

# PHP

TechGuide - Alura, FIAP e PM3

---

## Nível 1

### Lógica de Programação:

- Aprender lógica de programação, fundamental para o desenvolvimento de software
- Conhecer as bases para se criar, analisar e resolver problemas computacionais de forma estruturada e eficiente
- Entender o que são tipos de dados
- Declarar variáveis, considerando os diferentes tipos
- Conhecer os operadores de atribuição e comparação
- Usar estruturas condicionais
- Usar estruturas de repetição e laços
- Usar funções, passando parâmetros e argumentos

### Conteúdos

- **Artigo** Lógica de programação: o que é e para que serve?  
(<https://www.locaweb.com.br/blog/temas/codigo-aberto/logica-de-programacao-o-que-e/>)
- **Artigo** Lógica de programação: guia completo para iniciantes  
(<https://www.hostgator.com.br/blog/logica-de-programacao/>)
- **Site** Livro - Lógica de Programação  
(<https://www.casadocodigo.com.br/products/livro-programacao>)

- **YouTube** O MÍNIMO QUE VOCÊ PRECISA SABER ANTES DE PROGRAMAR!  
(<https://www.youtube.com/watch?v=BTENKdRVS2U>)
- **Podcast** Se você não aprender lógica de programação, você NÃO VAI CONSEGUIR VAGA  
(<https://open.spotify.com/episode/2XxEhO0cPfIGp3oGRgGJLX?si=FtgYSOvaSZGo-B31ex5ANQ>)

### Conteúdos Alura:

- **Curso** Formação A partir do zero: iniciante em programação  
(<https://cursos.alura.com.br/formacao-programacao>)

### PHP - Fundamentos:

- PHP é uma linguagem de programação que permite aos desenvolvedores web criar conteúdo dinâmico que interage com bancos de dados. O PHP é basicamente usado para desenvolver aplicações de software baseadas na web.
- Conhecer os tipos primitivos
- Declarar variáveis
- Usar estruturas condicionais ('if', 'else')
- Conhecer os operadores de atribuição e comparação ('=', '==', '===')
- Usar estruturas de repetição e laços ('while', 'for')
- Usar funções, passando parâmetros e argumentos
- Manipular arrays e listas

### Conteúdos

- **Site** Documentação PHP ([https://www.php.net/manual/pt\\_BR/index.php](https://www.php.net/manual/pt_BR/index.php))
- **Artigo** Introdução ao PHP (<https://medium.com/caf%C3%A9-com-java/introdu%C3%A7%C3%A3o-ao-php-91472783da7c>)
- **Artigo** Guia prático do Modern PHP: desenvolvimento e ecossistema  
(<https://medium.com/@FernandoDebrand/guia-pr%C3%A1tico-do-modern-php-desenvolvimento-e-ecossistema-c9715184e463>)

- **Artigo** Curso de PHP Básico  
(<https://medium.com/walternascimento-barroso-pt/curso-de-php-b%C3%A1sico-fa984be7c9b9>)
- **YouTube** Mjailton: O que é PHP ? Saiba tudo sobre esta Linguagem  
(<https://www.youtube.com/watch?v=va4Wlzo211Y>)
- **YouTube** Mjailton: Aprenda Como Rodar o PHP  
([https://www.youtube.com/watch?v=xKkipZ133\\_s](https://www.youtube.com/watch?v=xKkipZ133_s))

## Conteúdos Alura:

- **Podcast** Hipsters.tech: PHP não é coisa de sobrinho - Hipsters Ponto Tech 36 (<https://cursos.alura.com.br/extra/hipsterstech/php-nao-e-coisa-de-sobrinho-hipsters-36-a550>)
- **Podcast** Hipsters.tech: PHP8, Mudanças e Ecossistema - Hipsters Ponto Tech 241 (<https://cursos.alura.com.br/extra/hipsterstech/php8-mudancas-e-ecossistema-hipsters-ponto-tech-241-a789>)
- **YouTube** Alura: Ecossistema PHP | HipstersPontoTube  
(<https://www.youtube.com/watch?v=yD3BqXWHua4>)
- **Artigo** PHP - Uma Introdução à Linguagem  
(<https://www.alura.com.br/artigos/php-uma-introducao-linguagem>)
- **Artigo** Quando usar == ou === em php?  
(<https://www.alura.com.br/artigos/quando-usar-ou-em-php>)
- **Artigo** PHP: operação com números decimais  
(<https://www.alura.com.br/artigos/php-operacoes-com-numeros-decimais>)
- **Artigo** Trabalhando com arrays em PHP  
(<https://www.alura.com.br/artigos/trabalhando-com-arrays-em-php>)
- **Curso** Formação PHP (<https://cursos.alura.com.br/formacao-linguagem-php>)
- **Curso** Curso PHP: conceitos, lidando com dados, loops e mais  
(<https://cursos.alura.com.br/course/php-primeiros-passos>)
- **Curso** Curso PHP: manipulando coleções com Arrays  
(<https://cursos.alura.com.br/course/php-manipulando-colecoes-arrays>)

- **Curso** Curso PHP Strings: manipulando textos com PHP (<https://cursos.alura.com.br/course/php-strings-manipulando-textos-php>)
- **Desafio** 7 Days of Code: PHP (<https://7daysofcode.io/matricula/php>)

## PHP - Coleções:

- Uma coleção representa um grupo de objetos, conhecidos como seus elementos. Eles são como recipientes que agrupam vários itens em uma única unidade. Algumas coleções permitem a duplicação de elementos e outras não. Algumas são ordenadas e outras não ordenadas.
- Aprender os usos e diferenças entre arrays, listas, pilhas, hash maps, heaps
- Trabalhar com Iterators, Geradores e Collections

## Conteúdos

- **Site** Documentação: Arrays ([https://www.php.net/manual/pt\\_BR/language.types.array.php](https://www.php.net/manual/pt_BR/language.types.array.php))
- **Artigo** Como trabalhar com Iterators e Collections em PHP (<https://www.devmedia.com.br/como-trabalhar-com-iterators-e-collections-em-php/29016>)
- **Artigo** Collections em PHP (<https://imasters.com.br/back-end/collections-em-php>)
- **YouTube** Rodrigo Oliveira: Curso de PHP 7 - Arrays #1 ([https://www.youtube.com/watch?v=HJSad\\_aSZ6A&ab\\_channel=RodrigoOliveira](https://www.youtube.com/watch?v=HJSad_aSZ6A&ab_channel=RodrigoOliveira))
- **YouTube** WDEV: Arrays em PHP - criação, ordenação e manipulação de dados ([https://www.youtube.com/watch?v=0ODab\\_KrjCg&ab\\_channel=WDEV](https://www.youtube.com/watch?v=0ODab_KrjCg&ab_channel=WDEV))
- **YouTube** iMasters: Explorando Collections e Estruturas de Dados em PHP ([https://www.youtube.com/watch?v=VMfQTP4p55A&ab\\_channel=iMasters](https://www.youtube.com/watch?v=VMfQTP4p55A&ab_channel=iMasters))

## Conteúdos Alura:

- **Artigo** Curso PHP: manipulando coleções com Arrays (<https://www.alura.com.br/curso-online-php-manipulando-colecoes-arrays>)

- **Curso** Curso PHP: dominando as Collections  
(<https://www.alura.com.br/curso-online-php-dominando-collections>)

## Conceitos de Orientação a Objetos:

- A Programação Orientada a Objetos é um paradigma de programação de software baseado na composição e interação entre diversas unidades chamadas de 'objetos' e as classes, que contêm uma identidade, propriedades e métodos). Ela é baseada em quatro componentes da programação: abstração digital, encapsulamento, herança e polimorfismo.
- Como funcionam objetos
- Criar e utilizar construtores
- O que são classes
- Criar e utilizar métodos
- Como funciona encapsulamento
- O que é herança
- O que é polimorfismo
- Como funcionam interfaces
- O que são abstrações

## Conteúdos

- **Artigo** Princípios básicos da Programação Orientação a Objetos (POO)  
(<https://yanborowski.medium.com/princ%C3%ADpios-b%C3%A1sicos-da-programa%C3%A7%C3%A3o-orienta%C3%A7%C3%A3o-a-objetos-poo-62da3998b7ce>)
- **YouTube** FernandaDev: O que é Programação Orientada a Objetos?(POO)  
(<https://www.youtube.com/watch?v=QJjY2TNyl-8>)
- **YouTube** Giuliana Bezerra: Desmistificando os tipos de referência - Ep. 8  
(<https://youtu.be/D9hANYbPYCc>)

## Conteúdos Alura:

- **Podcast** Hipsters.tech: TechGuide - Orientação a Objetos – Hipsters Ponto Tech #350 (<https://www.hipsters.tech/techguide-orientacao-a-objetos->

[hipsters-ponto-tech-350/](https://hipsters-ponto-tech-350/))

- **Artigo** POO: o que é programação orientada a objetos?  
(<https://www.alura.com.br/artigos/poo-programacao-orientada-a-objetos>)
- **Site** Apostila: Java e Orientação a Objetos  
(<https://www.alura.com.br/apostila-java-orientacao-objetos>)
- **Site** Livro: Orientação a Objetos  
(<https://www.casadocodigo.com.br/products/livro-oo-conceitos>)
- **YouTube** Alura: Orientação a objetos (com Roberta Arcoverde)  
(<https://www.youtube.com/watch?v=jpuJ1qrluoU>)
- **YouTube** Alura: Imersão Java: Orientação a Objetos, APIs e além (e live coding!) (<https://www.youtube.com/watch?v=WdT90ffB-0Q>)
- **Curso** Formação Java e Orientação a Objetos  
(<https://cursos.alura.com.br/formacao-java>)
- **Curso** Formação C# e Orientação a Objetos  
(<https://cursos.alura.com.br/formacao-c-sharp-orientacao-objetos>)
- **Curso** Formação Python e orientação a objetos  
(<https://cursos.alura.com.br/formacao-Python-linguagem>)
- **Curso** Curso Orientação a Objetos com PHP: Classes, métodos e atributos  
(<https://www.alura.com.br/curso-online-php-oo-classes-metodos-atributos>)
- **Curso** Curso JavaScript: programação orientada a objetos  
(<https://www.alura.com.br/curso-online-javascript-passos-programacao-orientada-objetos>)

## PHP - Manipulação de Erros:

- O tratamento de erros refere-se aos procedimentos de resposta e recuperação de condições de erro presentes em um aplicativo de software. Em outras palavras, é o processo composto de antecipação, detecção e resolução de erros de aplicação, de programação ou de comunicação.
- Tratar exceções pré-definidas
- Uso de 'try' e 'catch'
- Criar exceções específicas

