">  
var c1=new Object;  
var c2=new Object;  
  
c1.nome = "Tizio";  
c2.nome = "Caio";  
  
document.write("<p>Il primo cliente si chiama " + c1.nome +  
".</p>");  
document.write("<p>Il secondo cliente si chiama " + c2.nome  
+ "</p>");  
  
</script>  
  
<!--  
notare:  
-->  
  
</body>  
</html>
```

- oggetti creati con una propria funzione

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0  
Transitional//EN">  
  
<html>  
<head>  
  <meta http-equiv="content-type" CONTENT="text/html;  
charset=ISO-8859-1">  
  <title>Javascript - oggetti creati con una propria  
funzione</title>  
<script language="Javascript1.3">  
  
function Cliente(nomerichevuto) {  
  this.nome = nomerichevuto;  
  //      this.mostra = mostra;  
  };  
  
/*  
function mostra() {  
  document.writeln("<p>Sono il cliente " + this.nome +  
"</p>");  
  }  
*/  
  
</script>  
</head>  
<body>
```

```

<script language="Javascript1.3">
var c1=new Cliente("Tizio");
var c2=new Cliente("Caio");
var c3=new Cliente("Sempronio");

document.writeln("<p>" + c1.nome +
"</p>");
document.writeln("<p>" + c2.nome +
"</p>");
document.writeln("<p>" + c3.nome +
"</p>");

/*
c1.mostra();
c2.mostra();
c3.mostra();
*/

</script>

<!--
notare:
- la funzione costruttrice potrebbe impostare più di una proprietà
- è possibile impostare dei metodi della classe
- è possibile avere un array di oggetti
-->

</body>
</html>

```

- aggiunta di un metodo tramite prototipo per gli oggetti stringa

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - aggiunta di un metodo tramite prototipo per gli
oggetti stringa</title>
<script language="Javascript1.3">

function intesta(livello) {
  html = "h" + livello;
  return "<" + html + ">" + this.toString() +
"</" + html + ">";
};

String.prototype.intesta = intesta;

</script>
</head>
<body>
<script language="Javascript1.3">
document.writeln("Testo di prova".intesta(1));

</script>

<!--
notare:
-->

</body>
</html>

```

- aggiunta di un metodo tramite prototipo per gli oggetti date

```

<script language="Javascript1.3">

function formattaAModoMio() {
    html = "strong";
    return "<" + html + ">"
    + "Addì " + this.getDate()
    + " del mese " + (this.getMonth()+1)
    + "° dell'anno " + this.getFullYear()
    + "</" + html + ">"
};

Date.prototype.formattaAModoMio = formattaAModoMio;

</script>
</head>
<body>
<script language="Javascript1.3">
adesso = new Date();
document.writeln(adesso.formattaAModoMio());

</script>

<!--
notare:
-->

</body>
</html>

```

- aggiunta di un metodo tramite prototipo per l'oggetto window

```

<script language="Javascript1.3">

function saluta(msg) {
    msg += "\n\nQuesto messaggio è stato generato dalla funzione
window.saluta()";
    alert(msg);
};

window.prototype.saluta = saluta;

</script>
</head>
<body>
<script language="Javascript1.3">
window.saluta("Ciao");
</script>

<!--
notare:
-->

</body>
</html>

```

## Passaggio di parametri alle funzioni

- variabili semplici

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - variabili semplici</title>

```

```

<script language="Javascript1.3">

function elabora(unNumero) {
    document.writeln("<p><strong>Sono nella
funzione</strong></p>");
    document.writeln("<p>Sto elaborando il numero " + unNumero
+ "</p>");
    unNumero=6;
    document.writeln("<p>Ora il numero è " + unNumero +
"</p>");
    document.writeln("<p><strong>Esco dalla
funzione</strong></p>");
}
</script>
</head>
<body>
<script language="Javascript1.3">
n=5;
document.writeln("<p>Il numero prima del richiamo della funzione
è " + n + "</p>");
elabora(n);
document.writeln("<p>Il numero dopo il richiamo della funzione è
" + n + "</p>");

</script>

<!--
notare:
-->
</body>
</html>

```

- array

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - array</title>
<script language="Javascript1.3">

function elabora(unArray) {
    document.writeln("<p><strong>Sono nella
funzione</strong></p>");
    document.writeln("<p>Sto elaborando il numero " +
unArray[0] + "</p>");
    unArray[0]=6;
    document.writeln("<p>Ora il numero è " + unArray[0] +
"</p>");
    document.writeln("<p><strong>Esco dalla
funzione</strong></p>");
}
</script>
</head>
<body>
<script language="Javascript1.3">
var n=Array;
n[0] = 5
document.writeln("<p>Il numero prima del richiamo della funzione
è " + n[0] + "</p>");
elabora(n);
document.writeln("<p>Il numero dopo il richiamo della funzione è
" + n[0] + "</p>");

</script>

```

```

<!--
notare:
-->
</body>
</html>

```

- oggetti

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - oggetti</title>
<script language="Javascript1.3">

function elabora(unOggetto) {
  document.writeln("<p><strong>Sono nella
funzione</strong></p>");
  document.writeln("<p>Sto elaborando l'oggetto " +
unOggetto.value + "</p>");
  unOggetto.value="Sono cambiato";
  document.writeln("<p>Ora l'oggetto è " + unOggetto.value +
"</p>");
  document.writeln("<p><strong>Esco dalla
funzione</strong></p>");
}
</script>
</head>
<body>
<script language="Javascript1.3">
var s=new Object();
s.value="Valore iniziale";
document.writeln("<p>Il valore del mio oggetto prima del
richiamo della funzione è " + s.value + "</p>");
elabora(s);
document.writeln("<p>Il valore del mio oggetto dopo il richiamo
della funzione è " + s.value + "</p>");

</script>

<!--
notare:
-->
</body>
</html>

