RESUMO DA AULA

19 Importar e exportar módulos JavaScript

Nesta aula, vamos explorar o conceito de m dulos no JavaScript. Em resumo, m dulos s o arquivos que cont m funcionalidades, classes ou vari veis que podemos importar e utilizar quando necess rio. Eles nos permitem organizar o c digo de maneira mais eficiente, garantindo reusabilidade e manuten o simplificada. O que s o m dulos? M dulos s o partes de c digo encapsuladas em arquivos JavaScript que podem ser importadas e utilizadas em outros arquivos. A grande vantagem de modularizar o c digo a facilidade de reutiliza o. Ou seja, com um m dulo, voc consegue usar o mesmo trecho de c digo em v rias partes do projeto, sem precisar copiar e colar o mesmo bloco repetidamente. Al m disso, m dulos facilitam a manuten o do c digo. Se um trecho de c digo precisar ser alterado, voc s precisa alterar o m dulo correspondente, e todos os arquivos que o importam j passar o a utilizar a vers o atualizada. Como criar e exportar um m dulo Vamos criar um m dulo que cont m algumas fun es matem ticas b sicas, como somar e multiplicar n meros. Primeiro, dentro da pasta xx-modulos-javascript, crie um arquivo chamado operacoes-matematicas.js e adicione a seguinte fun o para somar dois n meros: function somar(numero1, numero2) { return numero1 + numero2; } Agora, vamos tornar essa fun o "export vel" para que possamos utiliz -la em outros arquivos. Para isso, usamos a palavra-chave export: export function somar(numero1, numero2) { return numero1 + numero2; } Com o export, estamos dizendo para o JavaScript que queremos tornar essa fun o p blica e permitir que outros arquivos a importem e utilizem. Importando um m dulo Agora, no arquivo index.js, vamos importar a fun o somar que criamos no operacoes-matematicas.js. Para isso, usamos a sintaxe import: import { somar } from './operacoes-matematicas.js'; Observe a estrutura de importa o: { somar } indica que estamos importando especificamente a fun o somar. from './operacoes-matematicas.js' especifica de onde estamos importando a fun o. Nesse caso, estamos apontando para o arquivo operacoes-matematicas.js que est na mesma pasta do index.js (representado pelo ./). Ap s a

importa o, podemos usar a fun o somar normalmente no nosso c digo: console.log(somar(1, 3)); // Output: 4 Uma observa o importante: para usar m dulos no navegador, precisamos definir o tipo de script como module. Para isso, no arquivo index.html, altere a tag <script> para: <script type="module" src="index.js"></script> Sem essa configura o, o navegador n o consegue reconhecer as instru es de import e export e gerar um erro. Exportando v rias fun es de um mesmo m dulo Agora, vamos adicionar mais uma fun o ao m dulo operacoes-matematicas.js. Vamos criar uma fun o chamada multiplicar: function multiplicar(numero1, numero2) { return numero1 * numero2; } Se quisermos exportar ambas as fun es (somar e multiplicar), podemos usar uma exporta o agrupada: export { somar, multiplicar }; No index.js, atualizamos a importa o para incluir a nova fun o: import { somar, multiplicar } from './operacoes-matematicas.js'; console.log(somar(1, 3));

// Output: 4 console.log(multiplicar(2, 3)); // Output: 6 Dessa forma, conseguimos importar v rias fun es de um mesmo m dulo de maneira organizada e centralizada. Exporta es padr o e nomeadas No JavaScript, existem dois tipos principais de exporta es: Exporta o Nomeada (Named Export): Como o exemplo que usamos acima, onde especificamos exatamente quais fun es queremos exportar. A importa o feita utilizando {} para selecionar as fun es espec ficas. Exporta o Padr o (Default Export): Quando um m dulo tem apenas uma funcionalidade principal para exportar. Nesse caso, usamos export default: No arquivo operacoes-matematicas.js, podemos definir a fun o somar como a exporta o padr o: export default function somar(numero1, numero2) { return numero1 + numero2; } Com isso, no index.js, a importa o muda: import somar from './operacoes-matematicas.js'; Note que, ao usar export default, n o precisamos usar {} para importar a fun o. Isso porque s pode haver um export default por m dulo, tornando-o o "principal" recurso exportado.