- Aprender sobre error handlers
- Fazer o processo de Debug

## Conteúdos

- **Site** Documentação PHP - Exceções  
([https://www.php.net/manual/pt\\_BR/language.exceptions.php](https://www.php.net/manual/pt_BR/language.exceptions.php))
- **Artigo** Não vá dizer que Exception é o mesmo que Erro  
(<https://cristianounix.medium.com/n%C3%A3o-v%C3%A1-dizer-que-exception-%C3%A9-o-mesmo-que-erro-850ea83e72a>)
- **Artigo** Criando Exceptions para impressionar no Teste Técnico  
(<https://dev.to/he4rt/criando-exceptions-para-impressionar-no-teste-tecnico-2nie>)
- **YouTube** WDEV: Try, catch e finally - tratando exceptions com PHP  
(<https://www.youtube.com/watch?v=S-3-9UkVKi4>)
- **YouTube** daves tecnologia: Tratamento de Erros em PHP  
(<https://www.youtube.com/watch?v=rSvtPN5F3sk>)

## Conteúdos Alura:

- **Curso** Curso PHP Exceptions: tratamento de erros  
(<https://cursos.alura.com.br/course/php-exceptions-tratamento-erros>)

## PHP - Persistência:

- O conceito de "persistência de dados" refere-se a garantir que as informações inseridas na aplicação serão armazenadas em um meio em que possam ser recuperadas de forma consistente. Ou seja, são registros permanentes e que não são perdidos quando há o encerramento da sessão.
- Conhecer o PDO
- Fazer inserções, alterações e remoções em um banco de dados
- Fazer consultas em um banco de dados usando filtros

## Conteúdos

- **Artigo** PHP Data Object (PDO) de maneira Produtiva: boas práticas e produtividade na interação com a camada de dados  
(<https://sergiodanilojr.medium.com/php-data-object-pdo-de-maneira-produtiva-boas-pr%C3%A1ticas-e-produtividade-na-intera%C3%A7%C3%A3o-com-a-b86ec61e66f2>)
- **Artigo** Entendendo a classe PDO  
(<https://medium.com/trainingcenter/entendendo-e-utilizando-a-classe-pdo-836b3fcdf75>)
- **Artigo** Primeiros passos com PHP e PDO  
(<https://medium.com/@valdiney/primeiros-passos-com-php-e-pdo-191a9dccc210>)
- **YouTube** Professor Edson Maia: Persistência com PHP OO e PDO  
(<https://www.youtube.com/watch?v=5FPTKe5PXLo>)
- **YouTube** Daniel Alves: PHP - Classe de Persistência  
(<https://www.youtube.com/watch?v=ayo2Qa37Mrg>)

## Conteúdos Alura:

- **Artigo** Como acessar o banco de dados com PHP7: mysqli ou PDO?  
(<https://www.alura.com.br/artigos/como-acessar-o-banco-de-dados-com-php-7>)
- **Curso** Curso PHP e PDO: trabalhando com bancos de dados  
(<https://cursos.alura.com.br/course/php-pdo-banco-de-dados>)

## PHP - Arquivos e Streams:

- Os 'streams' (fluxos) são a forma de generalizar operações com arquivos, rede, compressão de dados e outras que compartilhem um conjunto comum de funções e usos. Em sua definição mais simples, um stream é um objeto que exibe um comportamento 'streamable'. Ou seja, ele pode ser lido ou escrito de uma forma linear, e pode ser capaz de buscar uma localização arbitrária dentro do fluxo.
- Abrir, ler e manipular arquivos



- Aplicar wrappers e filtros nos streams
- Usar streams
- Conhecer encodings e character sets

## Conteúdos

- **Artigo** Wrapper de Streams PHP (<https://dias.dev/2020-11-03-wraper-de-streams-php/>)
- **Artigo** Leitura de arquivos CSV com PHP (<https://medium.com/mestredev/leitura-de-arquivos-csv-com-php-33819d78d03b>)
- **Artigo** Classe de upload de arquivos com PHP — Como criar (<https://marcioluucas.medium.com/criando-classe-de-upload-de-arquivos-com-php-40892cc63b78>)
- **Artigo** Como ler arquivos JSON muito grandes com PHP (<https://herisonassuncao.medium.com/como-ler-arquivos-json-muito-grandes-com-php-1a269f78a8c6>)
- **YouTube** Marcelo Josué Telles: Como ler arquivos em PHP? (<https://www.youtube.com/watch?v=2SWBxr8BNtM>)
- **YouTube** WDEV: Upload de arquivos com PHP orientado a objetos (<https://www.youtube.com/watch?v=bIAyp12FuQY>)

## Conteúdos Alura:

- **Site** Upload de arquivos com PHP (<https://cursos.alura.com.br/extra/alura-mais/upload-de-arquivos-com-php-c205>)
- **Curso** Curso PHP I/O: trabalhando com arquivos e streams (<https://cursos.alura.com.br/course/php-io-arquivos-streams>)

## Nível 2

### PHP - Composer:

- O Composer é uma ferramenta para o gerenciamento de dependências em PHP. Ele permite que você declare as bibliotecas das quais seu projeto

depende e as gerenciará (instalar/atualizar) para você.

- Gerenciar dependências
- Incluir bibliotecas de terceiros

## Conteúdos

- **Site** Documentação Composer (inglês) (<https://getcomposer.org/doc/>)
- **Artigo** O que é PHP Composer? (<https://medium.com/@morkhusz/o-que-%C3%A9-php-composer-974960bccf0e>)
- **Artigo** O que é o Composer do PHP (<https://felipeelia.com.br/o-que-e-o-composer-do-php/>)
- **YouTube** Felipe Elia WP: COMPOSER (PHP) - O que é, como instalar e como usar (<https://www.youtube.com/watch?v=kQNZz7smW3k>)

## Conteúdos Alura:

- **Artigo** Primeiros passos com Composer (<https://www.alura.com.br/artigos/primeiros-passos-com-composer>)
- **Artigo** PHP Composer: autoloading, executáveis e scripts (<https://www.alura.com.br/artigos/avancando-com-o-composer>)
- **Curso** Curso PHP Composer: Dependências, Autoload e Publicação (<https://cursos.alura.com.br/course/php-composer>)

## PHP - Testes:

- O teste de software é o processo de avaliação e verificação de que um software realmente faz o que deveria fazer. Os benefícios dos testes incluem a prevenção de bugs, a redução dos custos de desenvolvimento e a melhoria do desempenho.
- Usar testes unitários
- Usar testes de integração
- Usar testes de comportamento (behavior)
- Usar mocks

## Conteúdos

- **Artigo** Testes Unitários: Fundamentos e Qualidade de Software! (<https://dev.to/wnqueiroz/testes-unitarios-fundamentos-e-qualidade-de-software-5af0>)
- **Artigo** Testes unitários com PHP 7 (<https://medium.com/@bernardoamaral/testes-unit%C3%A1rios-com-php-7-b018164cd284>)
- **Artigo** Os Diferentes Tipos de Testes (<https://medium.com/elefante-yogue/m1-aula-1-os-diferentes-tipos-de-testes-8943dfe1c476>)
- **Artigo** Uma Breve introdução a testes em PHP (<https://medium.com/trainingcenter/phpunit-e-cobertura-de-c%C3%B3digo-uma-breve-introdu%C3%A7%C3%A3o-a-testes-em-php-47c7ffb7b5ba>)
- **Artigo** Testes de Aceitação com PHP (<https://medium.com/trainingcenter/testes-de-aceita%C3%A7%C3%A3o-com-php-da06054c64dd>)
- **YouTube** Dias de Dev: PHPUnit - Crie testes de maneira profissional em PHP (<https://www.youtube.com/watch?v=1oTRpTPEyX0>)
- **YouTube** Dias de Dev: O que são Testes Automatizados (<https://www.youtube.com/watch?v=O81uVNg4nEk>)
- **YouTube** Dias de Dev: Pest - Uma nova forma de escrever testes em PHP (<https://www.youtube.com/watch?v=c7s4MW10GoY>)

## Conteúdos Alura:

- **Artigo** Tipos de testes: quais os principais e por que utilizá-los? (<https://www.alura.com.br/artigos/tipos-de-testes-principais-por-que-utiliza-los>)
- **Podcast** Hipsters.tech: Testes Automatizados - Hipsters 51 (<https://cursos.alura.com.br/extra/hipsterstech/testes-automatizados-hipsters-51-a535>)
- **YouTube** Alura: Pirâmide de Testes (<https://www.youtube.com/watch?v=gVJ1Lap-JYA>)

- **Curso** Curso PHP e TDD: testes com PHPUnit  
(<https://cursos.alura.com.br/course/phpunit-tdd>)
- **Curso** Curso Testes de integração com PHP: testando o acesso à API e ao banco de dados (<https://cursos.alura.com.br/course/php-testes-integracao>)
- **Curso** Curso PHP e Behavior Driven Development: BDD com Behat  
(<https://cursos.alura.com.br/course/php-introducao-bdd>)

## PHP - MVC:

- Model-view-controller (MVC) é um padrão de arquitetura de software comumente usado para desenvolver interfaces de usuário que dividem a lógica do programa relacionado em três elementos interligados. Isto é feito para separar as representações internas de informação das formas como a informação é apresentada para o usuário.
- Entender o conceito do padrão MVC (Model-View-Controller)
- Separar as responsabilidades entre as camadas MVC

## Conteúdos

- **Artigo** Construindo um simples framework MVC com PHP  
(<https://medium.com/@jardelgoncalves/construindo-um-simples-framework-mvc-com-php-349e9cacbeb1>)
- **Artigo** MVC - framework: usando a arquitetura sem código de terceiros  
(<https://medium.com/trainingcenter/mvc-framework-usando-a-arquitetura-sem-c%C3%B3digo-de-terceiros-bf95a744c66d>)
- **YouTube** WDEV: MVC em PHP - Conceito e início do projeto  
([https://www.youtube.com/watch?v=TmeyoTNu748&list=PL\\_zkXQGHYosGQwNkMMdhRZgm4GjspTnXs&index=1](https://www.youtube.com/watch?v=TmeyoTNu748&list=PL_zkXQGHYosGQwNkMMdhRZgm4GjspTnXs&index=1))
- **YouTube** Miriam TechCod: Aprenda MVC com PHP  
(<https://www.youtube.com/watch?v=dctNvYVwDtl&list=PLYGFJHWj9BYqyAxiT02orCWbta5IWsCCL&index=1>)

## Conteúdos Alura:

- **Curso** Curso PHP na Web: conhecendo o padrão MVC  
(<https://cursos.alura.com.br/course/php-web-conhecendo-padrao-mvc>)

