

# Variabili in PHP

Le variabili e la loro visualizzazione

# PHP

- Nel linguaggio PHP le variabili hanno un nome che inizia con il simbolo del dollaro.

```
$esempio="Prova";
```

- Le variabili possono essere di diversi tipi, ad esempio:

Tipo semplice

Tipo array

# Variabili semplici

- Sono dette anche **variabili scalari**, come le grandezze scalari della fisica, perché sono aree della memoria che contengono un solo valore.

```
$variabile=5;
```

\$variabile
5

- Possono essere visualizzate con il costrutto **echo**

```
echo $variabile;
```

```
echo "<p>".$variabile."</p>";
```

# Variabili array

- Sono di tipo strutturato, perché possono contenere **più di un valore**, in diverse posizioni.

- Esempio di variabile array

$\$r$

'cf'	'nome'	'cognome'	← Chiave
"RSS"	"Mario"	"Rossi"	← Valore

## Variabili array

\$r

'cf'	'nome'	'cognome'	← Chiave
"RSS"	"Mario"	"Rossi"	← Valore

- Ogni posizione è chiamata **chiave** o indice.
- Ogni chiave può essere una parola o un numero

```
$r = array(  
    'cf' => "RSS",  
    'nome' => "Mario",  
    'cognome' => "Rossi");
```

← Chiave Valore

Nota: attenzione al simbolo apostrofo e alle virgolette

## Rappresentazione

- Non è importante la rappresentazione grafica verticale o orizzontale. Si tratta solo di uno schema che rappresenta il contenuto della memoria del computer.

\$r

'cf'	'nome'	'cognome'
"RSS"	"Mario"	"Rossi"



'cf'	"RSS"
'nome'	"Mario"
'cognome'	"Rossi"



## foreach()

- foreach() permette di ripetere delle istruzioni percorrendo tutte le posizioni di un array
- Non è necessario conoscere la struttura dell'array

```
foreach($variabile as $c => $v)  
{  
    //istruzioni da ripetere  
}
```

- \$c e \$v sono variabili temporanee che contengono rispettivamente la chiave e il valore di ogni elemento dell'array

## Esempio foreach()

- Il seguente esempio mostra tutte le chiavi dell'array

```
foreach($r as $c => $v)  
{  
    echo $c." ";  
}
```

passo	\$c	\$v	output

'cf'	'nome'	'cognome'
"RSS"	"Mario"	"Rossi"

## Esempio foreach()

- Il seguente esempio mostra tutte le chiavi dell'array

```
foreach($r as $c => $v)
{
    echo $c." ";
}
```

- Il seguente esempio mostra tutti i valori dell'array

```
foreach($r as $c => $v)
{
    echo "<p>".$v."</p>";
}
```

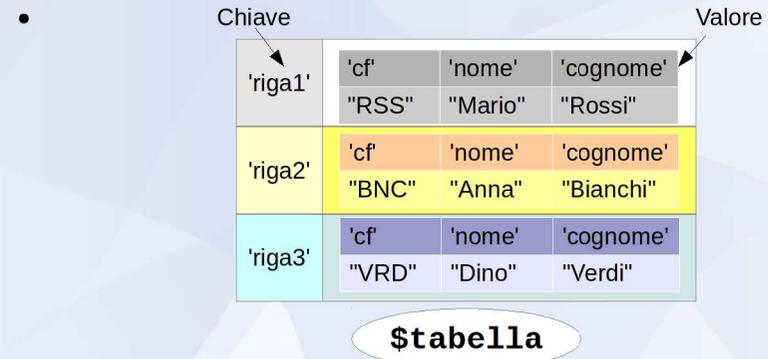
## Esempio foreach()

- Poiché per visualizzare un array si usa un foreach(), allora per visualizzare un array dentro un array si userà un foreach() dentro un foreach():
- Il foreach() più esterno percorre gli elementi dell'array più esterno, il più interno quelli dell'array più interno

```
foreach($stabella as $c_riga => $v_riga)
{
    foreach($v_riga as $c => $v)
    {
        echo "<p>".$v."</p>";
    }
}
```

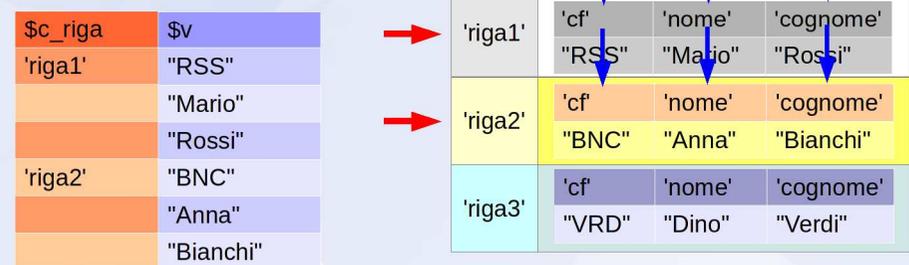
## Array di array

- È possibile che un array sia composto da altri array, cioè che ci siano array **dentro** un array.



