

Lógica de Programação

Manipulação de Arquivos

Manipulação de Arquivos em Java

Germinare Tech

Marcelo Modolo

GerminaTECH

Manipulação de Arquivos em Java

Marcelo Modolo

Introdução

- A manipulação de arquivos é uma tarefa comum em muitos programas Java
- As classes Scanner, FileWriter e PrintWriter são amplamente utilizadas para ler e escrever dados em arquivos de texto
- A classe Scanner do pacote java.util oferece métodos para ler linhas de texto, valores inteiros e valores de ponto flutuante de arquivos
- As classes FileWriter e PrintWriter do pacote java.io são usadas para escrever dados em arquivos

Classe Scanner

- Utiliza a classe File para indicar o caminho e o arquivo que será lido
- Sintaxe:

```
File arquivo = new File("caminho/do/arquivo.txt");  
Scanner entrada = new Scanner(arquivo);
```

Métodos da Classe Scanner

- `nextLine()`: Lê a próxima linha de texto como uma String
- `nextInt()`: Lê o próximo valor como um inteiro
- `nextDouble()`: Lê o próximo valor como um double
- `hasNextLine()`: Verifica se há mais linhas para serem lidas
- `hasNextInt()`: Verifica se há mais valores inteiros para serem lidos
- `hasNextDouble()`: Verifica se há mais valores double para serem lidos
- `close()`: Fecha o arquivo liberando os recursos associados

Exemplo Classe Scanner

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class ExemploLeituraArquivo {
    public static void main(String[] args) {
        String caminhoArquivo = "caminho/do/arquivo.txt";

        try {
            File arquivo = new File(caminhoArquivo);
            Scanner scanner = new Scanner(arquivo);
```

Exemplo Classe Scanner

```
while (scanner.hasNextLine()) {  
    String linha = scanner.nextLine();  
    System.out.println(linha);  
}  
  
scanner.close();  
} catch (FileNotFoundException e) {  
    System.out.println("O arquivo não foi encontrado: " + e.getMessage());  
}  
}  
}
```

Classes FileWriter e PrintWriter

- A classe FileWriter é usada para escrever caracteres em um arquivo.
- A classe PrintWriter é uma classe de saída que fornece métodos convenientes para escrever texto em um arquivo
- Necessário tratar as exceções adequadas ao trabalhar com arquivos, como IOException, ao usar as classes FileWriter e PrintWriter

Métodos da classe PrintWriter

- `write(String texto)`: Escreve uma sequência de caracteres no arquivo
- `println(String texto)`: Escreve uma linha de texto no arquivo, adicionando uma nova linha após o texto
- `printf()`: Escreve uma linha de texto formatado no arquivo
- `close()`: Fecha o arquivo após a gravação, liberando os recursos associados

Exemplo Classes FileWriter e PrintWriter

```
import java.io.FileWriter;
import java.io.PrintWriter;
import java.io.IOException;

public class Main {
    public static void main(String[] args) {
        String caminhoArquivo = "caminho/do/arquivo.txt";

        try {
            FileWriter fileWriter = new FileWriter(caminhoArquivo);
            PrintWriter printWriter = new PrintWriter(fileWriter);
```

Exemplo Classes FileWriter e PrintWriter

```
printWriter.println("Este é um exemplo de texto que será salvo no arquivo.");  
printWriter.println("Você pode adicionar quantas linhas quiser.");
```

```
printWriter.close();
```

```
System.out.println("Texto salvo com sucesso no arquivo.");
```

```
} catch (IOException e) {
```

```
    System.out.println("Ocorreu um erro ao salvar o arquivo: " + e.getMessage());
```

```
}
```

```
}
```

```
}
```

Perguntas???

