



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>)

### **C# - Fundamentos:**

- C# é uma linguagem de programação, multiparadigma, de tipagem forte, desenvolvida pela Microsoft como parte da plataforma .NET. O código fonte é compilado para Common Intermediate Language (CIL) que é interpretado pela máquina virtual Common Language Runtime (CLR). É projetada para funcionar na Common Language Infrastructure da plataforma .NET Framework.
- Conhecer os tipos primitivos
- Declarar variáveis, considerando os diferentes tipos
- 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 métodos
- Manipular arrays e listas
- Obter dados de uma API
- Fazer chamadas assíncronas, etc
- Criar construtores

## Conteúdos

- **Site** Microsoft Docs: Um tour pela linguagem C# (<https://docs.microsoft.com/pt-br/dotnet/csharp/tour-of-csharp/>)
- **Artigo** Iniciando com C# - O C# e o Ambiente de Desenvolvimento (<https://medium.com/blogmikaeriohana/iniciando-com-c-parte-1-o-c-e-o-ambiente-de-desenvolvimento-d0e9c3cb2692>)
- **YouTube** Diogo Costa Dev: O que é C# (<https://www.youtube.com/watch?v=KiJOC1mXuTc>)
- **YouTube** Márcio Fessini: C# - O que é .NET? (<https://www.youtube.com/watch?v=C-7YXpLsFml>)

## Conteúdos Alura:

- **Podcast** Hipster.tech: Presente e Futuro do .NET – Hipsters 26 (<https://www.hipsters.tech/passado-e-futuro-do-net-hipsters-26/>)
- **Artigo** Como separar palavras em String com C# (<https://www.alura.com.br/artigos/como-separar-palavras-em-string-com-c>)
- **Artigo** Lidando com data e hora C# (<https://www.alura.com.br/artigos/lidando-com-data-e-hora-c>)
- **Artigo** C#: entendendo classes e structs (<https://www.alura.com.br/artigos/c-sharp-entendendo-classes-e-structs>)
- **Site** Apostila: C# e Orientação a Objetos (<https://www.alura.com.br/apostila-csharp-orientacao-objetos>)
- **YouTube** Alura: O que é o C#? (<https://www.youtube.com/watch?v=0JWMVc1CqpA>)
- **Curso** Formação C# e Orientação a Objetos (<https://cursos.alura.com.br/formacao-c-sharp-orientacao-objetos>)
- **Curso** Formação Windows Forms com C# (<https://cursos.alura.com.br/formacao-windows-forms-csharp>)
- **Curso** Formação C# e Orientação a Objetos: coleções, arquivos e bibliotecas (<https://cursos.alura.com.br/formacao-avancando-c-sharp>)

- **Desafio** 7 Days of Code: C# (<https://7daysofcode.io/matricula/csharp>)
- **Desafio** 7 Days of Code: Windows Forms C# (<https://7daysofcode.io/matricula/windows-forms>)

## 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/>)
- **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>)

## C# - 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 Dictionary, List, Queue, SortedList e Stack
- Saiba trabalhar com ArrayList e HashTable
- Iterators

## Conteúdos

- **Site** Microsoft Docs: Coleções (C#) (<https://docs.microsoft.com/pt-br/dotnet/csharp/programming-guide/concepts/collections>)
- **Artigo** Collections (<https://wilsonsantosnet.medium.com/collections-76363e18ab41>)
- **Artigo** Um pouco sobre estruturas de dados em C# (<https://guih2127.medium.com/um-pouco-sobre-estruturas-de-dados-em-c-e4371fe651d8>)
- **YouTube** Andre Constantino: Coleções na linguagem C# ([https://www.youtube.com/watch?v=9hJ\\_V61NKG8](https://www.youtube.com/watch?v=9hJ_V61NKG8))
- **YouTube** Cléber dos Santos: C# trabalhando com Coleções ([https://www.youtube.com/watch?v=SR4-\\_gWRILU](https://www.youtube.com/watch?v=SR4-_gWRILU))