```

- stringhe

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - stringhe</title>
<script language="Javascript1.3">

function elabora(unaStringa) {
  document.writeln("<p><strong>Sono nella
funzione</strong></p>");
  document.writeln("<p>Sto elaborando la stringa " +
unaStringa + "</p>");
  unaStringa="Sono cambiata";
  document.writeln("<p>Ora la stringa è " + unaStringa +

```

```

"</p>");
        document.writeln("<p><strong>Esco dalla
funzione</strong></p>");
    }
</script>
</head>
<body>
<script language="Javascript1.3">
var s=new String("Stringa iniziale");
document.writeln("<p>La stringa prima del richiamo della
funzione è " + s + "</p>");
elabora(s);
document.writeln("<p>La stringa dopo il richiamo della funzione
è " + s + "</p>");

</script>

<!--
notare:
- le stringhe sono oggetti un po' anomali: vengono passati comunque per
valore
-->
</body>
</html>

```

## Oggetti di tipo Date

- metodi "get" a disposizione

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - metodi \"get\" a
disposizione</title>
    <script language="Javascript1.3">
oggi = new Date(); //imposta un oggetto data all'ora corrente

function paragraph(text) {
        document.writeln("<p>" + text +
"</p>");
    };

</script>
</head>
<body>
<script language="Javascript1.3">
paragraph("Oggi è il " + oggi);
paragraph("Giorno: " + oggi.getDate());
paragraph("Mese: " + oggi.getMonth()); // i mesi vanno da 0
(gennaio) a 11 (dicembre)
paragraph("Anno: " + oggi.getYear()); // in Netscape,
restituisce l'anno a due cifre
paragraph("Anno completo: " + oggi.getFullYear());
paragraph("Timestamp: " + oggi.getTime());
paragraph("Ore: " + oggi.getHours());
paragraph("Minuti: " + oggi.getMinutes());
paragraph("Secondi: " + oggi.getSeconds());

</script>

<!--
notare:
-->

```

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

- metodi "set" a disposizione

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - metodi \"set\" a
disposizione</title>
<script language="Javascript1.3">
fineanno = new Date("December 31, 2002 23:59:59"); // un secondo
a mezzanotte
// fineanno = new Date(2002, 11, 31, 23, 59, 59); // forma alternativa

adesso = new Date();

fraunasettimana = new Date();
fraunasettimana.setDate(fraunasettimana.getDate()+7);

function paragraph(text) {
    document.writeln("<p>" + text +
"</p>");
    };

</script>
</head>
<body>
<script language="Javascript1.3">
paragraph("Fine anno: " + fineanno);
paragraph("Giorno: " + fineanno.getDate());
paragraph("Mese: " + fineanno.getMonth()); // attenzione al
mese
paragraph("Anno: " + fineanno.getYear()); // in Netscape,
restituisce l'anno a due cifre
paragraph("Anno completo: " + fineanno.getFullYear());
paragraph("Timestamp: " + fineanno.getTime());
paragraph("Ore: " + fineanno.getHours());
paragraph("Minuti: " + fineanno.getMinutes());
paragraph("Secondi: " + fineanno.getSeconds());

paragraph("Mancano " + (fineanno.getTime() -
adesso.getTime())/1000 + " secondi alla fine dell'anno");

paragraph("Fra una settimana sarà il "
    + fraunasettimana.getDate() + "/"
    + (fraunasettimana.getMonth()+1) + "/"
    + fraunasettimana.getFullYear());

</script>

<!--
notare:
- sarebbe ovviamente preferibile definire una funzione per la
rappresentazione
  della data nel formato desiderato
- è anche possibile aggiungere un metodo all'oggetto predefinito Date
-->

</body>
</html>
```

- uso del timer

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - uso del timer</title>
<script language="Javascript1.3">
fineanno = new Date("January 1, 2003 00:00:00");

orologio = null;

function AggiornaContoAR() {
    adesso = new Date();
    document.all.conto.innerText = Math.round((fineanno.getTime() -
adesso.getTime())/1000);
    orologio = setTimeout("AggiornaContoAR()", 1000);
};

function Ferma() {
    clearTimeout(orologio);
};

</script>
</head>
<body onLoad="AggiornaContoAR();">
<p>E' con piacere che vi informiamo che mancano <span
id="conto"
style="background-color:yellow"></span>
  &nbsp;secondi alla fine dell'anno.</p>

<form><input type="button" onClick="Ferma()"
value="Ferma il conteggio"></form>

<!--
notare:
-->

</body>
</html>
```

## Math

- metodi e proprietà di Math

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - metodi e proprietà di Math</title>
<script language="Javascript1.3">
function calcola() {
    raggio = parseFloat(document.cerchio.raggio.value);
    document.cerchio.raggio.value = raggio;
    if (document.cerchio.raggio.value != "NaN") {
        document.cerchio.area.value = areaCerchio(raggio);
    }
    else {
        document.cerchio.area.value = "Input errato";
    };
};
```



```

function areaCerchio(raggio) {
    area = Math.PI*Math.pow(raggio,2);
    return area;
};

</script>
</head>
<body>
<form name="cerchio">
<h1>Calcola l'area del cerchio!</h1>
Raggio: <input type="text" name="raggio"
value="0" style="text-align: right"><br>
Area: <input type="text" name="area"
value="0" style="text-align: right"><br>
<input type="button" onClick="calcola();"
value="Calcola">

<!--
notare:
- il modulo può sicuramente essere reso più carino
- l'oggetto Math non va istanziato, si usa così com'è
- tra gli altri metodi dell'oggetto Math:
  .ceil(n), che restituisce n arrotondato per eccesso
  .floor(n), che restituisce n arrotondato per difetto
  .round(n), che restituisce n arrotondato in maniera naturale
  .random(), che genera un numero casuale
  .abs(n), valore assoluto di n
  .cos(n), .acos(n), .sin(n), .asin(n), .tan(n), .atan(n), funzioni
trigonometriche
  .exp(n), e elevato alla n
  .log(n), logaritmo naturale di n
  .max(n1, n2), .min(n1, n2)
  .sqrt(n), radice quadrata di n
  .pow(n,m), n elevato alla m
- tra le proprietà dell'oggetto Math:
  .E, e
  .PI, pi greco
-->

