Trabalho de férias

Enzo Gabriel Picoli Domiciano

**Aula 1- Começa aqui seu curso de PHP Moderno - ‪@CursoemVideo‬ de PHP - Gustavo Guanabara**

Ele dá a introdução ao curso introduzindo tópicos:

* Falar e explicar sobre o PHP pós 2020
* Apresentar a linguagem
* Mostar como ela funciona
* Instalar as ferramentas
* Testar configurações
* Estruturar o script
* Saida de dados
* Tipos de entrada
* Interação com formulários
* Operadores

Aula 2- Esse curso de PHP serve pra mim? - @CursoemVideo de PHP - Gustavo Guanabara

Esse curso não é para quem está com pressa ou para quem quer apenas copiar código, mas sim para quem quer aprender

**Aula 3- Lista TOP 6 livros de PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Nesta aula ele vai recomendar e apresentar algumas bibliografias

* Documentação oficial do PHP (php.net)
* Aprendendo PHP (livro)
* PHP moderno (livro) esse livro é para quem já está mais avançado e está migrando para uma versão moderna
* Learn PHP 8 (livro)
* PHP 8 Quick scripting reference (livro)
* PHP & MYSQL (livro)

**Aula 4- A evolução do PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Nesse vídeo ele vai nos mostrar a evolução do PHP

O PHP surgiu em 1995, criado por Rasmus Lerdorf, e tinha apenas funcionalidades básicas

Em 1997 ele fez e divulgou sua primeira atualização e foi oficialmente nomeado como PHP/FI 2.0, era uma linguagem standalone e tinha recursos bem limitados

Em 1998 foi a primeira versão colaborativa e carregava características mais relacionadas a uma linguagem

**Aula 5 - Por que um elefante é o mascote do PHP? - @CursoemVideo de PHP - Gustavo Guanabara**

Ele explica o porquê do elefante na logo do PHP, isso se deve ao artista Vicente Pontier, pois o próprio PHP parece com elefante, daí veio a ideia do elefante na logo.

**Aula 6- As versões do PHP e seus recursos - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula ele dá continuidade as versões do PHP da aula **Aula 4.**

Em 2000 foi lançado o PHP 4.0 e ele começou a funcionar na Zend Engine 1.0, teve também uma melhoria na performasse, um aumento na modularização, foi nessa versão também que foram implementados os super globais $\_GET, $\_POST, $\_SESSION etc.

Em 2004 PHP 5.0 foi a versão que mais durou e teve mais subversões, foi a que mais deu problema, era conhecida como uma linguagem ruim e de preguiçosos.

Usava a Zend Engine 2.0

Essa foi a primeira versão com orientação a objeto

Operador de esponenciação

Suporte a JSON (Serve para transferência rápida de dados sem muita formatação) que significa javascript object notation

Novo depurador PHPDBG

**Aula 7- O PHP vai morrer? Vale a pena estudar PHP? - @CursoemVideo de PHP - Gustavo Guanabara**

Na aula 7, o Guanabara fala sobre a linguagem PHP estar morrendo. Ele nos mostra alguns dados de gráficos que pessoas mostram para embasar a teoria de que o PHP está morrendo, mas ele diz “O PHP está morrendo assim como eu e você” mas isso não será hoje ou amanhã, mas daqui muito tempo. Ele nos mostra que os graficos usados por essas pessoas são do PHP 5.0 e essa versão do PHP realmente tem que morrer mesmo.

**Aula 8- Como funciona o PHP? - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula ele vai nos ensinar como o PHP funciona.

Ele nos da o seguinte exemplo:

Client-Side

O cliente faz uma pesquisa exemplo youtube.com o dispositivo que está sendo usada para fazer essa pesquisa, um celular ou computador vai fazer uma requisição, mas não diretamente para o servidor, pois no servidor não está salvo como youtube.com, mas com um número o IP, então temos que comunicar com um servidor DNS para que ele faça a busca desse endereço IP para que possa devolver essa informação a internet para que o cliente possa então fazer o contato com servidor e o servidor fazer uma cópia dos arquivos e enviar para o cliente.

