**Vídeo 01 - Começa aqui seu curso de PHP Moderno -**

**Vídeo 02 - Esse curso de PHP serve pra mim? - @CursoemVideo de PHP - Gustavo Guanabara**  
- Neste vídeo, ele mostra argumentos sobre o porque esse curso seria para mim (como por exemplo, se eu já sei o básico de HTML, CSS e JS, e quero continuar aprofundando mais ainda, PHP seria uma boa linguagem para isso, ou também se eu já tivesse programado em PHP porém nas versões antigas, já que esse curso é para ensinar o PHP moderno) e também o porque não seria (por exemplo, se você já sou uma pessoa com conhecimento mais avançado, se estou procurando um curso de códigos prontos, ou também se estou com pressa, ele afirma que esse curso não seria para você caso esteja buscando um curso com essas coisas). E no final ele propõe um acordo, de comprometimento, pede para que eu anote as coisas aprendidas, e se necessário repetir.

**Vídeo 03 - Lista TOP 6 livros de PHP - @CursoemVideo de PHP - Gustavo Guanabara**  
- Neste vídeo ele apresenta, assim como diz no título, recomendações de livros sobre o PHP. No começo do vídeo ele comenta que é importante pesquisar um pouco sobre a próxima aula antes de assisti-la, pois assim, seu desempenho pode ser melhor na aula, te ajudando a entender melhor o que está sendo passado.

**Vídeo 04 - A evolução do PHP - @CursoemVideo de PHP - Gustavo Guanabara**  
- Neste vídeo, assim como no título, nos é mostrado a história do PHP, começando desde sua criação por Rasmus Lerdof em 1995, suas diversificações de nomes devidos suas funcionalidades, sua abertura para open source devido a seu grande crescimento, suas grandes contribuições por anos feitas pelos dois israelenses Andi e Zeev (que mais para frente fundaram a ZEND, também grande contribuidora para o PHP) vindo junto a transformação real do PHP para uma linguagem de programação, passando do nome inicial PHP (Personal Home Page) para PHP (PHP: Hypertext Preprocessor). Um detalhe interessante dito por Gustavo no final do vídeo, é que Rasmus Lerdof, desenvolvedor principal da linguagem até então, não se encontra mais como desenvolvedor da linguagem, visto que agora está como funcionário na Yahoo, deixando o “cargo” de desenvolvedor principal para Andi e Zeev.

**Vídeo 05 - Por que um elefante é o mascote do PHP? - @CursoemVideo de PHP - Gustavo Guanabara**  
- Basicamente, Vicent o criador da imagem do elefante, que se chama elePHPant, diz que via na própria escrita do PHP um elefante, e por isso, decidiu fazer um elefante. Vicent acrescenta que o elefante é conhecido por ser um animal inteligente e amigável, trazendo esse significado para a linguagem através da mascote.

**Vídeo 06 - As versões do PHP e seus recursos - @CursoemVideo de PHP - Gustavo Guanabara**  
-

**Vídeo 07 - O PHP vai morrer? Vale a pena estudar PHP? - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, o Gustavo quebra a visão das pessoas que falam mal da linguagem, utilizando de argumentos e comparações com outras linguagens e com até coisas materiais. Ele diz que as pessoas que julgam o PHP como uma linguagem ruim, lenta e morta, são pessoas presas no tempo, que utilizam de argumentos super antigos, como de 1994, para justificar isso, mostrando dados sobre o PHP5 (que realmente apresentava diversos defeitos, criando essa imagem de uma linguagem ruim), porém não vendo que hoje em dia o PHP8 se mantém estável, e que ainda é utilizado por praticamente toda a web.

**Vídeo 08 - Como funciona o PHP? - @CursoemVideo de PHP - Gustavo Guanabara**  
-

**Vídeo 09 - É possível criar qualquer coisa com PHP? - @CursoemVideo de PHP - Gustavo Guanabara**  
- Por mais que seja meio óbvio, nesse vídeo o Gustavo apresenta argumentos para mostrar o porquê não dá para fazer tudo com PHP, utilizando de um exemplo cotidiano, como o de uma furadeira/parafusadeira, em que ela é ótima para furar e parafusar, porém para serrar com toda certeza não é indicada. A mesma coisa vale para o PHP, enquanto ele é ótimo para fazer aplicações webs, servidores, etc, por outro lado, ele não é nem um pouco viável para desenvolver um jogo por exemplo.

**Vídeo 10 - Como transformar seu computador em servidor PHP - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, ele mostra o que será necessário para utilizarmos o PHP em nossas máquinas (computadores), desde o navegador, IDE (vscode), até a solução AMP (Apache, MySql, PHP), que para podermos utilizar eles, usaremos nesse curso o XAMPP.

**Vídeo 11 - Seu servidor PHP no Windows - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, o Gustavo basicamente, ensina a baixar, passo a passo, tudo o que foi mostrado no vídeo anterior (XAMPP, VsCode, Chrome). Ele também mostra a solução para alguns possíveis problemas com incompatibilidade e depois ele faz o teste do servidor local (Localhost, proporcionado pelo XAMPP).

