

Guia de Estudos

Por Eve Nogueira

Sumário

Sumário	2
Comece por aqui	4
Internet	4
O que é HTTP?	5
Material de estudos	5
Navegadores	6
POSIX	6
Material de estudos - video	7
DNS	7
Sistema operacional	8
Material de estudos - video	9
Banco de dados	9
API	9
API REST	9
Material de estudo - video	10
JSON API	10
Material de estudo - video	10
Git e GitHub	10
Material de estudo - video	10
Material de estudo - sites	10
Front-end	11
HTML e CSS	11
Material de estudo - video	11
Material de estudo - site	11
JavaScript - TypeScript	11
Material de estudo - video	12
Typescript	12
Criação de projetos e portfólio	12
Frameworks	12
React	12
React - projetos e portfólio	12
Angular	12
Back-end	13
Java	13
Material de estudos - video	13
Prática	14
C#	14
Material de estudos - video	14
Prática	14
Python	15
Material de estudos - video	15

Material de estudos - Python para análise de dados e engenharia de dados	15
Material de estudos - Python para ciência de dados	15
Ruby	15
Material de estudos - video	16
DevOps	18
Linguagem de programação	18
Virtualização	18
Automação	18
Material de estudos - video	18
Containers	19
Material de estudos - video	19
Monitoramento e logs	19
Material de estudos - video	19
Cyber Security	19
Material de estudo - videos	20
Material de estudo - sites	20
Dados	20
Análise de dados	21
Material de estudos - video	21
SQL	21
Engenharia de dados	21
Material de estudos - video	22
Cloud	22
Ciência de dados	22
Sites para treinamento	23
Aprendizado Interativo:	23

Introdução

Primeiramente gostaria de agradecer por me acompanhar nas redes sociais. Se você chegou nesse guia é porque tem um interesse muito grande em entrar na área de tecnologia, seja como primeiro emprego ou transicionando carreira, ou apenas gostaria de aprender uma stack ou tecnologia nova e eu to aqui para apoiar você.

Acompanhe também

- Grupo no Telegram com desafios e projetos reais para você treinar os skills necessários do mundo real na área de tecnologia - <https://lastlink.com/p/C7A86FD4D/checkout-payment/>
- TikTok [Eve \(@eve.py\)](#) | TikTok
- Instagram <https://www.instagram.com/eveline.py>
- YouTube [Eve Nogueira - YouTube](#)
- E-mail suporte@evenogueira.com.br

Comece por aqui

Internet

A internet é uma rede global de computadores interconectados. Ela permite que pessoas, empresas e governos se comuniquem e compartilhem informações em todo o mundo. Simplificando, a internet funciona assim:

1. Você deve ter um dispositivo conectado à internet seja um computador celular ou tablet.

2. Este dispositivo se conecta ao seu Provedor de Serviços de Internet (ISP) via cabo ou conexão sem fio. Um provedor de serviços de Internet é uma empresa que provê serviços de conexão à internet.
3. Seu ISP conecta seu dispositivo a um router responsável por rotear seu tráfego de Internet para outras redes.
4. O router envia uma requisição de informações (como acessar um site ou enviar um e-mail) para um servidor da web. Um servidor web é um computador que armazena dados e informações na internet.
5. O servidor web processa sua solicitação e envia uma resposta de volta ao router. que será enviado de volta para o seu dispositivo.
6. Seu dispositivo exibe uma resposta do servidor da web em um navegador da web (como Google Chrome ou Firefox).
7. Todas essas interações são possíveis graças a um conjunto de protocolos de comunicação que regem a forma como os dados são transferidos na internet. Um dos protocolos mais importantes é o Hypertext Transfer Protocol (HTTP), que é usado para acessar sites e conteúdos da web.

Este é apenas um rápido resumo de como a internet funciona. Existem muitos outros componentes e tecnologias, como endereços IP, DNS, criptografia e muito mais. No entanto, espero que isso lhe dê uma ideia geral de como a internet funciona.

O que é HTTP?

HTTP (ou "Hypertext Transfer Protocol" em inglês) é um protocolo usado para enviar e receber informações pela internet. É a base para a comunicação entre os navegadores (como Chrome, Firefox, etc.) e o servidor web que hospeda os sites que visitamos. Quando você insere o endereço de um site na barra de endereços do navegador e pressiona Enter, o navegador envia uma solicitação HTTP ao servidor da Web apropriado, solicitando os arquivos que compõem o site. O servidor da Web responde com os arquivos solicitados e o navegador usa esses arquivos para exibir a página da Web que você está visualizando. O HTTP é responsável pela troca eficiente e segura de dados.

Material de estudos

- Documentação oficial - [RFC 7230 - Hypertext Transfer Protocol \(HTTP/1.1\): Message Syntax and Routing \(ietf.org\)](https://tools.ietf.org/html/rfc7230)
- [HTTP // Dicionário do Programador - YouTube](#)
- Livro em inglês download gratuito - [HTTP: The Definitive Guide \(PDF\) \(pdfroom.com\)](https://pdfroom.com/http-the-definitive-guide/)
- Serviço de requisições e respostas em HTTP para testar httpbin.org

Navegadores

Os navegadores da Web são aplicativos de software que permitem aos usuários acessar e navegar na internet. Eles são responsáveis por exibir sites, processar código HTML e JavaScript e executar outras funções que ajudam os usuários a interagir na internet. Aqui está uma explicação simples de como os navegadores da web funcionam:

1. Quando você abre o navegador da web, ele solicitará uma página da Web inserindo um URL ou clicando em um link em uma página existente.
2. O navegador usa o protocolo HTTP para enviar essa solicitação ao servidor da web. O servidor web responde retornando o conteúdo HTML, CSS e JavaScript da página.
3. Os navegadores analisam o código HTML e CSS para criar uma representação visual da página na tela.
4. Os navegadores executam código JavaScript para aumentar a interação com a página.
5. À medida que a página é carregada, o navegador envia requisições adicionais ao servidor web para carregar imagens, arquivos de vídeo e outros recursos necessários para que a página seja exibida corretamente.
6. O navegador também armazena em cache os dados da página para carregar mais rapidamente na próxima vez que você visitar o site.
7. Os navegadores da Web também possuem recursos adicionais, como a capacidade de armazenar senhas, preenchimento automático de formulário e bloqueio de pop-up.

Simplificando, um navegador da Web é uma interface que permite aos usuários interagir com a internet visualizando o conteúdo e executando scripts. Ele atua como uma ponte entre os usuários e a internet processando e exibindo informações de forma intuitiva.

POSIX

POSIX é um conjunto de padrões para sistemas operacionais, definidos pelo Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE) e pelo Comitê Técnico 1 da Organização Internacional de Normalização (ISO).

Esses padrões visam garantir a portabilidade de aplicativos e software entre sistemas operacionais diferentes, definindo uma interface padrão para a interação entre o sistema operacional e o software.

Entre os padrões POSIX estão o POSIX.1, que define as interfaces de programação de aplicativos (APIs) para serviços básicos do sistema operacional, como gerenciamento de arquivos, gerenciamento de processos e entrada/saída de dados. Outros padrões incluem POSIX.2, que define comandos e utilitários do shell, e POSIX.4, que define APIs para programação de rede.

Os sistemas operacionais compatíveis com POSIX incluem o Unix, o Linux, o macOS e o FreeBSD, entre outros. Esses sistemas operacionais oferecem suporte a APIs POSIX e, portanto, são capazes de executar aplicativos e software desenvolvidos de acordo com os padrões POSIX.

O uso de padrões POSIX ajuda a garantir que os aplicativos e software sejam portáteis entre diferentes sistemas operacionais, o que é especialmente importante em ambientes corporativos e de produção. Além disso, o uso de padrões POSIX também ajuda a promover a interoperabilidade entre diferentes sistemas operacionais e plataformas.

Material de estudos - video

- Básico sobre POSIX - <https://www.youtube.com/watch?v=6t1huk0qJ7s&t=1s>

DNS

O Domain Name System (DNS) é uma das principais tecnologias que sustentam a internet e é responsável por converter nomes de domínio (como google.com) em endereços IP (como 172.217.14.78) que os computadores podem entender. Aqui está uma explicação simples de como o ADN funciona:

1. Quando você digita um endereço da Web em seu navegador (como google.com), o navegador envia uma requisição ao servidor DNS local de seu computador para resolver o nome de domínio em um endereço IP.
2. Se o servidor DNS local já tiver armazenado em cache o endereço IP, ele o retornará ao navegador. Ocorrência contrário, ele envia uma solicitação ao servidor DNS de nível superior para obter o endereço IP.
3. O servidor DNS de nível superior verifica a solicitação e a encaminha para o servidor DNS autorizado responsável pelo domínio (por exemplo, o servidor DNS do Google no caso do google.com).
4. Um servidor DNS autoritativo verifica a requisição e retorna o endereço IP ao servidor DNS de nível superior.
5. O servidor DNS de nível superior retorna o endereço IP ao servidor DNS local do computador.

6. Um servidor DNS local armazena em cache o endereço IP para uso futuro e o retorna ao navegador.
7. Os navegadores usam endereços IP para enviar requisições HTTP aos servidores da web apropriados no caso do google.com, os servidores da web do Google).
8. O servidor web retorna o conteúdo da página solicitada ao navegador.

O processo de resolução de nomes de domínio pode parecer complicado, mas muitas vezes acontece em segundo plano sem que os usuários percebam. O DNS é uma parte crucial da internet porque permite que os usuários acessem sites e serviços usando nomes de domínio fáceis de lembrar, em vez de endereços IP complicados.

Sistema operacional

Um sistema operacional (SO) é um software que gerencia os recursos de um computador e fornece uma interface entre o usuário e o hardware. Ele é responsável pelo controle dos recursos de hardware, como memória, processador, dispositivos de entrada e saída, e pelo gerenciamento dos aplicativos que são executados no computador.

Existem vários tipos de sistemas operacionais, incluindo os de desktop, servidores, mobile e embarcados. Cada tipo é projetado para executar em diferentes tipos de dispositivos e atender a diferentes necessidades.

Os sistemas operacionais de desktop, como o Windows, MacOS e Linux, são projetados para computadores pessoais e laptops. Eles oferecem uma interface gráfica do usuário (GUI) que permite ao usuário interagir com o computador usando mouse e teclado.

Os sistemas operacionais de servidor, como o Windows Server e o Linux, são projetados para gerenciar servidores e serviços em rede. Eles são projetados para serem altamente confiáveis, escaláveis e seguros.

Os sistemas operacionais móveis, como o Android e o iOS, são projetados para smartphones e tablets. Eles oferecem uma interface de usuário projetada para dispositivos móveis e incluem recursos como gerenciamento de energia e integração com serviços em nuvem.