## Conteúdos Alura:

- **Artigo** Listas em C# (<https://www.alura.com.br/artigos/listas-em-csharp>)
- **Site** Apostila: C# e Orientação a Objetos - Facilitando o trabalho com coleções através das listas (<https://www.alura.com.br/apostila-csharp-orientacao-objetos/trabalhando-com-listas#facilitando-o-trabalho-com-colecoes-atraves-das-listas>)
- **Curso** Curso C#: consumindo API, gravando arquivos e utilizando o LINQ (<https://cursos.alura.com.br/course/c-sharp-consumindo-api-gravando-arquivos-linq>)
- **Curso** Curso C#: Trabalhando com Arrays e Coleções (<https://cursos.alura.com.br/course/csharp-arrays-colecoes>)

## NuGet:

- NuGet é um gerenciador de pacotes para a plataforma .NET. Ele define como os pacotes desta plataforma são criados, publicados e consumidos, e fornece ferramentas para cada uma dessas funções.
- Gerenciar pacotes
- Compartilhar bibliotecas

## Conteúdos

- **Site** Microsoft Docs: Uma introdução ao NuGet (<https://docs.microsoft.com/pt-br/nuget/what-is-nuget>)
- **Artigo** Criando pacotes NuGet (<https://alexalvess.medium.com/criando-pacotes-nuget-ec04dc6425a3>)
- **YouTube** Milton Sampaio - Programação: Nuget e pacotes .Net (<https://www.youtube.com/watch?v=6GYcRJbOJYk>)

## Conteúdos Alura:

- **Curso** Curso C#: criando e distribuindo bibliotecas em .NET (<https://cursos.alura.com.br/course/csharp-criando-distribuindo-bibliotecas-dotnet>)

## C# - System.IO:

- O namespace System.IO consiste em classes, estruturas, delegates e enumerações relacionadas com entrada e saída de dados (IO). Estas classes podem ser utilizadas para ler e escrever dados em arquivos ou streams de dados. Também contém classes para suporte a arquivos e diretórios.
- Ler dados de arquivos
- Escrever dados em arquivos
- Gerenciar arquivos com Using

## Conteúdos

- **Artigo** Trabalhando com a biblioteca System.IO no C# (<http://luisfelipetorres.com.br/2017/11/02/trabalhando-com-a-biblioteca->

[system-io-no-c/](#))

- **Site** Microsoft Docs: System.IO Namespace (<https://docs.microsoft.com/pt-br/dotnet/api/system.io?view=net-6.0>)
- **Site** Microsoft Docs: E/S de arquivo e de fluxo (<https://docs.microsoft.com/pt-br/dotnet/standard/io/>)
- **YouTube** Código Logo: C# Tutorial de IO 1 - Introdução (<https://www.youtube.com/watch?v=LYTZXQpy3RY>)
- **YouTube** Código Logo: C# Tutorial de IO 15 - Como Copiar Arquivo usando o FileStream (<https://www.youtube.com/watch?v=f9OOnzV6vIE>)

### Conteúdos Alura:

- **Curso** Curso C#: trabalhando com arquivos (<https://cursos.alura.com.br/course/c-sharp-trabalhando-arquivos>)

### C# - Gerenciamento de Memória:

- O gerenciamento automático de memória é um dos serviços que o Common Language Runtime oferece durante a Execução Gerenciada (Managed Execution). O coletor de lixo do Common Language Runtime gerencia a alocação e liberação de memória para uma aplicação.
- Entender como a memória é administrada
- Conhecer o Garbage Collector
- Entender sobre o Stack e o Heap gerenciado

### Conteúdos

- **Site** Microsoft Docs: Gerenciamento automático de memória (<https://docs.microsoft.com/pt-br/dotnet/standard/automatic-memory-management>)
- **Site** Microsoft Docs: Gerenciamento da memória e coleta de lixo (GC) no ASP.NET Core (<https://learn.microsoft.com/pt-br/aspnet/core/performance/memory?view=aspnetcore-8.0>)
- **Artigo** Gerenciamento de memória — C# (<https://medium.com/@mateusgoncalvesgeracino/gerenciamento-de->



[mem%C3%B3ria-c-bcb5f2c89464\)](#)

- **YouTube** Dias de Dev: Gerenciamento de memória - Stack vs Heap  
(<https://www.youtube.com/watch?v=7kJwVQGJCbw>)