**Vídeo 12 - Configurações importantes para VS Code - @CursoemVideo de PHP - Gustavo Guanabara**  
- Aqui nesse vídeo, o Guanabara mostra como criar/abrir a pasta do XAMPP corretamente no vscode e ele mostra também algumas configurações que ele considera importantes para as pessoas que estão começando aprender uma linguagem nova.

**Vídeo 13 - Melhores extensões para VS Code - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, como no título, o Gustavo nos indica extensões do Vs Code para nos auxiliar durante o desenvolvimento, como por exemplo, a extensão para transformar o Vs Code em português, o intelephense para auxiliar durante a escrita dos códigos, o PHP format para retornar as funcionalidades perdidas do HTML pelo fato de o arquivo estar em PHP e por último o PHP Awesome Snippets, que serve para que quando você estiver digitando um código, ele sugere uma forma de completar rapidamente o que você está digitando, agilizando o processo.

**Vídeo 14 - Primeiro programa em PHP - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, fizemos nossos primeiros programas, utilizando do HTML e do PHP, fazendo um simples “Hello, World!” com auxílio do ‘echo’, ‘\u{}’ para colocar um emoji da emojipedia, e mostrando também os dados do servidor com auxílio do ‘phpinfo()’.

**Vídeo 15 - O PHP por dentro - @CursoemVideo de PHP - Gustavo Guanabara**  
- No início desse vídeo, o Guanabara nos mostra algumas tags que era utilizada antigamente para representar o PHP (<? ?>, <% %>, <script language=”php”> </script>) e as que são utilizadas ultimamente: <?php ?>, e o <?= ?> para short tags. Após isso, ele nos explica como uma linha (phpinfo(), vista na aula passada) gera centenas de linhas de código no código fonte da página. Isso é devido o preprocessador do PHP, que olha o código antes de ser lançado para o cliente ver, e preprocessa ele para que ele veja apenas o que deve ver. Utilizando de exemplo na prática ele utilizou das propriedades echo para fazer uma simples frase: “Hoje é dia 24/07/25 e a hora atual é 20:09:05” e o date para determinar esse horário e a data. O preprocessador olha o código e exibe apenas a frase acima, retirando o código com o echo e o date da visão do cliente.

**Vídeo 16 - Variáveis e constantes em PHP - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, aprendemos sobre as variáveis, constantes, operador de atribuição, regras para identificadores para variáveis e tipos recomendados para colocar como nome às variáveis. Variáveis são como se fosse gavetas dentro de um armário gigantesco, que armazenam valores, e em que esses valores podem ser alterados a qualquer momento. Já as constantes, são como as variáveis, um lugar para armazenar valores, porém valores fixos, que não podem ser alterados até o fim do programa. Operador de atribuição é basicamente o =, que ao invés de ser chamado ‘igual’, ele é chamado ‘recebe’, ex: $nome = “Enzo” (variável nome recebe Enzo). Regras para identificadores, basicamente, podemos dizer que é como o nome da variável deve ser escrita, como por exemplo, é obrigatório ter o $ para declarar a variável, o segundo carácter pode ser apenas letra ou ‘\_’ jamais números, entre outros. E tipos recomendados de nomes, o Gustavo nos fala que é bom seguir um padrão para nome de variáveis desde o início do programa, até o final do mesmo, e como recomendação de como devemos escrever os nomes das variáveis, ele deixa como exemplos, o método Camel Case ($nomeCompleto) e o Snake Case ($nome\_completo).

**Vídeo 17 - Tipos primitivos do PHP - @CursoemVideo de PHP - Gustavo Guanabara**  
- Como dito no título, nesse vídeo aprendemos sobre os tipos primitivos, mais especificamente o tipo escalar, porém chegamos a ver algumas coisinhas apenas sobre os outros dois: compostos e especiais. Basicamente, os tipos primitivos são aqueles que definem o que é o valor dentro da variável, se o valor vai ser considerado inteiro, string, booleano ou float/double. Aprendemos também sobre a função var\_dump, que serve para ver o valor armazenado dentro da variável e qual o tipo primitivo dela, por exemplo, $nome = “Enzo” -> var\_dump($nome) -> String(4) “Enzo”.