## Laravel:

- Laravel é um framework PHP livre e open-source para o desenvolvimento de sistemas web que utilizam o padrão MVC (model, view, controller).
- Conhecer as características do framework Laravel
- Trabalhar com rotas
- Usar arquivos Blade
- Usar as suas funcionalidades ORM
- Lidar com injeção de dependências

## Conteúdos

- **Site** Documentação Laravel (inglês) (<https://laravel.com/docs/8.x/readme>)
- **Artigo** O que é Laravel? Porque usá-lo? (<https://medium.com/desenvolvendo-com-paixao/o-que-%C3%A9-laravel-porque-us%C3%A1-lo-955c95d2453d>)
- **Artigo** Como o Laravel funciona (<https://medium.com/jundevelopers/como-o-laravel-funciona-71136a262cee>)
- **Artigo** Começando com Laravel (<https://medium.com/@marcelgsantos/come%C3%A7ando-com-laravel-parte-01-e76567408b1f>)
- **YouTube** Gustavo Neitzke: Laravel - 02 | Criando o primeiro projeto ([https://www.youtube.com/watch?v=1ZCM6F3DRWc&list=PLSHNk\\_yA5fNhzzjOGcldSB1nyhv6NR0W1&index=2](https://www.youtube.com/watch?v=1ZCM6F3DRWc&list=PLSHNk_yA5fNhzzjOGcldSB1nyhv6NR0W1&index=2))
- **YouTube** Cristian Bernardes: Criando uma API Rest com Laravel em 10 minutos (<https://www.youtube.com/watch?v=daq5S5-55Lw>)

## Conteúdos Alura:

- **Artigo** Seja um artesão da web com Laravel, o framework MVC do PHP (<https://www.alura.com.br/artigos/seja-um-artesao-da-web-com-laravel-o-framework-mvc-do-php>)
- **Podcast** Hipsters.tech: PHP não é coisa de sobrinho - Hipsters Ponto Tech 36 (<https://cursos.alura.com.br/extra/hipsterstech/php-nao-e-coisa-de-36>)

[sobrinho-hipsters-36-a550](#))

- **Curso** Curso Laravel: criando uma aplicação com MVC (<https://cursos.alura.com.br/course/laravel-criando-aplicacao-mvc>)
- **Curso** Curso Laravel: validando formulários, usando sessões e definindo relacionamentos (<https://cursos.alura.com.br/course/laravel-formularios-sessoes-relacionamentos>)
- **Curso** Curso Laravel: transações, service container e autenticação (<https://cursos.alura.com.br/course/laravel-transacoes-service-container-autenticacao>)
- **Curso** Curso Laravel: e-mails, eventos assíncronos, uploads e testes (<https://cursos.alura.com.br/course/laravel-e-mails-eventos-uploads-testes>)
- **Curso** Curso Laravel: construindo APIs (<https://cursos.alura.com.br/course/laravel-construindo-apis>)

## PHP - Symfony:

- Symfony é um framework PHP de código aberto e de estrutura MVC que incorpora muitas funcionalidades e boas práticas de desenvolvimento do domínio PHP.
- Conhecer as principais características do framework
- Utilizar componentes
- Definir recursos e rotas
- Templates Twig

## Conteúdos

- **Site** Documentação Symfony (inglês) (<https://symfony.com/doc/current/index.html>)
- **Artigo** Como criar uma página com Symfony (<https://medium.com/code-maestro/como-criar-uma-p%C3%A1gina-com-symfony-91193e80b3d0>)
- **Artigo** Crud no Symfony com Doctrine (<https://medium.com/php-word/crud-no-symfony-com-doctrine-b7ddb820c332>)

- **YouTube** PoliCast: Symfony 5 - Criando seu primeiro projeto (<https://www.youtube.com/watch?v=gIWw77txato&t=10s>)
- **YouTube** Code Experts: API REST com Symfony 6 | Primeiros Passos e Setup Projeto (<https://www.youtube.com/watch?v=zhuYBGQGDBc>)

### Conteúdos Alura:

- **Podcast** Hipsters.tech: PHP não é coisa de sobrinho - Hipsters Ponto Tech 36 (<https://cursos.alura.com.br/extra/hipsterstech/php-nao-e-coisa-de-sobrinho-hipsters-36-a550>)
- **Curso** Formação Symfony e Doctrine (<https://cursos.alura.com.br/formacao-symfony-e-doctrine>)

### PHP - Comunicação com APIs:

- Uma API é uma interface que desenvolvedores de software utilizam para programar a interação com componentes ou recursos de software fora de seu próprio código. Uma definição ainda mais simples é que uma API é a parte de um componente de software que é acessível a outros componentes.
- Entender o que é uma API REST
- Conhecer os comandos básicos de comunicação HTTP
- Conhecer o cURL
- Construir APIs em Laravel
- Saber fazer requisições autenticadas para Web APIs

### Conteúdos

- **Artigo** cURL com PHP para APIs RESTful (<https://imasters.com.br/back-end/curl-com-php-para-apis-restfull>)
- **YouTube** Fred Tavares: Fazendo e Consumindo API com PHP ([https://www.youtube.com/watch?v=81Fy\\_TC6Msc](https://www.youtube.com/watch?v=81Fy_TC6Msc))
- **YouTube** WDEV: Consulta de CNPJs em PHP - Consumindo a API gratuita do Speedio (<https://www.youtube.com/watch?v=BE6LhjGlfrE>)

- **YouTube** WDEV: cURL - Enviando requisições HTTP com a libcurl do PHP (<https://www.youtube.com/watch?v=b1wBRZl6dQ8>)

### Conteúdos Alura:

- **Artigo** Curl: como usar e principais opções (<https://www.alura.com.br/artigos/curl-como-usar>)
- **Curso** Curso Laravel: construindo APIs (<https://www.alura.com.br/curso-online-laravel-construindo-apis>)

### PHP - XDebug:

- Xdebug é uma extensão PHP que fornece recursos de depuração e criação de perfil.
- Usar e aprimorar o var\_dump com XDebug
- Analisar e entender a stack (pilha) de execução
- Depurar o código
- Depurar remotamente com Docker

### Conteúdos

- **Site** Documentação XDebug (inglês) (<https://xdebug.org/docs/>)
- **Artigo** Laravel e Xdebug (<https://andrelugomes.medium.com/laravel-e-xdebug-795ac66457dc>)
- **Artigo** Ambiente de Desenvolvimento PHP com Docker + XDebug + VSCode (<https://medium.com/@alexcvcoelho/ambiente-de-desenvolvimento-php-com-docker-xdebug-vscode-94bda2b49be9>)
- **YouTube** Pensando em Software: PHP - Instalando e utilizando o XDebug (<https://www.youtube.com/watch?v=kQNZz7smW3k>)
- **YouTube** Code Easy: Configurar o XDebug para PHP com o VSCode (<https://www.youtube.com/watch?v=MDUAXMPYQPA>)
- **YouTube** Dias de Dev: XDebug e Docker - Configurar PHPStorm e VSCode para funcionar com XDebug usando Docker (<https://www.youtube.com/watch?v=kbq3FJOYmQ0>)



## Conteúdos Alura:

- **Artigo** Configurando o Xdebug em editores da JetBrains (<https://www.alura.com.br/artigos/configurando-xdebug-em-editores-jetbrains>)
- **Curso** Curso XDebug: ferramenta de debug e profiling (<https://cursos.alura.com.br/course/xdebug-ferramenta-debug-profiling>)

## Nível 3

### PHP Assíncrono:

- Na programação assíncrona as funções não são executadas em ordem. Com o assincronismo, podemos interromper do código para conseguirmos alguma outra informação necessária para a continuar a execução. Isso significa que o código espera por uma outra parte do código e, enquanto espera, executa as demais partes.
- Aprender a usar sockets
- Conhecer bibliotecas de programação reativa
- Aprender sobre corrotinas

### Conteúdos

- **Artigo** PHP assíncrono de forma nativa (<https://dias.dev/2020-09-16-php-assincrono-de-forma-nativa/>)
- **Artigo** Aplicações assíncronas com PHP, Swoole e Lumen (<https://medium.com/@erycson.nobrega/aplica%C3%A7%C3%B5es-ass%C3%ADncronas-com-php-swoole-e-lumen-bdce5eb200f0>)
- **Artigo** I/O assíncrono em PHP (<https://medium.com/@jefersondaniel/i-o-ass%C3%ADncrono-em-php-b38e7aa91ff9>)
- **YouTube** Dias de Dev: Programação assíncrona vs Programação paralela - Entenda a diferença (<https://www.youtube.com/watch?v=zLfXPSeCkB8>)
- **YouTube** PHPSP: PHPapo #19 - PHP assíncrono funciona? ([https://www.youtube.com/watch?v=02B1yEBGC2o&ab\\_channel=PHPSP](https://www.youtube.com/watch?v=02B1yEBGC2o&ab_channel=PHPSP))

- **YouTube** Dias de Dev: Coroutines - O que são corrotinas?  
(<https://www.youtube.com/watch?v=Nbt0eQHChol>)

### Conteúdos Alura:

- **Curso** Curso PHP Reativo: programação assíncrona em tempo real  
(<https://cursos.alura.com.br/course/php-reativo-programacao-assincrona-tempo-real>)
- **Curso** Curso Swoole PHP: servidor assíncrono e corrotinas com PHP  
(<https://cursos.alura.com.br/course/swoole-php-servidor-assincrono-corrotinas-php>)

### PHP Funcional:

- A programação funcional é um paradigma de programação baseado no uso de funções. As funções estão, de acordo com este paradigma, no epicentro de uma base de código.
- Entender a fundo sobre funções em PHP
- Conhecer funções de manipulação de arrays
- Realizar composição de funções usando pipelines

### Conteúdos

- **Artigo** PHP Funcional? Map, filter e reduce, conheça!  
(<https://medium.com/@jul101r4/php-funcional-map-filter-e-reduce-conhe%C3%A7a-ceaf24392376>)
- **YouTube** Dias de Dev: Programação funcional - Aplicação parcial de funções  
(<https://www.youtube.com/watch?v=N1EJTty-HYkk>)
- **YouTube** Dias de Dev: Mônadas - Como representar tipos opcionais  
(<https://www.youtube.com/watch?v=1FdK8adSo4Y>)

### Conteúdos Alura:

- **Curso** Curso PHP: Programação Funcional  
(<https://cursos.alura.com.br/course/php-programacao-funcional>)

## Object Calisthenics:

- Object Calisthenics são basicamente exercícios de programação, como um conjunto de 9 regras para ajudar a escrever melhor o código orientado a objetos.
- Conhecer as regras de Object Calisthenics
- Simplificar classes e métodos
- Manter o código mais curto e coeso