### Conteúdos Alura:

- **Curso** Curso .NET: gerenciamento de memória para otimização de performance (<https://cursos.alura.com.br/course/dotnet-gerenciamento-memoria-otimizacao-performance>)

### C# - 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

- **Site** Microsoft Docs: Passo a passo: Criar e executar testes de unidade para código gerenciado (<https://docs.microsoft.com/pt-br/visualstudio/test/walkthrough-creating-and-running-unit-tests-for-managed-code?view=vs-2022>)
- **Site** Microsoft Docs: Passo a passo: desenvolvimento controlado por teste usando o Gerenciador de Testes (<https://docs.microsoft.com/pt-br/visualstudio/test/quick-start-test-driven-development-with-test-explorer?view=vs-2022>)
- **Artigo** Testes Unitários: Fundamentos e Qualidade de Software!  
(<https://dev.to/wnqueiroz/testes-unitarios-fundamentos-e-qualidade-de-software-5af0>)

- **Artigo** Testes unitários em .NET com xUnit (<https://medium.com/unicoidtech/testes-unit%C3%A1rios-em-net-com-xunit-d7c78c593832>)
- **Artigo** .NET 6, C# 10 e implementação de testes: novos recursos para simplificar a codificação (<https://renatogroffe.medium.com/net-6-c-10-e-implementa%C3%A7%C3%A3o-de-testes-novos-recursos-para-simplificar-a-codifica%C3%A7%C3%A3o-5faec474ccc2>)
- **Artigo** Testes Unitários com C# (<https://aldocostasantos.wordpress.com/2012/12/21/testes-unitarios-parte01/>)
- **YouTube** Canal TI: Testes Unitários com C# (XUnit) (<https://www.youtube.com/watch?v=40TujcOnLdQ>)
- **YouTube** Matheus Rizzi: Mock de objetos para teste unitário com c# (<https://www.youtube.com/watch?v=ezxBMZQ8d84>)

## 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 (<https://cursos.alura.com.br/extra/hipsterstech/testes-automatizados-hipsters-51-a535>)
- **Curso** C#: aprenda a criar testes limpos com boas práticas (<https://cursos.alura.com.br/course/csharp-criar-testes-limpos-boas-praticas>)
- **Curso** Formação Testes em .NET (<https://www.alura.com.br/formacao-testes-em-dotnet>)

## ADO.NET:

- O ADO.NET é um conjunto de classes que expõem serviços de acesso a dados para desenvolvedores do .NET Framework. O ADO.NET fornece um conjunto rico de componentes para criar aplicativos distribuídos e de

compartilhamento de dados. Faz parte do .NET Framework, fornecendo acesso a dados de aplicativo relacionais e XML.

- Manipular bancos de dados
- Conhecer o DataSet e DataTable
- Realizar conexões
- Manipular documentos XML

## Conteúdos

- **Site** Microsoft Docs: ADO.NET (<https://docs.microsoft.com/pt-br/dotnet/framework/data/adonet/ado-net-overview>)
- **YouTube** Parmezani.NET: ADO.NET - Acesso ao banco de dados com classe genérica (<https://www.youtube.com/watch?v=GE5Q9sDS7rQ>)
- **YouTube** André Lima: Salvando arquivos no banco de dados com C# e ADO.NET (<https://www.youtube.com/watch?v=2H3Hc0ZFGjY>)

## Entity Framework Core:

- O Entity Framework Core é um mapeador objeto-relacional (ORM). O mapeamento objeto-relacional é uma técnica que permite aos desenvolvedores trabalhar com dados de forma orientada ao objeto, realizando o trabalho necessário para mapear entre objetos definidos em uma linguagem de programação da aplicação e dados armazenados em fontes de dados relacionais.
- Manipular bancos de dados usando objetos .NET
- Criar modelos correspondentes a um banco de dados
- Realizar consultas

## Conteúdos

- **Site** Microsoft Docs: Entity Framework Core (<https://docs.microsoft.com/pt-br/ef/core/>)
- **Artigo** Série: Entity Framework. Introdução (<https://medium.com/@alexandre.malavasi/s%C3%A9rie-entity-framework-introdu%C3%A7%C3%A3o-parte-1-13ab52b0bdaf>)

