

Javascript



Index.html
Hello World!

Javascript.js

```
<html>
  <head>
    <title>La mia prima applicazione JavaScript</title>
    <script language="Javascript" src="Javascript.js">
      <!-- commento //-->
    </script>
  </head>
  <body>
    <script>
      document.write("Hello, world!");
    </script>
  </body>
</html>
```

```
document.write("Hello, world! dal file esterno! ");
```

Sezione sito senza JS

```
<NOSCRIPT>  
  <META HTTP-EQUIV REFRESH CONTENT="0;  
    URL=altrapagina.htm">  
</NOSCRIPT>
```

Commenti

```
/*commento*/  
Int x ; // commento
```

Gestione Eventi
Passaggio del mouse o click

```
<a href="#"  
  onmouseover="alert('Hai passato il mouse')">  
  Passa il mouse  
</a>  
  
<a href="javascript:alert('Hai cliccato')">  
  Clicca qui  
</a>
```

Gli eventi sono descritti così

```
Oggetto.evento=handler;
```

Un uso principale del JavaScript in ambito Web è la scrittura di piccole funzioni integrate nelle pagine HTML

Tipi di variabili

Tipi di variabili

```
var mioCarattere , // le variabili non inizializzate sono di tipo nullo
    miaVariabile ; // minuscolo per variabili
var KOSTANTE_PI_GRECO = 3,14 ; // Maiuscolo per costanti
var pioveSiNo = False ;
var SomeWord= ' hello World' ;
var someNumber= 5 ;
console.log ( someNumber ) // Scrivere a console
console.log ( someWORD )
```

Le parole chiavi Riservate a Javascript

Abstract do if package throw boolean double implements private throws break
else import protected transient byte extends in public true case false instanceof
return try catch final int short var* char finally interface static void class float
long super while const* for native switch with continue function new
synchronized default goto null this.

Trappola Javascript

Temval=365+"10" ; Javascript converte **stringa** "10" in un **numero** 10
e ne fa la somma 375.
In altri linguaggi dava errore in Javascript no.
TRUCCO TIPO VARIABILE : Typeof(variabile_da_testare)
number, string, object

Differenza tra Variabili **LOCALI** E **GLOBALI**

1) le variabili globali si usano dentro il Documento HTML.

Precisamente in <SCRIPT> e <HEAD>.

Si definiscono in questa sezione perché devono essere inizializzati prima di ogni utilizzo. (Buona Regola)

2) Variabili locali hanno valore solo all'interno della funzione, function {} .

Array

```
nomematrice=new Array(num)
animals=new Object[]
animals[0]="rana";
animals[1]="anatra";
animals[2]="asino";
animals[3]="orso";
animals[4]="gallina";
var myCatFriends = ["bill", "tabby", "ceiling"]
var zeroThroughTen = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Funzione “ myFunction() ”

```
function myFunction()
{
    alert("Hello World!");
}
```

Messaggio
allerta con il nome John!

```
var person =
{
    firstName: "John",
    lastName: "Doe"
};

alert( person.firstName );
```

Metodi stringa

```
var T1 = new String ( " TeSto " );  
var T2 = T1.toLowerCase() // Restituisce "testo"  
var T3 = T1.toUpperCase() // Restituisce "TESTO"  
// Queste assegnazioni non modificano le variabili!
```

```
var testo = prompt ( " Inserisci un Testo " , "Hello , World!" ) ;  
var ultimoCarattere = testo.charAt ( testo.length - 1 ) ;  
alert ( ultimoCarattere ) ; // codice ascii non il carattere
```

Perchè -1 ?

0	1	2	3	4	5	6	7	8	9	10	11
H	e	l	l	o	,	w	o	r	l	d	!

Inverso di char AT

```
Var stringa ;  
For ( codice = "0".charCodeAt(0) ; codice <= "9".charCodeAt(0) codice ++ )  
{  
    stringa = stringa + string.fromCharCode ( codice ) ;  
}
```

Restituisce
la posizione
partendo da Zero
-1 = ricerca fallita

oggetto_string.**indexOf** (search , start)

```
var stringa = "Ma la volpe col suo balzo ha raggiunto il quieto fido"  
//ricordiamoci che JS converte automaticamente le stringhe  
var posiz = stringa.indexOf("v"); //contiene il valore 6)  
var posiz2 = stringa.indexOf("k"); // restituisce -1
```

Il metodo **lastIndexOf()** // inizia a contare dalla fine

Sezionamento stringhe.
mia_stringa.substr() ;