## Conteúdos

- **Artigo** Desenvolva um código melhor com Object Calisthenics ([https://medium.com/@rafaelcruz\\_48213/desenvolva-um-c%C3%B3digo-melhor-com-object-calisthenics-d5364767a9ba](https://medium.com/@rafaelcruz_48213/desenvolva-um-c%C3%B3digo-melhor-com-object-calisthenics-d5364767a9ba))
- **Artigo** Object Calisthenics de Jeff Bay (<https://dev.to/andrefilipp3/object-calisthenics-from-jeff-bay-501f>)
- **YouTube** Dias de Dev: Object Calisthenics - Apenas 1 nível de indentação por método ([https://www.youtube.com/watch?v=u\\_bg3k82uCE](https://www.youtube.com/watch?v=u_bg3k82uCE))
- **YouTube** Dias de Dev: Nunca use ELSE (<https://www.youtube.com/watch?v=z4fxFU5MoAA>)

## Conteúdos Alura:

- **Curso** Curso Object Calisthenics: exercitando a Orientação a Objetos (<https://cursos.alura.com.br/course/object-calisthenics-exercitando-orientacao-objetos>)

## Arquitetura de Microserviços:

- Microserviços são uma abordagem de arquitetura na qual o software consiste de pequenos serviços independentes que se comunicam entre si e são organizados de acordo com seus domínios de negócio.
- Aprender o conceito de arquitetura planejada para microserviços
- Realizar a comunicação usando APIs
- Melhorar a escalabilidade de um sistema

## Conteúdos

- **Artigo** Google Cloud: O que é a arquitetura de microsserviços?  
(<https://cloud.google.com/learn/what-is-microservices-architecture?hl=pt-br>)
- **Artigo** Amazon: O que são microsserviços?  
(<https://aws.amazon.com/pt/microservices/>)
- **YouTube** Otavio Lemos: Microsserviços | O mínimo que você precisa saber  
(<https://www.youtube.com/watch?v=eN8dFfTrEtQ>)
- **YouTube** Douglas Mugnos: O que é arquitetura de microsserviços? Como ela funciona? ([https://www.youtube.com/watch?v=\\_Oyy5PFOlcU](https://www.youtube.com/watch?v=_Oyy5PFOlcU))

## Conteúdos Alura:

- **Podcast** Hipsters.tech: TechGuide - Microsserviços – Hipsters Ponto Tech #349 (<https://www.hipsters.tech/techguide-microservicos-hipsters-ponto-tech-349/>)
- **YouTube** Alura: O que são Microsserviços?  
(<https://www.youtube.com/watch?v=jSnLOoGjQ80>)
- **Podcast** Hipsters.tech: Microserviços - Hipsters 17  
(<https://www.alura.com.br/podcast/microservicos-hipsters-17-a569>)
- **Podcast** Hipsters.tech: Microsserviços na Caelum - Hipsters On The Road 6  
(<https://www.alura.com.br/podcast/microservicos-na-caelum-hipsters-on-the-road-6-a436>)
- **Podcast** Hipsters.tech: Microsserviços autônomos na Usabilla - Hipsters On The Road 10 (<https://www.alura.com.br/podcast/microservicos-autonomos-na-usabilla-hipsters-on-the-road-10-a424>)
- **Site** Alura+: O que são Microsserviços?  
(<https://cursos.alura.com.br/extra/alura-mais/o-que-sao-microservicos--c699>)
- **Site** Alura+: Tipos de Microservices  
(<https://cursos.alura.com.br/extra/alura-mais/tipos-de-microservices-c698>)
- **Curso** Curso Microsserviços: padrões de projeto  
(<https://cursos.alura.com.br/course/microservicos-padroes-projeto>)

- **Curso** Curso Microserviços: explorando os conceitos (<https://cursos.alura.com.br/course/fundamentos-microservicos-aprofundando-conceitos>)
- **Curso** Curso Microserviços na prática: entendendo a tomada de decisões (<https://cursos.alura.com.br/course/Microservicos-pratica-tomada-decisoes>)

## Contêineres:

- Os contêineres são pacotes de software que contêm todos os elementos necessários para serem executados em qualquer ambiente. Gerenciamento de contêineres é uma área crucial na computação em nuvem e DevOps, que envolve o uso de tecnologias para automatizar o processo de criação, implantação, escalonamento e monitoramento de contêineres. Contêineres são unidades de software padronizadas que permitem aos desenvolvedores empacotar todas as dependências de um aplicativo (código, bibliotecas, configurações, etc.) em um único pacote. Isso permite que o aplicativo seja executado de forma consistente em qualquer ambiente de infraestrutura.
- A tecnologia de contêineres, como exemplificada pelo Docker, fornece um ambiente consistente e portátil para desenvolvimento, teste e implantação de aplicativos, o que é vital para o trabalho eficiente de engenharia de dados. Além disso, o Kubernetes, um sistema de orquestração de contêineres, permite o gerenciamento, a automação e a escalabilidade de aplicações baseadas em contêineres em ambientes de produção. Dominar esses conceitos e tecnologias possibilita a engenheiros de dados construir e manter pipelines de dados eficientes e confiáveis.
- O Kubernetes (também conhecido como k8s ou kube) é uma plataforma de orquestração de containers open source que automatiza grande parte dos processos manuais necessários para implantar, gerenciar e escalar aplicações em containers.
- Isolar seu software para funcionar independentemente
- Implantar software em clusters
- Modularizar seu sistema em pacotes menores
- Conhecer a plataforma Docker

- Conhecer Kubernetes

## Conteúdos

- **Site** IBM: Contêineres (<https://www.ibm.com/br-pt/cloud/learn/containers>)
- **Site** HPE: O que são contêineres? (<https://www.hpe.com/br/pt/what-is/containers.html>)
- **Site** Google Cloud: O que são contêineres? (<https://cloud.google.com/learn/what-are-containers?hl=pt-br>)
- **Site** Azure: O que é um contêiner? (<https://azure.microsoft.com/pt-br/resources/cloud-computing-dictionary/what-is-a-container/>)
- **Artigo** Red Hat: O que é Kubernetes (<https://www.redhat.com/pt-br/topics/containers/what-is-kubernetes>)
- **Artigo** Azure: O que é Kubernetes? (<https://azure.microsoft.com/pt-br/topic/what-is-kubernetes/>)
- **Artigo** Kubernetes - Guia de Referência Gratuito (<https://renatogroffe.medium.com/kubernetes-guia-de-refer%C3%A2ncia-gratuito-a2c1be8e7593>)
- **YouTube** Código Fonte TV: Containers // Dicionário do Programador (<https://www.youtube.com/watch?v=-pUZBovqRcU>)
- **YouTube** Código Fonte TV: Kubernetes // Dicionário do Programador (<https://www.youtube.com/watch?v=mVL0nOM3AGo>)
- **YouTube** Insight Lab: Revisão Docker - Curso de Introdução ao Kubernetes (<https://www.youtube.com/watch?v=bcRArpK00OU>)
- **YouTube** Pisani da Arch: Kubernetes - entenda e comece a arquitetar agora mesmo (<https://www.youtube.com/watch?v=XjaqTqm7g58>)

## Conteúdos Alura:

- **Artigo** O que é Kubernetes (<https://www.alura.com.br/artigos/o-que-e-kubernetes>)
- **Artigo** Criando volumes com Docker (<https://www.alura.com.br/artigos/criando-volumes-com-docker>)

- **Artigo** Docker Compose para compor uma aplicação (<https://www.alura.com.br/artigos/compondo-uma-aplicacao-com-o-docker-compose>)
- **YouTube** Alura: O que são Containers? ([https://www.youtube.com/watch?v=jv4\\_sLIGOS0](https://www.youtube.com/watch?v=jv4_sLIGOS0))
- **YouTube** Alura: Containers, Docker e Kubernetes (com Giovanni Bassi) (<https://www.youtube.com/watch?v=wxLvvMxzc1Q>)
- **Podcast** Hipsters.tech: Containers e Docker - Hipsters 75 (<https://www.alura.com.br/podcast/hipsterstech-containers-e-docker-hipsters-75-a511>)
- **Podcast** Hipsters.tech: Kubernetes - Hipsters 177 (<https://www.alura.com.br/podcast/kubernetes-hipsters-177-a389>)
- **Podcast** Hipsters.tech: Microsserviços e Kubernetes na Americanas.com - Hipsters On The Road 56 (<https://www.alura.com.br/podcast/microservicos-e-kubernetes-na-americanas-com-hipsters-on-the-road-56-a857>)
- **Curso** Formação Orquestração de containers com Kubernetes (<https://www.alura.com.br/formacao-kubernetes>)
- **Curso** Curso Docker: criando e gerenciando containers (<https://www.alura.com.br/curso-online-docker-criando-gerenciando-containers>)
- **Curso** Curso Kubernetes: Pods, Services e ConfigMaps (<https://www.alura.com.br/curso-online-kubernetes-pods-services-configmap>)
- **Curso** Curso Kubernetes: Deployments, Volumes e Escalabilidade (<https://www.alura.com.br/curso-online-kubernetes-deployments-volumes-escalabilidade>)
- **Curso** Curso Google Kubernetes Engine (<https://www.alura.com.br/curso-online-google-kubernetes-engine-primeiros-passos>)
- **Curso** Curso Kubernetes na Digital Ocean: gerenciando aplicações containerizadas (<https://www.alura.com.br/curso-online-kubernetes-digital-ocean>)

- **Curso** Curso Amazon EKS: gerenciando aplicações containerizadas com Kubernetes (<https://www.alura.com.br/curso-online-amazon-eks-kubernetes>)
- **Curso** Curso Azure: gerencie Kubernetes com AKS e ACR (<https://www.alura.com.br/curso-online-azure-gerencie-kubernetes-com-aks-acr>)

## Reflection e atributos:

- Os objetos de Reflection (reflexão) são usados para obter informações do tipo em tempo de execução. As classes que dão acesso aos metadados de um programa em execução estão no namespace System.Reflection.
- Escrever código que lê as informações e metadados de objetos em tempo de execução
- Obter nomes de classes em tempo de execução e criar objetos de uma classe

