

Stringhe e Oggetto String

In informatica una [stringa](#)¹ è una sequenza di uno o più caratteri alfanumerici.

Valori stringa in Javascript

In JavaScript una stringa o [string literals](#) si indica inserendola all'interno di doppie virgolette (") o apici (').

Bisogna però fare attenzione a chiudere una stringa con lo stesso apice con la quale è stata aperta; sono ad esempio stringhe valide:

```
"Hello, world ! 1234"  
"Peter O'Toole"
```

Per forzare la conversione da un numero ad una stringa basta usare la sintassi `numero.toString()`.

Sequenza di [escape](#)

Quando si lavora con i valori stringa o *string literals* capita che alcuni caratteri, come ad esempio gli apostrofi, o le doppie virgolette blocchino l'esecuzione di un'istruzione: questo accade perché c'è conflitto fra il carattere e le istruzioni di javascript.

Non è valida la seguente stringa:

```
'Peter O'Toole'
```

in quanto il parser la analizzerebbe e, arrivato a `O'` penserebbe che la stringa si chiuda, senza interpretare correttamente `Toole'`

Per questo motivo si usano le sequenze dette di *escape* o di controllo, che permettono di ottenere il risultato desiderato. È possibile dunque specificare che caratteri come " e ' non indicano la fine del testo ma un carattere qualsiasi facendoli precedere dal **carattere di commutazione** o *backslash* \.

Si tratta di anteporre tale carattere **senza spazi**, come di seguito esemplificato.

Ad esempio sono stringhe valide:

```
'Peter O\'Toole'  
"Questo libro è depositato su \"it.wikibooks\""
```

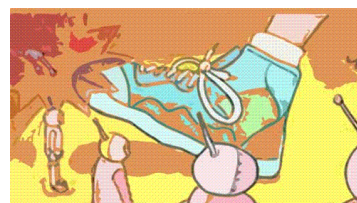
¹ Tratto da http://it.wikibooks.org/wiki/JavaScript/Tipi_di_dati_e_variabili

Oggetto String² in Javascript



Le **stringhe di caratteri** sono, in JavaScript, anche **oggetti istanza** di tipo **String**.

Un oggetto in JavaScript è una **struttura di dati** che comprende **attributi** (o dati dell'oggetto) e **metodi** (o funzioni dell'oggetto).



Metodi

L'oggetto **String**, nativo in JavaScript, dispone del metodo **costruttore** e di altri **metodi** che rappresentano le azioni eseguibili dagli oggetti istanza: permettono di effettuare numerose operazioni sulle stringhe quali ricerca, isolamento di un carattere e altro ancora.

Il **costruttore** serve per creare una nuova **istanza** di oggetto **String** con la sintassi:

```
var nome_variabile = new String(stringa);
```

Tale costruttore prende come parametro la stringa che sarà manipolata nel corso dello script.

L'oggetto String è un po' particolare, in quanto corrisponde ad un "doppione" del tipo di dato primitivo *string*, analizzato precedentemente.

Dal momento che JavaScript **effettua automaticamente la conversioni dei dati** quando necessario, la differenza è dal punto di vista pratico, in quanto:

- se si crea una stringa con la normale sintassi

```
var variabile = "testo";
```

e successivamente si vuole trattarla come un oggetto String usando metodi o proprietà della classe String, JavaScript converte automaticamente la stringa in un oggetto String

- se si ha un oggetto String si può recuperare la sua stringa - cioè il valore letterale - poichè JavaScript converte automaticamente l'oggetto String in una stringa contenente la stringa indicata nel costruttore.

Si noti che i metodi dell'oggetto **String** lavorano sulla stringa contenuta nell'oggetto ma restituiscono il valore desiderato senza modificare il valore dell'oggetto stringa.

Attributi

L'oggetto String dispone di una proprietà rilevante, la proprietà `length` che restituisce il numero di caratteri contenuti in una stringa:

```
var esempio = "Sono una stringa primitiva"
alert (esempio.length); //restituisce 26
```

Nell'esempio appena visto, nella seconda riga la variabile `esempio` viene **convertita** in un **oggetto String** per accedere alla proprietà `length`.

Sempre per quanto appena detto, potremmo semplicemente scrivere:

```
alert ("Sono una stringa primitiva".length); //restituisce 26
```

² Tratto da http://it.wikibooks.org/wiki/JavaScript/Oggetto_String