</body>
</html>

```

## Stringhe

- metodi e proprietà delle stringhe

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - metodi e proprietà delle
stringhe</title>
<script language="Javascript1.3">
function elabora() {
    var s=new String;
    s=document.modulo.stringain.value;
    document.modulo.stringaout.value = s.length;
//    document.modulo.stringaout.value = s.toUpperCase();
//    document.modulo.stringaout.value = s.toLowerCase();
//    document.modulo.stringaout.value = s.charAt(3);
//    document.modulo.stringaout.value = s.indexOf("p", 2);
//    document.modulo.stringaout.value = s.lastIndexOf("p", 2);
//    document.modulo.stringaout.value = s.substring(2,5);
};

```

```

</script>
</head>
<body>
<form name="modulo">
<h1>Opera sulle stringhe!</h1>
Stringa di input: <input type="text"
name="stringain" value="prova"><br>
Stringa di output: <input type="text"
name="stringaout" value=""><br>
<input type="button" onClick="elabora();"
value="Elabora">
</form>

<!--
notare:
- il modulo può sicuramente essere reso più carino
-->

</body>
</html>

```

- metodi split e join

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - metodi split e join</title>
<script language="Javascript1.3">
function elabora() {
    var s=new String;
    var out = new String;
    var nomi = new Array();
    s=document.modulo.stringain.value;
    nomi = s.split(" ");
    out = "Trovati " + nomi.length + " nomi.\n";
    out += "Eccoli:\n";
    for (i=0;i<nomi.length;i++) {
        out += i + " ) " + nomi[i] + "\n";
    };

    out += "Riuniamo i nomi separandoli con punti e virgola:\n";
    out += nomi.join(";");
    document.modulo.testoout.value = out;
};

</script>
</head>
<body>
<form name="modulo">
<h1>Opera sulle stringhe! (continua...)</h1>
Stringa di input: <input type="text"
name="stringain" value="Tizio Caio
Sempronio"><br>
Area di output:<br><textarea name="testoout"
rows="10" cols="60"></textarea><br>
<input type="button" onClick="elabora();"
value="Elabora">
</form>

<!--
notare:
-->

</body>

```

```
</html>
```

## Il Document Object Model (DOM) di Internet Explorer

- il DOM di Internet Explorer
- riferimenti tramite identificatori
- riferimenti ai campi di un modulo
- conoscere le proprietà di un oggetto

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0  
Transitional//EN">  
  
<html>  
<head>  
  <meta http-equiv="content-type" CONTENT="text/html;  
charset=ISO-8859-1">  
  <title>Javascript - conoscere le proprietà di un  
oggetto</title>  
<script language="Javascript1.3">  
  
function showobject(anObject) {  
    document.write("<h2>Object: " + anObject.nodeName +  
"</h2>");  
    for (p in anObject) {  
        document.writeln("<strong>" + p + ":"  
</strong><em>" + anObject[p] +  
"</em><br>");  
    }  
}  
  
</script>  
  
</head>  
<body>  
  <h1>  
    Il Document Object Model di Internet Explorer  
  </h1>  
<hr>  
<script language="Javascript1.3">  
  showobject(document);  
  showobject(navigator);  
</script>  
</body>  
</html>
```

## Finestre e frames

- aprire una finestra 'pop-up'

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0  
Transitional//EN">  
  
<html>  
<head>  
  <meta http-equiv="content-type" CONTENT="text/html;  
charset=ISO-8859-1">  
  <title>Javascript - aprire una finestra \'pop-up\'</title>  
<script language="Javascript1.3">  
  
function showinfo() {  
    finestra = window.open("", "finestrainfo",  
"width=600, height=400, toolbar=0, status=1");  
    finestra.document.write("Hai scritto <strong>" +  
document.moduloinfo.testo.value + "</strong> nella  
casellina.");  
    finestra.document.write("<br><br>");  
}
```

```

        finestra.document.write("<a href=\"#\""
onClick=\"javascript:window.close();\">Chiudi questa
finestra</a>");

    }

</script>

</head>
<body>
    <h1>
        Il Document Object Model di Internet Explorer
    </h1>
    <form name="moduloinfo">
Nome dell'oggetto: <input type="text" name="testo"
value="Testo di prova"><br>
<input type="button" value="Mostra info"
onClick="javascript:showinfo()" name="azione">
</form>

<hr>
<!--
sono impostabili all'apertura della nuova finestra:
- toolbar
- location
- directories
- status
- menubar
- scrollbars
- resizable
-->

</body>
</html>

```

- l'oggetto window.opener

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - l'oggetto window.opener</title>
<script language="Javascript1.3">

function infowindow() {
    if (document.moduloinfo.azione.value == "Mostra info") {
        finestra = window.open("", "finestrainfo",
"width=600, height=400, toolbar=0, status=1");
        finestra.document.write("Hai scritto <strong>" +
document.moduloinfo.testo.value + "</strong> nella
casellina.");
        finestra.document.write("<br><br>");
        return "Nascondi le info";
    }
    else {
        finestra.close();
        return "Mostra info";
    }
}

</script>

</head>
<body>
    <h1>
        Il Document Object Model di Internet Explorer

```

```

        </h1>
<form name="moduloinfo">
Nome dell'oggetto: <input type="text" name="testo"
value="Testo di prova"><br>
<input type="button" value="Mostra info"
onClick="this.value=infowindow()" name="azione">
</form>

<hr>
</body>
</html>

