



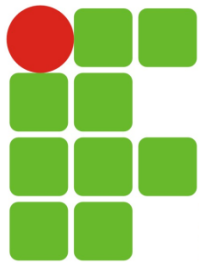
INSTITUTO FEDERAL
Farroupilha
Campus Avançado Uruguaiana

PHP

Arrays (Vetores)

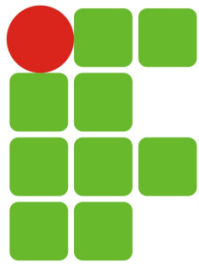
Prof. Leandro Martins Dallanora
leandro.dallanora@iffarroupilha.edu.br

Acesse o Manual do PHP Online em Português:
http://www.php.net/manual/pt_BR/



O que é um *array*?

- Arrays, também conhecidos como vetores, são variáveis que servem para guardar vários valores de forma uniforme na memória.
- Ver mais em:
 - http://php.net/manual/pt_BR/language.types.array.php



O que é um *array*?

- Enquanto uma variável simples pode armazenar um único valor por vez:

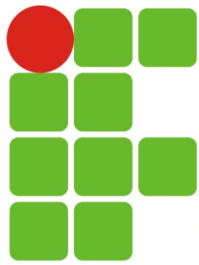
\$cor = “Vermelho”;

Vermelho

- Um array pode armazenar vários valores:

\$cor = array (“Verde”, “Vermelho”, “Azul”);

0	1	2
Verde	Vermelho	Azul

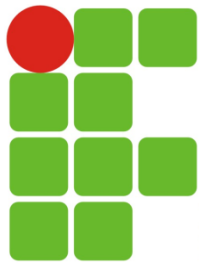


O que é um *array*?

- Cada item armazenado em um *array* é chamado de elemento do *array*, e é formado por chave e valor. No exemplo abaixo, temos um array com 3 elementos.

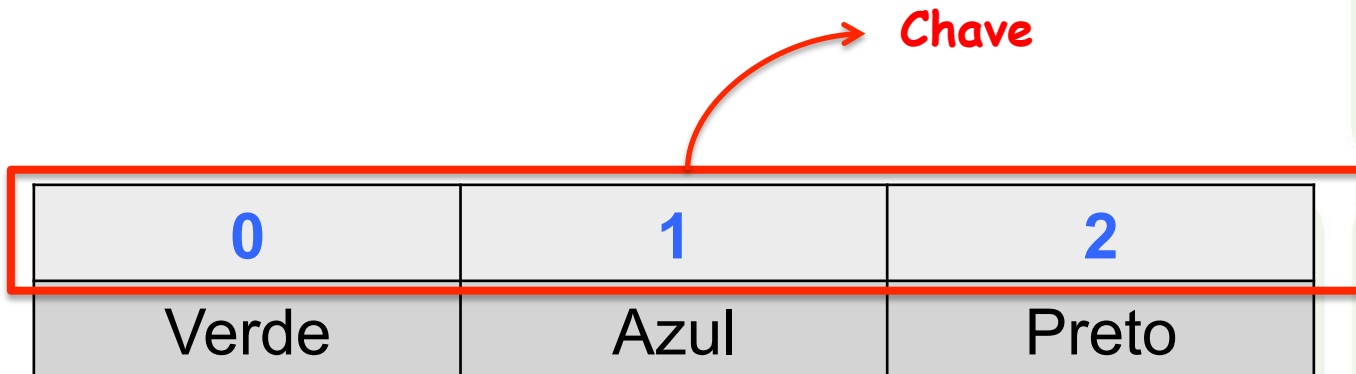
0	1	2
Verde	Azul	Preto

Elemento

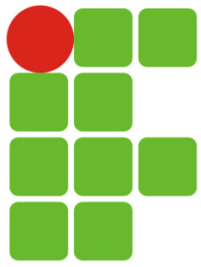


O que é um *array*?