Server-Side

No Server-Side é um pouco diferente, se no Client-Side o processamento de informações é no dispositivo do cliente no Server-Side quem terá o trabalho de processar essas informações será o servidor, mas temos uma diferença pois no Server-Side temos arquivos que não podem ser enviados para o cliente, esses arquivos que não podem ser enviados (os scripts) precisaram de interpretador e o PHP entra nessa função, então o servidor faz um processamento, após esse processamento são gerados arquivos em HTML e CSS, e uma cópia deles é enviado para o cliente.

O profissional que trabalha com Client-Side é um desenvolvedor de Front-End, já quem trabalha na área do Server-Side é desenvolvedor de Back-End, e se você trabalha com os dois é que chamamos de desenvolvedor FullStack

**Aula 9- É possível criar qualquer coisa com PHP? - @CursoemVideo de PHP - Gustavo Guanabara**

Na aula 9 Guanabara ele começa montando uma linha de raciocino:

Ele utiliza uma furadeira como exemplo, ela é perfeita para parafusos ou para fazer um furo, mas ela tem o mesmo desempenho para lixar por exemplo ou pregar um prego, da mesma forma o PHP, para criar um site ou um blog PHP é perfeito, para criar uma loja online para vender produtos perfeito,assim como a furadeira o PHP tem a sua area de atuação, dá para criar jogo com PHP sim, mas não terá um resultado tão bom quanto ferramentas que são para esse tipo de situação

“A linguagem é a ferramenta, mas nem toda ferramenta serve para tudo”.

**Aula 10- Como transformar seu computador em servidor PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula ele nos fala do precisaremos para conseguir começar desenvolver em PHP, o XAMPP que funciona no LAMP (Linux), WAMP (Windows) e MAMP

**Aula 11- É possível treinar PHP no smartphone - @CursoemVideo de PHP - Gustavo Guanabara**

Na aula 11 ele vai ensinar a treinar o PHP no celular.

**Aula 12- Seu servidor PHP no Windows - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula ele vai ensinar a instalar as ferramentas para Windows.

**Aula 13- Seu servidor PHP no Linux - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula ele vai ensinar a instalar as ferramentas para Linux.

**Aula 14- Seu servidor PHP no MacOS - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula ele vai ensinar a instalar as ferramentas para MacsOS.

**Aula 15- Configurações importantes para VS Code - @CursoemVideo de PHP - Gustavo Guanabara**

Nesta aula nós vamos criar a pasta onde faremos os exercícios. Logo em logo em seguida ele nos mostra como acessar essa pasta pelo Visual Studio Code. E colocar configurações importantes para mexermos com PHP

**Aula 16- Melhores extensões para VS Code - @CursoemVideo de PHP - Gustavo Guanabara**

**Aula 17- Primeiro programa em PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula fizemos o clássico “Olá Mundo” em PHP.

<?php

echo "Olá, Mundo!";

?>

**Aula 18- O PHP por dentro - @CursoemVideo de PHP - Gustavo Guanabara**

Nesse vídeo Guanabara vai nos explicar o porquê de uma única linha de PHP pode ser convertida em quase mil linhas de HTML.

Gustavo explica que em um código PHP tem várias linhas de programação, mas o que será enviado para o “cliente” será apenas a versão já processada, e para que tenha 1 linha em PHP e tenha quase mil linhas de html é porque por trás desta única linha tem várias outras linhas trabalhando para fazer aquela função.

Ele nos mostra que nas versões mais antigas dos PHP, para usar o código em PHP era da mesma forma do JS

<script language=”PHP”>

</script>

Hoje nas versões mais modernas se usa a “super tag”

<?php echo “Enzo Gabriel” ?>

Quando nós temos apenas um comando dentro da super tag, que seja echo ou print, nós podemos fazer da seguinte forma:

<?=“Enzo Gabriel” ?>

E essa simplificação se chama short tag PHP

Podemos usar também uma versão reduzida que se chama “short open tag”

<? ?>

**Aula 19- Como configurar o php.ini - @CursoemVideo de PHP - Gustavo Guanabara**

Na aula 19 nós vamos configurar o PHP, porque por exemplo, no MAC não estava sendo mostrado em que linha o erro estava. E por isso vamos configurar da maneira adequada.

