

# CURSO DE ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Christopher Steudel

Gustavo Do Espirito Santo

Vinicius Baggio

Vitor César

**VETORES E FUNÇÕES EM PHP**

**VETORES**

**Vetores definição:**

Vetor ou *array* é um conjunto de dados que pode assumir os mais diversos tipos de dados. Vetores são utilizados para armazenar mais de um valor em uma única variável, isso é comparável a uma variável que pode armazenar apenas um valor. Cada item do vetor tem um número ligado a ele, chamado de índice numérico, que permite acesso a cada valor armazenado na variável.

**Vetor Unidimensional:**

Um vetor unidimensional é uma estrutura de dados que armazena uma coleção de elementos do mesmo tipo em uma única dimensão. Este tipo de estrutura de dados funciona como se fosse uma lista de itens que deseja armazenar e cada um destes itens podem ser acessado através de um índice número que começa em 0.

Características principais:

* Estrutura homogênea: Todos os elementos do vetor devem possuir o mesmo tipo de dado (por exemplo, todos inteiros, todos strings, etc.).
* Acesso por índice: Cada elemento é identificado por sua posição (índice) dentro do vetor. O primeiro elemento tem o índice 0, o segundo tem o índice 1, e assim por diante.
* Armazenamento sequencial: Os elementos são armazenados em posições de memória contiguas, o que permite o acesso rápido a qualquer um deles.

Como criar:

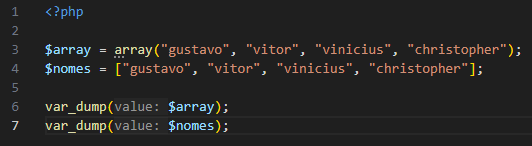


Figura 1 - Fonte: elaborada pelo autor

Exemplo do output:

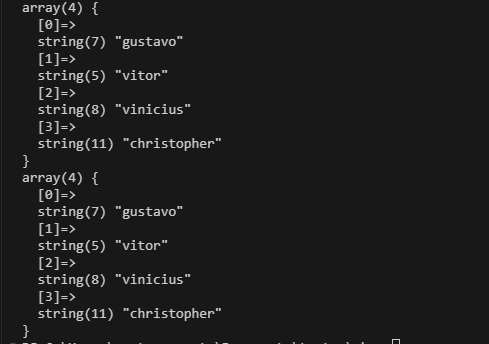


Figura 2 - Fonte: elaborada pelo autor

Para percorrer esses vetores e acessar seus índices, podemos fazer de duas maneiras diferentes:

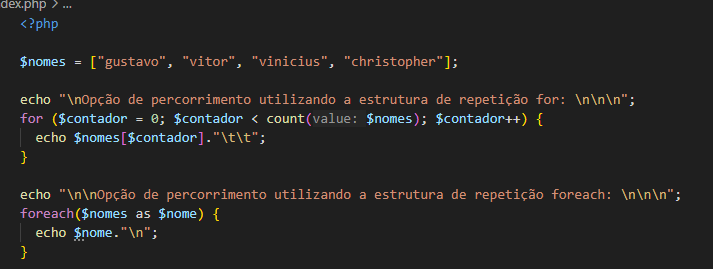


Figura 3 - Fonte: elaborada pelo autor

Exemplo do output:

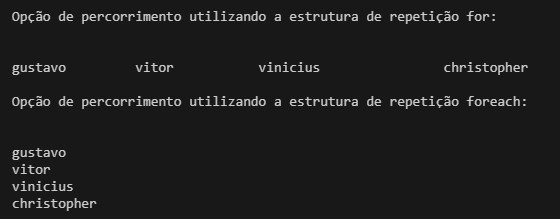


Figura 4 - Fonte: elaborada pelo autor

**Vetor Multidimensional:**

Um vetor multidimensional é uma estrutura de dados que permite armazenar um conjunto de elementos do mesmo tipo em mais de uma dimensão, como uma tabela bidimensional ou um cubo tridimensional. Cada elemento é acessado através de múltiplos índices, sendo um para cada dimensão.

Caracaterísticas principais:

* Múltiplas dimensões: Em vez de uma única linha de elementos, vetores multidimensionais têm linhas e colunas.
* Estrutura de vetores: Um array bidimensional, por exemplo, é essencialmente um “vetor de vetores”.
* Acesso por índices: Para acessar um elemento específico, você fornece um índice para cada dimensão (por exemplo, matriz[linha][coluna]).

Como criar:

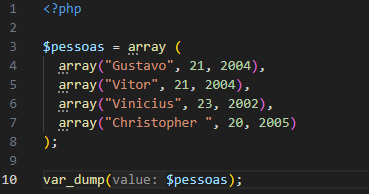
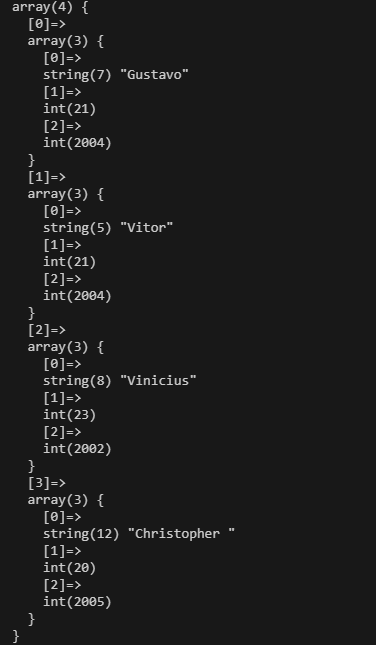


Figura 5 - Fonte: elaborada pelo autor

Exemplo output:



Como percorrer este vetor multidimensional?