## Conteúdos

- **Site** Microsoft Docs: Reflection (C#) (<https://docs.microsoft.com/pt-br/dotnet/csharp/programming-guide/concepts/reflection>)
- **Site** Acervo Lima: Reflexão em Python (<https://acervolima.com/reflexao-em-python/>)
- **Artigo** Reflection in C# (inglês) (<https://medium.com/@ghadeer.kenawi/reflection-in-c-fa27d8f827f8>)
- **Artigo** Reflection em Javascript (<https://medium.com/trainingcenter/reflection-em-javascript-73fc0e702e2>)
- **YouTube** Marco Diniz: C# em 5 minutos - Reflection ([https://www.youtube.com/watch?v=EX\\_0lf5zyrU](https://www.youtube.com/watch?v=EX_0lf5zyrU))
- **YouTube** Kerplunk: C# Orientação a Objeto - Reflection (<https://www.youtube.com/watch?v=fl1ChSv2yqI>)
- **YouTube** Dev Coffee BR: C# Orientação a Objeto - Reflection (<https://www.youtube.com/watch?v=XkgeNMvZyzo>)



## Conteúdos Alura:

- **Curso** Curso Java Reflection: simplifique a conversão de objetos (<https://cursos.alura.com.br/course/java-reflection-simplifique-conversao-objetos>)
- **Curso** Curso C# Reflection: manipule dinamicamente tipos e assemblies (<https://cursos.alura.com.br/course/csharp-reflection-manipule-dinamicamente-tipos-assemblies>)

## Habilidade Auxiliar: Infraestrutura e boas práticas

### Git e GitHub - Fundamentos:

- Git é um sistema de controle de versão distribuído gratuito e de código aberto projetado para lidar com tudo, desde projetos pequenos a muito grandes com velocidade e eficiência.
- GitHub é um serviço de hospedagem para desenvolvimento de software e controle de versão usando Git.
- Criar um repositório
- Clonar um repositório
- Fazer commit, push e pull de e para o repositório
- Reverter um commit
- Criar branches e pull requests
- Lidar com merge e conflitos

### Conteúdos

- **Site** GitHub Documentação (<https://docs.github.com/pt>)
- **Site** GitHub Pages Documentação (<https://docs.github.com/pt/pages/getting-started-with-github-pages/about-github-pages>)
- **Site** Git School - Visualizing Git (<https://git-school.github.io/visualizing-git/>)
- **Site** Dangit, Git!?! (<https://dangitgit.com/>)

- **YouTube** Rafaella Ballerini: O que é Git e GitHub? - definição e conceitos importantes 1/2 (<https://www.youtube.com/watch?v=DqTITcMq68k>)
- **YouTube** Rafaella Ballerini: Como usar Git e GitHub na prática! - desde o primeiro commit até o pull request! 2/2 (<https://www.youtube.com/watch?v=UBAX-13g8OM>)
- **YouTube** Mario Souto - Dev Soutinho: Git: Entendendo de vez como funciona do melhor e mais visual jeito possível (<https://www.youtube.com/watch?v=4-tfJ-ZyA0Q>)
- **YouTube** Mario Souto - Dev Soutinho: Como colocar seu projeto no ar DE GRAÇA via GitHub! | Hospedagem com GitHub Pages ([https://www.youtube.com/watch?v=BU-w2\\_Aae54](https://www.youtube.com/watch?v=BU-w2_Aae54))
- **YouTube** CodandoTV(Rods) - 5 coisas que você precisa saber sobre Git ([https://youtu.be/MqogPzjQyCY?si=VL\\_mT8EowuemptmU](https://youtu.be/MqogPzjQyCY?si=VL_mT8EowuemptmU))
- **YouTube** CodandoTV(Rods) - README de Sucesso: Transforme seu Projeto ou seu Perfil em Destaque no GitHub ([https://youtu.be/v9ZM2PVzctM?si=49ah\\_HRPlwpSEX4A](https://youtu.be/v9ZM2PVzctM?si=49ah_HRPlwpSEX4A))

### Conteúdos Alura:

- **Artigo** Git e Github: O que são, Como Configurar e Primeiros Passos (<https://www.alura.com.br/artigos/o-que-e-git-github>)
- **Artigo** Mais git com o hub: a linha de comando do Github (<https://www.alura.com.br/artigos/github-na-linha-de-comando>)
- **Podcast** Hipsters 184: Guia do Iniciante em Github (<https://cursos.alura.com.br/extra/hipsterstech/guia-do-iniciante-em-github-hipsters-184-a378>)
- **Site** GitHub: diferentes maneiras de compartilhar seu projeto (<https://cursos.alura.com.br/extra/alura-mais/github-diferentes-maneiras-de-compartilhar-seu-projeto-c2002>)
- **Site** Websérie: Git e Github para Sobrevivência (<https://www.alura.com.br/webseries/git-e-github-para-sobrevivencia>)
- **Podcast** Hipsters 109: Git e Github (<https://www.alura.com.br/podcast/hipsterstech-git-e-github-hipsters-109->)

[a474](#))

- **YouTube** Alura: Git e Github para Sobrevivência 01: Como o Git funciona? (<https://www.youtube.com/watch?v=BAmvmaKQklQ>)
- **Curso** Curso Git e GitHub: compartilhando e colaborando em projetos (<https://cursos.alura.com.br/course/git-github-compartilhando-colaborando-projetos>)
- **Curso** Curso Git e GitHub: dominando controle de versão de código (<https://cursos.alura.com.br/course/git-github-dominando-controle-versao-codigo>)
- **Desafio** 7 Days of Code: GitHub (<https://7daysofcode.io/matricula/github>)

## HTTP - Fundamentos:

- HTTP significa Hyper Text Transfer Protocol. A comunicação entre computadores cliente e servidores web é feita enviando solicitações HTTP e recebendo respostas HTTP.
- Entender a diferença dos verbos HTTP
- Testar os requests e ver os status codes no navegador
- Saber fazer uma requisição HTTP na linha de comando com WGET
- Baixar uma imagem com WGET
- Fazer um post

## Conteúdos

- **Site** MDN Web Docs: Uma visão geral do HTTP (<https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Overview>)
- **Site** MDN Web Docs: Métodos de requisição HTTP (<https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Methods>)
- **Site** HTTP Cats (<https://http.cat/>)
- **Site** HTTP Dogs (<https://http.dog/>)
- **YouTube** Fabiano Gabardo Lemos: Requisições HTTP - GET, POST, PUT, PATCH DELETE (<https://www.youtube.com/watch?v=kncOJZrnkTg>)

- **YouTube** Programador a Bordo: Protocolo HTTP e TCP/IP - Introdução (<https://www.youtube.com/watch?v=V4XZ81vRGtM>)

### Conteúdos Alura:

- **Artigo** HTTP: Desmistificando o protocolo da Web (<https://www.alura.com.br/artigos/desmistificando-o-protocolo-http-parte-1>)
- **Artigo** Métodos de requisição do HTTP (<https://www.alura.com.br/artigos/metodos-de-requisicao-do-http>)
- **Artigo** Qual é a diferença entre HTTP e HTTPS? (<https://www.alura.com.br/artigos/qual-e-diferenca-entre-http-e-https>)
- **Podcast** Hipsters.tech: HTTP/2: magia com o novo protocolo - Hipsters 13 (<https://www.alura.com.br/podcast/http-2-magia-com-o-novo-protocolo-hipsters-13-a573>)
- **Curso** Curso HTTP: Entendendo a web por baixo dos panos (<https://www.alura.com.br/curso-online-http-entendendo-web-por-baixo-dos-panos>)

### JSON:

- JSON significa JavaScript Object Notation (notação de objeto JavaScript). É um formato de texto para armazenar e transmitir dados.
- Criar um objeto
- Transformar um objeto em uma string
- Transformar uma string em objeto
- Manipular um objeto

### Conteúdos

- **Site** MDN Web Docs: JSON ([https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global\\_Objects/JSON](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Global_Objects/JSON))
- **Site** MDN Web Docs: Trabalhando com JSON (<https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/Objects/JSON>)

- **Artigo** Introdução ao JSON (<https://juliocarneiro.medium.com/introdu%C3%A7%C3%A3o-ao-json-7825b1a550ff>)
- **YouTube** Marco Bruno: O que é JSON e como criar um objeto (<https://www.youtube.com/watch?v=oCY5YEEjlwE>)

### Conteúdos Alura:

- **Artigo** O que é JSON? (<https://www.alura.com.br/artigos/o-que-e-json>)
- **Curso** Curso MySQL e JSON: persistindo JSON de maneira eficiente (<https://www.alura.com.br/curso-online-mysql-json-persistencia>)

### Linha de comando - Fundamentos:

- CLI é um programa de linha de comando que aceita entradas de texto para executar funções do sistema operacional.
- Conhecer os principais comandos

### Conteúdos

- **Site** Microsoft Docs: Como usar argumentos de linha de comando para terminal do Windows (<https://docs.microsoft.com/pt-br/windows/terminal/command-line-arguments?tabs=windows>)
- **YouTube** Estevan Maito: Linha de comando básica - 8 comandos mais frequentes - WINDOWS e LINUX (<https://www.youtube.com/watch?v=rKBqXJZocYk>)
- **YouTube** Ninja do Linux: Comandos básicos da linha de comando do Linux ([https://www.youtube.com/watch?v=rs\\_yshFGu8E](https://www.youtube.com/watch?v=rs_yshFGu8E))

### Conteúdos Alura:

- **Artigo** CMD: dicas para trabalhar no prompt do Windows (<https://www.alura.com.br/artigos/cmd-dicas-para-trabalhar-no-prompt-do-windows>)
- **Artigo** Como configurar variáveis de ambiente no Windows, Linux e macOS (<https://www.alura.com.br/artigos/configurar-variaveis-ambiente-windows-linux-macos>)

- **Site** Primeiras aulas do Curso Windows Prompt: Trabalhando na linha de comando (<https://www.alura.com.br/conteudo/windows-prompt-utilizando-cmd>)
- **Curso** Curso Windows Prompt: Trabalhando na linha de comando (<https://www.alura.com.br/curso-online-prompt>)
- **Curso** Curso Terminal: aprenda comandos para executar tarefas (<https://cursos.alura.com.br/course/terminal-comandos-executar-tarefas>)

## Cloud - Fundamentos:

- Cloud, ou computação em nuvem é a distribuição de serviços de computação pela Internet usando um modelo de preço pago conforme o uso. Uma nuvem é composta de vários recursos de computação, que abrangem desde os próprios computadores (ou instâncias, na terminologia de nuvem) até redes, armazenamento, bancos de dados e o que estiver em torno deles. Ou seja, tudo o que normalmente é necessário para montar o equivalente a uma sala de servidores, ou mesmo um data center completo, estará pronto para ser utilizado, configurado e executado.
- Conhecer a diferença entre IaaS, PaaS e SaaS
- Conhecer os maiores provedores de cloud
- Especializar-se em algum provedor