**Aula 20- Variáveis e constantes em PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Nessa aula vamos ver sobre Variável e Constante

Variável obviamente é algo que varia, já a constante é algo que vai permanecer da mesma forma até o final da execução

Ele nos dá um exemplo prático para mostrar e exemplificar melhor como a variável pode mudar ao longo do código:

<?php

$nome = “Enzo”;

$sobrenome = “Gabriel”;

echo “Muito prazer, $nome $sobrenome!”;

?>

Para mostrar que a variável pode mudar ele faz uma pequena mudança:

<?php

$nome = “Enzo”;

$sobrenome = “Gabriel”;

//

//

//

$nome = “Luiz”;

$sobrenome = “Fernado”;

echo “Muito prazer, $nome $sobrenome!”;

?>

Dessa forma o nome que será imprimido na execução será mudado de “Muito prazer, Enzo Gabriel” para “Muito prazer, Luiz Fernado”.

Logo adiante ele cria uma constante:

<?php

$nome = “Enzo”;

$sobrenome = “Gabriel”;

Const PAIS = “Brasil”;

echo “Muito prazer, $nome $sobrenome! Você mora no” . PAIS;

?>

Muito prazer, Enzo Gabriel! Você mora no Brasil

Dessa forma ela nos mostra como atribuir uma constante. Logo depois ele nos mostra que não é possível mudar o valor da constante, para exemplificar ele faz a mesma coisa que fez com a variável para mostras que não da certo.

**Aula 21- Tipos primitivos do PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Na aula 21 o Gustavo vai nos explicar o que são os tipos primitivos.

Eles se dividem em três tipos:

-Escalares

-Compostos

-Especiais

Ele nos dá um exemplo da aula anterior

$sobrenome = “Gabriel”;

$idade = “16”;

$peso = “60”;

$casado = “false”;

\_\_\_\_\_

Escalares:

$sobrenome

É do tipo string, que é uma sequência de letras, números e símbolos, sempre representados em aspas.

$idade

É do tipo INT ou INTEGER, um valor numérico inteiro

$peso

É do tipo Float, double ou real que não é mais considerado nas versões mais recentes, que seve para os números quebrados.

$casado

É do tipo bool ou boolean, que aceitam true ou false (verdadeiro ou fals)

Esses são os quatro tipos primitivos, do tipo escalar

\_\_\_\_\_\_\_\_

**Aula 22- Manipulação de strings com PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Nesta aula ele nos mostra que existem 2 tipos de strings a com aspas duplas (“”)e a com aspas simples (’’) e cada uma tem sua função, no caso da aspa dupla ele tem a função de abrigar uma variável.

Ele nos da um exemplo:

<?php

$nome = “Enzo”;

$sobrenome = “Gabriel”;

echo “$nome $sobrenome!”;

?>

Neste exemplo da certo e nós temos a impressão Enzo Gabriel, mas quando ele troca o tipo da string temos um erro ao invés de aparecer Enzo Gabriel, aparece Enzo $sobrenome, isso ocorre porque a aspa simples ela não pode abrigar uma variável

Mais adiante ele nos explica que as strings só abrigam variáveis e que constantes não, para isso ele nos da outro exemplo:

<?php

$nome = “Enzo”;

$sobrenome = “Gabriel”;

Const ESTADO = “Paraná”

echo “$nome $sobrenome!” .ESTADO;

?>

Dessa forma a constante também será apresentada.

**Aula 23- Obtendo dados de formulários com PHP (parte 1) - @CursoemVideo de PHP - Gustavo Guanabara**

Nesta aula vamos aprender como criar um formulário.

<header>

<h1>Apresente-se para nós</h1>

</header>

<section>

<form action="cad.php" method="get">

<label for="idnome">Nome:</label>

<input type="text" name="nome" id="idnome">

<label for="idsobrenome">Sobrenome:</label>

<input type="text" name="sobrenome" id="idsobrenome">

<input type="submit" value="Enviar">

</form>

</section>

Foi esse o código que usamos para criar o formulário, e o CSS pegamos no Git Hub do Gustavo.

**Aula 24- Obtendo dados de formulários com PHP (parte 2) - @CursoemVideo de PHP - Gustavo Guanabara**

Na aula passada tivemos um pequeno problema na hora de enviar as informações do formulário dizia que para o local que tinha sido enviadas as informações não aviam sido encontradas, nesta aula vamos resolver esse problema.

O problema era, que para o formulário estava sendo envido ainda não existia, para isso criamos o local para onde essas informações serão processadas.