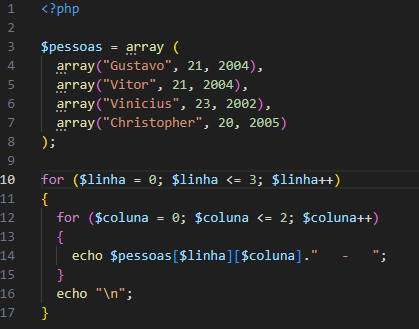


Figura 6 - Fonte: elaborada pelo autor

Exemplo do output:

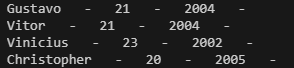


Figura 7 - Fonte: elaborada pelo autor

**Funções para manipulação de vetores:**

No PHP existem várias funções natívas para manipulação de vetores, aqui estão alguns exemplos de funções que podem ser úteis no dia a dia:

* count: Conta o número de elementos.
* array\_push: Insere um novo elemento dentro do vetor na última possições + 1.
* array\_merge: Faz a união de dois vetores em um único só.
* array\_pop: Remove e retorna o último elemento.
* array\_unshift: Adiciona um ou mais elementos no início do vetor.
* array\_splice: Remove uma porção de um vetor e a substitui por algo mais.
* array\_pad: Preenche um vetor com um elemento até um tamanho especificado.

**FUNÇÕES**

**Definição:**

Uma função é um bloco de código reutilizável que executa uma tarefa específica. Ao definir uma função, você a nomeiae agrupa as instruções dentro dela. Sempre que a tarefa precisar ser realizada, o programa pode “chamar” essa função pelo nome, evitando a repetição do mesmo código. Funções também podem receber dados e retornar um resultado.

A principal importância da utilização de funções é evitar a replicação de código, reduzindo assim a sua redundância.

Estrutura básia:

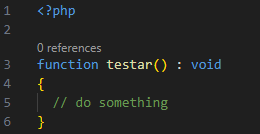


Figura 8 - Fonte: elaborada pelo autor

**Parâmetros e retorno de valores:**

Os parâmetros são variáveis declaradas entre parênteses no cabeçalho de uma função, que recebem valores quando a função é chamada.

Exemplo:

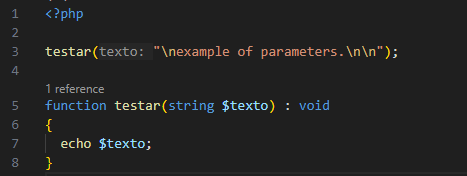


Figura 9 - Fonte: elaborada pelo autor

Output:



Figura 10 - Fonte: elaborada pelo autor

Uma função pode retornar algum elemento ou não. As funções que não retornam nenhum elemento chamos de “*void”* (vazia), e as funções que retornam elementos nós adicionamos o tipo de retorno.

Exemplo:

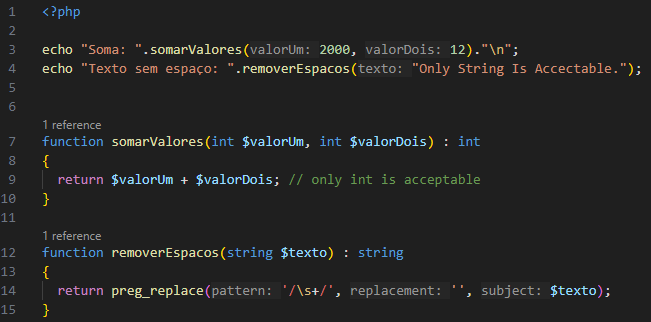


Figura 11 - Fonte: elaborada pelo autor

Output:



Figura 12 - Fonte: elaborada pelo autor

**Funções internas ou nativas:**

São as funções já prontas que vêm com a linguagem de programação. Elas fazem parte da biblioteca padrão e podem ser usadas sem precisar criar nada.

Vantagens de utilizar funções nativas:

* Já são otimizadas e testadas.
* Economizam tempo de desenvolvimento.
* Evitam a necessidade de reinventar funções básicas.

**Funções definidas pelo usuário:**

Para realizar tarefas específicas que não existem prontas na linguagem e/ou definir a sua própria regra.

Vantagens de utilizar funções definidas:

* Permitem organizar melhor o código.
* Evitam repetição (reutilização lógica).
* Facilitam a manutennção e leitura do programa.

**Boas práticas:**

É importante que as funções possuam nomes significativos e descritivos, indicando claramente ação que realizam. Além disso, recomenda-se que cada função seja curta e objetiva, executando apenas uma tarefa específica, o que facilita a leitura e os testes do programa. Outro ponto fundamental é evitar a repetição de código, utilizando funções para centralizar trechos que se repetem em diferentes partes do sistema. A adoção de boas práticas contribui para um desenvolvimento mais eficiente, legível e sustentável, promovendo a qualidade e a manutenção do código a lingo prazo.

REFERÊNCIAS

< https://www.php.net/manual/en/language.types.array.php>

Acessado em 16 out. 2025.

< https://www.w3schools.com/php/php\_arrays\_multidimensional.asp>

Acessado em 16 out. 2025.

<https://www.php.net/manual/pt\_BR/functions.user-defined.php>

Acessado em 20 out. 2025.

<https://www.ionos.com/pt-br/digitalguide/sites-de-internet/desenvolvimento-web/function-php/>

Acessado em 20 out. 2025.