## Conteúdos

- **Artigo** Computação em nuvem (<https://medium.com/sysadminas/computa%C3%A7%C3%A3o-em-nuvem-515930304cf9>)
- **Artigo** O que é cloud? (<https://gabriel-faraday.medium.com/o-que-%C3%A9-cloud-991109e708c6>)
- **YouTube** Gabs Ferreira: Por que investir e estudar cloud? (<https://www.youtube.com/watch?v=Z45BTNeZ1I0>)
- **YouTube** Andre Iacono: O que é MICROSOFT AZURE? Qual Certificação começar em 2022? (<https://www.youtube.com/watch?v=f-oVzkvMwnE>)

- **YouTube** AWS: O que é a AWS? (<https://www.youtube.com/watch?v=8JI9wQ8sUdQ>)
- **YouTube** O que é Google Cloud e por que aprender? (<https://www.youtube.com/shorts/Lzq3f1DHWcl>)
- **Artigo** AWS vs Google Cloud vs Azure: o que cada um tem de melhor? (<https://medium.com/data-hackers/aws-vs-google-cloud-vs-azure-o-que-cada-um-tem-de-melhor-52107174f7b7>)
- **YouTube** Código Fonte TV: Azure (A plataforma Cloud da Microsoft) (<https://www.youtube.com/watch?v=YgE-sZaCuJ0>)
- **YouTube** Mundo da Cloud: AWS do Zero ao Expert (<https://www.youtube.com/watch?v=HiBCv9DolxI&list=PLtL97Owd1gkQ0dfqGW8OtJ-155Gs67Ecz>)

### Conteúdos Alura:

- **Podcast** Hipsters.tech: TechGuide - Fundamentos Cloud – Hipsters Ponto Tech #348 (<https://www.hipsters.tech/techguide-fundamentos-cloud-hipsters-ponto-tech-348/>)
- **Artigo** Cloud: o que é, História e Guia da computação em nuvem (<https://www.alura.com.br/artigos/cloud>)
- **Podcast** Hipsters.tech: Uma jornada Para o Cloud - Hipsters Deep Dive 005 (<https://www.alura.com.br/podcast/uma-jornada-para-o-cloud-hipsters-deep-dive-005-a1100>)
- **Podcast** Hipsters.tech: Histórias do Cloud - Hipsters 04 (<https://www.alura.com.br/podcast/historias-do-cloud-hipsters-04-a582>)
- **Artigo** Heroku, Vercel e outras opções de cloud como plataforma (<https://www.alura.com.br/artigos/heroku-vercel-outras-opcoes-cloud-plataforma>)
- **Artigo** AWS: Guia sobre o que é Amazon Web Services, seus Serviços e Certificações (<https://www.alura.com.br/artigos/aws>)
- **Artigo** Terraform: criando máquinas na Azure (<https://www.alura.com.br/artigos/terraform-maquinas-na-azure>)

- **YouTube** Alura: O que é cloud? (<https://www.YOUTUBE.com/watch?v=wev9fMrg-TU>)
- **YouTube** Alura: AWS, Google Cloud e Azure: Por onde começar? | Hipsters.Talks (<https://www.YOUTUBE.com/watch?v=z9k6rsdmWc0&t=300s>)
- **YouTube** Alura: Certificação em Cloud: Azure, AWS, Google | Hipsters.Talks ([https://www.YOUTUBE.com/watch?v=W4K82n\\_WK5g&t=290s](https://www.YOUTUBE.com/watch?v=W4K82n_WK5g&t=290s))
- **Curso** Formação Começando em Cloud Computing (<https://cursos.alura.com.br/formacao-cloud-computing>)
- **Curso** Formação Amazon Web Services (<https://cursos.alura.com.br/formacao-amazon-web-services>)
- **Curso** Formação Google Certified Associate Cloud Engineer (<https://cursos.alura.com.br/formacao-google-certified-associate-cloud-engineer>)
- **Curso** Formação Certificação AWS Certified Cloud Practitioner (<https://cursos.alura.com.br/formacao-aws-certified-cloud-practitioner>)
- **Curso** Formação Containers com AWS ECS e EKS (<https://cursos.alura.com.br/formacao-containers-aws>)
- **Curso** Formação Google Cloud Platform (<https://cursos.alura.com.br/formacao-google-cloud>)
- **Curso** Formação Certificação Google Certified Associate Cloud Engineer (<https://cursos.alura.com.br/formacao-google-certified-associate-cloud-engineer>)
- **Curso** Formação Azure (<https://cursos.alura.com.br/formacao-conhecendo-azure>)
- **Curso** Formação Certificação AZ-900: Microsoft Azure Fundamentals (<https://cursos.alura.com.br/formacao-certificacao-az-900-microsoft-azure-fundamentals>)

## SQL - Fundamentos:

- SQL (Structured Query Language, traduzindo, Linguagem de Consulta Estruturada) é uma linguagem de programação padronizada que é usada



para gerenciar bancos de dados relacionais e realizar várias operações sobre os dados neles contidos.

- Conhecer os comandos mais comuns do SQL
- Usar SELECT para consultar uma tabela
- Usar INSERT para inserir dados em uma tabela
- Usar UPDATE para atualizar uma tabela
- Usar DELETE para remover dados de uma tabela
- Usar JOIN para conectar os dados de múltiplas tabelas
- Conhecer as cláusulas (FROM, ORDER BY, etc)

## Conteúdos

- **Artigo** SQL — O que é e como funciona na prática? (<https://medium.com/pravaler-digital-team/sql-o-que-%C3%A9-e-como-funciona-na-pr%C3%A1tica-6ae7a322554a>)
- **Artigo** SQL: comandos básicos (<https://medium.com/nerdzao/sql-comandos-b%C3%A1sicos-483a8dc12909>)
- **YouTube** CreateSe: Comandos básicos do SQL ([https://www.youtube.com/watch?v=\\_rZxrqn0X-4](https://www.youtube.com/watch?v=_rZxrqn0X-4))

## Conteúdos Alura:

- **Artigo** O que é SQL? (<https://www.alura.com.br/artigos/o-que-e-sql>)
- **Artigo** SQL JOIN: Aprenda INNER, LEFT, RIGHT, FULL e CROSS (<https://www.alura.com.br/artigos/join-em-sql>)
- **YouTube** Alura: O que é SQL e NoSQL? (<https://www.youtube.com/watch?v=aure5d3B88g>)
- **Curso** Formação Conhecendo SQL (<https://www.alura.com.br/formacao-conhecendo-sql>)
- **Curso** Curso SQL com MySQL: manipule e consulte dados (<https://cursos.alura.com.br/course/mysql-manipule-dados-com-sql>)
- **Curso** Formação SQL com MySQL Server da Oracle (<https://cursos.alura.com.br/formacao-oracle-mysql>)

- **Artigo** SQL e NoSQL: trabalhando com bancos relacionais e não relacionais (<https://www.alura.com.br/artigos/sql-nosql-bancos-relacionais-nao-relacionais>)
- **Curso** Microsoft SQL Server 2022: conhecendo SQL (<https://cursos.alura.com.br/course/microsoft-sql-server-conhecendo-sql>)
- **Curso** Curso de PostgreSQL (<https://cursos.alura.com.br/course/introducao-postgresql-primeiros-passos>)
- **Curso** SQL com Oracle Database: manipule e consulte dados (<https://cursos.alura.com.br/course/introducao-sql-oracle-manipule-consulte-dados>)
- **Podcast** SQL e MySQL (<https://cursos.alura.com.br/extra/hipsterstech/sql-e-mysql-hipsters-ponto-tech-404-a9270>)

## SOLID:

- O Solid possui cinco princípios considerados como boas práticas no desenvolvimento de software que ajudam os programadores a escrever os códigos mais limpos, separando as responsabilidades, diminuindo acoplamentos, facilitando na refatoração e estimulando o reaproveitamento do código.

## Conteúdos

- **Artigo** Princípios de S.O.L.I.D em C# — Guia prático (<https://medium.com/beelabacademy/princ%C3%ADpios-de-s-o-l-i-d-em-c-guia-pr%C3%A1tico-cbb1e6584284>)
- **YouTube** Filipe Deschamps: SOLID fica fácil com Essas Ilustrações (<https://www.youtube.com/watch?v=6SfrO3D4dHM>)
- **YouTube** Central dotNET: Princípios de SOLID com C# (<https://www.youtube.com/watch?v=iU4BMUcjg8g>)
- **YouTube** Felipe Pinheiro - Tech: Princípios SOLID, o que é DIP? (<https://www.youtube.com/watch?v=q0BGgQJcp7w>)
- **YouTube** Dev Eficiente: SOLID - Uma reflexão sobre o Princípio da responsabilidade única ([https://www.youtube.com/watch?v=GGe0o\\_v5vjM](https://www.youtube.com/watch?v=GGe0o_v5vjM))

## Conteúdos Alura:

- **Artigo** SOLID: o que é e quais os 5 princípios da Programação Orientada a Objetos (POO) (<https://www.alura.com.br/artigos/solid>)
- **Podcast** SOLID: Código bom e bonito - Hipsters Ponto Tech 219 (<https://www.alura.com.br/podcast/hipsterstech-solid-codigo-bom-e-bonito-hipsters-ponto-tech-219-a649>)
- **YouTube** Alura: Clean Code e Solid ([https://www.youtube.com/watch?v=XV9B4LX\\_re8](https://www.youtube.com/watch?v=XV9B4LX_re8))
- **Curso** Curso C#: aplique princípios SOLID (<https://cursos.alura.com.br/course/csharp-aplique-principios-solid>)
- **Curso** Curso Boas práticas de programação: melhore o código de uma API Java (<https://cursos.alura.com.br/course/boas-praticas-programacao-melhore-codigo-api-java>)
- **Curso** Curso SOLID com PHP: princípios da programação orientada a objetos (<https://cursos.alura.com.br/course/solid-php-principios-orientacao-a-objetos>)
- **Curso** Curso iOS: escrevendo código de qualidade com SOLID em Swift (<https://www.alura.com.br/curso-online-ios-escrevendo-codigo-qualidade-solid-swift>)
- **Curso** SOLID com TypeScript: aplicando boas práticas em orientação a objetos (<https://www.alura.com.br/curso-online-solid-typescript-boas-praticas-orientacao-objetos>)
- **Curso** Formação Boas práticas em C# (<https://cursos.alura.com.br/formacao-boas-praticas-c-sharp>)
- **Curso** Formação Boas práticas em PHP (<https://cursos.alura.com.br/formacao-boas-praticas-php>)
- **Curso** Formação Boas práticas em Java (<https://cursos.alura.com.br/formacao-boas-praticas-java>)
- **Site** Livro Casa do Código: Desbravando SOLID - Práticas avançadas para códigos de qualidade em Java moderno (<https://www.casadocodigo.com.br/products/livro-desbravando-solid>)

