

## Linguaggio PHP - Scheda N. 02 - Vettori e Sottoprogrammi

<p style="text-align: center;"><b>come Creare un Vettore in PHP</b></p>	<p>Al momento della creazione di un Vettore <b>non è necessario indicarne né tipo né dimensione</b>.</p> <p>E' possibile creare un <i>vettore vuoto</i> e <i>aggiungere successivamente</i> gli elementi.</p> <p>Per <b>Creare un Vettore Vuoto</b>:</p> <pre>&lt;nome-vettore&gt; = [ ]           ... oppure... &lt;nome-vettore&gt; = array ( )</pre> <p>Per <b>Inizializzare un Vettore con una Serie di Dati</b>:</p> <pre>&lt;nome-vettore&gt; = [ &lt;valore&gt;, &lt;valore&gt;, ... ]</pre>	<p><i>Esempi:</i></p> <pre>\$VetDati = [ ]; \$VetNomi = array ( );</pre> <hr/> <pre>\$Colori = [ 'Rosso',             'Verde',             'Giallo',             'Blu'             ];</pre>
<p style="text-align: center;"><b>come Accedere o Aggiungere Elementi ad un Vettore in PHP</b></p>	<p><b>Accesso al Singolo Elemento</b> di un Vettore:</p> <pre>&lt;nome-vettore&gt; [ &lt;indice&gt; ]</pre> <p>Il Vettore è una Struttura Dati in <b>Base Zero</b>, ossia il <i>primo elemento è in posizione 0</i>.</p> <p>Per <b>Creare un Nuovo Elemento</b> in una data posizione di un Vettore, basta <b>assegnarvi un valore</b>:</p> <pre>&lt;nome-vettore&gt; [ &lt;indice&gt; ] = &lt;valore&gt;</pre> <p>Se l'elemento in posizione &lt;indice&gt; <i>non esiste</i>, allora <b>viene creato</b> e in esso viene memorizzato il &lt;valore&gt;.</p>	<p><i>Esempi:</i></p> <pre>\$VetNomi [ 0 ] = 'Mario';</pre> <pre>\$Pos = 5 ; echo \$VetNomi [ Pos ] ;</pre> <hr/> <pre>\$VetNomi = [ ] ;</pre> <pre>// crea l'elemento in posizione 0 ... \$VetNomi [ 0 ] = 'Giovanni' ;</pre>
<p style="text-align: center;"><b>come usare i Vettori Associativi</b></p>	<p>In un <b>Vettore Associativo</b>, l'<b>accesso al singolo elemento avviene utilizzando una Chiave</b> (in genere, una <i>stringa</i>), anziché un indice numerico.</p> <p><b>Inizializzare un Vettore Associativo:</b></p> <pre>&lt;nome-vettore&gt; = [ &lt;chiave&gt; =&gt; &lt;valore&gt;,                   &lt;chiave&gt; =&gt; &lt;valore&gt;, ... ]</pre> <p><b>Creazione o Accesso ad un Elemento:</b></p> <pre>&lt;nome-vettore&gt; [ &lt;chiave&gt; ] = &lt;valore&gt;</pre> <p>Se l'elemento relativo alla &lt;chiave&gt; specificata <b>non esiste</b>, allora <b>l'elemento viene creato</b> e in esso viene memorizzato il &lt;valore&gt;. Se invece, <b>esiste</b>, ne viene semplicemente <b>modificato il valore corrente</b>.</p> <p>N.B.: le variabili superglobali <b>\$_GET</b> e <b>\$_POST</b> sono dei <b>Vettori Associativi</b> in cui le <b>Chiavi</b> sono i <i>nomi dei parametri</i> e i <b>Valori</b> sono i rispettivi <i>valori dei parametri</i>.</p>	<p><i>Esempi:</i></p> <pre>\$Stipendi [ 'Mario' =&gt; 1200 ,             'Anna' =&gt; 1500 ,             'Carlo' =&gt; 1300 ] ;</pre> <pre>// questa istruzione visualizzerà 1300 ... echo \$Stipendi [ 'Carlo' ] ;</pre> <hr/> <pre>// crea l'elemento 'Marco' e il suo stipendio \$Stipendi [ 'Marco' ] = 1700 ;</pre>