Os sistemas operacionais embarcados, como o FreeRTOS e o VxWorks, são projetados para dispositivos incorporados, como sistemas de controle industrial, sistemas de navegação por satélite e dispositivos médicos. Eles são projetados para serem altamente confiáveis e eficientes em termos de recursos.

Alguns dos principais recursos de um sistema operacional incluem o gerenciamento de memória, gerenciamento de processos, gerenciamento de arquivos, segurança e rede. Os sistemas operacionais modernos também incluem recursos avançados, como suporte a multitarefa, virtualização e suporte a contêineres.

- Comandos do Prompt do Windows - [Comandos do Prompt do Windows Que Você Deveria Saber](#)
- Comando do Prompt do Linux - [30 Comandos do Terminal Linux BÁSICOS que você PRECISA SABER para ser um HACKER](#)
- Comando do Prompt do Mac - [Dicas e comandos diversos para o Terminal do Mac!](#)

Banco de dados

Um banco de dados é um sistema de armazenamento de informações que permite organizar, gerenciar e recuperar dados de forma eficiente. Existem dois tipos principais de banco de dados: SQL e NoSQL.

O SQL é um tipo de banco de dados relacional que utiliza a linguagem SQL (Structured Query Language) para gerenciar dados em tabelas relacionais. Esse tipo de banco de dados é usado principalmente para aplicações de negócios, como sistemas de gerenciamento de estoque e de vendas.

Já o NoSQL é um tipo de banco de dados não relacional que não utiliza tabelas para armazenar dados. Em vez disso, ele usa outros métodos de armazenamento, como documentos, gráficos ou pares chave-valor. O NoSQL é utilizado principalmente para aplicações que precisam de escalabilidade e flexibilidade, como em redes sociais e aplicativos de jogos.

API

Uma API (Interface de Programação de Aplicativos) é um conjunto de rotinas, protocolos e ferramentas para construir softwares. Ela define como diferentes componentes de software devem interagir entre si, permitindo a comunicação e a integração entre aplicativos e plataformas diferentes.

API REST

API REST é um estilo arquitetural para a criação de serviços web que utiliza os protocolos HTTP e HTTPS para a comunicação entre o cliente e o servidor. Essas APIs são projetadas para serem escaláveis, flexíveis e interoperáveis, permitindo que diferentes sistemas e plataformas se comuniquem de forma eficiente e segura.

Material de estudo - video

- Curso API Rest - <https://www.youtube.com/watch?v=VqVQ7vHY32o&list=PLf8x7B3nFTI17WeEVj405tHlStiq1kNBX>

JSON API

APIs JSON são APIs que utilizam o formato de dados JSON (JavaScript Object Notation) para representar e transmitir informações entre o cliente e o servidor. Esse formato é leve, fácil de ler e interpretar, além de ser amplamente suportado por diversas linguagens de programação e plataformas. As APIs JSON são usadas em diversas aplicações, incluindo serviços web, aplicativos móveis e Internet das Coisas (IoT).

Material de estudo - video

- JSON API em 20 minutos - <https://www.youtube.com/watch?v=BWPUSXzSWA8>

Git e GitHub

Git é um sistema de controle de versão distribuído amplamente usado para rastrear mudanças em projetos de software. Ele permite que várias pessoas trabalhem em um mesmo projeto simultaneamente, mantendo um histórico completo das mudanças, facilitando a colaboração e a manutenção do código.

Material de estudo - video

Conteúdo gratuito e em português de Git e Github:

- [Curso de Git e Github COMPLETO 2021 \[Iniciantes\] + Desafios + Muita Prática - YouTube](#)
- [Curso gratuito Git e Github #1 - Introdução - O que é git? - YouTube](#)

Material de estudo - sites

Sites para treinar Git:

- [Learn Git Branching](#)

- [Visualizing Git \(git-school.github.io\)](https://git-school.github.io/Visualizing-Git/)

Front-end

HTML e CSS

HTML (Hypertext Markup Language) é uma linguagem de marcação usada para criar páginas da web. Ele fornece uma estrutura básica para o conteúdo da página, incluindo cabeçalho, corpo e rodapé. Os elementos HTML são identificados por tags, que indicam ao navegador como renderizar o conteúdo. Os atributos HTML são usados para fornecer informações adicionais sobre os elementos, como a cor do texto ou a largura de uma imagem. HTML é frequentemente combinado com outras tecnologias, como CSS (Cascading Style Sheets) e JavaScript, para criar páginas da web mais interativas e visualmente atraentes. Para criar uma página da web em HTML, basta digitar o código HTML em um editor de texto simples e salvá-lo com a extensão .html. Em seguida, abra o arquivo HTML em um navegador da web para visualizar a página.

Material de estudo - video

Conteúdo gratuito (siga a ordem):

- [Começa aqui o novo @CursoemVideo de HTML5 e CSS3 - YouTube](#)

Material de estudo - site

Sites interativos para treinar HTML e CSS.

- [HTML Tutorial \(w3schools.com\)](https://www.w3schools.com/html/)
- [Flexbox Froggy - Pesquisar \(bing.com\)](https://bing.com/search?q=Flexbox+Froggy)
- [CSS Diner - Where we feast on CSS Selectors! \(flukeout.github.io\)](https://flukeout.github.io/)
- [Grid Garden - A game for learning CSS grid \(cssgridgarden.com\)](https://cssgridgarden.com/)
- [Frontend Mentor | Challenges](https://frontendmentor.io/challenges)

JavaScript - TypeScript

JavaScript é uma linguagem de programação que permite adicionar interatividade e dinamismo a sites e aplicativos. É executado diretamente no navegador do usuário e pode

ser usado para criar animações, validar formulários, alterar o conteúdo de uma página sem recarregá-la e muito mais.