- Cada elemento tem um índice associado (também chamado de chave) que é utilizado para acessar o elemento.



0	1	2
Verde	Azul	Preto

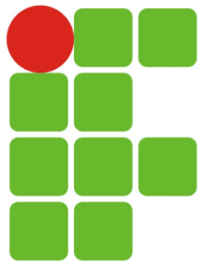


O que é um *array*?

- É um valor, que é o conteúdo do elemento. O valor que guardamos.

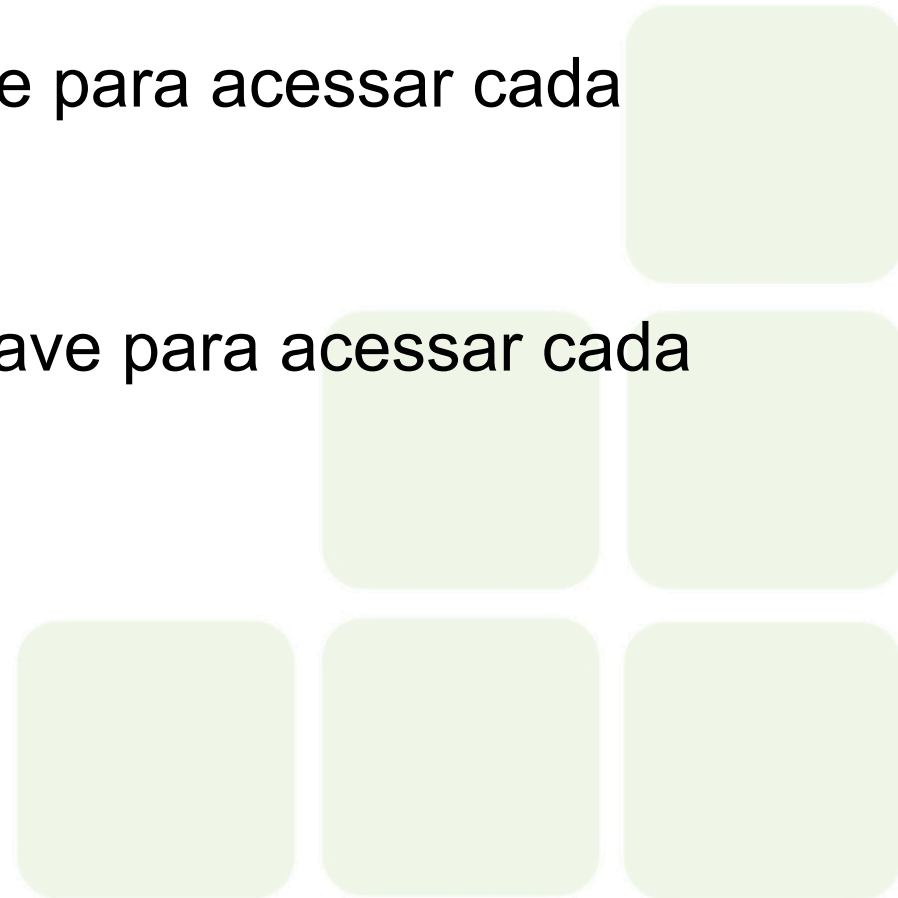
0	1	2
Verde	Azul	Preto

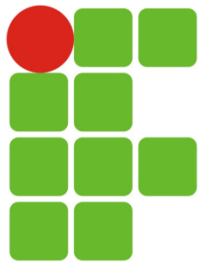
Valor



O que é um *array*?