<header>

<h1>Resultado do processamento</h1>

</header>

<main>

<?php

$nome = $\_GET["nome"] ?? "sem nome";

$sobrenome = $\_GET["sobrenome"] ?? "desconhecido";

echo "<p>É um prazer te conhecer, <strong>$nome $sobrenome</strong>!</p>";

?>

<p>

<a href="javascript:history.go(-1)">Voltar para a página anterior</a>

</p>

</main>

Esse foi o código que usamos para criar esse local onde serão processadas as informações enviadas, e o CSS é o mesmo da página onde está o formulário.

**25- Expressões Aritméticas com PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Gustavo começa o vídeo de uma forma bem diferente (findo que está no show do milhão) voltando, ele nos faz uma pergunta “Para a Linguagem PHP , a expressão 50/2 +3 \*\*2 tem como resultado:

1. 784
2. 4.545454...
3. 34
4. 100”

E ele nos diz que dependo da ordem que resolvermos todas estão corretas, mas qual é a resposta que teríamos do PHP.

No começo da explicação ele nos dá a seguinte conta, 5+2/2=? Temos 2 respostas possíveis 3.5 ou 6, a correta é 6, qual resposta o PHP nos daria, o resultado da conta que o PHP nos dá é 3.5 zueira, é 6 porque qualquer linguagem de programação deve obedecer a ordem de precedência.

Agora voltando a pergunta do começo, do Silvio Santos da shopp, o resultado daquela conta (o verdadeiro) seria 34, porque seguindo a ordem de precedência, potência vem primeiro depois divisão e multiplicação e por último adição e subtração.

Mas teria como eu mudar a ordem da precedência? Sim sabe como, colocando um parêntese, após fazermos essa mudança a expressão ficaria assim 50/ (2 +3) \*\*2, então teríamos que resolver o que está dentro do parêntese primeiro resultando em 2.

A ordem de precedência é a seguinte:

1° ( )

2° \*\* (Potência)

3° \* / % (Multiplicação, divisão e porcentagem)

4° - + (Adição e subtração)

**26- Funções Aritméticas do PHP - @CursoemVideo de PHP - Gustavo Guanabara**

Na aula passada Guanabara nos mostrou como é que o PHP resolve uma conta e os exemplos que ele usou foram 50/2 +3 \*\*2, 5+2/2 e 50/ (2 +3) \*\*2, mas nos ainda não vimos algumas coisas como raiz quadrada, cosseno, operação de bases etc. E tudo isso podemos fazer em PHP, mas não por operador.

Iremos utilizar funções, e são alguns exemplos de operadores aritméticos que podem ser resolvidos por função:

abs( ), Que é o valor absoluto de um número, nada mais é que sua distância até zero na reta numérica, ignorando o sinal.

base\_convert( ), Que é o que transforma um número de um sistema numérico para outro.

ceil( ), floor( ), round( ), Fazem um arredondamento ceil para cima, floor para baixo e round é o arredondamento normal aritmético.

Aula 27-

**Aula 28- Cinco desafios PHP para iniciantes - ‪@CursoemVideo‬ de PHP**

Nesta Gustavo nos dá introdução sobre os desafios, nos mostrando o que iremos fazer a partir desse ponto.

**Aula 29- Desafio 001: Antecessor e Sucessor - ‪@CursoemVideo‬ de PHP**

Está é a primeira aula com desafios e nesta aula vamos fazer um código que quando colocado um número ele irá mostrar o antecessor e o sucessor.

E esse foi o código que usamos:

Index.html

<main>

<h1>Informe um número</h1>

<form action="resposta.php" method="get">

<label for="num">Número: </label>

<input type="number" name="num" id="num">

<input type="submit" value="Calcular">

</form>

</main>

Resposta.php

<main>

<h1>Resultado Final</h1>

<p>

<?php

$n = $\_REQUEST["num"] ?? 0;

$a = $n - 1;

$s = $n + 1;

echo "O número escolhido foi <strong>$n</strong>";

echo "<br>O seu <em>antecessor</em> é $a";

echo "<br>O seu <em>sucessor</em> é $s";

?>

</p>

<button onclick="javascript:history.go(-1)">&#x2B05; Voltar</button>

</main>

</body>

**Aula 30- Desafio 002: Sorteador de números - ‪@CursoemVideo‬ de PHP**

Nessa aula vamos criar um código que gera um número aleatório.

