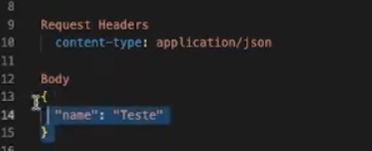
Body é o corpo da requisição

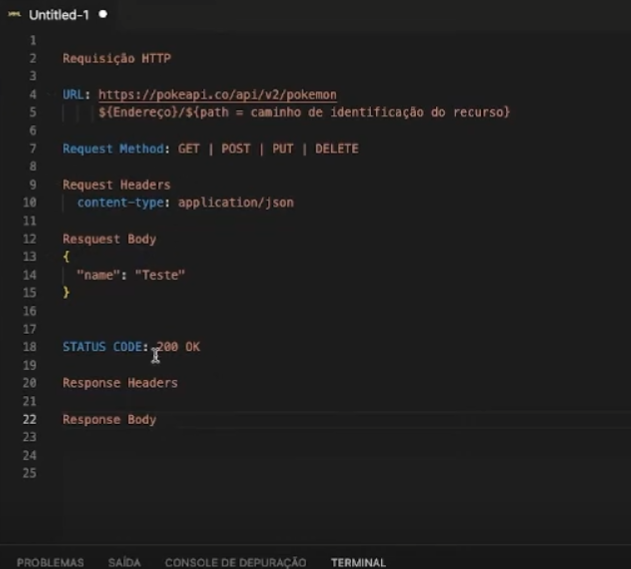
Vamos usar ele para enviar os dados para o servidor, nesse caso vamos usar ele para fazer um put, post, patch...

Onde vamos enviar no caso uma requisição para atualizar um dado e no corpo dela vamos colocar os dados.





Configuração completa de tudo que volta do servidor.



Claro, Alleph! Aqui está um resumo em JavaScript que representa a estrutura básica de uma requisição HTTP para a PokéAPI, com comentários explicativos para facilitar o entendimento e reutilização:

// Requisição HTTP para PokéAPI usando fetch const urlBase = "https://pokeapi.co/api/v2"; const recurso = "pokemon"; // Caminho de identificação do recurso const endpoint = `${urlBase}/${recurso}`; const options = { method: "GET", // Pode ser GET, POST, PUT ou DELETE headers: { "Content-Type": "application/json" }, // Para métodos como POST ou PUT, você pode incluir um corpo: // body: JSON.stringify({ name: "Teste" }) }; // Executa a requisição fetch(endpoint, options) .then(response => { console.log("Status Code:", response.status); // Exemplo: 200 OK return response.json(); }) .then(data => { console.log("Dados recebidos:", data); // Corpo da resposta }) .catch(error => { console.error("Erro na requisição:", error); });

Esse script pode ser salvo como pokeapi-request.js e adaptado conforme o tipo de requisição que você quiser testar. Se quiser que eu transforme isso em uma função reutilizável ou adicione suporte para outros métodos, é só dizer!

// Requisição HTTP para PokéAPI usando fetch

const urlBase = "https://pokeapi.co/api/v2";

const recurso = "pokemon"; // Caminho de identificação do recurso

const endpoint = `${urlBase}/${recurso}`;

const options = {

method: "GET", // Pode ser GET, POST, PUT ou DELETE

headers: {

"Content-Type": "application/json"

},

// Para métodos como POST ou PUT, você pode incluir um corpo:

// body: JSON.stringify({ name: "Teste" })

};

// Executa a requisição

fetch(endpoint, options)

.then(response => {

console.log("Status Code:", response.status); // Exemplo: 200 OK

return response.json();

})

.then(data => {

console.log("Dados recebidos:", data); // Corpo da resposta

})

.catch(error => {

console.error("Erro na requisição:", error);

});