```
var stringa = "Questo è un esempio";  
var str2 = stringa.substr(0, 6);  
//restituisce "Questo"  
var str3 = stringa.substr(7);  
//restituisce "è un esempio"
```

Mettere al posto di x y
mia_stringa.replace() ;

```
var s = "L'utente Pinco ha modificato questa pagina"  
var s2 = s.replace("Pinco", "Pallino");  
//s2 ora contiene  
"L'utente Pallino ha modificato questa pagina"  
//s1 contiene ancora  
"L'utente Pinco ha modificato questa pagina"
```

Dividere → split ()
mia_stringa.split();

restituisce :
stringa di stringhe → array

```
stringa.split(separatore);  
"a,e,i,o,u".split(",");  
//restituisce "a", "e", "i", "o", "u"  
"a,e,i,o,u,".split(",");  
//attenzione: restituisce "a", "e", "i", "o", "u", ""
```

Metodi matematica
Math.modulo
Math.abs();

Math.Abs()	: restituisce il valore assoluto.
Math.Min()	: restituiscono il minimo
Math.Max()	: restituiscono il massimo
Math.Sqrt()	: la radice quadrata
Math.pow(b, n)	: potenza ennesima di b
sin(), cos(), tan(), log()	: Trigonometria classica.
Math.random()	: Genera un Numero casuale

Array

```
var descCampi = new Array ("ID", "Modello", "Prezzo")
var prod1 = new Array (1, "3000 plus", 35);
var prod2 = new Array (5, "4000 minus, 12);
```

Concatena

```
var a = new Array (1, 2, 3);
var b = new Array (4, 5, 6);
var c = a.concat(b); //1, 2, 3, 4, 5, 6
```

Ordine alfabetico

```
var a = new Array ("e", "a", "u", "i", "o");
a.sort() /ora le vocali sono ordinate nell'array a
// questo metodo modifica l'array ( non è come string)
MAIUSCOLE poi minuscole
```

Array inverso reverse()

```
var vettore = new Array (5); // nuovo array con 5 locazioni
var vocali = new Array ("A", "E", "I", "O", "U");
vocali.reverse(); //ora contiene U O I E A
```

Copiare sezioni di array Slice ()

```
var vettore = new Array (5);
//crea un array contenente 5 elementi

var a = new Array ("A", "E", "I", "O", "U");
var b = a.slice(1,4); //b contiene gli elementi E I O
```

Inizializzazione di un array

```
var a = new Array ("A", "E", "I", "O", "U");
for (indice in a)
{
a[indice] = ""; // stringa nulla,annulla il valore
} // l'array contiene 5 elementi vuoti!
```

Cicli di controllo

Struttura di controllo SE

```
If ( espressione )
{
    // ISTRUZIONI
}
else
{
    Istruzioni
}
```

Struttura di controllo SWITCH (Multi SE)

```
switch ( espressione )
{
    case costante_1 :           // ISTRUZIONI
        break ;
    case costante_2 :           // ISTRUZIONI
        break ;
    case costante_ENNE :       // ISTRUZIONI
        break ; // se manca break Javascript va in LOOP!
}
```

Operatore TERNARIO

condizione ? esp1 : esp2

Controlla l'espressione se è vera passa la 1° esp1.
Se è Falsa passa alla 2° espressione esp2.

```
var anni = prompt('Quanti anni hai?', 20);
var msg = "Ciao, vedo che sei  
" + (anni >= 18 ? "maggiorenne" : "minorenne");
alert(msg);
```


Cicli di controllo FOR WHILE

Struttura di controllo FOR

```
for ( Inizializzazione ; condizione ; incremento )  
{  
    // ISTRUZIONI  
}
```

Esempio

```
For (i=0 ; i<10 ; i++)  
{ matrice[i] = 0  
}
```

Struttura di controllo WHILE

```
while ( espressione )  
{  
    // ISTRUZIONI _ RIPETE FINO A CHE È VERA  
}
```

Struttura di controllo DO-WHILE

```
do  
{  
    // ISTRUZIONI  
} while ( CONDIZIONE );
```

L'istruzione CONTINUE Interrompe un flusso di controllo per evitare i LOOP (Ciclo infinito)

Che cosa sono gli oggetti

Gli oggetti sono una classe di tipologia per risolvere un determinato problema.
Un oggetto può contenere delle proprietà che definiscono la struttura.

Struttura oggetto