Antes de iniciarmos ele nos fala que existem 3 tipos de fazer esse tipo de exercício:

1° Com a tag rand( ), que é a mais antiga sendo ela de 1951

Para ela ser aplicada seria assim:

<main>

<h1>Trabalhando com números aleatórios</h1>

<?php

$min = 0;

$max = 100;

$num = mt\_rand($min, $max);

echo "<p>Gerando um número aleatório entre $min e $max... <br>O valor gerado foi <strong>$num</strong></p>"

?>

</main>

2° Temos também mt\_rand( ), uma versão mais nova sendo ela de 1997

Para usarmos ela é da seguinte forma:

<?php

<?php

$min = 0;

$max = 100;

$num = mt\_rand($min, $max);

Echo "Gerar um número entre $min e $max... \n”;

Echo “o número foi $num

?>

Da mesma forma, mas so que com uma pequena mudança

3° E a última randon\_int( ), ele gera números aleatórios criptográficamete

<?php

$min = 0;

$max = 100;

$num = andon\_int ($min, $max);

Echo "Gerar um número entre $min e $max... \n”;

Echo “o número foi $num

?>

**Aula 31- Desafio 003: Conversor de Moedas básico - ‪@CursoemVideo‬ de PHP**

Nesta aula vamos fazer um conversor de moedas básico

Index.php

<h1>Conversor de Moedas</h1>

<?php

$cotação = 5.17;

$real = $\_REQUEST["din"] ?? 0;

$dólar = $real / $cotação;

$padrão = numfmt\_create("pt\_BR", NumberFormatter::CURRENCY);

echo "<p>Seus " . numfmt\_format\_currency($padrão, $real, "BRL") . " equivalem a <strong>" . numfmt\_format\_currency($padrão, $dólar, "USD") . "</strong></p>";

?>

<button onclick="javascript:history.go(-1)">Voltar</button>

</mai

Index.html

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Desafio PHP</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<main>

<h1>Conversor de Moedas</h1>

<form action="conversao.php" method="get">

<label for="din">Quantos R$ você tem na carteira?</label>

<input type="number" name="din" id="din" step="0.01">

<input type="submit" value="Converter">

</form>

</main>

**Aula 32- Desafio 004: Conversor de Moedas avançado - ‪@CursoemVideo‬ de PHP**

Nessa aula fizemos um conversor de moeda mais avançado, que utiliza de uma PI

Index.php

<main>

<h1>Conversor de Moedas</h1>

<?php

$inicio = date("m-d-Y", strtotime("-7 days"));

$fim = date("m-d-Y");

$url = '<https://olinda.bcb.gov.br/olinda/servico/PTAX/versao/v1/odata/>' .

'CotacaoDolarPeriodo(dataInicial=@dataInicial,dataFinalCotacao=@dataFinalCotacao)?' .

'@dataInicial=\'' . $inicio . '\'&@dataFinalCotacao=\'' . $fim . '\'' .

'&$top=1&$orderby=dataHoraCotacao%20desc&$format=json&' .

'$select=cotacaoCompra,dataHoraCotacao';

$dados = json\_decode(file\_get\_contents($url), true);

$cotação = $dados["value"][0]["cotacaoCompra"];

$real = $\_REQUEST["din"] ?? 0;

$dólar = $real / $cotação;

$padrão = numfmt\_create("pt\_BR", NumberFormatter::CURRENCY);

echo "<p>Seus " . numfmt\_format\_currency($padrão, $real, "BRL") . " equivalem a

<strong>" . numfmt\_format\_currency($padrão, $dólar, "USD") . "</strong></p>";

?>

<button onclick="javascript:history.go(-1)">Voltar</button>

</main>

Index.html

<main>

<h1>Conversor de Moedas</h1>

<form action="index.php" method="get">

<label for="din">Quantos R$ você tem na carteira?</label>

<input type="number" name="din" id="din" step="0.01">

<input type="submit" value="Converter">

</form>

</main>

**Aula 33- Desafio 005: Analisando um número Real - ‪@CursoemVideo‬ de PHP**

**Nessa aula fizemos um código que examina o número e nos informa quais são os números inteiros e quais os fracionados**

Index.php

<h1>Analizador de Número Real</h1>

<?php

$num = $\_POST["n"] ?? 0;

echo "<p>Analizando o número <strong>" . number\_format($num, 3, ",", ".") . "</strong> informado pelo usuário:</p>";

