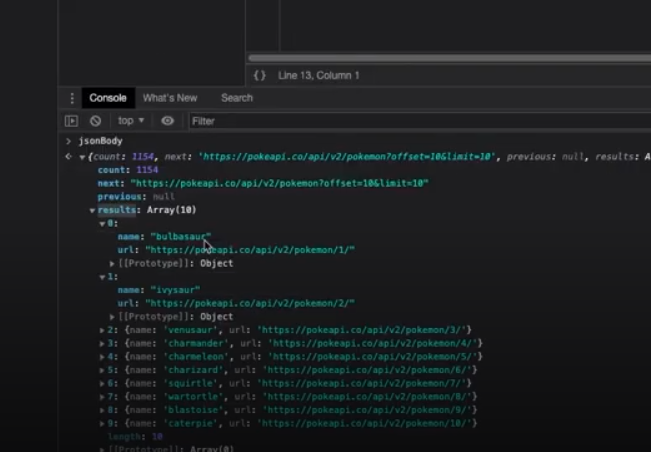
Agora que já sabemos o que devemos fazer, vamos consumir a API



Sabemos que em results, vamos ter a lista com os itens do pokemon

Então sabemos que é ele que precisamos manipular

Entao vamos entrar no URL -> converter para JSON -> Entrar nos convertidos e pegar o results que é onde esta nossos pokemons

E jogar para a listaDePokemons o resultado.

const offset = 0 // Qual pagina vamos carregar, no caso a pagina 0

const limit = 10 // Limite de 10 itens nessa pagina

const url = `https://pokeapi.co/api/v2/pokemon?offset=${offset}&limit=${limit}` // URL de onde vamos puxar.

fetch(url)

    .then((response) => response.json()) // Pegamos o response e estamos retornando em formato JSON

    .then((jsonBody) => jsonBody.results) // Results é onde fica nossa lista de pokemons

    .then((listaPokemons) => {

        debugger

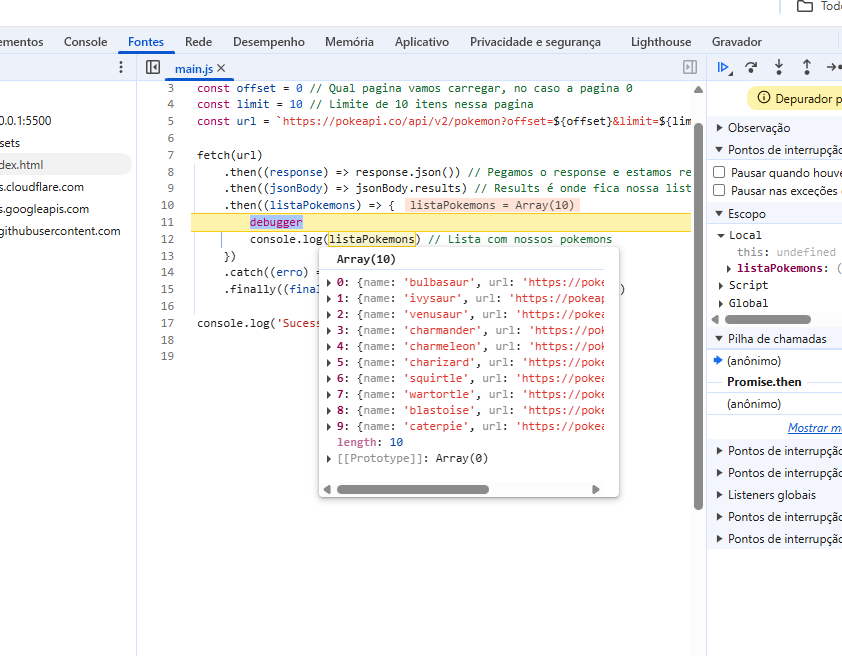
        console.log(listaPokemons) // Lista com nossos pokemons

    })

    .catch((erro) => console.log(erro))

    .finally((final) => console.log('Codigo finalizado' + final))

console.log('Sucesso')



Agora sim vamos ter o array com 10 pokemons

Agora vamos uma função para retornar o pokemon e adicionar no nosso HTML e vamos fazer isso.

const offset = 0 // Qual pagina vamos carregar, no caso a pagina 0

const limit = 10 // Limite de 10 itens nessa pagina

const url = `https://pokeapi.co/api/v2/pokemon?offset=${offset}&limit=${limit}` // URL de onde vamos puxar.

function converterPokemonParaHTML(pokemon) {

    return `

        <li class="pokemon">

                <span class="number">#001</span>

                <span class="name">${pokemon.name}</span>

                <div class="detail">

                    <ol class="types">

                        <li class="type">grass</li>

                        <li class="type">poison</li>

                    </ol>

                    <img src="https://raw.githubusercontent.com/PokeAPI/sprites/master/sprites/pokemon/other/dream-world/1.svg"

                         alt="${pokemon.name}">

                </div>

        </li>

    `

}

fetch(url)

    .then((response) => response.json()) // Pegamos o response e estamos retornando em formato JSON

    .then((jsonBody) => jsonBody.results) // Results é onde fica nossa lista de pokemons

    .then((listaPokemons) => {

        debugger

        console.log(listaPokemons) // Lista com nossos pokemons

    })

    .catch((erro) => console.log(erro))

    .finally((final) => console.log('Codigo finalizado' + final))

console.log('Sucesso')

Aqui estamos fazendo a conversão de apenas um item, mas vamos usar o for, para saber quantos itens vem dentro da lista e quanto vamos colocar la.

Agora vamos chamar ela, quando os dados já estão prontos.

const offset = 0 // Qual pagina vamos carregar, no caso a pagina 0

const limit = 10 // Limite de 10 itens nessa pagina

const url = `https://pokeapi.co/api/v2/pokemon?offset=${offset}&limit=${limit}` // URL de onde vamos puxar.

function converterPokemonParaHTML(pokemon) {

    return `

        <li class="pokemon">

                <span class="number">#001</span>

                <span class="name">${pokemon.name}</span>

                <div class="detail">

                    <ol class="types">

                        <li class="type">grass</li>

                        <li class="type">poison</li>

                    </ol>

                    <img src="https://raw.githubusercontent.com/PokeAPI/sprites/master/sprites/pokemon/other/dream-world/1.svg"

                         alt="${pokemon.name}">

                </div>

        </li>

    `

}

fetch(url)

    .then((response) => response.json()) // Pegamos o response e estamos retornando em formato JSON

    .then((jsonBody) => jsonBody.results) // Results é onde fica nossa lista de pokemons

    .then((listaPokemons) => {

        console.log(listaPokemons) // Lista com nossos pokemons

        for (let index = 0; index < listaPokemons.length; index++) {

            const pokemon = listaPokemons[index];

            console.log(converterPokemonParaHTML(pokemon))

        }

    })

    .catch((erro) => console.log(erro))

    .finally((final) => console.log('Codigo finalizado' + final))

console.log('Sucesso')

Agora vamos colocar o conjunto de LI que criamos dentro do nosso HTML