

# Corso Javascript

a.s. 2015/2016

Prof. Aldo Guastafierro

# introduzione

Le pagine web basate esclusivamente sul linguaggio HTML sono dette “**pagine statiche**” poiché i loro contenuti sono costanti e con esse l’utente non può interagire (tranne che per il click sui collegamenti ipertestuali e ben altre poche azioni).

Si parla invece di pagine “**interattive**” quando la pagina web è in grado di rispondere alle azioni dell’utente compiendo elaborazioni in risposta ad esse come, ad esempio:

- il cambio di immagini al passaggio del mouse;
- la visualizzazione di messaggi di avvertimento o risposte ad azioni effettuate;
- Il controllo dei dati inseriti in un modulo (form);

Per realizzare tali effetti vengono introdotti, all’interno di pagine HTML, un insieme di istruzioni (dette **script**) scritte in particolari linguaggi di programmazione che rendono interattiva la pagina stessa.

Questi insiemi di istruzioni saranno eseguiti dal **browser** (e quindi sul **client**) al momento dell’apertura della pagina o in seguito a particolari azioni che l’utente effettua sulla pagina, pertanto vengono detti “**script lato client**”. I due linguaggi più diffusi per la scrittura di tali script sono **Javascript e Vbscript**.

# Inserire uno script in una pagina HTML

Gli script javascript possono essere inclusi direttamente nelle pagine web tramite il tag *script*.

Esempio :

```
<script language="javascript">
```

*Istruzioni*

```
</script>.
```

# Generazione di HTML tramite Javascript

Per inserire dei contenuti all'interno della pagina HTML tramite javascript si utilizza l'istruzione **document.write("...")**. Ciò che è racchiuso tra parentesi viene visualizzato dal browser nella pagina stessa.

Es.

```
<html>
<head> <title> Esempio1 </title> </head>
<body>
<script language="javascript">
    document.write("Questo testo è stato inserito tramite uno <b> script
    </b>");

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

# Visualizzazione di finestre di dialogo tramite Javascript

## alert()

**alert:** visualizza una finestra di dialogo contenente un messaggio ed un bottone di OK.

La sintassi è la seguente:

```
alert("messaggio");
```

dove "messaggio" può essere qualsiasi stringa.

*alert* viene di solito usato quando si vuol dare una comunicazione istantanea all'utente.

Non è richiesta in questo caso un'interazione. La comunicazione è a senso unico (dal sito all'utente), quindi *alert* non ritorna alcun valore allo script.

Es:

```
<HTML>
  <HEAD>
  </HEAD>
  <BODY>
<SCRIPT language=Javascript>
  alert("Dato non valido");
</SCRIPT>
</BODY>
</HTML>
```



# Visualizzazione di finestre di dialogo tramite Javascript

## confirm()

**confirm**: visualizza una finestra di dialogo con un messaggio e due bottoni: OK e Annulla.

La sintassi è la seguente:

```
confirm("messaggio");
```

dove "messaggio" può essere qualsiasi stringa.

*confirm* viene usato quando si voglia far scegliere all'utente se effettuare o no una certa operazione

*confirm* ritorna un valore true se l'utente ha cliccato su OK o false se ha cliccato su Annulla.

Es.

```
<HTML>  
  <HEAD>  
  </HEAD>  
  <BODY>  
    <SCRIPT>  
      confirm("Vuoi confermare i dati inseriti?");  
    </SCRIPT>  
  
  </BODY>  
</HTML>
```



# Visualizzazione di finestre di dialogo tramite Javascript

## prompt()

**prompt**: visualizza una finestra di dialogo con un messaggio, due bottoni (**OK** e **Annulla**) ed un campo testo in cui l'utente può digitare qualcosa.

La sintassi di **prompt** è la seguente:

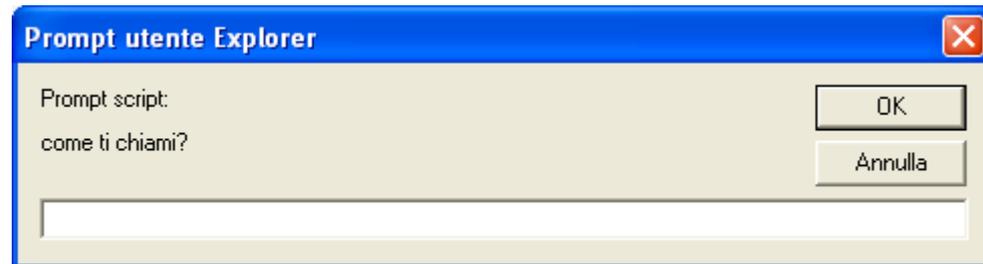
```
prompt("messaggio",testoDefault);
```

"messaggio" è ancora una volta una stringa con la richiesta che si fa all'utente, mentre *testoDefault* (opzionale) è un testo preimpostato nella risposta che compare già selezionato (ad esempio un valore di default), cosicché viene immediatamente sostituito appena l'utente scrive qualcosa. Se *testoDefault* non viene dichiarato nel box compare la scritta "undefined".

Se l'utente clicca su **OK** la variabile associata a **prompt** assume il valore del testo immesso (o del *testoDefault*), se invece l'utente clicca su **Annulla** la variabile associata conterrà **null**.

Esempio:

```
<HTML>
  <HEAD>
  </HEAD>
  <BODY>
  <SCRIPT>
    var nome = prompt("Come ti chiami?");
  </SCRIPT>
  </BODY>
</HTML>
```



# Le variabili

Come ogni altro linguaggio di programmazione i dati che caratterizzano il problema o l'aspetto della realtà che si deve rappresentare sono memorizzati all'interno di **variabili**. Tuttavia in javascript non è obbligatorio dichiarare le variabili prima di utilizzarle, anche se è comunque consigliato farlo.

Per dichiarare le variabili si utilizza la parola chiave *var* seguita dal nome della variabile, senza specificare il tipo.

Es:     *var x;*

# Esercitazione N°1

Scrivere una pagina web in cui, utilizzando delle finestre di dialogo vengono acquisiti due numeri. I due numeri vengono poi sommati ed il risultato visualizzato sia all'interno della pagina che in una ulteriore finestra di dialogo

```
<html>
  <head> <title> Javascript - Esercizitazione N°1 </title> </head>
  <body>
    <script language="javascript">
      var x1, x2, somma;
      x1= prompt("inserisci un numero",""); (Nota 1)
      x2= prompt("inserisci un numero","");
      confirm("sei proprio sicuro di voler fare la somma?");
      somma=eval(x1) + eval(x2); (Nota 2)
      alert("la somma di " + x1 + " e " + x2 + " è " + somma);
      document.write("la somma di " + x1 + " e " + x2 + " è " + somma); (Nota 3)
    </script>
  </body>
</html>
```

**Note:**

Nell'istruzione `x1= prompt("inserisci un numero","");` il valore digitato dall'utente viene memorizzato nella variabile `x1`

Tramite la funzione `eval`, quello che viene letto da tastiera sotto forma di stringa di caratteri viene tradotto in formato numerico

Per concatenare in una stessa espressione dei contenuti fissi e dei valori memorizzati in variabili si utilizza il simbolo `+` : `alert("la somma di " + x1 + " e " + x2 + " è " + somma);`