- **YouTube** CodandoTV(Rods): SOLID e o 'D' do Principio da Inversão de Dependência usando a Injeção de Dependência para ajudar (<https://youtu.be/xpcCDCH5k4Y>)

## Design Patterns:

- Na engenharia de software, um "padrão de projeto" (Design Pattern em inglês) é uma solução geral e reutilizável para um problema que ocorre normalmente dentro de um determinado contexto de projeto de software.
- Conhecer e aplicar os principais Design Patterns

## Conteúdos

- **Artigo** Design Patterns — O que são e quais os benefícios? (<https://djesusnet.medium.com/design-patterns-gof-o-que-s%C3%A3o-e-quais-os-benef%C3%ADcios-9cd0cfdd6ebf>)
- **Artigo** Design Patterns (<https://medium.com/xp-inc/design-patterns-727494af001d>)
- **YouTube** Matheus Castiglioni: O que são design patterns (padrões de projetos) e para que servem? (<https://www.YOUTUBE.com/watch?v=KNWd1aWtsMw>)
- **YouTube** RinaldoDev: Design Patterns - o que são? Por que aprender a utilizar? (<https://www.YOUTUBE.com/watch?v=AWOf6Wo6gtg>)

## Conteúdos Alura:

- **Podcast** Hipsters.tech: TechGuide - Design Patterns – Hipsters Ponto Tech #347 (<https://www.hipsters.tech/techguide-design-patterns-hipsters-ponto-tech-347/>)
- **Artigo** Design patterns: Breve introdução aos padrões de projeto (<https://www.alura.com.br/artigos/design-patterns-introducao-padroes-projeto>)
- **Podcast** Hipsters.tech: Design Patterns - Hipsters 206 (<https://www.alura.com.br/podcast/design-patterns-hipsters-206-a345>)
- **Curso** Curso Design Patterns em Java I: boas práticas de programação (<https://www.alura.com.br/curso-online-introducao-design-patterns-java>)

- **Curso** Curso Design Patterns C# I: boas práticas de programação (<https://www.alura.com.br/curso-online-design-patterns-dotnet>)
- **Curso** Curso Design Patterns em PHP: padrões comportamental (<https://www.alura.com.br/curso-online-php-design-pattern-comportamental>)
- **Curso** Curso Design Patterns em PHP: padrões estruturais (<https://www.alura.com.br/curso-online-php-design-pattern-estrutural>)
- **Curso** Curso Design Patterns em PHP: padrões criacionais (<https://www.alura.com.br/curso-online-php-design-pattern-criacional>)

## Clean Architecture:

- A Clean Architecture (Arquitetura Limpa) é uma forma de desenvolver software, de tal forma que apenas olhando para o código fonte de um programa, você deve ser capaz de dizer o que o programa faz.

## Conteúdos

- **Artigo** Descomplicando a Clean Architecture (<https://medium.com/luizalabs/descomplicando-a-clean-architecture-cf4dfc4a1ac6>)
- **YouTube** Como DEV ser!: Entenda CLEAN ARCHITECTURE de uma vez por todas! (<https://www.youtube.com/watch?v=HynTfTli4mw>)
- **YouTube** Full Cycle: O que é Clean Architecture (<https://www.youtube.com/watch?v=dQys-mnOtQg>)
- **YouTube** CodandoTV(Rods) - Simplificando o Clean Architecture +MVVM na sua aplicação mobile - Guia Completo (<https://youtu.be/8ehIZfyN1S0?si=I7-I5l4zsnYLI5jD>)

## Conteúdos Alura:

- **Site** Clean Architecture (Arquitetura Limpa) - O que é? (<https://cursos.alura.com.br/extra/alura-mais/clean-architecture-arquitetura-limpa-o-que-e--c204>)
- **Podcast** Hipsters.tech: Clean Architecture - Hipsters Ponto Tech 254 (<https://www.hipsters.tech/clean-architecture-hipsters-ponto-tech-254/>)

- **YouTube** Alura: Arquitetura de sistemas (<https://www.youtube.com/watch?v=oedWxgAZc2A>)
- **Site** Primeiras aulas do curso Java e Clean Architecture: descomplicando arquitetura de software (<https://www.alura.com.br/conteudo/java-clean-architecture>)
- **Curso** Curso Java e Clean Architecture: descomplicando arquitetura de software (<https://www.alura.com.br/curso-online-java-clean-architecture>)
- **Curso** Curso PHP e Clean Architecture: descomplicando arquitetura de software (<https://cursos.alura.com.br/course/php-introducao-clean-architecture>)

## Conceitos de Design Orientado a Domínio (Domain-Driven Design - DDD):

- O Design Orientado a Domínio (DDD) é uma abordagem ao projeto e desenvolvimento de software que é primeiramente informado pelos requisitos de negócios. Os componentes do programa (objetos, classes, matrizes, etc.) indicam a indústria, setor ou domínio empresarial em que o negócio opera.
- Modelar domínios de forma efetiva
- Basear projetos complexos em modelos do domínio
- Conhecer os blocos de construção de DDD

## Conteúdos

- **Artigo** Domain-Driven Design, do início ao código (<https://medium.com/cwi-software/domain-driven-design-do-in%C3%ADcio-ao-c%C3%B3digo-569b23cb3d47>)
- **Artigo** Introdução ao DDD (<https://medium.com/xp-inc/introdu%C3%A7%C3%A3o-ao-ddd-2cad38e4234a>)
- **YouTube** Dias de Dev: O que é Domain-Driven Design (<https://www.youtube.com/watch?v=SNdGV0c40yo>)
- **YouTube** Otavio Lemos: DDD + Clean Architecture (<https://www.youtube.com/watch?v=cZuyPPrRGAw>)

## Conteúdos Alura:

- **YouTube** Alura: Domain Driven Design (<https://www.youtube.com/watch?v=n40Z1c9Ryog>)
- **Site** Domain-Driven Design (DDD) - O que é? (<https://cursos.alura.com.br/extra/alura-mais/domain-driven-design-ddd-o-que-e--c283>)
- **Site** Primeiras aulas do curso Java e Domain Driven Design: apresentando os conceitos (<https://www.alura.com.br/conteudo/java-domain-driven-design-conceitos>)
- **Curso** Curso Java e Domain Driven Design: apresentando os conceitos (<https://www.alura.com.br/curso-online-java-domain-driven-design-conceitos>)
- **Site** Primeiras aulas do curso PHP e Domain Driven Design: apresentando os conceitos (<https://www.alura.com.br/conteudo/domain-driven-design-php>)
- **Curso** Curso PHP e Domain Driven Design: apresentando os conceitos (<https://www.alura.com.br/curso-online-domain-driven-design-php>)

## Habilidade Auxiliar: Front-end

### HTML - Fundamentos:

- HTML é uma linguagem de marcação que define a estrutura do seu conteúdo. HTML consiste em uma série de elementos que você usa para mostrar algo de uma determinada maneira ou agir de uma certo modo. As tags podem criar um hiperlink de uma palavra ou imagem para outro lugar, podem colocar palavras em itálico, podem aumentar ou diminuir a fonte e assim por diante.
- Aprender quais tags são necessárias para um HTML básico
- Criar um parágrafo de texto
- Exibir uma imagem
- Conhecer a diferença entre 'h1', 'h2', 'h3', etc

- Criar um texto com hyperlink
- Criar um formulário com campos relevantes
- Criar uma lista de itens ordenada ou não ordenada
- Criar uma lista de itens dentro de uma lista suspensa (dropdown list)
- Conectar com um arquivo de CSS
- Criar uma tabela
- Adicionar IDs e classes

## Conteúdos

- **Site** MDN Web Docs: HTML básico ([https://developer.mozilla.org/pt-BR/docs/Learn/Getting\\_started\\_with\\_the\\_web/HTML\\_basics](https://developer.mozilla.org/pt-BR/docs/Learn/Getting_started_with_the_web/HTML_basics))
- **YouTube** Rafaella Ballerini: 5 MINUTOS DE HTML PARA INICIAR EM PROGRAMAÇÃO! (<https://www.youtube.com/watch?v=3oSlqlqzN3M>)
- **YouTube** Rafaella Ballerini: LANDING PAGE COM HTML e CSS! (<https://www.youtube.com/watch?v=IIF6vD-RlJE>)
- **YouTube** Rafaella Ballerini: FORMULÁRIOS COM HTML e CSS! (<https://www.youtube.com/watch?v=wwqOJ2o84S4>)

## Conteúdos Alura:

- **Artigo** HTML: o que é, a importância para a Web, como aprender e um Guia para iniciantes (<https://www.alura.com.br/artigos/html>)
- **Alura+** Alura+: Fazendo deploy de um projeto na Vercel (<https://cursos.alura.com.br/extra/alura-mais/fazendo-deploy-de-um-projeto-na-vercel-c9239>)
- **Artigo** HTML, CSS e Javascript, quais as diferenças? (<https://www.alura.com.br/artigos/html-css-e-js-definicoes>)
- **Curso** Formação HTML e CSS (<https://www.alura.com.br/formacao-html-e-css>)
- **Curso** Apostila: Desenvolvimento Web com HTML, CSS e JavaScript (<https://www.alura.com.br/apostila-html-css-javascript>)



- **Desafio** 7 Days of Code: HTML e CSS  
(<https://7daysofcode.io/matricula/html-css>)

## CSS - Fundamentos:

- Cascading Style Sheets (CSS) é uma linguagem usada para descrever a apresentação de um documento escrito em uma linguagem de marcação como HTML ou XML. CSS pode ser usado para estilos de texto de documentos muito básicos — por exemplo, para alterar a cor e o tamanho de títulos e links. Ele pode ser usado para criar um layout — por exemplo, transformar uma única coluna de texto em um layout com uma área de conteúdo principal e uma barra lateral para informações relacionadas. Pode até ser usado para efeitos como animações.
- Aprender a estrutura visual de uma página, com 'margin' e 'padding'
- Estabelecer o tamanho com 'width' e 'height'
- Aprender sobre a posição de um elemento ('static', 'relative' ou 'absolute')
- Aprender sobre o 'display' de exibição de um elemento ('block', 'inline', 'inline-block')
- Aprender a posicionar imagens em relação ao texto
- Aprender sobre alinhamento
- Aprender sobre estilo de fontes
- Aprender as diferenças e vantagens de usar as diferentes unidades de medida em CSS (% , relativas, etc)
- Conectar com os elementos (IDs, classes) de um arquivo HTML
- Alterar características de um elemento quando o mouse passar por cima dele ('hover')
- Aprender box-sizing
- Aprender Flexbox
- Aprender Grid