```

- i frames

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - i frames</title>
</head>

<frameset cols="30%,*">
  <frame src="principale.php?title=indice"
name="principale">
  <frame src="documento.php?title=doc1"
name="secondario">
</frameset>

```

I seguenti collegamenti portano ai documenti che dovrebbero apparire dentro i riquadri. Seguirli può avere senso solo per analizzarne il codice sorgente:

- ◆ indice

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - indice</title>
  <script language="Javascript1.3">

function showobject(anObject) {
    document.write("<h2>Object: " + anObject.nodeName +
"</h2>");
    for (p in anObject) {
        document.writeln("<strong>" + p + ":
</strong><em>" + anObject[p] +
"</em><br>");
    }
}

</script>

</head>

<body>
<h1>Contenuto del riquadro principale</h1>
<p>Il riquadro in cui sono si chiama <strong>
<script language="Javascript1.3">
document.write(self.name);
</script>
</strong></p>

```

```

<p>Il riquadro a fianco si chiama <strong>
<script language="Javascript1.3">
document.write(parent.frames[1].name);
</script>
</strong></p>

```

```

<p>I link che intendono agire su riquadri diversi devono specificare
l'attributo
<em>target</em>, indicando il nome del riquadro dove far aprire
il
<a href="documentodiverso.php?title=doc2"
target="secondario">nuovo documento</a>.</p>

</body>
</html>

```

#### ◆ doc1

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - doc1</title>
  <script language="Javascript1.3">

function showobject(anObject) {
    document.write("<h2>Object: " + anObject.nodeName +
"</h2>");
    for (p in anObject) {
        document.writeln("<strong>" + p + ":
</strong><em>" + anObject[p] +
"</em><br>");
    }
}

</script>

</head>

<body>
<h1>Contenuto del riquadro secondario</h1>
<p>Il riquadro in cui sono si chiama <strong>
<script language="Javascript1.3">
document.write(self.name);
</script>
</strong></p>

<p>Il riquadro a fianco si chiama <strong>
<script language="Javascript1.3">
document.write(parent.frames[0].name);
//document.write(parent.principale.name); //valido anche così
</script>
</strong></p>

</body>
</html>

```

#### ◆ doc2

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>

```

```

<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - doc2</title>
<script language="Javascript1.3">

function showobject(anObject) {
    document.write("<h2>Object: " + anObject.nodeName +
"/>");
    for (p in anObject) {
        document.writeln("<strong>" + p + ":
</strong><em>" + anObject[p] +
"/><em><br>");
    }
}

</script>

</head>

<body>
<h1>Contenuto diverso del riquadro secondario</h1>
<p>Sono un file diverso adesso: cambia il
<em>contenuto</em>, ma rimane sempre
lo stesso <em>riquadro</em>!</p>

<p>Il riquadro in cui sono si chiama <strong>
<script language="Javascript1.3">
document.write(self.name);
</script>
</strong></p>

<p>Il riquadro a fianco si chiama <strong>
<script language="Javascript1.3">
document.write(parent.frames[0].name);
</script>
</strong></p>

</body>
</html>

```

## Funzioni predefinite

- eval

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - eval</title>
<script language="Javascript1.3">

    x = "document.writeln(\"test\")";

</script>

</head>

<body>

<script language="Javascript1.3">

    eval(x);

```

```

</script>

<!--
esercizio:
- realizzare una pagina web con una casella di testo in cui l'utente
  possa scrivere un'istruzione Javascript da eseguire tramite eval
-->

<!--
da notare:
- in molti script la funzione eval viene usata per gestire le differenze
  tra browser nelle definizioni dei modelli di documento; si compongono
  delle stringhe diverse (ad esempio document.all.test.style.visibility
  piuttosto che document.layers['test'].visibility) e poi le si esegue
-->

</body>
</html>

```

## Operazioni sui moduli

- spostamento di opzioni

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - spostamento di opzioni</title>
<script language= "JavaScript">

function spostaOpzioni(daM, aM) {
  for (i=0; i<daM.length ; i++){
    if (daM.options[i].selected == true ) {
      aM.options[aM.length]= new Option(daM.options[i].text);
    }
  }

  for (i = (daM.length -1); i>=0; i--){
    if (daM.options[i].selected == true ) {
      daM.options[i] = null;
    }
  }
}

</script>
</head>
<body>

<h1>Passaggio di elementi da una lista ad un'altra</h1>

<form method="get" name="miomodulo">
<table bgcolor=white border=1 cellpadding=5 cellspacing=2 align=center
>
<tr>
<td align=center>
  <select name=menu1 size=10 multiple>
    <option>Elemento 1</option>
    <option>Elemento 2</option>
    <option>Elemento 3</option>
    <option>Elemento 4</option>
    <option>Elemento 5</option>
  </select><br>
  <p align=center><input type="button"
onClick="spostaOpzioni(m1,m2)" value=" >>

```



```

"></p>

</td>
<td align=center>

    <select name=menu2 size=10 multiple >
    <option>Elemento 6</option>
    </select><br>
    <p align=center><input type="button"
onClick="spostaOpzioni(m2,m1)" value=" << "
></p>

</td>
</tr>
</table>
</form>

<script language="JavaScript">
var m1 = document.miomodulo.menu1;
var m2 = document.miomodulo.menu2;
</script>

</body>
</html>