- **YouTube** ProgrAmada Mente: Introdução ao Entity Framework Core (<https://www.youtube.com/watch?v=OlsM0LkIYH0>)
- **YouTube** A Escola de Software: O que é Entity Framework Core? (<https://www.youtube.com/watch?v=ZaVyLFTDuIQ>)

### Conteúdos Alura:

- **Curso** Curso .NET: persistindo dados com Entity Framework Core (<https://cursos.alura.com.br/course/dot-net-persistindo-dados-entity-framework-core>)
- **Curso** Formação: Aprofunde em.NET com Entity Framework Core (<https://cursos.alura.com.br/formacao-dot-net-avancando-ef-core>)

### Estruturas de Dados:

- No contexto dos computadores, uma estrutura de dados é uma forma específica de armazenar e organizar os dados na memória do computador para que esses dados possam ser facilmente recuperados e utilizados de forma eficiente quando necessário posteriormente.
- Conhecer as principais estruturas de dados
- Implementar as principais estruturas de dados

### Conteúdos

- **Artigo** Estrutura de dados (<https://medium.com/php-word/estrutura-de-dados-6afb386a6fc1>)
- **Artigo** Estrutura de Dados e Algoritmo - parte 1 (<https://medium.com/@PaoloProdossimoLopes/01-estrutura-de-dados-e-algoritmo-9cb6a9b14aa>)
- **Artigo** Estrutura de Dados e Algoritmo - parte 2 (<https://medium.com/@PaoloProdossimoLopes/02-estrutura-de-dados-e-algoritmo-deba0b384a0f>)
- **YouTube** Mario Souto - Dev Soutinho: Estruturas de dados com JavaScript (<https://www.youtube.com/watch?v=MweeZn1rR8s>)

- **YouTube** Fabio Akita: Árvores (<https://www.youtube.com/watch?v=9GdesxWtOgs>)
- **YouTube** Loiane Groner: Estrutura de Dados com Java | Lista Encadeada ([https://www.youtube.com/watch?v=RW0oD2L\\_tSg](https://www.youtube.com/watch?v=RW0oD2L_tSg))
- **YouTube** Loiane Groner: Estrutura de Dados com Java | Filas ([https://www.youtube.com/watch?v=RdoLTgJSV\\_c](https://www.youtube.com/watch?v=RdoLTgJSV_c))
- **YouTube** Loiane Groner: Estrutura de Dados com Java | Pilhas (<https://www.youtube.com/watch?v=ZdU4wMyiTSs>)

### Conteúdos Alura:

- **Site** Apostila Python: Estruturas de dados (<https://www.alura.com.br/apostila-python-orientacao-a-objetos/estrutura-de-dados>)
- **Artigo** Estruturas de dados: uma introdução (<https://www.alura.com.br/artigos/estruturas-de-dados-introducao>)
- **Artigo** Curso Estrutura de Dados: computação na prática com Java (<https://www.alura.com.br/artigos/estrutura-dados-computacao-na-pratica-com-java>)
- **Podcast** Hipsters.tech: Algoritmos e estrutura de dados - Hipsters 186 (<https://www.alura.com.br/podcast/hipsterstech-algoritmos-e-estrutura-de-dados-hipsters-186-a375>)
- **Curso** Curso PHP: dominando as Collections (<https://cursos.alura.com.br/course/php-dominando-collections>)
- **Curso** Curso Python: crie a sua primeira aplicação (<https://cursos.alura.com.br/course/python-crie-sua-primeira-aplicacao>)
- **Desafio** 7 Days of Code: Estruturas de Dados (<https://7daysofcode.io/matricula/estruturas-de-dados>)

## Nível 2

### CLR:

- O Common Language Runtime (CLR) é o componente de máquina virtual da plataforma .NET da Microsoft que gerencia a execução de programas .NET.
- Entender como o CLR funciona
- Entender o gerenciamento de memória
- Conhecer a CIL e o JIT

### Conteúdos

- **Site** Microsoft Docs: Visão geral do CLR (Common Language Runtime) (<https://docs.microsoft.com/pt-br/dotnet/standard/clr>)
- **Artigo** Acervo Lima: Common Language Runtime (CLR) em C# (<https://acervolima.com/common-language-runtime-clr-em-c/>)
- **Artigo** Common Language Runtime(CLR) DotNet (inglês) (<https://medium.com/@mirzafarrukh13/common-language-runtime-dotnet-83e0218edcae>)
- **YouTube** Easy Technology: Overview of the Common Language Runtime CLR (inglês) (<https://www.youtube.com/watch?v=ttggRyMPU9c>)