## Esempio foreach()

```
foreach($stabella as $c_riga => $v_riga)
{
    foreach($v_riga as $c => $v)
    {
        echo "<p>".$v."</p>";
    }
}
```



## Tabelle in HTML

- Ricordiamo brevemente la forma delle tabelle in HTML: tr=table row, td=table data.
- **NOTA BENE:** Mario e Rossi sono i dati e sono contenuti dentro elementi di tipo <td>

```
<table>
  <tr>
    <td> Mario </td>
    <td> Rossi </td>
  </tr>
</table>
```

## Esempio di tabella

```
foreach($tabella as $c_riga => $v_riga)
{
    foreach($v_riga as $c => $v)
    {
        echo "<td>".$v."</td>";
    }
}
```

- Un array può contenere decine e decine di elementi, e il linguaggio PHP ci può aiutare costruendo automaticamente delle tabelle usando solo due foreach().
- Se il foreach() più interno percorre gli elementi più interni, come il nome e il cognome, si può usare <td> al posto di <p>.

## Esempio di tabella

```
foreach($tabella as $c_riga => $v_riga)
{
    echo "<tr>";
    foreach($v_riga as $c => $v)
    {
        echo "<td>".$v."</td>";
    }
    echo "</tr>";
}
```

- Le righe sono percorse dal foreach() più esterno, quindi <tr> deve essere usato proprio lì.

## Esempio di tabella

```
echo "<table>";
foreach($tabella as $c_riga => $v_riga)
{
    echo "<tr>";
    foreach($v_riga as $c => $v)
    {
        echo "<td>".$v."</td>";
    }
    echo "</tr>";
}
echo "</table>";
```

- Resta da aggiungere solo <table>. Poiché <table> non va ripetuto, va scritto fuori del foreach(), sopra e sotto.

## Esempio di tabella

- Usando il codice PHP appena visto, è possibile visualizzare un array di array, qualsiasi sia il suo numero di righe e qualsiasi sia il suo numero di colonne. Nel nostro esempio si vedrebbe la seguente tabella, dove mancano solo i titoli sopra le colonne.
- Dove si trovano i termini CF, NOME e COGNOME?

RSS	Mario	Rossi
BNC	Anna	Bianchi
VRD	Dino	Verdi

## Le chiavi

- Tutti gli elementi hanno le stesse chiavi CF, NOME e COGNOME, quindi è sufficiente prendere le chiavi del primo elemento.
- Per poter accedere al primo elemento, non si può usare `foreach()`.
- Se non si usa `foreach()` è necessario conoscere la sua chiave, che in questo caso sarebbe `'riga1'`

'riga1'	'cf'	'nome'	'cognome'
	"RSS"	"Mario"	"Rossi"

## Le chiavi

- Tutti gli elementi hanno le stesse chiavi CF, NOME e COGNOME, quindi è sufficiente prendere le chiavi del primo elemento.
- Per poter accedere al primo elemento, non si può usare `foreach()`.
- Se non si usa `foreach()` è necessario conoscere la sua chiave, che in questo caso sarebbe `'riga1'`

'riga1'	'cf'	'nome'	'cognome'
	"RSS"	"Mario"	"Rossi"

## Le chiavi

- `key()` è una funzione PHP che permette di accedere alla prima chiave di un array, quindi scrivendo:

```
key($var)
```

```
key($tabella)
```

- si ottiene `'riga1'`. Se si ottiene una chiave, si può mettere tra parentesi quadrate:

```
$tabella[key($tabella)]
```

- Questa scrittura corrisponde a

```
$tabella['riga1']
```

'riga1'	'cf'	'nome'	'cognome'
	"RSS"	"Mario"	"Rossi"

## Le chiavi



```
echo "<table>";
foreach($tabella as $c_riga => $v_riga)
{
    echo "<tr>";
    foreach($v_riga as $c => $v)
    {
        echo "<td>".$v."</td>";
    }
    echo "</tr>";
}
echo "</table>";
```

- La riga con le chiavi va aggiunta subito dopo <table>

## Le chiavi

```
echo "<table>";
echo "<tr>";
foreach($tabella[key($tabella)] as $c => $v)
{
    echo "<th>".$c."</th>";
}
echo "</tr>";
//continua...
```

## Copyright

Fabio Proietti (c) 2013

Licenza:

<http://creativecommons.org/licenses/by-sa/3.0/>

Immagini:

<http://openclipart.org>