Material de estudo - video

Conteúdo gratuito que recomendo:

- [Curso Grátis de JavaScript Moderno - YouTube](#)
- [Make a JavaScript Drum Kit in Vanilla JS! #JavaScript30 1/30 - YouTube](#)
- [Curso Grátis de JavaScript Moderno - YouTube](#)
- [APRENDA FETCH API DE JAVASCRIPT COM PROJETO - YouTube](#)
- [Curso de NPM - Aula 1: O que é o NPM? - YouTube](#)
- [JavaScript Strict Mode #AluraMais](#)

Typescript

- [Typescript - TypeScript - Zero to Hero | O que é TypeScript? #01](#)

Criação de projetos e portfólio

- [Criando projeto completo com HTML, CSS e JavaScript - YouTube](#)
- [TELA DE LOGIN COM TEMA DARK | HTML + CSS - YouTube](#)

Frameworks

React

- [Curso React: Introdução - #01 - 2021 - YouTube](#)
- [Testando aplicações React com Jest & Testing Library - Decode #012 - YouTube](#)
- [Curso gratuito Tailwind CSS #1 - Introdução e instalação - YouTube](#)

React - projetos e portfólio

- [LANDING PAGE COM HTML E CSS - YouTube](#)
- [Crie um Quiz com React.js - Projeto de React para iniciantes - YouTube](#)
- [CRIANDO PROJETO DO ZERO COM REACTJS 🚀 - YouTube](#)
- [PROJETO DE FILMES COM REACT E API DO TMDB \(REACT ROUTER, REACT HOOKS\) - YouTube](#)

Angular

- Curso Angular - [Curso Angular 13: Introdução - #01 - 2022](#)

Back-end

Backend é a parte do software que não é visível para o usuário final, mas que desempenha um papel fundamental no funcionamento de aplicativos e sites. Em outras palavras, é a parte do software que lida com as funções "nos bastidores", como armazenamento de dados, gerenciamento de conexões, processamento de solicitações do usuário e execução de lógica de negócios.

O backend é geralmente composto por uma combinação de tecnologias e linguagens de programação, como bancos de dados, servidores web, linguagens de programação como Python, Java ou Ruby e frameworks como Django ou Ruby on Rails. É responsabilidade dos desenvolvedores de backend garantir que todas essas peças sejam integradas e funcionem juntas para fornecer uma experiência confiável e eficiente para o usuário final.

Java

Java é uma linguagem de programação popular e amplamente utilizada para desenvolvimento de software. Foi criada em 1995 pela Sun Microsystems e hoje é mantida pela Oracle Corporation. Java é uma linguagem orientada a objetos, o que significa que ela permite que os desenvolvedores organizem o código em "objetos" que possuem propriedades e métodos. Isso torna o código mais modular e fácil de gerenciar.

Uma das características mais importantes do Java é sua portabilidade. O código Java pode ser executado em qualquer plataforma que possua uma máquina virtual Java (JVM), o que significa que um programa Java pode ser executado em uma variedade de sistemas operacionais sem precisar ser reescrito.

Além disso, Java é uma linguagem segura, com recursos incorporados de segurança, como a verificação de tipos e a verificação de segurança de bytecode.

Java é amplamente utilizado para desenvolvimento de aplicativos de desktop, aplicativos web, sistemas de gerenciamento de banco de dados e até mesmo para o desenvolvimento de aplicativos móveis para o sistema operacional Android. Há também uma grande comunidade de desenvolvedores Java e muitas bibliotecas e frameworks disponíveis para facilitar o desenvolvimento de software com Java.

Material de estudos - video

- [Curso de Java #01 - História do Java - Gustavo Guanabara - YouTube](#)
- [Curso POO Teoria #01a - O que é Programação Orientada a Objetos - YouTube](#)
- [Tratamento de exceções : aprenda na prática : exemplo Java - YouTube](#)
- [Aprofundando no Java: domine as Collections - YouTube](#)
- [Curso de Java - Socket - Programação Iniciante - YouTube](#)

- [Aprenda Como Conectar JAVA com BANCO DE DADOS usando JDBC - YouTube](#)
- [JAVA com JPA #1 - Criando do zero projeto JPA com Hibernate - YouTube](#)
- [Introdução a Servlets e JSP - YouTube](#)
- [Spring Boot | Curso Completo 2022 - YouTube](#)
- [Apache Maven para programadores Java iniciantes - YouTube](#)
- [#01 Teste Unitário em Java - Conceito - YouTube](#)

Prática

- [Java Tutorial \(w3schools.com\)](#)

C#

C# (pronuncia-se "C sharp") é uma linguagem de programação moderna e orientada a objetos desenvolvida pela Microsoft. Ela foi projetada para ser simples, eficiente e segura. Assim como Java, C# é uma linguagem compilada, o que significa que o código é traduzido em código de máquina antes da execução. C# é frequentemente usado para desenvolvimento de aplicativos Windows e jogos, bem como para o desenvolvimento de aplicativos Web usando o framework ASP.NET.