### LINQ:

- O LINQ (Language-Integrated Query) é o nome de um conjunto de tecnologias com base na integração de recursos de consulta diretamente na linguagem C#.
- Criar consultas
- Conhecer as cláusulas Select e Where
- Consultar coleções de objetos em memória
- Mapear o banco de dados com Linq to Sql

## Conteúdos

- **Site** Microsoft Docs: LINQ (Consulta Integrada à Linguagem) (C#)  
(<https://docs.microsoft.com/pt-br/dotnet/csharp/programming-guide/concepts/linq/>)
- **Site** Microsoft Docs: Introdução a consultas LINQ (C#)  
(<https://docs.microsoft.com/pt-br/dotnet/csharp/programming-guide/concepts/linq/introduction-to-linq-queries>)
- **Artigo** LINQ: Menos esforço, mais dados (<https://lucas-eschechola.medium.com/linq-menos-esfor%C3%A7o-mais-dados-7354640afba1>)
- **YouTube** Willian Menezes: C# - Introdução ao LINQ  
(<https://www.youtube.com/watch?v=ja1nORsswSU>)
- **YouTube** Willian Menezes: C# - LINQ - Criando a primeira consulta  
(<https://www.youtube.com/watch?v=gC4A54H-EdQ>)
- **YouTube** André Secco: Recursos do C# - LINQ  
(<https://www.youtube.com/watch?v=z7SsqZEDKts>)

## Conteúdos Alura:

- **Curso** Curso C#: consumindo API, gravando arquivos e utilizando o LINQ  
(<https://cursos.alura.com.br/course/c-sharp-consumindo-api-gravando-arquivos-linq>)
- **Curso** Curso C# LINQ: explorando ferramentas de consulta  
(<https://cursos.alura.com.br/course/chsharp-linq-explorando-ferramentas-consulta>)

## C# - Serialização:

- Serialização é o processo de converter um objeto em um stream de bytes para armazenar o objeto ou transmiti-lo à memória, a um banco de dados ou a um arquivo. Seu principal objetivo é salvar o estado de um objeto a fim de poder recriá-lo quando necessário.
- Enviar um objeto para um aplicativo remoto usando um serviço Web
- Passar um objeto como uma cadeia de caracteres JSON ou XML

- Passar informações específicas do usuário ou de segurança entre aplicativos

## Conteúdos

- **Site** Microsoft Docs: Serialização (C#) (<https://docs.microsoft.com/pt-br/dotnet/csharp/programming-guide/concepts/serialization/>)
- **Site** Microsoft Docs: Serialização básica (<https://docs.microsoft.com/pt-br/dotnet/standard/serialization/basic-serialization>)
- **Artigo** O que é serialização/serialization? (<https://medium.com/@thaisdalencar/o-que-%C3%A9-serializa%C3%A7%C3%A3o-serialization-fc0887bd0970>)
- **YouTube** Miguel Jordan: Aprenda a serializar e deserializar objetos em C# (<https://www.youtube.com/watch?v=wWuagfG-1sY>)
- **YouTube** Derek Banas: C# Serialization (inglês) (<https://www.youtube.com/watch?v=jbwjbbc5PjI>)

## ASP.NET Core:

- ASP.NET Core é um framework de código aberto e multi-plataforma para a construção de aplicações baseadas na nuvem, tais como aplicações web, aplicações IoT e backends mobile.
- Criar aplicações e serviços Web
- Manter uma aplicações MVC
- Desenvolver de interface do usuário da Web do lado do cliente
- Criar uma API Web

## Conteúdos

- **Site** Microsoft Docs: Visão geral do ASP.NET Core (<https://docs.microsoft.com/pt-br/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-6.0>)
- **Artigo** ASP.NET Core: dicas úteis para o dia a dia de um Desenvolvedor - Parte 1 (<https://renatogroffe.medium.com/asp-net-core-dicas-%C3%BAteis-para-o-dia-a-dia-de-um-desenvolvedor-parte-1-fd2e4b797472>)