- Podemos dividir os arrays em:
 - **Numéricos:** onde a chave para acessar cada elemento, é um número;
 - **Associativos:** onde a chave para acessar cada elemento, é uma palavra;





Arrays Numéricos

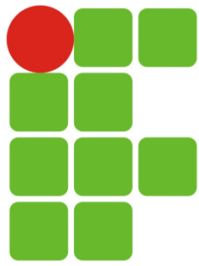
- Vamos **criar** um array que contenha nomes de cores.

```
$cores = array( “Verde”, “Azul”, “Preto” );
```

- Nosso array vai ficar assim:

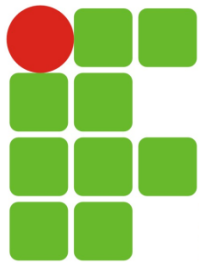
0	1	2
Verde	Azul	Preto

- Quando o índice é omitido, um índice numérico inteiro é automaticamente gerado, começando do 0 (zero).



Arrays Numéricos

- Para **acessar** um elemento de um *array*, utiliza-se o nome do *array* e uma chave.
- A chave indica qual valor está sendo acessado.
- A chave é colocada entre colchetes depois do nome.



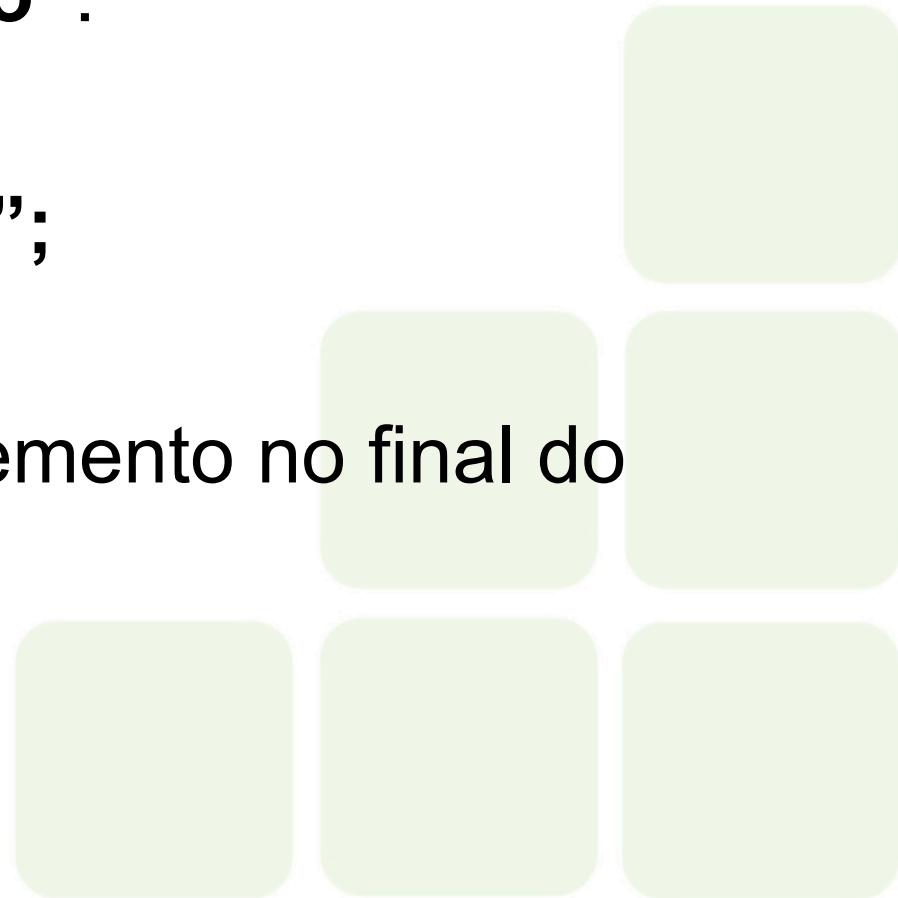
Arrays Numéricos

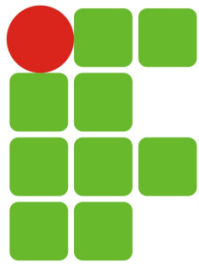
- Para **trocar** o primeiro elemento de “Verde” para “Vermelho”:

\$scores[0] = “Vermelho”;

- Para **acrescentar** um elemento no final do *array*:

\$scores[3] = “Branco”;