**Vídeo 18 - Manipulação de strings com PHP - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, aprendemos mais sobre os diferentes tipos de string: Double quoted, Single quoted, Heredoc e Nowdoc, concatenação das strings, mistura de aspas duplas com simples em uma mesma string e sobre as sequências de escape. Agora, vou dar uma breve explicação sobre cada tópico.  
 Basicamente, sobre os tipos de strings, o double quoted (aspas duplas) seria a string com interpretação do que está dentro, ou seja, se eu tenho um Unicode de um emoji qualquer no meio da string, se a string estiver entre aspas duplas, ela irá interpretar o código Unicode e colocar o emoji na apresentação final. Já a single quoted (aspas simples), é justamente o contrário da double quoted, ela não apresenta a interpretação do conteúdo. O tipo Heredoc é praticamente a double quoted só que especificamente para frases maiores, já a Nowdoc, assim como a single quoted, é o inverso da Heredoc porém utiliza da mesma estrutura adicionando apenas as aspas simples (echo <<< ‘TESTE’ - Nowdoc; echo <<< TESTE - Heredoc). OBS: os tipos Heredoc e Nowdoc são raramente utilizados.  
 Sobre os demais, a concatenação das strings, ou seja, o ato de juntar duas strings separadas, diferente de outras linguagens que utilizam do ‘+’ com operador de concatenação, no PHP é utilizado o ‘.’. Misturas de aspas duplas e aspas simples em uma mesma string, geralmente é utilizada para apresentar o apelido de pessoas. E por último as sequências de escape, é utilizada para mostrar um carácter especial, como por exemplo, utilizada em um dos exemplos do vídeo, para mostrar o emoji é necessário utilizar o \u{}, essa função é chamada de sequência de escape.

**Vídeo 19 - Obtendo dados de formulários com PHP (parte 1) - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, nós começamos a criação do formulário que será utilizado nas próximas aulas e tivemos uma breve explicação sobre como funciona o method=”get” e o method=”post”, que basicamente, o get ele faz com que as informações enviadas do formulário apareçam na URL, e o post faz com que elas não apareçam, porém as duas enviam as informações igualmente.

**Vídeo 20 - Obtendo dados de formulários com PHP (parte 2) - @CursoemVideo de PHP - Gustavo Guanabara**  
- Nesse vídeo, continuamos o formulário feito na aula passada, adicionando códigos ao sla.php, fazendo com que as informações passadas do formulário, aparecesse na página para o usuário com uma simples mensagem de comprimento. Além disso, utilizamos o operador de coalescência nula (??), que serve para definir que se, por exemplo, a variável $nome não tiver nenhum valor atribuído a ela, mostra uma mensagem diferente.

**Vídeo 21 - Expressões Aritméticas com PHP - @CursoemVideo de PHP - Gustavo Guanabara**- Nesse vídeo, o Gustavo deu uma breve explicada sobre as expressões aritméticas, ele abordou sobre a ordem de precedência, modificadores da ordem de precedência e provando que funciona na prática.

**Vídeo 22 - Funções Aritméticas do PHP - @CursoemVideo de PHP - Gustavo Guanabara**- Nesse vídeo, assim como no título, o Guanabara nos mostra as funções aritméticas no PHP, como por exemplo, a função para calcular a hipotenusa é hypot(), para calcular a raiz quadrada é sqrt(), entre outras funções que estão no github.

**Vídeo 23 - Operadores Aritméticos do PHP - @CursoemVideo de PHP - Gustavo Guanabara**- Nesse vídeo, o Gustavo começa mostrando meio que uma curiosidade, demonstrando a evolução do PHP, utilizando do exemplo: “2” + “2” retornaria que resultado? O PHP, diferente do JS, ele consegue converter a string em inteiro, retornando 4 em inteiro (se utilizarmos do var\_dump(), confirmamos isso), porém se tiver por exemplo, “2 dólares” + “12 reais”, na versão atual será retornado um erro, já nas versões mais antigas do PHP, como no PHP7 ou no PHP5, ele pega apenas os números e retorna a soma dos mesmos (2 + 12 = 14). Após isso, ele mostra os operadores aritméticos: + (adição), - (subtração), / (divisão real), \* (multiplicação), % (resto da divisão ou módulo) e \*\* (potenciação).

**Vídeo 24 - Cinco desafios PHP para iniciantes -  de PHP**- Nesse vídeo, foi proposto para nós 5 desafios para colocar em prática o que aprendemos e também para aprender outras funcionalidades/funções.

**Vídeo 25 - Desafio 001: Antecessor e Sucessor -  de PHP**- Nesse vídeo, vemos a resolução do desafio 001 do vídeo anterior e vimos que a forma de adicionar um ícone no HTML, diferente do PHP que utiliza do \u{}, no HTML é utilizado o &#x.

**Vídeo 26 - Desafio 002: Sorteador de números -  de PHP**- Nesse vídeo, vemos a solução do desafio 002 e aprendemos sobre os diferentes tipos de geradores de números aleatórios, sendo eles: rand() – o mais antigo, mt\_rand() – o mais moderno e 4x vezes mais rápido que o anterior e o random\_int() – utilizado para a geração de números criptograficamente seguros (e também é função mais lenta dentre as outras). Além disso, vimos como fazer a parte da geração de um novo número pelo botão, recarregando a página, segue o código: <button onclick=”javascript:document.location.reload()”></button>.

**Vídeo 27 - Desafio 003: Conversor de Moedas básico -  de PHP**- Neste vídeo, vimos a solução do desafio 002 e aprendemos sobre formatar números de duas diferentes maneiras, utilizando do number\_format() e da biblioteca de formatação de moedas com internacionalização (intl). Além disso, aprendemos a como habilitar a biblioteca.