```

## Convalida di dati nei moduli

- uso di button e metodo submit

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - uso di button e metodo submit</title>
<script language="Javascript1.3">
function riferisci() {
    if (document.miomodulo.miacasella.value == "vai") {
        document.miomodulo.submit();
    }
    else {
        window.alert("Nella casella di testo hai inserito il valore: "
+ document.miomodulo.miacasella.value
+ "\nDovresti invece inserire il valore \"vai\"");
    };
}
</script>
</head>
<body>
<form name="miomodulo"
action="http://192.168.1.127/www.tissino.it/esempi/pagineweb/foo.php"
method="get">
Inserisci un valore: <input type="text"
name="miacasella"><br>
<input type="button" value="Riferisci"
onClick="riferisci();">
</form>
<!--
notare:
- input type del pulsante: button != submit
- il controllo va fatto comunque sul lato server
- \n nelle stringhe
-->
</body>
</html>

```

## • uso del metodo focus

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - uso del metodo focus</title>
<script language="Javascript1.3">
function riferisci() {
    for(var i=1;i<=5;i++) {
        campo = document.miomodulo.elements["num"+i];
        if (isNaN(parseFloat(campo.value))) {
            alert("Non è numerico il valore del campo " +i);
            campo.focus();
            campo.select();
            break;
        }
    }
}
</script>
</head>
<body>
<form name="miomodulo"
action="http://192.168.1.127/www.tissino.it/esempi/pagineweb/foo.php"
method="get">
<input type="text" value="100"
style="text-align:right" name="num1"><br>
<input type="text" value="200"
style="text-align:right" name="num2"><br>
<input type="text" value="300"
style="text-align:right" name="num3"><br>
<input type="text" value="400"
style="text-align:right" name="num4"><br>
<input type="text" value="500"
style="text-align:right" name="num5"><br>
<p>Prova a inserire un valore non numerico in una casella e a premere
il pulsante.</p>

<input type="button" value="Riferisci"
onClick="riferisci();">
</form>
<!--
notare:
- funzione isNaN
- uso dell'oggetto campo per abbreviare il codice
- metodo focus
- metodo select
-->
</body>
</html>
```

## • gestore di eventi onBlur

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - gestore di eventi onBlur</title>
<script language="Javascript1.3">
function controlla(campo) {
    n = parseFloat(campo.value);
    if (isNaN(n)) {
```

```

        campo.select();
        alert("Devi inserire un valore numerico in questo campo!\n"
+ "(" + campo.value + ")" non lo è
sicuramente...");
    }
    else {
        campo.value=n;
    };
};

function riferisci() {
    alert("non faccio nulla di particolare");
};
</script>
</head>
<body>
<form name="miomodulo"
action="http://192.168.1.127/www.tissino.it/esempi/pagineweb/foo.php"
method="get">
<input type="text" value="100"
style="text-align:right" name="num1"
onBlur="javascript:controlla(this)";><br>
<input type="text" value="200"
style="text-align:right" name="num2"
onBlur="javascript:controlla(this)";><br>
<input type="text" value="300"
style="text-align:right" name="num3"
onBlur="javascript:controlla(this)";><br>
<input type="text" value="400"
style="text-align:right" name="num4"
onBlur="javascript:controlla(this)";><br>
<input type="text" value="500"
style="text-align:right" name="num5"
onBlur="javascript:controlla(this)";><br>
<p>Prova a inserire un valore non numerico in una casella e a premere
il pulsante.</p>

<input type="button" value="Riferisci"
onClick="riferisci();">
</form>
<!--
notare:
- uso dell'evento Blur (perdita del focus)
- esercizio: anziché impedire di mettere quel valore, chiedere conferma
  ("Il valore inserito non è corretto. Sei sicuro di quello che
  fai?"
  con la funzione confirm
-->
</body>
</html>

```

- gestore di eventi onSubmit

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - gestore di eventi onSubmit</title>
<script language="Javascript1.3">
function indirizzoEmailValido(indirizzo) {
    return (indirizzo.indexOf("@") > 0)
};

function controllaDati() {
    if (!indirizzoEmailValido(document.miomodulo.email.value)) {
        document.miomodulo.email.select();
    }
}

```

```

        alert("Sorry, indirizzo email non valido");
        return false;
    }
    else
        return true;
};
</script>
</head>
<body>
<form name="miomodulo"
action="http://192.168.1.127/www.tissino.it/esempi/pagineweb/foo.php"

method="get" onSubmit="javascript:return
controllaDati(this);">
<input type="text" value="indirizzo@email"
name="email"><br>
<p>Iscriviti alla nostra mailing list!</p>
<input type="submit" value="Sì, mi iscrivo!">
</form>
<!--
notare:
- uso dell'evento Submit per il modulo anziché per il pulsante
- return controllaDati(), non solo controllaDati()
- il controllo della sola presenza del segno @ non è granché, con le
espressioni
  regolari si può fare di meglio
-->
</body>
</html>

```

#### • controllo dell'input dell'utente

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - controllo dell'input
dell'utente</title>
<script>
function controlla() {
    with (document.modulo) {
        if (nome.value=="") {
            alert("Non hai inserito il nome");
            nome.focus();
            return false;
        };

        var sintassi_in = false;
        for (i=0;i<sintassi.length;i++) {
            sintassi_in |= sintassi[i].checked;
        };

        if (!sintassi_in) {
            alert("Non hai risposto alla domanda sulla sintassi");
            sintassi[0].focus();
            return false;
        };

        if (browser.value=="") {
            alert("Non hai risposto alla domanda sul browser");
            browser.focus();
            return false;
        };

        var istruzioni_in = false;
        for (i=0;i<istruzioni.length;i++) {

```

```

        istruzioni_in = istruzioni_in || istruzioni[i].selected;
    };

    if (!istruzioni_in) {
        alert("Non hai risposto alla domanda sulle istruzioni");
        istruzioni.focus();
        return false;
    };