- **YouTube** Glaucia Lemos: Workshop - ASP.NET Core para Iniciantes (<https://www.youtube.com/watch?v=njImcXxSHE4>)
- **YouTube** Teacher Mauro Duarte: C# Introdução ao Asp.Net Core MVC (<https://www.youtube.com/watch?v=lwVvrRIS7al>)

### Conteúdos Alura:

- **Podcast** Hipsters.tech: ASP.NET Core – Hipsters 108 (<https://www.hipsters.tech/asp-net-core-hipsters-108/>)
- **Podcast** Hipsters.tech: .NET Core no Stack Overflow – Hipsters On The Road 20 (<https://www.hipsters.tech/net-core-no-stack-overflow-hipsters-on-the-road-20/>)
- **Artigo** Como criar uma calculadora de IMC com ASP.NET Core Blazor (<https://www.alura.com.br/artigos/como-criar-calculadora-imc-com-asp-net-core-blazor>)
- **YouTube** Alura: Criando sua primeira aplicação .NET Core no Linux (<https://www.youtube.com/watch?v=UZQqXPuKr1s>)
- **Curso** Curso .NET 6: criando uma web API (<https://cursos.alura.com.br/course/dotnet-criando-web-api>)
- **Curso** Formação ASP.Net Core: crie aplicações com C#, .NET, Entity Framework e LINQ (<https://cursos.alura.com.br/formacao-dotnet>)

### Dapper:

- O Dapper é um produto de mapeamento objeto-relacional (ORM) para a plataforma Microsoft .NET. Ele fornece um framework para mapear um modelo de domínio orientado a objetos para bancos de dados relacionais.
- Realizar consultas a bancos de dados como 'select', 'insert', 'update', 'delete'
- Manipular bancos de dados

### Conteúdos

- **Artigo** Descomplicando o Dapper com .Net Core (<https://medium.com/jundevelopers/descomplicando-o-dapper-com-net->

[core-11118941e255\)](#)

- **Artigo** .NET 5 + Dapper: exemplos de implementação (<https://renatogroffe.medium.com/net-5-dapper-exemplos-de-implementa%C3%A7%C3%A3o-c3a9ce2be802>)
- **Artigo** Introdução ao Dapper (<http://www.andrealveslima.com.br/blog/index.php/2017/10/04/introducao-ao-dapper/>)
- **YouTube** Leonardo Buta: Dapper - Introdução, conceitos e prática com .NET Core (<https://www.youtube.com/watch?v=MnfxKnY3bDI>)

### Conteúdos Alura:

- **YouTube** Alura: O que é o Dapper? (<https://www.youtube.com/watch?v=-USSV3LfdYw>)

### Injeção de Dependências:

- Injeção de Dependências é um padrão de projeto no qual uma classe solicita dependências de fontes externas ao invés de criá-las.
- Evitar o alto nível de acoplamento de código dentro de uma aplicação
- Implementar a inversão de controle

### Conteúdos

- **Artigo** Injeção de Dependência (<https://medium.com/@eduardolanfredi/inje%C3%A7%C3%A3o-de-depend%C3%Aancia-ff0372a1672>)
- **YouTube** CódigoEscola: O que é injeção de dependência na prática? ([https://www.youtube.com/watch?v=F\\_Rts61IE3g](https://www.youtube.com/watch?v=F_Rts61IE3g))
- **YouTube** Code Easy: Injeção de Dependência, Entenda o que é e para que serve (<https://www.youtube.com/watch?v=plyNfpgHpFU>)
- **YouTube** DevSuperior: Injeção de dependência e inversão de controle (<https://www.youtube.com/watch?v=O07XFebgw-g>)

## Conteúdos Alura:

- **Artigo** Services e injeção de dependência no Angular: o que são e como funcionam? (<https://www.alura.com.br/artigos/services-injecao-dependencia-angular-o-que-sao-como-funcionam>)
- **Artigo** Injeção de dependência no Android com o Hilt (<https://www.alura.com.br/artigos/injecao-de-dependencia-do-android-com-o-hilt>)
- **Site** Injeção de dependências: O que é? (<https://cursos.alura.com.br/extra/alura-mais/injecao-de-dependencias-o-que-e--c224>)
- **Curso** Curso C# Reflection: manipule dinamicamente tipos e assemblies (<https://cursos.alura.com.br/course/csharp-reflection-manipule-dinamicamente-tipos-assemblies>)

