

Trattamento delle stringhe in php

Concatenamento

per concatenare due o più stringhe si usa il `.` (punto)

P.es: `$c=$a.$b;`

Se si desidera includere un carattere separatore, ad esempio lo spazio, si può anche usare il **join**:

`$risultato=join($carattere, $s1, $s2, ...);`

unisce le variabili `$s1`, `$s2`, ecc. in un'unica variabile `$risultato`, inserendo il carattere `$carattere` in mezzo ad ognuna

Conversione da ASCII a carattere

`char(codiceASCII)`

restituisce un carattere corrispondente al *codice ASCII* fornito tra parentesi

Conversione da carattere a ASCII

`ord(carattere)`

restituisce il codice ASCII corrispondente al *carattere* fornito tra parentesi.

Se invece di un solo carattere, la variabile ne contiene molti, viene comunque preso in considerazione solo il primo

Conversione in minuscolo

`strtolower(stringa)`

converte in minuscolo la *stringa* passata come argomento

Conversione in maiuscolo

`strtoupper(stringa)`

converte in maiuscolo la *stringa* passata come argomento

Lunghezza di una stringa

`strlen(stringa)`

restituisce la lunghezza della *stringa*

Eliminazione degli spazi

`trim(stringa)`

elimina gli eventuali spazi iniziali e finali da una stringa.

Può essere utile quando si devono confrontare due stringhe, per essere sicuri che qualche spazio in più non le renda diverse anche se contengono la stessa parola o frase.

Estrazione di caratteri

Prima di tutto è necessario sapere che, per calcolare la posizione dei caratteri nella stringhe bisogna partire a

Trattamento delle stringhe in php

contare dal numero 0. Il primo carattere è quindi il numero 0, il secondo il numero 1 e così via.

La stringa può essere usata come un array di caratteri, indicando tra parentesi graffe la posizione del carattere che si vuole estrarre. quindi, per esempio:

\$nome{0}

restituisce il primo carattere della variabile \$nome

Estrazione di sottostringhe

substr(stringa, posizione, lunghezza)

estrae da *stringa* una sottostringa partendo da *posizione* e prendendo tanti caratteri quanti sono indicati dall'argomento *lunghezza*

Esempio: substr("parola", 2, 3)

restituisce tre caratteri a partire dalla posizione 2 della stringa, cioè **rol** Se la *lunghezza* non è indicata, viene presa tutta la stringa a partire da *posizione* in avanti

Esempio: substr("parola", 2)

restituisce tutta la stringa a partire dalla posizione 2 della stringa, cioè elimina i primi caratteri. Il risultato è **rola** Se la *posizione* è negativa, per calcolare il punto di partenza si parte dalla fine della stringa, e si prendono gli ultimi caratteri. Se non è indicato l'argomento *lunghezza*, si prendono gli ultimi *posizione* caratteri, se invece *lunghezza* è maggiore di 0 si prendono *lunghezza* caratteri andando in avanti verso la fine della stringa

Esempio: substr("parola", -2)

restituisce gli ultimi 2 caratteri della stringa, cioè la

substr("parola", -4, 2)

prende 3 caratteri a partire dal quartultimo della stringa, cioè **ro**

L'argomento *lunghezza* può essere usato anche in negativo: in questo caso indica di togliere altrettanti caratteri dalla fine della stringa.

Esempio: substr("parola", 0, -2)

Indica di prendere la stringa a partire dal primo carattere, tralasciando gli ultimi 2. Il risultato è **paro**

substr("parola", 2, -2)

Indica di prendere la stringa a partire dal terzo carattere, tralasciando gli ultimi 2. Il risultato è **ro**

substr("parola", -4, -2)

Indica di prendere la stringa a partire dal quartultimo carattere, tralasciando gli ultimi 2. Il risultato è sempre **ro**

N.B.: se il punto di partenza risulta fuori dalla stringa. il risultato sarà *falso*. Se la *lunghezza* è negativa ed indica di togliere più caratteri di quanti ce ne siano, il risultato sarà una stringa vuota.

Se si desidera invece spezzare una stringa in più parti, per esempio suddividere una frase in parole, si può usare **split** in abbinamento con la funzione **list()**:

Trattamento delle stringhe in php

list(\$a1, \$a2, ... , \$aN) = split(\$carattere, \$frase, N);

Tra le parentesi della funzione list va inserito l'elenco di variabili che conterranno i risultati.

(Naturalmente, al posto di list si può specificare un array).

La variabile \$carattere conterrà il carattere separatore, quindi probabilmente verrà preso lo spazio.

\$frase è la variabile da suddividere.

N (facoltativo) è il numero massimo di spezzoni da creare: se si ha una lista finita è meglio specificarlo, se invece il risultato viene inserito in un array dinamico, non è necessario

P. es:

list(\$nome,\$cognome,\$telefono)=split(" ", \$record, 3);

Il record viene suddiviso nei singoli campi usando lo spazio come carattere separatore

\$vettore=split(" ", \$frase);

La frase viene suddivisa in parole usando lo spazio come elemento separatore: Ciascuna delle parole sarà contenuta in un elemento dell' array : \$vettore[0], \$vettore[1], ...

Ricerca di una sottostringa

strpos(stringa, sottostringa)

cerca la *sottostringa* indicata nella *stringa*, e restituisce la posizione in cui viene trovata (il primo carattere).

Se la sottostringa non è contenuta nella stringa data, la funzione restituisce *false*.