    // inizia il controllo sulle risposte fornite

    var risposte_corrette = 0;

    if (sintassi[1].checked) risposte_corrette++;

    if (browser.value=="ly") risposte_corrette++;

    var istruzioni_rc = true;

    istruzioni_rc &= istruzioni[0].selected;
    istruzioni_rc &= istruzioni[1].selected;
    istruzioni_rc &= !istruzioni[2].selected;
    istruzioni_rc &= istruzioni[3].selected;
    istruzioni_rc &= istruzioni[4].selected;

    if (istruzioni_rc) risposte_corrette++;

    msg = "Caro " + nome.value + ", hai risposto correttamente
a "
    + risposte_corrette + " domande.";

    if (risposte_corrette==3) msg+= " Complimenti!";

    alert(msg);

    submit();
    }
}
</script>

</head>

<body>
<h1>Esercizio conclusivo corso Javascript</h1>
<form action="../pagineweb/foo.php" method="get"
name="modulo">
<fieldset>
<legend>Informazioni personali</legend>
<table border="0">
<tr>
<td><label for="nome"
accesskey="N"><u>N</u>ome</label></td>
<td><input type="text" name="nome"
size="30" id="nome"></td>
</tr>
</table>
</fieldset>
<fieldset>
<legend>Sintassi di Javascript</legend>
<table border="0">
<tr>
<td rowspan="2">Quale linguaggio ricorda la sintassi di
Javascript?<br>
<input type="radio" name="sintassi"
value="pascal"
accesskey="P"><u>P</u>ascal
<input type="radio" name="sintassi"
value="cpp" accesskey="C"><u>C</u>++

```

```

<input type="radio" name="sintassi"
value="vb" accesskey="V"><u>V</u>isual
Basic
</td>
</tr>
</table>
</fieldset>
<fieldset>
<legend>Browser</legend>
<table border="0">
<tr>
<td rowspan="2">Quale tra i seguenti
<u>b</u>rowser non supporta Javascript?<br>
<select name="browser" accesskey="B">
<option value=""></option>
<option value="ns6">Netscape 6</option>
<option value="ns4">Netscape Navigator 4</option>
<option value="ie">Internet Explorer</option>
<option value="ly">Lynx</option>
<option value="op3">Opera 3</option>
</select>
</td>
</tr>
</table>
</fieldset>
<fieldset>
<legend>Istruzioni</legend>
<table border="0">
<tr>
<td rowspan="2">Quali tra le seguenti sono
<u>i</u>struzioni javascript valide?<br>
<select name="istruzioni" accesskey="I" multiple
size="5">
<option value="1">alert("ciao")</option>
<option value="2">alert(saluto)</option>
<option value="3">alert "ciao"</option>
<option
value="4">window.alert("ciao")</option>
<option
value="5">window.saluta("ciao")</option>
</select>
</td>
</tr>
</table>
</fieldset>
<fieldset>
<legend>Invio</legend>
<table border="0">
<tr>
<td rowspan="2">
<input type="button" value="controlla"
onClick="controlla();">
</td>
</tr>
</table>
</fieldset>
</form>

</body>
</html>

```

## Espressioni regolari

- test

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

```

```

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - test</title>
<script language="Javascript1.3">

function stringaValida(str) {
  var patternvalido = "[a-z]+$";
  // la stringa sarà valida se conterrà solo lettere minuscole (in
qualsiasi numero)

  var reg = new RegExp(patternvalido);
  return str.match(reg);
  // sarebbe valido anche
  // return reg.test(str);
  };

function controllaDati() {
  msg = stringaValida(document.miomodulo.stringa.value) ? " " :
"<b>non</b> ";
  msg = "La stringa " + msg + "è valida";
  document.all.esito.innerHTML = msg;
  };

</script>
</head>
<body>
<form name="miomodulo" method="get">
Stringa da controllare: <input type="text" value="stringa
da controllare" name="stringa"><br>
<input type="button" value="Controlla!"
onClick="controllaDati();">
</form>
<p id="esito"></p>

<!--
notare:
-->
</body>
</html>

```

- replace

```

<script language="Javascript1.3">

function applica() {
  var reg = new RegExp(document.miomodulo.stringa_reg.value, 'g');
  //notare il parametro 'g' (global)

  vecchia = document.miomodulo.stringa_in.value;
  sost = document.miomodulo.stringa_sost.value;

  msg = vecchia.replace(reg, sost);

  document.all.esito.innerHTML = msg;
  };

</script>
</head>
<body>
<form name="miomodulo"
action="http://192.168.1.127/www.tissino.it/esempi/pagineweb/foo.php"

method="get">
Stringa iniziale: <input type="text" value="testo
iniziale" name="stringa_in"><br>
Espressione regolare: <input type="text" value="ini"

```

```

name="stringa_reg"><br>
Stringa da sostituire: <input type="text"
value="INI" name="stringa_sost"><br>
<input type="button" value="Applica"
onClick="applica();">
</form>
<p id="esito"></p>

<!--
notare:
-->
</body>
</html>

```

### • controllo indirizzo email con RegExp

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - controllo indirizzo email con
RegExp</title>
<script language="Javascript1.3">
function indirizzoEmailValido(indirizzo) {
  if (window.RegExp) {
    var nonvalido = "(@.*@)|(\.\.\.)|(@\.\.)|(\.\.)|(^\\..)";
    var valido =
    "^.+\\@([\\?])[a-zA-Z0-9\\-\\.]+\\.([a-zA-Z]{2,4}|[0-9]{1,3})(\\?)?$";
    var regnv = new RegExp(nonvalido);
    var regv = new RegExp(valido);
    if (!regnv.test(indirizzo) && regv.test(indirizzo))
      return true;
    return false;
  }
  else {
    if(indirizzo.indexOf("@") >= 0)
      return true;
    return false;
  }
};

function controllaDati(modulo) {
  if (!indirizzoEmailValido(modulo.email.value)) {
    modulo.email.select();
    alert("Si prega di inserire un indirizzo email valido");
    return false;
  }
  else
    return true;
};

</script>
</head>
<body>
<form name="miomodulo" action="../pagineweb/foo.php"

method="get" onSubmit="javascript:return
controllaDati(this);">
<input type="text" value="indirizzo@email"
name="email"><br>
<p>Iscriviti alla nostra mailing list!</p>
<input type="submit" value="Sì, mi iscrivo!">
</form>
<!--
notare:
- uso dell'evento Submit per il modulo anziché per il pulsante
- return controllaDati(), non solo controllaDati()

```