## C# - Multithreading:

- Multithreading é a capacidade de realizar múltiplas operações ao mesmo tempo. Operações com o potencial de atrasar outras operações podem ser executadas em threads separadas.
- Executar múltiplas tarefas simultaneamente
- Entender como threads são executadas
- Aprender como fazer uma thread esperar em um ponto específico

## Conteúdos

- **Site** Microsoft Docs: Usando threads e threading (<https://docs.microsoft.com/pt-br/dotnet/standard/threading/using-threads-and-threading>)
- **Artigo** Multi-threading em C# (programação concorrente) (<https://marcionizzola.medium.com/multi-threading-em-c-programa%C3%A7%C3%A3o-concorrente-3d872ce4c8ae>)
- **Artigo** Implementando Threads em C# ([https://medium.com/@samuell\\_santos/implementando-threads-em-c-4d4e6b14a398](https://medium.com/@samuell_santos/implementando-threads-em-c-4d4e6b14a398))

- **YouTube** Código Logo: C# Avançado - Introdução à Threads (<https://www.youtube.com/watch?v=DEYkxz1QEDY>)
- **YouTube** Abre Chaves: Desenvolvimento em C# - Conceitos Básicos de Threads (<https://www.youtube.com/watch?v=dljWM-Yf-LQ>)

## Nível 3

### C# - Delegates e Eventos:

- Delegates são objetos que são usados como ponteiros de função para se referirem a um método atribuído a eles.
- Eventos são a ação realizada que altera o estado de um objeto. Eventos são declarados usando delegates - eles fornecem um encapsulamento aos delegates.
- Entender o conceito de delegate
- Criar uma referência para uma função com uma certa lista de parâmetros
- Entender o conceito de evento
- Manipular diferentes tipos de eventos

### Conteúdos

- **Site** Microsoft Docs: Distinção entre Delegates e Eventos (<https://docs.microsoft.com/pt-br/dotnet/csharp/distinguish-delegates-events>)
- **Site** Microsoft Docs: Manipular e gerar eventos (<https://docs.microsoft.com/pt-br/dotnet/standard/events/>)
- **Artigo** Delegates em C#: o que são, o que comem, onde vivem e pra que servem? (<https://medium.com/@alexandre.malavasi/delegates-em-c-o-que-s%C3%A3o-o-que-comem-onde-vivem-e-pra-que-serv-12b5336f3d9a>)
- **YouTube** Matrix Code: C# Delegates e Eventos (<https://www.youtube.com/watch?v=Sice1GmFiuo>)
- **YouTube** Ricardo Maroquio: Eventos e Delegates (<https://www.youtube.com/watch?v=0xxbsSIYDE8>)

## C# - Métodos anônimos e lambda expressions:

- Métodos anônimos são métodos sem um nome que podem ser definidos usando a palavra-chave delegate.
- Lambda expressions são usadas como funções anônimas, mas você não precisa especificar o tipo do valor que você digita, tornando-as mais flexíveis de usar.
- Criar funções anônimas que você pode usar para criar delegates
- Criar funções locais que podem ser passadas como argumentos

### Conteúdos

- **Site** Microsoft Docs: Lambda expressions and anonymous functions (<https://learn.microsoft.com/pt-br/dotnet/csharp/language-reference/operators/lambda-expressions>)
- **Site** Microsoft Docs: Delegates e lambdas (<https://docs.microsoft.com/pt-br/dotnet/standard/delegates-lambdas>)
- **Site** Microsoft Docs: Delegates com métodos nomeados versus anônimos (<https://docs.microsoft.com/pt-br/dotnet/csharp/programming-guide/delegates/delegates-with-named-vs-anonymous-methods>)
- **Site** Acervo Lima: Método anônimo em C# (<https://acervolima.com/metodo-anonimo-em-c/>)
- **Artigo** Apreendendo a usar Lambda Expression em C# — Guia Rápido (<https://medium.com/beelabacademy/apreendendo-a-usar-lambda-expression-em-c-guia-r%C3%A1pido-7316b2cbcec1>)
- **Artigo** Usando DELEGATE e LAMBDA no C# (<https://medium.com/@pedropini/usando-delegate-e-lambda-no-c-c87251a22d37>)
- **YouTube** Marco Diniz: C# em 5 minutos - Lambda Expressions (<https://www.youtube.com/watch?v=Ftbrn82BPcc>)
- **YouTube** Coding Night: Introdução a consultas LINQ e expressões Lambda no C# (<https://www.youtube.com/watch?v=-yxrPTlvpz4>)