C# é considerada uma linguagem fácil de aprender, pois possui uma sintaxe clara e concisa. Além disso, a linguagem oferece recursos poderosos de gerenciamento de memória, como garbage collection, que reduzem a probabilidade de bugs de memória e vazamentos de memória. A linguagem C# é compatível com a plataforma .NET da Microsoft, o que significa que os desenvolvedores podem usar uma ampla variedade de bibliotecas e frameworks para criar aplicativos. Alguns dos frameworks mais populares incluem o .NET Framework e o .NET Core. C# é uma linguagem bem estabelecida e amplamente usada, com uma grande comunidade de desenvolvedores. Isso significa que há muitos recursos disponíveis para os desenvolvedores, incluindo documentação detalhada, fóruns de suporte e bibliotecas de código aberto.

Material de estudos - video

- [Curso Programação Completo C# - Aula 01 - YouTube](#)
- [C# POO - 01 O que é programação orientada a objetos - YouTube](#)
- [Introdução - Teste Unitário com C# - Parte 01 - Curso Completo - YouTube](#)
- [Projeto do curso de .NET Core C# | Mod 01 - Aula 01 - YouTube](#)
- [ASP.Net MVC - C# - Criando uma aplicação do Zero - YouTube](#)
- [Aula 01 - Curso completo Asp.net Core - YouTube](#)

Prática

- [C# Tutorial \(C Sharp\) \(w3schools.com\)](#)

Python

Python é uma linguagem de programação interpretada, de alto nível e orientada a objetos. Ela é conhecida por sua sintaxe clara e concisa, que torna o código fácil de ler e escrever.

Python é usada em uma ampla variedade de aplicações, desde desenvolvimento web até ciência de dados e automação de tarefas. Ela é uma das linguagens de programação mais populares atualmente e possui uma grande comunidade de desenvolvedores.

Uma das características distintivas de Python é sua ênfase na legibilidade do código. A linguagem usa indentação significativa para delimitar blocos de código, o que torna o código mais fácil de entender e visualmente mais limpo. Além disso, Python possui uma ampla variedade de bibliotecas de código aberto, que simplificam o processo de desenvolvimento de software.

Outra característica importante de Python é sua flexibilidade. Ela é uma linguagem interpretada, o que significa que não é necessário compilar o código antes da execução. Isso permite que os desenvolvedores possam testar e iterar rapidamente no código. Além disso, Python é compatível com uma ampla variedade de plataformas, o que significa que o mesmo código pode ser executado em diferentes sistemas operacionais.

Em resumo, Python é uma linguagem de programação popular, de alto nível, orientada a objetos e flexível, que é usada em muitas aplicações diferentes

Material de estudos - video

- Django - [Como Sair do Zero em Django no Python \[Passo a Passo Primeiro Site\]](#)
- Flask - [Como Criar e Publicar um Site em Python com Flask](#)
- [Curso Python #01 - Seja um Programador - YouTube](#)

Material de estudos - Python para análise de dados e engenharia de dados

- Pandas - [Curso de Pandas - Python - Introdução ao curso \(versão ago/2020\)](#)
- Numpy - [Como Sair do Zero com a Biblioteca Numpy no Python](#)
- SQL Alchemy - [Como usar Banco de Dados em Python sem precisar de SQL | SQLAlchemy ORM | Tutorial parte 1/2](#)

Material de estudos - Python para ciência de dados

- Matplotlib - [Como sair do ZERO em Gráficos no Python \[Matplotlib\]](#)
- Seaborn - [Como sair do ZERO em Gráficos no Python \[Seaborn\]](#)
- SciPy - <https://www.youtube.com/watch?v=CY8eqENHCr0>
- Scikit-learn - [Aula 01 - Aprendizado de Máquina com Python e scikit-learn](#)
- TensorFlow - [Aula 01 - Aprendizado de Máquina com Python e scikit-learn](#)

Ruby

Ruby é uma linguagem de programação dinâmica e orientada a objetos que foi criada em 1995 por Yukihiro "Matz" Matsumoto. É uma linguagem de script interpretada e de alto nível que é frequentemente usada para desenvolvimento web, mas também pode ser usada para automação de tarefas, desenvolvimento de jogos, entre outras aplicações.

Algumas das características mais importantes do Ruby incluem sua sintaxe clara e concisa, tipagem dinâmica e coleta de lixo automática de memória. Além disso, a linguagem é altamente extensível e suporta metaprogramação, o que significa que os desenvolvedores podem escrever código que modifica ou estende a própria linguagem.

Aqui estão alguns links úteis para aprender Ruby:

- Documentação oficial do Ruby: <https://www.ruby-lang.org/pt/documentation/>
- Ruby Programming (Codecademy): <https://www.codecademy.com/learn/learn-ruby>
- Learn Ruby the Hard Way (Zed Shaw): <https://learncodethehardway.org/ruby/>
- Ruby on Rails Tutorial (Michael Hartl): <https://www.railstutorial.org/book>
- Ruby Weekly: newsletter semanal com as últimas notícias e recursos relacionados a Ruby: <https://rubyweekly.com/>
- RubyGems: repositório de pacotes Ruby para instalação de bibliotecas e frameworks: <https://rubygems.org/>

Material de estudos - video

- Ruby para iniciantes - [Ruby Para Iniciantes #Aula 00 - O Poder da Linguagem Ruby](#)
- Ruby On Rails - [Ruby on Rails para iniciantes - Aula 01 - A história do Ruby / Ruby on Rails](#)
-

Ambos os tipos de banco de dados possuem vantagens e desvantagens. O SQL é altamente estruturado e escalável, com suporte para transações complexas e consultas ad-hoc. Já o NoSQL é mais flexível, com suporte para grandes volumes de dados e alta disponibilidade.

