



# 01. Leitura com java.io

📅 Date	@10/10/2022
▼ Categoria	Java
▼ Curso	Java e java.io: Streams Reader e Writers

## Revisão

Primeira data @February 21, 2022

---

## Tópicos

- Introdução
  - Estabelecendo a entrada
  - Abertura de arquivos
  - InputStream e Reader
  - Leitura de linhas
  - Leitura de arquivos
  - Mãos na massa: Leitura de arquivo
  - O que aprendemos
- 

## Estabelecendo a entrada

## Criando um Fluxo de entrada com arquivos

```
public class TesteLeitura
{
    public static void main (String[] args) throws FileNotFoundException{
        //Criando um fluxo concreto com arquivo
        FileInputStream fis = new FileInputStream("nomeDoArquivo.txt");
        //Transformando os dados binarios em caracteres
        InputStreamReader isr = new InputStreamReader(fis);
        //Juntar muitos caracteres
        BufferedReader br = new BufferedReader(isr);

        //O metodo .readLine() vai guarda uma linha do nosso arquivo.
        String linha = br.readLine();
        //Aqui foi mostrado a primeira linha do nosso arquivo
        System.out.println(linha);
        //O Fechamento de recurso e uma boa pratica
        br.close()
    }
}
```

## Abertura de arquivos

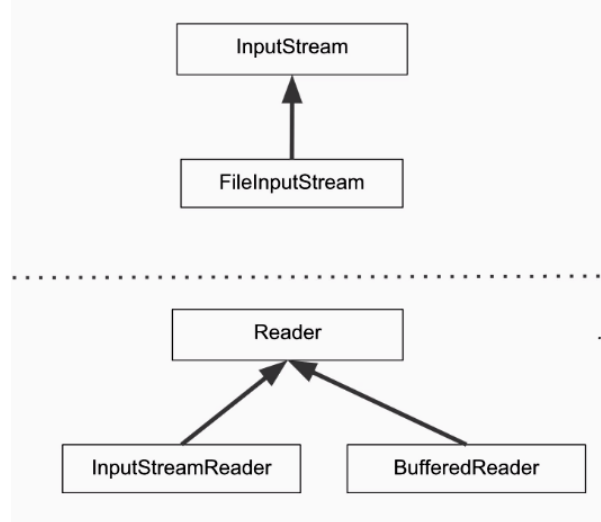
Ao utilizar um `FileInputStream` devemos adicionar um `throws FileNotFoundException` ao método `main()` ou tratar a criação do `FileInputStream` com um bloco `try-catch`. Logo percebemos que essa classe tem uma exceção e checked, e essa exceção é lançada quando o arquivo indicado não existe.

## InputStream e Reader

```
while(linha != null){
    System.out.println(linha); // Exibindo a linha atual
    linha = br.readLine(); //Ler a proxima linha
}
```

- Stream é focado em ler bytes,
  - `InputStream` (Abstract)

- Reader é focado em ler caracteres
  - Reader(Abstract)
- InputStream e Reader seguem um padrão de projeto chamado template



## 6. Leitura de arquivos

```

public class TesteLeitura {

    public static void main(String[] args) throws IOException {

        InputStream fis = new FileInputStream("LivroDeJava.txt");
        Reader isr = new InputStreamReader(fis);
        Reader br = new BufferedReader(isr);

        String linha = br.readLine();

        while (linha != null) {
            System.out.println(linha);
            linha = br.readLine();
        }

        br.close();
    }
}

```

O código não compila pois `br` é uma referência do tipo `Reader` e essa classe não possui o método `readLine()`. `BufferedReader` até é uma subclasse de `Reader`, no entanto, como precisamos do método `readLine()` e `Reader` não possui esse método, então é necessário que a referência seja do tipo `BufferedReader`.

## O que aprendemos

Nesta aula aprendemos sobre alguns tópicos relacionados a leitura com Java IO:

1. Abertura de arquivos;
  2. Leitura linha a linha do arquivo e detecção de seu fim;
  3. Utilização de *exceptions* pelo Java IO caso a operação de leitura não saia como esperado;
  4. Classes abstratas e concretas para leitura.
-