AULA 04

Como criar variáveis em PHP? Como declarar variáveis em PHP? Quais são os tipos primitivos do PHP? Como realizar conversões em PHP? A resposta para essas e muitas outras perguntas você vai ter nessa quarta aula do Curso Grátis de PHP para Iniciantes.

Antes de mais nada, é importante que você entenda os conceitos de variáveis em Algoritmos e saiba utilizá-las corretamente. Nós temos uma aula no Curso Grátis de Algoritmos que fala sobre variáveis, assista esse vídeo antes de prosseguir com essa aula, caso você não esteja familiarizado com esse conceito.

Comandos de saída do PHP:

Os comandos ECHO, PRINT e PRINTF servem para gerar saídas na tela. Um exemplo simples desse comando é:

echo “Estou aprendendo variáveis em PHP”;

A pronúncia correta do comando echo é ECO (aquele som que se repete quando gritamos dentro de cavernas, lembra?).

Note que no comando acima, foi utilizada uma tag HTML para a quebra de linha. Isso é possível, já que o PHP se integra com essa tecnologia. Fique à vontade para utilizar tags de marcação hipertexto dentro dos seus comandos de saída em PHP.

Usando variáveis

Variáveis são espaços na memória do computador que podem conter valores. Variáveis simples armazenam apenas um valor de cada vez, dependendo do seu tipo.

Todas as variáveis no PHP possuem um $ na frente e seguem as mesmas regras de construção de nomes de identificadores:

– Devem começar com uma letra (após o $)

– Podem conter letras e números

– Não podem conter caracteres acentuados

– Não podem conter símbolos como % # \* & etc (exceto \_ e $)

OBS: O PHP faz a diferenciação entre as letras maiúsculas e minúsculas, por isso recomendamos que você utilize apenas identificadores com letras minúsculas em seus scripts PHP.

Declaração de variáveis

No PHP não existe a necessidade de declarar variáveis. Os tipos serão atribuídos automaticamente de acordo com o valor que a variável receber. Esse processo é conhecido como coerção. Isso pode ser até empolgante no início, mas é preciso prestar bastante atenção para não fazer nada errado.

Você pode forçar um tipo primitivo a uma variável por TYPECAST, utilizando (int) (real) (float) (double) (string) antes do valor na atribuição.

Não existe typecast para variáveis lógicas. Os valores booleanos são considerados números, sendo o valor 1 atribuído para true e vazio para false.

Atribuição de valores

Atribuir valor é colocar um conteúdo em uma variável. Utilizamos o operador de atribuição = para realizar essa tarefa. Veja alguns exemplos:

$idade = 15;

$salario = 1500.25;

$nome = “Gustavo”;

$casado = true;

Concatenação de valores

Concatenar é juntar valores para tratá-los em conjunto. A concatenação em PHP é realizada pelo operador ponto (.) como no trecho de código a seguir.

$idade = 18;

$nome = “Maria”;

echo $nome . “ tem “ . $idade . “ anos “;

No script acima, será exibido “Maria tem 18 anos”, já que acontecerão concatenações.

Porém, já que os identificadores PHP começam com um $, é possível realizar o comando de saída utilizando um modo simplificado:

echo “$nome tem $idade anos”; O comando acima pode ser usado sem problemas, contanto que sejam utilizadas aspas duplas.

Aula 5

Como fazer contas no PHP? Como realizar somas, multiplicações e mais? Exponenciações em PHP? Raiz quadrada em PHP? Os operadores aritméticos do PHP são:

+ é o operador de adição

– é o operador de subtração

\* é o operador de multiplicaçã

o / é o operador de divisão Real

% é o operador de módulo (resto da divisão)

O código a seguir, vai somar dois números:

$n1 = 3;

$n2 = 2;

$s = $n1 + $n2;

echo “A soma entre $n1 e $n2 é igual a $s”;

O código acima vai mostrar na tela a mensagem

A soma entre 3 e 2 é igual a 5

Outra coisa importante a saber é a ordem de precedência de operadores aritméticos em PHP. Sempre em uma expressão, os operadores que serão executados são, na ordem:

Em primeiro lugar, parênteses ( )

Em segundo lugar, operadores de multiplicação, divisão e módulo \* / %

Em seguida, as adições e subtrações + –

Por exemplo, considerando o código a seguir:

$media = $nota1 + $nota2 / 2;

A média será calculada de maneira errada, já que segundo a ordem de precedência, a divisão será feita antes. O correto seria escrever:

$media = ($nota1 + $nota2) / 2;

Note que o uso dos parênteses muda bastante as coisas.

Obtendo valores da URL com PHP

Vamos analisar a URL abaixo:

http://localhost/aula05/operadores.php?a=3&b=2

No link acima, o arquivo operadores.php está sendo chamado dentro da pasta aula05 do servidor local. Na linha, serão passados dois valores: a valendo 3 e b valendo 2.