```
var nome_oggetto = new nome_classe ();  
  
var nome_oggetto = new nome_classe (parametri_del_costruttore);  
  
var mia_macchina = new macchina ("Utilitaria", "Rosso");
```

Proprietà

```
nome_classe.proprietà
```

Metodi

```
nome_oggetto.metodo_esempio ()
```

Struttura with

```
with (la_mia_macchina)  
{  
    .istanza (20) //20 euro di benzina  
    alert(.lunghezza);  
    classe.istanza_2(50);  
    //Per altri oggetti devo specificare il nome.  
}
```

I cookie

Definizione : In informatica, i cookie sono piccoli file di testo memorizzati sul client. che contengono informazioni dei siti web (server)

Per ottenere le informazioni ineterne dei cookie bisogna accedere alle proprietà.

document.cookie

Restituisce una stringa contenente coppie di nome-valore.
Le informazioni di ciascun dato sono separate da " ;"punto e virgola e spazio.

Esempio:

```
document.cookie="NomeUtente=Ramac;expires=Tue, 28 August 2007 00:00:00"
```

NomeUtente=Paolo;UltimaVisita=Giovedì 30 Gennaio 2020 13:34:12

Nome cookie

;
Separatore

Valore

;
Separatore

Scadenza
Experidate
Parametro
Opzionale

Funzione
per
Creare
i
cookie

```
function impostaCookie (nome, valore, scadenza) {  
  if (scadenza == "") {  
    var oggi = new Date();  
    oggi.setMonth(oggi.getMonth() + 3);  
    //restituisce la data nel formato necessario  
    scadenza = oggi.toGMTString();  
  }  
  valore = escape(valore);  
  document.cookie=nome + "=" + valore + ";expires=" + scadenza;  
}
```

Funzione
per
se esistono

Oppure
Non esistono

i
cookie

```
function valoreCookie (nome)  
{  
  var valore=document.cookie; //ottiene la stringa di cookie  
  var inizioCookie=valore.indexOf(" " + nome + "=");  
  //trova il cookiedesiderato  
  if (nizioCookie == -1) { //se non esiste, magari è all'inizio della stringa  
    inizioCookie = valore.indexOf(nome + "=");  
  }  
  if ( inizioCookie == -1) // ← il Cookie non esiste proprio  
  {  
    //il cookie non esiste proprio  
    valore = null;  
  }  
  if (inizioCookie >= 0) // ←Se il Cookie esiste  
  {  
    inizioCookie = valore.indexOf("=", inizioCookie) + 1;  
    //qui inizia la  
    stringa del valore  
    var fineCookie = valore.indexOf(";", inizioCookie);  
    //qui finisce  
    if (fineCookie == -1) // ← se non viene trovato, allora è l'ultimo cookie  
    {  
      fineCookie = valore.length;  
      valore = unescape(valore.substring(inizioCookie, fineCookie)); //elimina i  
      caratteri commutati  
    }  
  }  
  return valore;  
}
```

Keyboard

Keydown

Keyup

keypress

input

Eventi

Tastiera
Mouse

Mouse

auxclick

dblclick

mouseover

mousemove

mouseenter

mousewheel

contextmenu

click

mouseup

mousedown

mouseout

mouseleave

wheel

Catturare l'input tastiera

```
<!DOCTYPE html>
<html>
<body>

<p>Premi un un qualsiasi tasto della tastiera! </p>

<input type="text" size="40" onkeydown="myFunction(event)">

<script>
function myFunction(event) {
    var x = event.key;

    if (x == "a" || x == "A") {
        alert ("You pressed the 'A' key!");
    }
    else
        alert (x)
    }
</script>

</body>
</html>
```

Cambiare il paragrafo <P> tramite Identificativo
Al click del mouse fai apparire l'orario

My First JavaScript

Click me to display Date and Time.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>My First JavaScript</h2>
```

```
<button type="button"Onclick=  
"document.getElementById('demo').innerHTML = Date()" >
```

```
Click me to display Date and Time.</button>
```

```
<p id = "demo" > </p>
```

```
</body>
```

```
</html>
```

My First JavaScript

Click me to display Date and Time.

Thu Apr 23 2020 23:04:36 GMT+0200 (Ora legale dell'Europa centrale)

Funzioni ritardate

Visualizza dopo
3000 milli-secondi (3 secondi)
il caricamento
della pagina

```
var tim = setTimeout('alert("ciao!")', 3000);
```

Funzioni programmate

```
var timer = setInterval ("aggiornaPagina()", 1000);  
clearInterval(timer); // per eliminarlo
```

Gestire le Date

```
var d = new Date();  
new Date()  
new Date(year, month, day, hours, minutes, seconds, milliseconds)  
new Date(milliseconds)  
new Date(date string)  
var d = new Date(2020, 12, 24, 10, 30, 25, 0);
```


BOM DOM

La differenza tra BOM e DOM è semplice.