Em resumo, a escolha entre SQL e NoSQL depende das necessidades específicas da aplicação. O SQL é uma boa opção para aplicações de negócios que precisam de um alto nível de estruturação e suporte para transações complexas. Já o NoSQL é ideal para aplicações que precisam de escalabilidade e flexibilidade, como em redes sociais e aplicativos de jogos.

- Curso modelagem de dados - [Modelagem de Dados - Conceitos de Bancos de Dados](#)

SQL Server

O SQL Server é um sistema de gerenciamento de banco de dados relacional desenvolvido pela Microsoft. Ele é projetado para armazenar e recuperar dados solicitados por outras aplicações de software.

Para pequenos projetos ou necessidades simples, o SQL Server pode ser instalado em um computador pessoal ou em um servidor de baixa potência. Ele pode ser usado para armazenar informações em pequena escala, como um banco de dados para um pequeno negócio ou para uso pessoal.

O SQL Server é uma opção popular para muitos desenvolvedores, devido à sua facilidade de uso e versatilidade. Ele oferece recursos avançados para gerenciamento de dados, como a capacidade de criar índices, gerenciar transações e realizar consultas complexas.

No entanto, é importante lembrar que, para projetos maiores ou com necessidades mais complexas, pode ser necessário utilizar outras ferramentas ou recursos de banco de dados mais avançados.

- Curso SQL Server - [SQL SERVER - 01 - SQL Server Express Edition - Instalando no seu computador](#)

MongoDB

MongoDB é um banco de dados NoSQL (não relacional) orientado a documentos, projetado para armazenar, recuperar e gerenciar grandes volumes de dados não estruturados. Ele é desenvolvido pela MongoDB Inc. e é usado por muitas empresas para aplicativos baseados em nuvem e móveis.

Ao contrário de bancos de dados relacionais, que usam tabelas com linhas e colunas, o MongoDB armazena dados em documentos BSON (Binary JSON) que podem ser aninhados. Isso permite que o MongoDB armazene dados sem uma estrutura rígida, o que o torna mais flexível para armazenar diferentes tipos de dados.

O MongoDB suporta uma variedade de recursos, como consultas ad-hoc, indexação, replicação e balanceamento de carga automático. Ele também suporta consultas geoespaciais e processamento de agregação.

O MongoDB é escalável horizontalmente, o que significa que é possível aumentar a capacidade de armazenamento e processamento do banco de dados adicionando mais servidores ao cluster. Ele também é altamente disponível, o que significa que os dados podem ser acessados e atualizados mesmo em caso de falha de servidor.

O MongoDB é compatível com muitas linguagens de programação, incluindo Java, C++, Python, Ruby e muitas outras. Ele também é executado em várias plataformas, incluindo Windows, Linux e MacOS.

- MongoDB para iniciantes - [Curso de MongoDB fundamental | Aprenda MongoDB em 1 hora - 2021](#)

-

DevOps

DevOps é um conjunto de práticas e metodologias que visam a colaboração e integração entre as equipes de desenvolvimento de software e operações de infraestrutura, com o objetivo de entregar produtos de software de forma mais rápida, eficiente e com qualidade. Essas práticas incluem automação de processos, monitoramento constante do ambiente, cultura de feedback e comunicação aberta entre as equipes envolvidas. A adoção do DevOps pode trazer diversos benefícios, como redução de falhas e retrabalho, aumento da eficiência e melhorias na experiência do usuário final.

Linguagem de programação

Não há uma linguagem de programação específica que seja considerada a "melhor" para DevOps, pois cada linguagem tem suas próprias vantagens e desvantagens. O importante é escolher uma linguagem que atenda às necessidades específicas do projeto e que seja fácil de usar e entender para a equipe de desenvolvimento e operações. Algumas das linguagens mais comuns para automação de DevOps incluem Python, Bash, Ruby, PowerShell e Go. Python, por exemplo, é popular devido à sua facilidade de uso, bibliotecas de automação e grande comunidade de usuários. Bash é frequentemente usado para scripts de linha de comando e automação de tarefas de sistema, enquanto Ruby é conhecido por sua simplicidade e flexibilidade. A escolha da linguagem certa dependerá das necessidades do projeto e das habilidades da equipe.

Virtualização

- Material de estudo - curso [Como criar uma máquina virtual - Aula 01](#)

Automação

Material de estudos - video

- Curso programação em Bash - [Intensivo de Programação em Bash #1: Introdução](#)
- Ansible -
 - Primeiros passos - [Primeiros passos no Ansible.](#)

- Curso Ansible - [Ansible - Aula 1](#)
- Terraform - https://www.youtube.com/watch?v=bIPF_hzmQGE&list=PLWQmZVQayUUlgSmOj3GPH2BJcn0hOzlaP
- Jenkins
 - Introdução CI/CD - [O básico de Pipelines CI/CD com Jenkins](#)

Containers

Material de estudos - video

- Docker - [Preparando ambiente de estudos/desenvolvimento Docker | Curso Docker #001](#)
- Kubernetes
 - Introdução - [#01 Revisão Docker - Curso de Introdução ao Kubernetes](#)
 -

Monitoramento e logs

Material de estudos - video

- Splunk -
 - Basico - [Splunk básico - apresentação](#)
 -
- Grafana
 - Primeiros passos - [AULA GRAFANA - Primeiros passos com o viewer open source GRAFANA](#)

Cyber Security

Trabalhar com segurança cibernética envolve proteger sistemas de computadores, redes e dados contra ataques maliciosos de hackers, criminosos cibernéticos e outras ameaças digitais. Isso pode incluir o desenvolvimento e implementação de políticas de segurança, a configuração de firewalls e outras medidas de proteção, a realização de testes de segurança, a detecção e resposta a incidentes de segurança e a educação dos usuários sobre práticas seguras de computação. Os profissionais de segurança cibernética precisam ter conhecimentos técnicos sólidos, habilidades de resolução de problemas e um forte

senso de ética e responsabilidade para proteger informações confidenciais e sistemas críticos.