Para pegar esses valores no script PHP, use o código

$valor1 = $\_GET[“a”];

$valor2 = $\_GET[“b”];

No código acima, o parâmetro a (3) passado pela URL será armazenado na variável $valor1. De maneira similar, a variável $valor2 vai conter o valor do parâmetro b (2).

Obs: use sempre $\_GET com todas as letras maiúsculas.

Funções Matemáticas em PHP abs() : Retorna o valor absoluto de um número.

Ex: abs(-5) = 5

pow() : Calcula uma potência

. Ex: pow(3,2) = 32 = 9

sqrt() : Calcula a raiz quadrada de um número.

Ex: sqrt(25) = 5

round() : Arredonda valores.

Ex: round(3.8) = 4

intval() : Trunca um número.

Retorna a parte inteira de um valor.

Ex: abs(8.7) = 8

number\_format() : Formata um número Real.

Ex: number\_format(3258.754, 2, “,”, “.”) = 3.258. 75.

Obs: Ainda existem os métodos ceil() e floor() para arredondamentos para cima e para baixo, respectivamente. A função round() vai usar as regras de arredondamento.

AULA 6

Uma atribuição acontece quando queremos colocar algum valor dentro de uma variável, seja ele um número ou string estática, o resultado de uma expressão, o retorno de uma função ou o conteúdo de outra variável.

== Operadores de Atribuição do PHP

Os operadores de atribuição do PHP são += -= \*= /= %= e .=

Vejamos alguns exemplos:

$c = $c + 5; $c += 5;

$c = $c – $a; $c -= $a;

$c = $c + 1; $c += 1;

Na última linha da tabela acima, você verifica a adição de apenas uma unidade na variável. Nesses casos, podemos usar os operadores de incremento.

== Operadores de Incremento ou Decremento

$c = $c + 1; $c += 1; $c++;

Qualquer uma das formas acima é válida. De maneira similar, temos:

$c = $c – 1; $c -= 1; $c–;

A forma de utilizar o operador de incremento/ decremento faz toda diferença se ele aparece antes ou depois da variável. Assim:

$c ++ $c–

vão apresentar resultados diferentes de acordo com a situação. Durante a aula, vai ser explicado melhor como utilizar pré-incremento, pós-incremento, pré-decremento e pós-decremento em PHP.

 == Comentários em PHP

Existem três tipos de comentários em PHP. Os comentários inline // e # transformam tudo o que está após o(s) símbolo(s) será considerado comentário.

Existe também o comentário multiline, que vai criar comentários que ocupem várias linhas:

<?php

      /\* Esse comentário vai ocupar várias linhas do seu código e todas serão ignoradas \*/

     $a += 1; // Esse é um comentário de uma linha

     $b ++; # Esse aqui também é

?>

== Variáveis Referenciadas

Colocar um caractere & na frente de uma variável vai criar um ponteiro em PHP. Ela não será uma variável de fato, mas será uma referência à variável original. Considere o código:

<?php

     $x = 3;

     $y = $x;

     $z = &$x;

?>

A variável $x vai receber 3. A variável $y vai receber o valor que está dentro da variável $x. A variável $z vai ser uma ligação com a variável $x. Mais tarde, qualquer modificação em $y não vai alterar o valor de $x. Porém, se mudarmos o valor de $z, o valor de $x será afetado, já que existe uma relação entre as duas.

== Variáveis de Variáveis (variáveis variantes)

Colocar um segundo $ na frente de uma variável também possui um efeito bem peculiar. Ele vai criar uma variável dinamicamente, dependendo do conteúdo da variável original. Essas são as variáveis de variáveis em PHP.

<?php

    $nome = “gustavo”;

    $$nome = “professor”;

?>

Com essas linhas, teremos uma variável $nome, como o conteúdo “gustavo” e a linha de baixo vai criar uma variável $gustavo, com o conteúdo “professor”.

Veja a aula em vídeo e tire todas as suas dúvidas sobre esses conteúdos sobre atribuição de valores em PHP.

AULA 08

Nessa oitava aula do Curso de PHP, vamos aprender como aumentar a interatividade dos nossos scripts PHP com formulários HTML5.

Formulários em HTML5

Se você não sabe como lidar com formulários HTML5, não se esqueça de assistir antes as duas aulas que tratam desse assunto no Curso em Vídeo Grátis de HTML5.