## Conteúdos

- **Site** MDN Web Docs: Iniciando com CSS ([https://developer.mozilla.org/pt-BR/docs/Learn/CSS/First\\_steps/Getting\\_started](https://developer.mozilla.org/pt-BR/docs/Learn/CSS/First_steps/Getting_started))
- **Site** MDN Web Docs: Seletores ([https://developer.mozilla.org/pt-BR/docs/Learn/CSS/Building\\_blocks/Selectors](https://developer.mozilla.org/pt-BR/docs/Learn/CSS/Building_blocks/Selectors))
- **Site** MDN Web Docs: Cascata e herança ([https://developer.mozilla.org/pt-BR/docs/Learn/CSS/Building\\_blocks/Cascade\\_and\\_inheritance](https://developer.mozilla.org/pt-BR/docs/Learn/CSS/Building_blocks/Cascade_and_inheritance))
- **Site** MDN Web Docs: Propriedade 'display' (<https://developer.mozilla.org/pt-BR/docs/Web/CSS/display>)
- **Site** MDN Web Docs: Conceitos básicos de Flexbox ([https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS\\_Flexible\\_Box\\_Layout/Basic\\_Concepts\\_of\\_Flexbox](https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_Flexible_Box_Layout/Basic_Concepts_of_Flexbox))
- **Site** MDN Web Docs: Grid Layout ([https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS\\_Grid\\_Layout/Basic\\_Concepts\\_of\\_Grid\\_Layout](https://developer.mozilla.org/pt-BR/docs/Web/CSS/CSS_Grid_Layout/Basic_Concepts_of_Grid_Layout))
- **Artigo** Entendendo CSS: Display Inline, Block e Inline-Block (<https://dev.to/sucodelarangela/entendendo-css-display-inline-block-e-inline-block-lic>)
- **Artigo** Entendendo CSS: Pseudo-Classes e Pseudo-Elementos (<https://dev.to/sucodelarangela/entendendo-css-pseudo-classes-e-pseudo-elementos-b83>)
- **Artigo** Como aplicar opacidade em background-image sem afetar textos (<https://dev.to/sucodelarangela/como-aplicar-opacidade-em-background-image-sem-afetar-textos-31fj>)
- **YouTube** Rafaella Ballerini: O QUE É CSS? (<https://www.youtube.com/watch?v=LWU2OR19ZG4>)
- **YouTube** Rafaella Ballerini: FLEXBOX CSS! Como posicionar elementos na página web - parte 1 ([https://www.youtube.com/watch?v=KbjLtEgmZ\\_E](https://www.youtube.com/watch?v=KbjLtEgmZ_E))
- **YouTube** Rafaella Ballerini: FLEXBOX CSS! Como posicionar elementos na página web - parte 2 ([https://www.youtube.com/watch?v=hjz6ezV9\\_uc](https://www.youtube.com/watch?v=hjz6ezV9_uc))
- **YouTube** Matheus Castiglioni: Arquitetura CSS - BEM (<https://www.youtube.com/watch?v=yKPXW9aSxQI>)

- **YouTube** Matheus Castiglioni: CSS Grid Layout (<https://www.youtube.com/watch?v=HBIBNAtFcdw>)
- **YouTube** Mario Souto - Dev Soutinho: Como centralizar no CSS (<https://www.youtube.com/watch?v=Cu-HP-gvggg>)

### Conteúdos Alura:

- **Podcast** Hipster 09 - CSS: Cansei de Ser Simples (<https://www.hipsters.tech/css-cansei-de-ser-simples-hipsters-09/>)
- **Artigo** CSS: o que é, como usar no HTML e um Guia para iniciar (<https://www.alura.com.br/artigos/css>)
- **Artigo** HTML, CSS e Javascript, quais as diferenças? (<https://www.alura.com.br/artigos/html-css-e-js-definicoes>)
- **Artigo** Guia de unidades no CSS (<https://www.alura.com.br/artigos/guia-de-unidades-no-css>)
- **Artigo** CSS: Guia do Flexbox (<https://www.alura.com.br/artigos/css-guia-do-flexbox>)
- **Artigo** Como fazer Grids e a Responsividade na Web (<https://www.alura.com.br/artigos/como-fazer-grids-e-a-responsividade-na-web>)
- **Artigo** Como criar animações incríveis com CSS (<https://www.alura.com.br/artigos/animacoes-com-css>)
- **Artigo** Houdini CSS: um jeito mágico de criar estilos personalizados (<https://www.alura.com.br/artigos/houdini-css>)
- **YouTube** Alura: CSS FlexBox: Dicas para começar (<https://www.youtube.com/watch?v=326-B1avuYo>)
- **YouTube** Alura: Box Model e Sizing no CSS (<https://www.youtube.com/watch?v=qcNUxyOWXlw>)
- **YouTube** Alura: Curso de HTML5 e CSS3 (<https://www.youtube.com/watch?v=78AyiuxYceg>)
- **Curso** Apostila: Desenvolvimento Web com HTML, CSS e JavaScript (<https://www.alura.com.br/apostila-html-css-javascript>)

- **Curso** Formação HTML e CSS (<https://www.alura.com.br/formacao-html-e-css>)
- **Curso** Formação CSS: aprofunde em estilos (<https://cursos.alura.com.br/formacao-css-estilos>)
- **Curso** HTML e CSS: responsividade com mobile-first (<https://www.alura.com.br/curso-online-html-css-responsividade-mobile-first>)
- **Curso** CSS: cursos para transformar designs com Grid, Flexbox e Sass (<https://cursos.alura.com.br/formacao-css-cursos-transformar-designs-grid-flexbox-sass>)
- **Desafio** 7 Days of Code: HTML e CSS (<https://7daysofcode.io/matricula/html-css>)
- **Desafio** 7 Days of Code: Responsividade (<https://7daysofcode.io/matricula/responsividade>)

## JavaScript - Fundamentos:

- JavaScript é a linguagem de programação mais popular do mundo e é uma das principais tecnologias da World Wide Web, juntamente com HTML e CSS. Ela possui tipagem dinâmica, orientação a objetos baseada em protótipos e funções de primeira classe. Ela é multi-paradigma e suporta estilos de programação orientados a eventos, funcionais e imperativos.
- Conhecer os tipos primitivos
- Declarar variáveis, considerando a diferença entre 'var', 'let' e 'const'
- Usar estruturas condicionais ('if', 'else')
- Conhecer os operadores de atribuição e comparação ('=', '==', '===')
- Usar estruturas de repetição e laços ('while', 'for')
- Usar funções, passando parâmetros e argumentos
- Manipular arrays e listas
- Aprender o conceito de Orientação a Objetos
- Fazer um CRUD

- Obter dados de uma API
- Fazer chamadas assíncronas usando 'Async/Await', 'Promise', etc

## Conteúdos

- **Site** MDN Web Docs: Um primeiro mergulho no JavaScript ([https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First\\_steps](https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps))
- **Site** MDN Web Docs: Trabalhando com texto — strings em JavaScript ([https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First\\_steps/Strings](https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/Strings))
- **Site** MDN Web Docs: if...else (<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Statements/if...else>)
- **Site** MDN Web Docs: while (<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Statements/while>)
- **Site** MDN Web Docs: for (<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Statements/for>)
- **Site** MDN Web Docs: Arrays ([https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First\\_steps/Arrays](https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/Arrays))
- **Site** MDN Web Docs: Template strings ([https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Template\\_literals](https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Reference/Template_literals))
- **Artigo** Definindo Funções em Javascript (<https://blog.matheuscastiglioni.com.br/definindo-funcoes-em-javascript/>)
- **YouTube** Mario Souto - Dev Soutinho: Como manipular arrays e objetos em JavaScript (<https://www.youtube.com/watch?v=yS7AcF-xRUg>)
- **YouTube** Marco Bruno: O que é Json e como criar um objeto (<https://www.youtube.com/watch?v=oCY5YEEjlwE>)
- **YouTube** Mario Souto - Dev Soutinho: Como usar Async/Await? Promises no JavaScript? (<https://www.youtube.com/watch?v=q28lfkBd9F4>)
- **YouTube** Mario Souto - Dev Soutinho: Como pegar dados de uma API? Como fazer AJAX? | Pegando dados de serviços via JavaScript (<https://www.youtube.com/watch?v=85vJXFpXLQw>)

## Conteúdos Alura:

- **Artigo** JavaScript: o que é, como aprender e um Guia da linguagem mais popular do mundo (<https://www.alura.com.br/artigos/javascript/>)
- **Podcast** Hipster 38 - O Reino encantado do JavaScript (<https://www.hipsters.tech/o-reino-encantado-do-javascript-hipsters-38/>)
- **Podcast** Hipster 169 - JavaScript: manual de sobrevivência 2020 (<https://www.hipsters.tech/javascript-manual-de-sobrevivencia-2020-hipsters-169/>)
- **Podcast** Hipster 236 - Evolução do JavaScript (<https://www.hipsters.tech/evolucao-do-javascript-hipsters-ponto-tech-236/>)
- **Artigo** Strings com JavaScript: o que são e como manipulá-las (<https://www.alura.com.br/artigos/strings-com-javascript-o-que-sao-e-como-manipular>)
- **Artigo** Como utilizar operadores de comparação em Javascript (<https://www.alura.com.br/artigos/operadores-matematicos-em-javascript>)
- **Artigo** Para que serve um Array? (<https://www.alura.com.br/artigos/javascript-para-que-serve-array>)
- **Artigo** Quando usar forEach e map? (<https://www.alura.com.br/artigos/javascript-quando-devo-usar-foreach-e-map>)
- **Artigo** Manipulação de array com map, filter e reduce (<https://www.alura.com.br/artigos/manipulacao-de-array-com-map-filter-e-reduce>)
- **Curso** Formação Desenvolva aplicações Web com JavaScript (<https://cursos.alura.com.br/formacao-javascript-front-end>)
- **YouTube** Alura: O que é JavaScript? (<https://www.youtube.com/watch?v=NaVSbnnV75Q>)
- **YouTube** Alura: Destructuring em JavaScript (<https://www.youtube.com/watch?v=f8a-qwKC5yk>)

- **YouTube** Alura: Classes x funções no Javascript  
(<https://www.youtube.com/watch?v=iohhj-k9L6s>)
- **Desafio** 7 Days of Code: JavaScript e DOM  
(<https://7daysofcode.io/matricula/javascript-e-dom>)
- **Desafio** 7 Days of Code: JavaScript e DOM com API  
(<https://7daysofcode.io/matricula/javascript-e-dom-api>)
- **Desafio** 7 Days of Code: JavaScript e DOM - Wordle  
(<https://7daysofcode.io/matricula/javascript-e-dom-wordle>)

---

TechGuide - Alura  
Alura, PM3 e FIAP  
O Techguide.sh é um projeto open source