Material de estudo - videos

- [Curso de Hardware #01 - Principais Componentes de um PC](#)
- [Como vai ser o Curso de Redes? - Curso Redes #01](#)
- [Aula 1 - Fundamentos técnicos da Criptografia com Nathalia Sautchuk | Curso Criptografia e Sociedade](#)
- [\[Videoaula\] Incidentes de Segurança: Ameaças x Vulnerabilidades x Riscos](#)
- Videoaula] Incidentes de Segurança: Principais Tipos de Ataques - <https://www.youtube.com/watch?v=avtLJcyMfng>
- [\[Videoaula\] Identificação, Combate e Resposta a Incidentes de Segurança](#)

Material de estudo - sites

- [HackTheBox](#)
- [TryHackMe](#)
- Ambientes Vulneráveis para treinamento - [GitHub - vulhub/vulhub: Pre-Built Vulnerable Environments Based on Docker-Compose](#)
- Competição de Segurança Cibernética - [picoCTF](#)

Dados

A área de análise, ciência e engenharia de dados está relacionada à utilização de técnicas computacionais para coletar, processar, analisar e interpretar grandes volumes de dados, com o objetivo de obter insights valiosos e tomar decisões informadas.

A análise de dados envolve a aplicação de métodos estatísticos e matemáticos para identificar padrões e tendências nos dados.

Já a ciência de dados combina técnicas de análise de dados com conhecimentos de programação, estatística e ciência da computação para criar modelos preditivos e algoritmos de aprendizado de máquina capazes de automatizar processos e identificar relações complexas nos dados.

A engenharia de dados, por sua vez, está relacionada à gestão e processamento de grandes volumes de dados, incluindo a coleta, armazenamento, limpeza e integração de diferentes fontes de dados. Isso envolve a utilização de tecnologias como bancos de dados, ferramentas de ETL (Extract, Transform, Load) e plataformas de processamento de dados

em larga escala, como o Hadoop. Em conjunto, a análise, ciência e engenharia de dados são essenciais para a tomada de decisões informadas em diversas áreas, desde a gestão empresarial e financeira até a pesquisa científica e médica.


Análise de dados

A análise de dados é uma disciplina ampla que abrange uma variedade de técnicas e metodologias. Algumas das principais áreas de análise de dados incluem:




- **Análise descritiva:** a análise descritiva é a técnica mais básica de análise de dados e envolve a utilização de estatísticas descritivas, como média, mediana e desvio padrão, para descrever características dos dados.
- **Análise exploratória de dados:** a análise exploratória de dados envolve a utilização de técnicas gráficas e estatísticas para explorar e visualizar padrões nos dados. Isso pode incluir a criação de gráficos e visualizações de dados para ajudar a identificar tendências e padrões.
- **Análise inferencial:** a análise inferencial envolve a utilização de técnicas estatísticas para fazer inferências a partir dos dados, como a realização de testes de hipóteses ou a criação de modelos de regressão.
- **Análise preditiva:** a análise preditiva envolve a utilização de algoritmos de aprendizado de máquina e modelos estatísticos para prever o comportamento futuro dos dados.
- **Análise de séries temporais:** a análise de séries temporais envolve a utilização de técnicas estatísticas para modelar e prever o comportamento de dados que mudam ao longo do tempo.

A análise de dados é uma habilidade cada vez mais importante em muitas profissões, desde a ciência e engenharia até a gestão de negócios e marketing. É uma maneira poderosa de extrair insights e tomar decisões informadas a partir de grandes volumes de dados.

Material de estudos - video

-  [Aula 01 - Recursos Essenciais e Produtividade no Excel](#)

SQL

-  [SQL SERVER - 01 - SQL Server Express Edition - Instalando no seu computador](#)
- [Python](#) - Clique para acessar o conteúdo
-  [Curso Básico de Power BI - Do Zero Ao Dashboard - 100% Gratuito!](#)
-  [\[Power BI\] Introdução ao Power BI](#)

Engenharia de dados




Engenharia de dados é uma área da tecnologia que se concentra no desenvolvimento de sistemas e infraestruturas de dados para coletar, armazenar, processar e analisar grandes volumes de dados. O objetivo é fornecer informações úteis para apoiar as decisões de negócios, bem como melhorar a eficiência e a produtividade.

Os engenheiros de dados são responsáveis por projetar e implementar soluções de armazenamento de dados, como bancos de dados relacionais, bancos de dados NoSQL e sistemas de arquivos distribuídos. Eles também trabalham com ferramentas ETL (Extração, Transformação e Carga), como Apache NiFi, Talend e AWS Glue, para mover dados de uma fonte para outra, limpando e transformando os dados no processo.



Além disso, os engenheiros de dados também são responsáveis por criar soluções escaláveis e duráveis de armazenamento e processamento de dados, incluindo arquiteturas de dados como Data Warehousing, Data Lake e Lambda Architecture. Eles trabalham com tecnologias de processamento distribuído, como Apache Spark e Apache Flink, para processar grandes volumes de dados com eficiência.