Formulários HTML5 – Parte 1: [http://youtu.be/metoFY-x\_yg](https://www.youtube.com/watch?v=metoFY-x_yg)

Formulários HTML5 – Parte 2: [http://youtu.be/lwuDJN9Udfc](https://www.youtube.com/watch?v=lwuDJN9Udfc)

Formulários HTML5 e JavaScript: [http://youtu.be/YY5hqlcbfoI](https://www.youtube.com/watch?v=YY5hqlcbfoI)

Lá você vai aprender como escrever códigos específicos para a criação de formulários completos e vai estudar todos os novos controles de formulários adicionados ao HTML5, como o range, color, number, date, etc.

Por exemplo, vamos considerar um formulário que use o método GET e envie um valor V para o arquivo DADOS.PHP:

Obs: O método GET envia dados de um formulário diretamente pela URL.

Interligando o formulário HTML com o script PHP

Para interligar o formulário, vamos usar a cláusula $\_GET

<?php

     $valor = $\_GET[“v”];

     echo “Digitou $valor”;

?>

Obs: O $\_GET sempre vai utilizar todas as letras maiúsculas. Obs: Se por acaso seu formulário utilizar o método post, você vai precisar usar $\_POST.

Interligando outros tipos de controles de formulários HTML5 com PHP

A integração de qualquer controle de formulário PHP pode ser feita através do parâmetro NAME de cada um deles. Durante a aula, criamos um formulário com vários tipos de controles e interligamos eles ao arquivo PHP.

Pegando o ano atual com PHP Para obter o ano atual no PHP, utilizamos

$idade = date(“Y”) – $ano;

AULA 09

A estrutura condicional em PHP é representada da seguinte forma:

if ($idade = 18) {

     $vota = true;

} else {

    $vota = false;

}

Estruturas condicionais aninhadas

Quando colocamos uma condicional dentro da outra, dizemos que estamos aninhando estruturas (termo que se refere a ninho). Para aninhar blocos, utilizamos uma sintaxe semelhante à anterior:

if ($peso 50) {

    $tipo = “muito magro”;

} else {

    if ($peso =50 && $peso 70) {

          $tipo = “peso normal”;

     } else {

        $tipo = “acima do peso”;

     }

}

Simplificando estruturas condicionais aninhadas

No PHP, podemos substituir uma cláusula else seguida de outro if (como feito acima) por uma estrutura elseif {} que vai se comportar exatamente da mesma maneira, mas usará menos blocos.

if ($tipo == “nacional”) {

     $imposto = 0;

} elseif ($tipo == “importado”) {

     $imposto = 60;

} elseif ($tipo == “mercosul”) {

     $imposto = 20;

} else {

     $imposto = 80;

}

Na próxima aula veremos como criar estruturas condicionais com múltiplos valores utilizando o comando switch. Até lá!

AULA 11

Vamos agora começar as Estruturas de Repetição em PHP, partindo da estrutura WHILE (enquanto).

A Estrutura While (enquanto), também conhecida como Estrutura de Repetição com Teste Lógico no início, realiza o teste de uma expressão lógica sempre na primeira linha da estrutura. Vamos ver como realizar uma contagem progressiva de 1 até 10, utilizando a estrutura while em PHP.

$c = 1;

while ($c = 10) {

    echo $c; $c++;

}

Durante a aula, veremos também como realizar outras contagens, como por exemplo a contagem regressiva.

Outra coisa que vamos aprender durante essa aula de repetição é criar código HTML dinamicamente, utilizando scripts PHP. No exemplo apresentado, vamos criar várias caixas de texto com nomes e etiquetas diferentes usando uma estrutura de repetição em PHP.

AULA 16

Nessa aula, veremos uma lista de funções para Strings usando PHP. São funções internas que já existem na linguagem. A lista de funções de manipulação de Strings que serão vistas nessa aula é composta pelas instruções:

Função printf() : Permite exibir uma string com itens formatados.

Função print\_r() : Exibe coleções, objetos e variáveis compostas (vetores e matrizes) de maneira organizada.

Função wordwrap() : Cria quebras de linha ou divisões em uma string em um tamanho especificado.

Função strlen() : Permite verificar o tamanho de uma string, contando seus caracteres (inclusive espaços em branco).

Função trim() : Elimina espaços em branco antes e depois de uma string.

Função ltrim() : Elimina espaços no início de uma string.

Função rtrim() : Elimina espaços em branco no final de uma string.

Função str\_word\_count() : Conta quantas palavras uma string possui.

Função explode() : Quebra uma string e coloca os itens em um vetor.

Função str\_split() : Coloca cada letra de uma string em uma posição de um vetor.

Função implode() : Transforma um vetor inteiro em uma string.

Função chr() : Retorna um caractere de acordo com seu código ASCII passado como parâmetro.

Função ord() : Retorna o código ASCII de um caractere passado como parâmetro.

[**Aula Anterior**](https://www.cursoemvideo.com/curso/php-basico/aulas/php/modulos/estrutura-condicional-switch/)