$int = (int) $num;

$fra = $num - $int;

echo "<ul>";

echo "<li>A parte inteira do número é <strong>" . number\_format($int, 0, ",", ".") . "</strong></li>";

echo "<li>A parte fracionária do número é <strong>" . number\_format($fra, 3, ",", ".") . "</strong></li>";

echo "</ul>";

?>

<button onclick="javascript:history.go(-1)">Voltar</button>

</main>

Index.html

<main>

<h1>Analizador de Número Real</h1>

<form action="index.php" method="post">

<label for="n">Número Real: </label>

<input type="number" name="n" id="n" step="0.001">

<input type="submit" value="Analisar">

</form>

</main>

**Aula 34- Superglobais no PHP - ‪@CursoemVideo‬ de PHP - Gustavo Guanabara**

**Aula 35- Formulários retroalimentados - ‪@CursoemVideo‬ de PHP - Gustavo Guanabara**

**Aula 36- Oito desafios PHP para iniciantes - ‪@CursoemVideo‬ de PHP**

Nesta aula Guanabara apresenta e fala um pouco sobre os proximos desafios que iremos fazer

**Aula 37- Desafio 006: A anatomia de uma divisão - ‪@CursoemVideo‬ de PHP**

Nessa aula de número 37 (graças a Deus está acabando) iremos fazer um código que realize divisões

Index.php

<?php

$dividendo = $\_GET['d1'] ?? 0;

$divisor = $\_GET['d2'] ?? 1;

?>

<main>

<h1>Anatomia de uma Divisão</h1>

<form action="" method="get">

<label for="d1">Dividendo</label>

<input type="number" name="d1" id="d1" min="0" value="<?= $dividendo ?>">

<label for="d2">Divisor</label>

<input type="number" name="d2" id="d2" min="1" value="<?= $divisor ?>">

</form>

</main>

<section>

<?php

$quociente = intdiv($dividendo, $divisor);

$resto = $dividendo % $divisor;

?>

<table class="divisao">

<tr>

<td><?= $dividendo ?></td>

<td><?= $divisor ?></td>

</tr>

<tr>

<td><?= $resto ?></td>

<td><?= $quociente ?></td>

</tr>

</table>

</section>

**Aula 38- Desafio 007: Salário Mínimo com PHP - ‪@CursoemVideo‬ de PHP**

Na aula 38 vamos fazer um código que calcule quantos salários-mínimos quem estiver calculando ganha

Index.php

<?php

$minimo = 1380.60;

$salario = $\_GET['sal'] ?? 0;

?>

<main>

<h1>Informe seu salário</h1>

<form action="<?= $\_SERVER['PHP\_SELF'] ?>" method="get">

<label for="sal">Salário (R$)</label>

<input type="number" name="sal" id="sal" value="<?= $salario ?>" step="0.01">

<p>

Considerando o salário mínimo de

<strong>R$ <?= number\_format($minimo, 2, ",", ".") ?></strong>

</p>

<input type="submit" value="Calcular">

</form>

</main>

<section>

<h2>Resultado Final</h2>

<?php

$tot = intdiv($salario, $minimo);

$dif = $salario % $minimo;

echo "<p>Quem recebe um salário de R$" . number\_format($salario, 2, ",", ".") .

" ganha <strong>$tot salários mínimos</strong> + R$ " .

number\_format($dif, 2, ",", ".") . ".</p>";

?>

</section>

**Aula 39- Desafio 008: Exercício das Raízes - ‪@CursoemVideo‬ de PHP**

Nesta aula vamos fazer um código que calcula raiz quadrada e raiz cúbica

Index.php

<?php

$numero = $\_GET['num']??1;

?>

<main>

<h1>Informe um número</h1>

<form action="<?= $\_SERVER['PHP\_SELF'] ?>" method="get">

<label for="num">Número</label>

<input type="number" name="num" id="num" value="<?= $numero ?>">

<input type="submit" value="Calcular Raízes">

</form>

</main>

<section>

<h2>Resultado final</h2>

<?php

$rq = $numero \*\* (1 / 2);