## 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=wxLvMxz1Q>)

- **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/microsservicos-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>)



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

### Conteúdos Alura:

- **Podcast** Hipsters.tech: TechGuide - Microserviços – Hipsters Ponto Tech #349 (<https://www.hipsters.tech/techguide-microservicos-hipsters-ponto-tech-349/>)
- **YouTube** Alura: O que são Microserviços? (<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: Microserviços 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: Microserviços 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 Microserviços?  
(<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 Microserviços: 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>)

## 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>)

### MAUI:

- .NET Multi-platform App UI (.NET MAUI) é um framework multi-plataforma para criar aplicativos nativos móveis e desktop com C# e XAML.
- Criar aplicativos móveis e desktop nativos com C# e XAML
- Criar aplicativos multiplataforma
- Compartilhar o layout e o design da interface do usuário entre plataformas

### Conteúdos

- **Site** Microsoft Docs: O que é o MAUI do .NET?  
(<https://docs.microsoft.com/pt-br/dotnet/maui/what-is-maui>)
- **Artigo** All About .NET MAUI (inglês) (<https://medium.com/volosoft/all-about-net-maui-5a1e774803ac>)
- **YouTube** Jose Async: .NET MAUI - Por onde começar  
([https://www.youtube.com/watch?v=vV\\_KVf\\_OQaI](https://www.youtube.com/watch?v=vV_KVf_OQaI))

- **YouTube** balta.io: O futuro do Cross Platform com MAUI (<https://www.youtube.com/watch?v=wjPx7BPEpQk>)
- **YouTube** James Montemagno: Learn .NET MAUI - Full Course for Beginners | Build cross-platform apps in C# (inglês) ([https://www.youtube.com/watch?v=DuNLR\\_NJv8U](https://www.youtube.com/watch?v=DuNLR_NJv8U))

## Habilidade Auxiliar: Infraestrutura

### 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>)

## Habilidade Auxiliar: Boas práticas

### 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>)

## Clean Code:

- Aplicar técnicas simples que visam facilitar a escrita e leitura de um código
- Refatorar seu código para que fique mais claro

## Conteúdos

- **Artigo** Clean Code: O que é? Porque usar? (<https://medium.com/desenvolvendo-com-paixao/1-clean-code-o-que-%C3%A9-porque-usar-1e4f9f4454c6>)
- **YouTube** Filipe Deschamps: CLEAN CODE 1 - Introdução (<https://www.youtube.com/watch?v=9w3o9NHXqu0>)
- **YouTube** Filipe Deschamps: CLEAN CODE 2 - O que é código limpo? ([https://www.youtube.com/watch?v=ZVtGbO\\_CnfA](https://www.youtube.com/watch?v=ZVtGbO_CnfA))
- **Livro** Código limpo: habilidades práticas do Agile software (<https://www.amazon.com.br/C%C3%B3digo-limpo-Robert-C-Martin/dp/8576082675>)

## Conteúdos Alura:

- **Artigo** Clean Code: O que é, Casos de Uso, Exemplo de Código Limpo (<https://www.alura.com.br/artigos/o-que-e-clean-code>)
- **YouTube** Alura: Clean Code e Solid ([https://www.youtube.com/watch?v=XV9B4LX\\_re8](https://www.youtube.com/watch?v=XV9B4LX_re8))

- **Livro** Orientação a Objetos e SOLID para Ninjas  
([https://www.casadocodigo.com.br/products/livro-oo-solid?\\_pos=2&\\_sid=fe732c657&\\_ss=r](https://www.casadocodigo.com.br/products/livro-oo-solid?_pos=2&_sid=fe732c657&_ss=r))

## 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>)