**Lendo Arquivos com Scanner**

prendemos a salvar um arquivo, via classe PrintWriter, mas caso você queira fazer o caminho contrário, ou seja, ler o conteúdo de um arquivo existente em Java, pode fazer isso com a utilização da classe Scanner.

A classe Scanner é uma classe padrão do Java que permite ler dados de diferentes fontes, incluindo arquivos. Para ler um arquivo com essa classe, basta criar uma instância dela e passar como argumento um objeto do tipo File, contendo o caminho do arquivo. Por exemplo:

import java.io.File;

import java.io.FileNotFoundException;

import java.util.Scanner;

public class LeituraDeArquivo {

public static void main(String[] args) {

try {

File arquivo = new File("arquivo.json");

Scanner scanner = new Scanner(arquivo);

while (scanner.hasNextLine()) {

String linha = scanner.nextLine();

System.out.println(linha);

}

scanner.close();

} catch (FileNotFoundException e) {

System.out.println("Arquivo não encontrado!");

}

}

}

COPIAR CÓDIGO

No código anterior, estamos lendo um arquivo chamado arquivo.json e imprimindo o conteúdo do arquivo linha por linha no console. O método hasNextLine() verifica se há mais linhas a serem lidas, enquanto o método nextLine() lê a próxima linha do arquivo.

A classe Scanner também pode ser usada para ler dados de entrada de outras fontes, como o teclado e strings. Além disso, ela oferece muitas opções para personalizar a forma como os dados são lidos, incluindo a capacidade de usar expressões regulares para analisar o texto.

No código anterior, estamos lendo um arquivo chamado arquivo.json e imprimindo o conteúdo do arquivo linha por linha no console. O método hasNextLine() verifica se há mais linhas a serem lidas, enquanto o método nextLine() lê a próxima linha do arquivo.

A classe Scanner também pode ser usada para ler dados de entrada de outras fontes, como o teclado e strings. Além disso, ela oferece muitas opções para personalizar a forma como os dados são lidos, incluindo a capacidade de usar expressões regulares para analisar o texto