

Roadmap JavaScript

- #JavaScript

Nível Iniciante:

1. Introdução ao JavaScript:

- Entenda o que é JavaScript, seu papel no desenvolvimento web e sua história.
- Aprenda como incluir JavaScript em documentos HTML.

1. Sintaxe Básica:

- Estude variáveis, tipos de dados (números, strings, booleanos), operadores e expressões.
- Compreenda a importância de pontos e vírgulas no JavaScript.

1. Estruturas de Controle:

- Aprenda sobre estruturas de controle de fluxo, como condicionais (if-else) e loops (for, while).
- Entenda a importância das declarações de bloqueio.

1. Funções:

- Crie funções simples.
- Aprenda sobre parâmetros e retorno de funções.
- Explore escopo de variáveis (escopo global e local).

1. Arrays e Objetos:

- Entenda como trabalhar com arrays e objetos em JavaScript.
- Aprenda sobre métodos de array como push, pop, splice, etc.

1. Manipulação do DOM:

- Saiba como selecionar elementos HTML com JavaScript.
- Aprenda a manipular conteúdo, estilos e atributos do DOM.

1. Eventos:

- Familiarize-se com a captura e o tratamento de eventos.
- Aprenda a usar ouvintes de eventos (event listeners).

Nível Intermediário:

1. Manipulação de Strings e Datas:

https://www.youtube.com/watch?v=IAsJw5V_mn0 – Conheça os principais métodos para strings no Javascript

<https://www.youtube.com/watch?v=rFMss-lWUjw&t=45s> - Operando com Datas em JavaScript

- Explore métodos para trabalhar com strings.
- Entenda como manipular datas e horários em JavaScript.

2. Aprenda sobre Arrow Functions:

<https://www.youtube.com/watch?v=khlpCir4auw> - Arrow Functions em JavaScript

- Compreenda a sintaxe das funções de seta (=>).
- Entenda como elas simplificam a escrita de funções em JavaScript.

ES6 e Recursos Modernos:

<https://www.youtube.com/watch?v=RtfBx90R070&list=PLlAbYrWSYTiPQ1BE8klOtheBC0mtL3hEi> – Ver esse 1, pois é justamente ES6 moderno - Curso JS Moderno (ES6+)

<https://www.youtube.com/watch?v=ruoHSuTKp-U> - Desestruturação no javascript com exemplos - Destructuring Assignment JS

- Explore recursos modernos do JavaScript, como let/const, classes, destructuring, etc.
- Familiarize-se com módulos ES6 para organizar seu código.

1. Trabalhando com Promises:

<https://www.youtube.com/watch?v=WUmAAxH9n-A&t=120s> - Curso de JavaScript Assíncrono - Callbacks, Promises & Async/Await

- Aprenda sobre assincronia em JavaScript.
- Compreenda o uso de Promises para lidar com operações assíncronas.

1. AJAX e Requisições HTTP:

<https://www.youtube.com/watch?v=QExV413mers&t=133s> - AJAX na Prática: Requisições HTTP com JavaScript

<https://www.youtube.com/watch?v=GngMMN93jik> - JavaScript - Como realizar requisições AJAX

https://www.youtube.com/watch?v=OQdXCDRvRU&list=PL2bJNatYYfGHTTOv1_ivXa2C_zLp_MW6&pp=0gcJCbEEOCosWNin – (Analisar depois

esses curso de javascript, com jquery e o ajax claro) - Fundamentos Essenciais: JavaScript, jQuery e AJAX

- Saiba como fazer requisições AJAX para buscar e enviar dados em uma aplicação web.
- Explore bibliotecas como Axios ou o uso nativo do `fetch`.

<https://www.youtube.com/watch?v=1O5GyaGcp0s> - FETCH API na Prática: Consumindo API com JavaScript

1. Manuseio de Exceções:

<https://www.youtube.com/watch?v=5dQ6FQ79-WQ> – Exceções: try, catch, finally com Javascript

<https://www.youtube.com/watch?v=PknztTIBGUk> - Como usar TRY CATCH e THROW ERROR no JAVASCRIPT? AULA para INICIANTES

- Compreenda como lidar com erros e exceções em JavaScript.
- Aprenda sobre o uso de blocos try-catch.

Nível Avançado:

1. Frameworks e Bibliotecas:

- Escolha um framework JavaScript, como React, Angular ou Vue, para aprender e criar aplicações mais complexas.

Resposta: VOU APRENDER REACT.JS e Possivelmente

- Explore bibliotecas populares como jQuery ou lodash, conforme necessário.

1. Gerenciamento de Estado:

- <https://www.youtube.com/watch?v=qnVuDePDavA> - Gerenciando estado em aplicações React com Redux 2020

<https://www.youtube.com/watch?v=unfHrB2sGGM> - Gerenciamento de estado no React (por que não uso mais Redux) com React

1. Testes e Depuração:

- Saiba como escrever testes unitários e de integração para seu código JavaScript.
- Aprenda a usar ferramentas de depuração, como o DevTools do navegador.

1. Segurança em JavaScript:

- Compreenda as práticas de segurança em JavaScript, como evitar injeção de código (XSS) e outros ataques comuns.

1. Otimização de Desempenho:

- Aprenda a otimizar o desempenho de suas aplicações JavaScript, incluindo carregamento rápido de página e minimização de solicitações de rede.

1. Integração com Back-End:

- Explore como integrar JavaScript com um servidor back-end, seja com Node.js ou outro framework/server.