```
- il controllo della sola presenza del segno @ non è granché
-->
</body>
</html>
```

## Gestione degli errori

- gestore eventi onerror

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - gestore eventi onerror</title>
<script language="Javascript1.3">
<!-- //

function gestisciErrore() {
    return true;
}

//window.onerror = gestisciErrore;
//-->
</script>

</head>
<body onload="funzioneCheNonEsiste();">
<p>Questa pagina contiene un errore JS.</p>
<!--
notare:
- togliere il doppio slash di commento all'istruzione window.onError
- se la funzione gestisciErrore() restituisce true IE non visualizza errori

- onerror, non onError!
-->
</body>
</html>
```

- onerror: alert

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - onerror: alert</title>
<script language="javascript1.3">
<!-- //

function gestisciErrore() {
    alert("Il codice Javascript di questa pagina non funziona.
Peccato.");
    return true;
}

window.onerror = gestisciErrore;

function funzioneConErrore() {
    a = Math.swe(); // non esiste il metodo swe() di Math
    return true;
}

}
```

```

//-->
</script>

</head>
<body onload="funzioneConErrore();">

<!--
notare:
- togliere il doppio slash di commento all'istruzione window.onerror
-->
</body>
</html>

```

- **onerror: ridirezione**

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - onerror: ridirezione</title>
<script language="javascript1.3">
<!-- //

function gestisciErrore() {
    window.location = "paginaerrori.php";
    return true;
}

window.onerror = gestisciErrore;

function funzioneConErrore() {
    a = Math.swe(); // metodo che non esiste, provoca un errore
    return true;
}

//-->
</script>

</head>
<body onload="funzioneConErrore();">

<!--
notare:
-
-->
</body>
</html>

```

- **onerror: assenza disabilitazione controllo**

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - onerror: assenza disabilitazione
controllo</title>
<script type="text/javascript">

```

```

<!-- //

function gestisciErrore() {
    alert("Si è verificato un errore nel codice Javascript");
    return false;
}

window.onerror = gestisciErrore;

function funzioneConErrore() {
    a = Math.swe();
    return true;
}

//-->
</script>

</head>
<body onload="funzioneConErrore();">
<p>Questa pagina contiene un errore JS.</p>
<!--
notare:
-
-->
</body>
</html>

```

- **onerror: info sull'errore**

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - onerror: info sull\'errore</title>
<script language="javascript1.3">
function gestisciErrore(messaggio, indirizzo, riga) {

    alert("Si è verificato un errore nel codice Javascript alla riga
" + riga
    + "\ndel file " + indirizzo + ";\ndescrizione dell'errore:
\n" + messaggio);

    return true;
}

window.onerror = gestisciErrore;

function funzioneConErrore() {
    a = Math.swe();
    return true;
}

//-->
</script>

</head>
<body onload="funzioneConErrore();">

<!--

```

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

- gestione con try – catch

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - gestione con try - catch</title>
<script language="Javascript1.3">

function showobject(anObject) {
    document.write("<h2>Object: " + anObject.nodeName +
"</h2>");
    for (p in anObject) {
        document.writeln("<strong>" + p + ":
</strong><em>" + anObject[p] +
"</em><br>");
    }
}

function dividi() {
    try {
        c = Math.swe() // 2/0;
        document.writeln("Risultato: " + c);
    }
    catch (errore_verificato) {
        window.alert("Si è verificato l'errore \" +
errore_verificato.name + "\"\n(" +
errore_verificato.description + ")");
        // showobject(errore_verificato);
    }
    finally {
        alert("Queste operazioni vengono eseguite comunque");
    }
}

</script>
</head>
<body>
<script language="Javascript1.3">
dividi();
</script>

<!--
notare:
- try: prova a fare l'operazione
- catch: esegui nel caso si sia verificato un errore
- finally: esegui comunque
-->
</body>
</html>
```

- try – catch: esempio concreto

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
```

```

<title>Javascript - try - catch: esempio concreto</title>
<script language="Javascript1.3">

function showobject(anObject) {
    document.write("<h2>Object: " + anObject.nodeName +
"</h2>");
    for (p in anObject) {
        document.writeln("<strong>" + p + ":
</strong><em>" + anObject[p] +
"</em><br>");
    }
}

function infosullimmagine() {
    try {
        cmd
        = "document.all.spazioinfo.innerText = document.images."
        + document.miomodulo.nomeimmagine.value
        + "."
        + document.miomodulo.proprieta.value;

        alert("Sto per eseguire " + cmd);
        eval(cmd);
    }
    catch (errore_verificato) {
        window.alert("Si è verificato l'errore \" +
errore_verificato.name + "\"\n(" +
errore_verificato.description + ")");
        //      showobject(errore_verificato);
    }
    finally {
        //      alert("Queste operazioni vengono eseguite comunque");
    }
}

</script>
</head>
<body>

<br>

<br>

<form name="miomodulo">
<h1>Informazioni sulle immagini</h1>
Nome dell'immagine: <input type="text"
name="nomeimmagine"><br>
Proprietà che vuoi conoscere: <input type="text"
name="proprieta"><br>
<input type="button" value="Dammi informazioni"
onClick="infosullimmagine();">
</form>
<hr>
<p id="spazioinfo"></p>
<!--
notare:
-->
<p>Le due immagini si chiamano "qr" e "qv". Le
proprietà delle immagini sono, ad esempio, src, width, height,
alt.</p>
</body>
</html>

```

# Cookies

- impostazione di un cookie

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - impostazione di un cookie</title>
<script language="Javascript1.3">

function impostaUnCookieGenerico(nome, valore, scadenza) {
  strCookie = nome + "=" + escape(valore);
  if (scadenza)
    strCookie += "; expires=" + scadenza.toGMTString();
  document.cookie = strCookie;
  return "Ho impostato il cookie <strong>" + strCookie +
"</strong>";
}

function impostaIlMioCookie() {
  dataScadenza = new Date();
  dataScadenza.setMonth(dataScadenza.getMonth()+1); // fra un mese
  info = impostaUnCookieGenerico(
    document.miomodulo_set.nomecookie.value,
    document.miomodulo_set.valorecookie.value,
    dataScadenza
  );
  document.all.spazioinfo.innerHTML = info;
}