Arrays Numéricos

- Para **mostrar** o conteúdo do *array*:

```
echo $scores[0];
```

```
echo $scores[1];
```

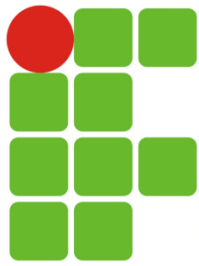
```
echo $scores[2];
```

```
echo $scores[3];
```

- Ou ainda, se quisermos **remover** um elemento:

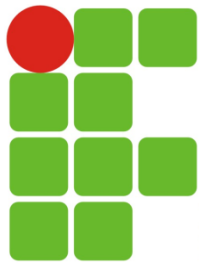
```
unset($scores[3]);
```

Isso irá remover o elemento com da posição 3



Arrays Numéricos

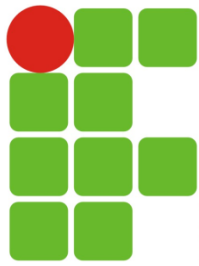
- Como com outras variáveis de PHP, os *arrays* não precisam ser inicializados ou criados antecipadamente.
- Os *arrays* são criados de forma automática na primeira vez que você os utiliza.
- Exemplo:
 - **`$scores[0] = “Verde”;`**
 - **`$scores[1] = “Azul”;`**
 - **`$scores[2] = “Preto”;`**



Arrays Numéricos

- Como o *array* está indexado por uma sequência de números, podemos utilizar o comando de repetição **for** para exibir seu conteúdo:
- Exemplo:

```
for ( $i = 0; $i <= 3; $i++ ){  
    echo "$scores[ $i ] <br>";  
}
```

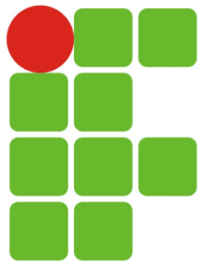


Arrays Numéricos

- Também podemos usar o laço **foreach**, especialmente projetado para usar com os *arrays*.
- Exemplo:

```
foreach ( $scores as $valor )  
    echo “$valor <br>”;
```

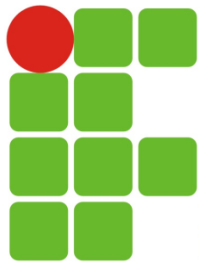
- Assim as chaves do *array* não precisam iniciar em zero e podem estar fora de ordem.



Arrays Associativos

- O código a seguir cria um *array associativo* com a relação parental como chaves e os nomes de cada familiar como valores.

```
$familia = array("Pai"=>"Peter",  
"Mãe"=>"Lois",  
"Filho"=>"Stewie",  
"Filha"=>"Meg");
```

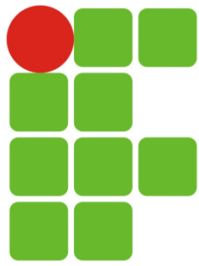


Arrays Associativos

- O array **\$familia**, ficará assim:

The diagram illustrates an associative array structure. A red box highlights a table with two rows. The first row contains the keys: **Pai**, **Mãe**, **Filho**, and **Filha**. The second row contains the corresponding values: **Peter**, **Lois**, **Stewie**, and **Meg**. A red arrow labeled **Chave** points to the first row, and another red arrow labeled **Valor** points to the second row.