BOM →	Browser Object Model	→	Modello Navigatore internet
DOM →	Document Object Model	→	Modello documento

Il metodo alert che abbiamo chiamato semplicemente alert() è figlia di window.
Quindi se vogliamo scriverlo in forma estesa è window.alert()

```
window.defaultStatus = "Testo personalizzato"; // scritta al caricamento completo della pagina
window.status = "Testo personalizzato"; // sezione della pagina
alert() // messaggio di allerta
prompt() // inserimento testo
window.open("url", "titolo", "parametri") // Apre una nuova finestra
```

Nome parametro	Valori
copyHistory	yes, no
width, height, top, left	Numero intero
Location	yes, no
menubar	yes, no
resizable	yes, no
scrollbars	yes, no
status	yes, no
toolbar	yes, no

TTS

```
<!DOCTYPE html>
<html lang="it" dir="ltr">
<head>
  <title> HEHello world TTSt</title>

  <script type="text/javascript">
    function parla1()
    {
      var msg = new SpeechSynthesisUtterance('Hello world');
      window.speechSynthesis.speak(msg);
    }

    function parla_da_ID()
    {
      alert ( testo=document.getElementById('Topolino').value );
      var msg = new SpeechSynthesisUtterance( document.getElementById('Topolino').value );
      a = window.speechSynthesis.speak( msg );
    }

  </script>

</head>

<body>

  <h2> TTS parla </h2>
  <input value="Ciao Paolo come stai?" id="Topolino">
  <button type="button" onclick="parla1() " name ="parla1" >
  <button type="button" onclick="parla_da_ID()" value="PAOLO" >

</body>
</html>
```

Al click cambia il testo!

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1 onclick="changeText(this)">Click on this text!</h1>
```

```
<script>
```

```
function changeText(id) {  
  id.innerHTML = "Ooops!";
```

```
}
```

```
</script>
```

```
</body>
```

```
</html>
```

Esempio di una funzione che ritorna un numero (f)

Inserisci in input un numero > 0 e calcola il fattoriale!

```
<!DOCTYPE html>
<html lang="it" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title> Fattoriale in Javascript </title>
    <script type="text/javascript">
      CODICE JAVASCRIPT
    </script>
  </head>
  <body>

    <h2> Calcolo Fattoriale </h2>
    <form name="miof">
      numero
      <input type="text" size=5 name="num" >
      <input type="button" value="Calcola"
        onclick="calcolaFatt(document.miof.num.value)"
      >

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

```
function fatt(n)
{
  f = 1 ;
  if (n < 0 )
    return - 1;
  i = 1 ;
  do
  {
    f= f * i ;
    i= i + 1 ;

  } while (i<=n);
  return f;
}

function calcolaFatt(n)
{
  f=fatt(n);
  if ( f== -1 )
    alert("Errore fattoriale "-1" !");
  else
    alert ("Il fattoriale di " + n + " = " + f) ;
}
```

Come evitare alcuni errori comuni

- Leggi attentamente il tuo codice per eventuali errori di battitura
- Assicurati che tutte le parentesi siano chiuse (può aiutare in questo un buon editor)
- Ricorda che JavaScript è *case-sensitive*: Name e name sono due variabili differenti
- Non usare parole riservate come nomi di variabili o di funzioni
- Ricordati di effettuare l'*escape* degli apici singoli/doppi:
- Quando converti i numeri usando la funzione `parseInt` ricorda che "08" o "09" sono valutati come numeri in base otto, a causa dello zero.
- Ricorda che JavaScript è indipendente dalla piattaforma, ma lo è dal browser!

Parole riservate

break case catch continue default delete
else false finally for function if in
instanceof new null return switch this
throw true try typeof var void while with

Indice delle pagine

Introduzione a javascript	Pagina 1
Descrizione Javascript	Pagina 2
Tipi di variabili	Pagina 3
Array funzioni alert	Pagina 4
Metodi stringa	Pagina 5
Metodi stringa & matematica	Pagina 6
Metodi Array	Pagina 7
Cicli di controllo 1 di 2	Pagina 8
Cicli di controllo 2 di 2	Pagina 9
Gestione oggetti	Pagina 11
Cookie	Pagina 11
Gestione Cookie	Pagina 12
Introduzione agli eventi tastiera e mouse	Pagina 12
Catturare l'input tastiera	Pagina 13
Metodo click su un testo H1	Pagina 14
Funzioni ritardate e Date	Pagina 15
BOM vs DOM	Pagina 16
Sintesi vocale su Google Chrome	Pagina 17
Esempio cambio testo con un click	Pagina 18
Esempio funzioni al lavoro	Pagina 19
Errori comuni	Pagina 20
Indice delle pagine	Pagina 21