</script>
</head>
<body>
<form name="miomodulo_set">
<h1>Imposta cookie</h1>
Nome del cookie: <input type="text"
name="nomecookie"><br>
Valore da assegnare al cookie: <input type="text"
name="valorecookie"><br>
<input type="button" value="Imposta un cookie"
onClick="impostaIlMioCookie();">
</form>

<hr>

<p id="spazioinfo"></p>
<!--
notare:
- la data di scadenza può essere ovviamente impostata in altro modo
-->
</body>
</html>
```

- lettura dei cookies

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
  <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
  <title>Javascript - lettura dei cookies</title>
```

```

<script language="Javascript1.3">

function leggiCookies() {
    str = "I cookies di mia competenza costituiscono una stringa di
"
        + document.cookie.length
        + " caratteri.";
    str += "<br>Eccola: <strong>" + document.cookie +
"</strong>.";
    return str;
}

function leggiIMieiCookies() {
    document.all.spazioinfo.innerHTML = leggiCookies();
}

</script>
</head>
<body>

<form name="miomodulo_get">
<h1>Leggi cookies</h1>
<input type="button" value="Leggi il valore del
cookie" onClick="leggiIMieiCookies();">
</form>
<hr>

<p id="spazioinfo"></p>
<!--
notare:
-->
</body>
</html>

```

- lettura di un particolare cookie

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - lettura di un particolare
cookie</title>
<script language="Javascript1.3">

function leggiUnCookieGenerico(nome) {
    var ricerca = nome + "=";
    // quello che devo cercare nella stringa

    if (document.cookie.length > 0) {
        //se esistono i cookies

        inizio = document.cookie.indexOf(ricerca);
        if (inizio != -1) {
            inizio += ricerca.length;
            // mi metto all'inizio del valore
            fine = document.cookie.indexOf(";", inizio);
            // trovo la fine del valore
            fine = (fine==-1) ? document.cookie.length : fine;
            //.. che potrebbe essere anche la fine della stringa

            return unescape(document.cookie.substring(inizio, fine));
            // missione compiuta!
        }
    }
    return "";
}

```

```

        // se proprio non c'erano cookies
    }

function leggiIlMioCookie() {
    document.all.spazioinfo.innerHTML = leggiUnCookieGenerico(
        document.miomodulo_get.nomecookie.value);
}

</script>
</head>
<body>
<form name="miomodulo_get">
<h1>Leggi cookie specifico</h1>
Nome del cookie: <input type="text"
name="nomecookie"><br>
<input type="button" value="Leggi il valore del
cookie" onClick="leggiIlMioCookie();">
</form>
<hr>

<p id="spazioinfo"></p>
<!--
notare:
- potrei aver bisogno di informazioni diverse nel caso di assenza
cookies,
  assenza valore per un cookie, valore nullo per un cookie-.
-->
</body>
</html>

```

- ricordarsi il nome del visitatore

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">

<html>
<head>
    <meta http-equiv="content-type" CONTENT="text/html;
charset=ISO-8859-1">
    <title>Javascript - ricordarsi il nome del
visitatore</title>
<script language="Javascript1.3">

function impostaUnCookieGenerico(nome, valore, scadenza) {
    strCookie = nome + "=" + escape(valore);
    if (scadenza)
        strCookie += "; expires=" + scadenza.toGMTString();
    document.cookie = strCookie;
}

function leggiUnCookieGenerico(nome) {
    var ricerca = nome + "=";

    if (document.cookie.length > 0) {
        inizio = document.cookie.indexOf(ricerca);
        if (inizio != -1) {
            inizio += ricerca.length;
            fine = document.cookie.indexOf(";", inizio);
            fine = (fine== -1) ? document.cookie.length : fine;
            return unescape(document.cookie.substring(inizio, fine));
        }
    }
    return "";
}

function impostaCookie() {

```



```

        scadenza = new Date();
        scadenza.setFullYear(scadenza.getFullYear()+1);
        impostaUnCookieGenerico("nome", document.utente.mionome.value,
scadenza);
        alert("Ho impostato un cookie per ricordarmi di te. Aggiorna la
pagina per verificarlo.");
    };

function cancellaCookie() {
    scadenza = new Date();
    scadenza.setFullYear(scadenza.getFullYear()-1); //data già passata
    impostaUnCookieGenerico("nome", "", scadenza);
    alert("Ho cancellato il cookie che avevo impostato per ricordarmi di
te. Aggiorna la pagina per verificarlo.");
};

function salutaUtente(nome) {
    document.writeln("<p>Ciao, <strong>" + nome +
"</strong>, ben tornato.</p>");
    document.writeln("<p>Se non sei " + nome
+ ", <a
href=\"javascript:cancellaCookie()\">cancella il
cookie</a>.</p>");
};

function mostraFormUtente() {
    document.writeln("<p>Hey, non ci conosciamo ancora. Se mi dici
chi sei, mi ricorderò di te.</p>");
    document.writeln("<form name=\"utente\">");
    document.writeln("Mi chiamo: <input type=\"text\"
name=\"mionome\"><br>");
    document.writeln("<input type=\"button\"
value=\"Ricordati di me\"
onClick=\"impostaCookie()\">");
    document.writeln("</form>");
};

</script>
</head>
<body>
<script language="Javascript1.3">
nomeUtente = leggiUnCookieGenerico("nome");
if (nomeUtente == '')
    mostraFormUtente()
else
    salutaUtente(nomeUtente);
</script>

<p id="spazioinfo"></p>
<!--
esercizi:
- far ricordare anche il colore preferito per lo sfondo della pagina, e
impostarlo
automaticamente;
-->
</body>
</html>

```

---

**Loris Tissino**

email [loris@tissino.it](mailto:loris@tissino.it)

fax +39 02 700 35667