Para se tornar um engenheiro de dados, é importante ter conhecimento em várias tecnologias e ferramentas, bem como habilidades em matemática e estatística, programação, arquitetura de software, resolução de problemas, modelagem de dados e compreensão do negócio.

Material de estudos - video

-  Introdução à Arquitetura de Dados - Parte 1 (Overview)
- [SQL](#) - Clique para acessar o conteúdo
- [Python](#) - Clique para acessar o conteúdo
- Apresentação do Curso | Fundamentos de Big Data e Apache Hadoop #01 - https://www.youtube.com/watch?v=NmzNlov-HOI&list=PLFJLvHW_AAoCgaQLMu9tgYyOtc0m3-tDV
-  Curso de MongoDB fundamental | Aprenda MongoDB em 1 hora - 2021
-  UNAERP - Minicurso Apache NiFi - 04/11/2020
-  Processando dados com Spark na Prática - Parte 1

Cloud

-  O que é AWS e como aprender mais com o Curso Gratuito Amazon Web Servi...
-  TUTORIAL GOOGLE BIGQUERY E GOOGLE DATASTUDIO COM CASO ...

Ciência de dados



Ciência de dados é uma área multidisciplinar que utiliza métodos estatísticos, matemáticos e de computação para extrair conhecimento e insights de dados complexos. O objetivo é ajudar as empresas a tomar decisões baseadas em dados e a encontrar soluções para problemas complexos.

Os cientistas de dados coletam, limpam e analisam grandes conjuntos de dados, usando ferramentas como Python, R, SAS e SQL. Eles aplicam técnicas estatísticas, aprendizado de máquina e mineração de dados para extrair informações valiosas dos dados.

Os cientistas de dados trabalham em vários setores, incluindo finanças, saúde, varejo, tecnologia e governo. Eles colaboram com equipes de negócios para entender os objetivos do negócio e fornecer insights relevantes e acionáveis. Eles também desenvolvem modelos e algoritmos preditivos para prever tendências e comportamentos futuros.

Para se tornar um cientista de dados, é importante ter habilidades em matemática, estatística, programação e análise de dados. Além disso, é importante ter uma compreensão do negócio e habilidades de comunicação para trabalhar efetivamente com outras equipes e líderes empresariais.

Material de estudos - video

- [SQL](#) - Clique para acessar o conteúdo
- [Python](#) - Clique para acessar o conteúdo
-  Preencher Dados Faltantes - Machine Learning 01
-  Curso de NPM - Aula 1: O que é o NPM?

Sites para treinamento

Aprendizado Interativo:

- Codecademy: [Codecademy](#) - plataforma de aprendizado interativo para cursos em diversas linguagens de programação
- FreeCodeCamp: [freeCodeCamp](#) - plataforma de aprendizado interativo para áreas como desenvolvimento web, ciência de dados e aprendizado de máquina
- DataCamp: [DataCamp](#) - plataforma de aprendizado interativo focada em ciência de dados e análise de dados
- edX: [edX](#) - plataforma de aprendizado online que oferece cursos em diversas áreas, incluindo ciência da computação e tecnologia

Competições e Desafios:

- HackerRank: [HackerRank](#) - plataforma de programação competitiva que oferece desafios de programação e competições
- CodeWars: [Codewars](#) - plataforma de desafios de programação que oferece oportunidades para melhorar suas habilidades em diversas linguagens de programação
- TopCoder: [Topcoder](#) - plataforma de programação competitiva que oferece desafios em diversas áreas, incluindo algoritmos e estruturas de dados

Recursos de Aprendizado:

- Stack Overflow: [Stack Overflow](#) - fórum online de perguntas e respostas relacionadas à programação e tecnologia

- W3Schools: [W3Schools](#) - plataforma de recursos e tutoriais para aprendizado de desenvolvimento web
- Mozilla Developer Network: [MDN Web Docs](#) - plataforma de recursos e tutoriais para desenvolvimento web e tecnologias relacionadas

Desenvolvimento Web:

- CodePen: [CodePen](#) - plataforma de desenvolvimento web que permite compartilhar e testar código HTML, CSS e JavaScript
- CSS-Tricks: [CSS-Tricks](#) - plataforma de recursos e tutoriais para aprendizado de CSS
- Bootstrap: [Bootstrap](#) - framework de desenvolvimento web que ajuda a criar sites responsivos e com boa aparência

Desenvolvimento Mobile:

- Swift Playgrounds: [Swift Playgrounds - Apple](#) - plataforma de aprendizado interativo para a linguagem de programação Swift, usada no desenvolvimento de aplicativos iOS
- Android Developers: [Android Developers](#) - plataforma de recursos e tutoriais para desenvolvimento de aplicativos Android
- React Native: [React Native](#) - framework de desenvolvimento mobile que permite criar aplicativos para iOS e Android com código JavaScript

Ciência de Dados:

- Kaggle: [Kaggle](#) - plataforma de competições e projetos relacionados a ciência de dados e aprendizado de máquina
- Coursera: [Coursera](#) - plataforma de aprendizado online que oferece cursos em diversas áreas, incluindo ciência de dados e aprendizado de máquina
- TensorFlow: [TensorFlow](#) - biblioteca de aprendizado de máquina de código aberto desenvolvida pelo Google

Aprendizado de Máquina:

- Fast.ai: [Fast.ai](#) - plataforma de aprendizado de máquina que oferece cursos práticos e aplicados
- PyTorch: [PyTorch](#) - biblioteca de aprendizado de máquina de código aberto desenvolvida pelo Facebook
- Machine Learning Mastery: [Machine Learning Mastery](#)