$rc = $numero \*\* (1 / 3);

echo "<p>Analisando o <strong>número $numero</strong>, temos:</p>";

echo "<ul><li>A sua raiz quadrada é <strong>" . number\_format($rq, 3, ",", ".") . "</strong></li>";

echo "<li>A sua raiz cúbica é <strong>" . number\_format($rc, 3, ",", ".") . "</strong></li></ul>";

?>

</section>

**Aula 40- Desafio 009: Entendendo as Médias - ‪@CursoemVideo‬ de PHP**

Na aula 40 (finalmente só falta 4) vamos fazer um código que calcula média aritmética

Index.php

<?php

$valor1 = $\_GET['v1'] ?? '';

$peso1 = $\_GET['p1'] ?? '';

$valor2 = $\_GET['v2'] ?? '';

$peso2 = $\_GET['p2'] ?? '';

?>

<main>

<h1>Médias Aritméticas</h1>

<form action="<?= $\_SERVER['PHP\_SELF'] ?>" method="get">

<label for="v1">1º Valor</label>

<input type="number" name="v1" id="v1" required value="<?= $valor1 ?>">

<label for="p1">1º Peso</label>

<input type="number" name="p1" id="p1" min="1" required value="<?= $peso1 ?>">

<label for="v2">2º Valor</label>

<input type="number" name="v2" id="v2" required value="<?= $valor2 ?>">

<label for="p2">2º Peso</label>

<input type="number" name="p2" id="p2" min="1" required value="<?= $peso2 ?>">

<input type="submit" value="Calcular Médias">

</form>

</main>

<section>

<?php

$ma = ($valor1 + $valor2) / 2;

$mp = ($valor1 \* $peso1 + $valor2 \* $peso2) / ($peso1 + $peso2);

?>

<h2>Cálculo das Médias</h2>

<p>Analisando os valores <?= $valor1 ?> e <?= $valor2 ?>:</p>

<ul>

<li>A <strong>Média Aritmética Simples</strong> entre os valores é igual a <?= number\_format($ma, 2, ",", ".") ?>.</li>

<li>A <strong>Média Aritmética Ponderada</strong> com pesos <?= $peso1 ?> e <?= $peso2 ?> é igual a <?= number\_format($mp, 2, ",", ".") ?>.</li>

</ul>

</section>

**Aula 41- Desafio 010: Máquina do Tempo - ‪@CursoemVideo‬ de PHP**

Aqui vamos fazer um código que calcula sua idade

Index.php

<?php

$atual = date("Y");

$nasc = $\_GET['nasc'] ?? '2000';

$ano = $\_GET['ano'] ?? $atual;

?>

<main>

<h1>Calculando a sua idade</h1>

<form action="<?= $\_SERVER['PHP\_SELF'] ?>" method="get">

<label for="nasc">Em que ano você nasceu?</label>

<input type="number" name="nasc" id="nasc" min="1900" value="<?= $nasc ?>">

<label for="ano">Quer saber sua idade em que ano? (atualmente estamos em <strong><?= $atual ?></strong>)</label>

<input type="number" name="ano" id="ano" min="1900" value="<?= $ano ?>">

<input type="submit" value="Qual será minha idade?">

</form>

</main>

<section>

<?php

$idade = $ano - $nasc;

?>

<h2>Resultado</h2>

<p>

Quem nasceu em <?= $nasc ?> vai ter

<strong><?= $idade ?></strong> anos

em <?= $ano ?>!

</p>

</section>

**Aula 42- Desafio 011: Reajustar Preços - ‪@CursoemVideo‬ de PHP**

Na 42 vamos fazer um código que mostra o quanto um valor aumenta de acordo com o quanto a porcentagem aumenta

Index.php

<?php

$preco = $\_REQUEST['preco'] ?? '0';

$reaj = $\_REQUEST['reaj'] ?? '0';

?>

<main>

<h1>Reajustador de Preços</h1>

<form action="<?= $\_SERVER['PHP\_SELF'] ?>" method="get">

<label for="preco">Preço do Produto (R$)</label>

<input type="number" name="preco" id="preco" min="0.10" step="0.01" value="<?=$preco?>">

<label for="reaj">Qual será o percentual de reajuste?