<p>come <b>Scorrere con un Ciclo un Vettore Associativo</b></p>	<p>Non avendo indici numerici, la <b>scansione ciclica di un Vettore Associativo</b> deve avvenire come segue:</p> <pre><b>foreach ( &lt;vettore&gt; as &lt;variabile-valore&gt; )</b> <b>{ &lt;istruzioni ... &gt; }</b></pre> <p>Il ciclo ripete le istruzioni per un <b>numero di passi pari al numero di elementi</b> del vettore del ciclo: ad ogni passo, la <b>&lt;variabile-valore&gt; assume un diverso valore</b> del vettore.</p> <p>E' possibile anche la forma:</p> <pre><b>foreach( &lt;vettore&gt; as &lt;var-chiave&gt; =&gt; &lt;var-valore&gt; )</b> <b>{ &lt;istruzioni ... &gt; }</b></pre> <p>Qui, ad ogni passo, <b>&lt;var-chiave&gt; e &lt;var-valore&gt; contengono i dati relativi ad una diversa coppia chiave/valore</b> del vettore.</p>	<p><i>Esempi:</i></p> <pre><b>\$Stipendi</b> = [ 'Mario'=&gt;1200 ,               'Anna' =&gt;1500 , 'Carlo'=&gt;1300 ] ; <b>\$Somma</b> = 0; <b>foreach ( \$Stipendi as \$Importo )</b> <b>    \$Somma = \$Somma + \$Importo ;</b> echo 'Somma Stipendi: ' . <b>\$Somma</b> ;</pre> <p>nella pagina sarà inserito questo HTML ... <b>Somma Stipendi: 4000</b></p> <hr/> <pre><b>foreach ( \$Stipendi as \$Nome =&gt; \$Imp )</b> echo <b>\$Nome</b> . ':' . <b>\$Imp</b> . '&lt;br /&gt;' ;</pre> <p>nella pagina sarà inserito questo HTML ... <b>Mario: 1200</b> <b>Anna: 1500</b> <b>Carlo: 1300</b></p>
<p><b>Altre Operazioni sui Vettori</b></p>	<pre><b>count ( &lt;vettore&gt; )</b> <b>in_array ( &lt;valore&gt;, &lt;vettore&gt; )</b> <b>array_keys_exists ( &lt;chiave&gt;, &lt;vett.ass.&gt; )</b> <b>array_keys ( &lt;vett.ass.&gt; )</b> <b>array_values ( &lt;vett.ass.&gt; )</b></pre>	<p>numero di elementi nel vettore (int) esistenza di un valore nel vettore (bool) esistenza di una chiave nel vett.ass. (bool) vettore delle sole chiavi del vett.ass. (vettore) vettore dei soli valori del vett.ass. (vettore)</p>
<p>come <b>Richiamare un Sottoprogramma in PHP</b></p>	<p>Per <b>Richiamare un Sottoprogramma</b> (ossia per richiederne l'esecuzione) si usa la solita regola:</p> <pre>&lt;nome-sottopr.&gt; ( &lt;parametro&gt;, &lt;parametro&gt;, ... )</pre> <p>N.B.: anche in PHP, i sottoprogrammi possono <i>restituire un risultato (funzioni) o non restituire nulla (procedure)</i>.</p>	<pre>// richiama un sottoprogramma che NON restituisce un risultato (procedura) ... <b>Saluto ( ) ;</b></pre> <hr/> <pre>// richiama un sottoprogramma che restituisce un risultato (funzione) ... <b>\$Num = 8;</b> <b>\$Ris = Doppio ( \$Num ) ;</b></pre>
<p>come <b>Dichiarare un Sottoprogramma in PHP</b></p>	<p>La dicitura <b>function</b> dichiara sia per sottoprogrammi <i>procedura</i> (void) che <i>funzione</i> (con risultato).</p> <pre><b>function &lt;nome&gt; ( &lt;parametro&gt;, &lt;parametro&gt;, ... )</b> <b>{ &lt;istruzioni ... &gt; }</b></pre> <p>L'istruzione <b>return</b> provoca la <b>restituzione di un risultato</b> e termina l'esecuzione del sottoprogramma.</p> <p>N.B.: se non si utilizza l'istruzione <i>return</i>, il sottoprogramma non restituisce risultati ed è quindi una procedura (void).</p>	<pre>// dichiarazione di un sottoprogramma che NON restituisce un risultato (procedura) ... <b>function Saluto ( )</b> <b>{ echo 'CIAO' ; }</b></pre> <hr/> <pre>// dichiarazione di un sottoprogramma che restituisce un risultato (function) ... <b>function Doppio ( \$Num )</b> <b>{ return \$Num * 2 ; }</b></pre>
<p>come passare dei <b>Parametri</b> a un <b>Sottoprogramma</b> in PHP</p>	<p>Sia nella <b>Chiamata</b> (<i>parametri effettivi</i>) che nella <b>Dichiarazione</b> (<i>parametri formali</i>), <b>i Parametri vanno semplicemente elencati</b>, separati da virgola e senza indicarne il tipo.</p> <p>In dichiarazione, un <b>Parametro di Output</b> (<i>passaggio per riferimento</i>) va preceduto dal <b>carattere &amp;</b>.</p>	<pre>// procedura con paramtro di output. ... <b>function Media ( \$A, \$B, &amp;\$M )</b> <b>{ \$M = (\$A + \$B) / 2 ; }</b> ... // chiamata ... <b>\$A = 5; \$B = 7; \$M = 0;</b> <b>Media ( \$A, \$B, \$M ) ;</b> echo <b>\$M</b> ; // ... genera 6 sulla pagina</pre>