Pai	Mãe	Filho	Filha
Peter	Lois	Stewie	Meg



Arrays Associativos

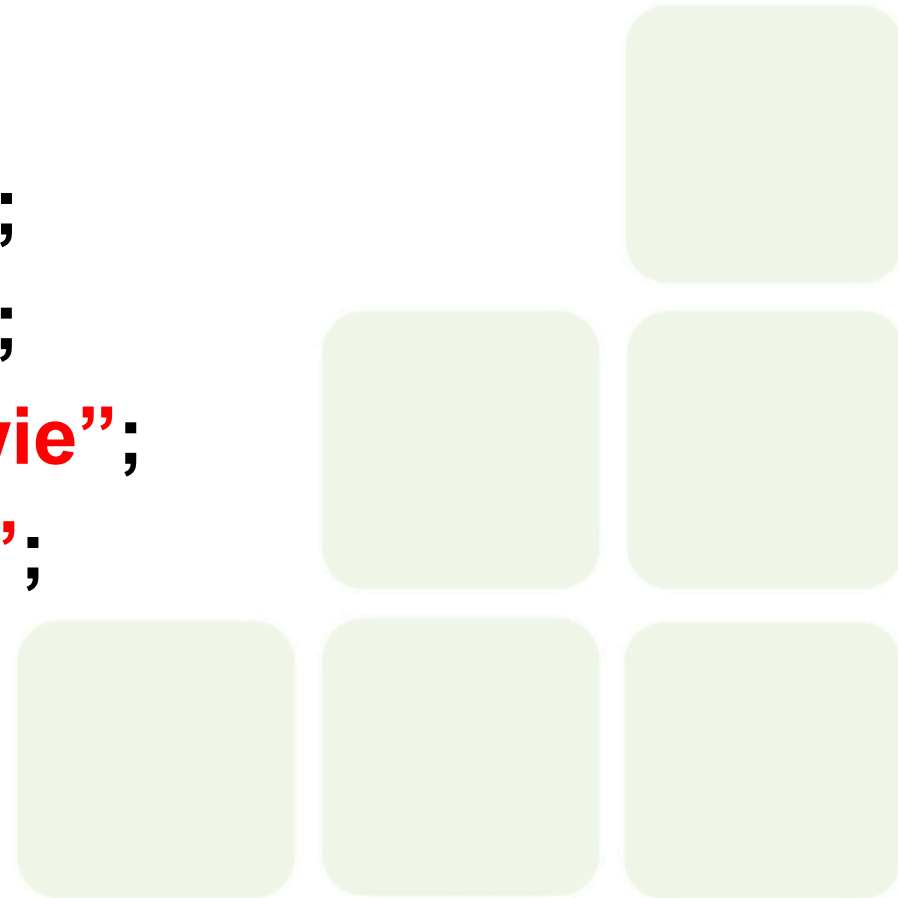
- Esse mesmo *array* pode ser criado da seguinte maneira:

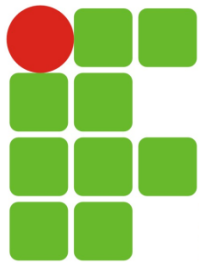
```
$familia[“Pai”] = “Peter”;
```

```
$familia[“Mãe”] = “Lois”;
```

```
$familia[“Filho”] = “Stewie”;
```

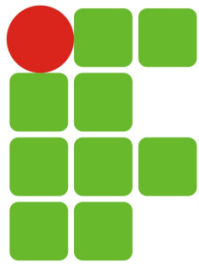
```
$familia[“Filha”] = “Meg”;
```





Arrays Associativos

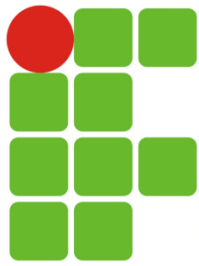
- Para acessar um elemento do array, utilizamos o nome da variável e a chave entre colchetes.
- Como os índices no *array* não são números, não podemos utilizar um contador em um **for** para percorrer os elementos do *array*.
- Mas podemos utilizar o comando **foreach**.



Arrays Associativos

- O comando **foreach** pode ser usado para determinar o valor de cada elemento do *array*, como no exemplo abaixo:

```
foreach ( $familia as $nome ){  
    echo "$nome <br>";  
}
```



Arrays Associativos

- O comando **foreach** também pode ser usado para determinar a chave e o valor de cada elemento do *array*.
- Mais comum em arrays associativos.

```
foreach( $familia as $familiar => $nome ){  
    echo "$familiar - $nome <br>";  
}
```