(<strong><span id="p">0</span>%</strong>)

</label>

<input type="range" name="reaj" id="reaj" min="0" max="100" step="1" oninput="mudaValor()" value="<?=$reaj?>">

<input type="submit" value="Reajustar">

</form>

</main>

<?php

$aumento = $preco \* $reaj / 100;

$novo = $preco + $aumento;

?>

<section>

<h2>Resultado do Reajuste</h2>

<p>

O produto que custava R$

<?= number\_format($preco, 2, ",", ".") ?>,

com <strong><?= $reaj ?>%</strong> de aumento

vai passar a custar

<strong>R$<?= number\_format($novo, 2, ",", ".") ?></strong>

a partir de agora.

</p>

</section>

<script>

mudaValor();

function mudaValor() {

p.innerText = reaj.value;

}

reaj.addEventListener('input', mudaValor);

</script>

**Aula 43- Desafio 012: Calculadora de Tempo - ‪@CursoemVideo‬ de PHP**

Só falta um, nessa aula vamos fazer um código que calcula quanto tempo, semanas, dias, horas, minutos e segundos, de acordo com o número de segundos que queremos calcular

Index.php

<?php

$total = $\_REQUEST['seg'] ?? 0;

?>

<main>

<h1>Calculadora de Tempo</h1>

<form action="<?= $\_SERVER['PHP\_SELF'] ?>" method="get">

<label for="seg">Qual é o total de segundos?</label>

<input type="number" name="seg" id="seg" min="0" step="1" required value="<?= $total ?>">

<input type="submit" value="Calcular">

</form>

</main>

<?php

$sobra = $total;

$semana = (int)($sobra / 604800);

$sobra = $sobra % 604800;

$dia = (int)($sobra / 86400);

$sobra = $sobra % 86400;

$hora = (int)($sobra / 3600);

$sobra = $sobra % 3600;

$minuto = (int)($sobra / 60);

$sobra = $sobra % 60;

$segundo = $sobra;

?>

<section>

<h2>Totalizando tudo</h2>

<p>Analisando o valor que você digitou, <strong><?=number\_format($total, 0, ",", ".")?></strong> segundos

equivalem a um total de:</p>

<ul>

<li><?=$semana?> semanas</li>

<li><?=$dia?> dias</li>

<li><?=$hora?> horas</li>

<li><?=$minuto?> minutos</li>

<li><?=$segundo?> segundos</li>

</ul>

</section>

**Aula 44- Desafio 013: Simulando um Caixa Eletrônico - ‪@CursoemVideo‬ de PHP**

ALELUIA O ULTIMOO, nessa aula vamos fazer um código que nos mostra quais notas e quantas delas iremos peguar de acordo com o valor do saque, que é o que vamos preencher.

Index.php

<style>

img.nota {

height: 50px;

margin: 5px;

}

</style>

</head>

<body>

<?php

$saque = $\_REQUEST['saque'] ?? 0;

?>

<main>

<h1>Caixa Eletrônico</h1>

<form action="<?=$\_SERVER['PHP\_SELF']?>" method="get">

<label for="saque">Qual valor você deseja sacar? (R$)<sup>\*</sup></label>

<input type="number" name="saque" id="saque" step="5" required

value="<?=$saque?>">

<p style="font-size: 0.6em"><sup>\*</sup>Notas disponíveis:

R$100, R$50, R$10 e R$5</p>

<input type="submit" value="Sacar">

</form>

</main>

<?php

$resto = $saque;

// Saque de R$100

$tot100 = floor($resto / 100);

$resto %= 100;

// Saque de R$50

$tot50 = floor($resto / 50);

$resto %= 50;

// Saque de R$10

$tot10 = floor($resto / 10);

$resto %= 10;

// Saque de R$5

$tot5 = floor($resto / 5);

$resto %= 5;

?>

<section>

<h2>Saque de R$<?=number\_format($saque, 2, ",", ".")?> realizado</h2>

<p>O caixa eletrônico vai te entregar as seguintes notas:</p>

<ul>

<li><?=$tot100?> x R$100</li>

<li><?=$tot50?> x R$50</li>

<li><?=$tot10?> x R$10</li>

<li><?=$tot5?> x R$5</li>

</ul>

</section>

</body>