

## Extraíndo interface

Atualmente, para criarmos um novo leitor, como por exemplo, um leitor de XML, precisamos fazer a extensão da nossa classe `Leitor`. Entretanto, se observamos apenas essa extensão não garante o comportamento da **leitura de pagamentos**.

Em outras palavras, precisamos criar uma interface que vai permitir obrigar qualquer leitor de pagamento a implementar o método `ler()`, ou seja, o mesmo método contido na classe `LeitorPagamento`, portanto, esse método vai seguir a seguinte assinatura:

```
List<Pagamento> ler(int codigo) throws IOException
```

Essa assinatura assegura que na leitura do arquivo que foi enviado via o código teremos a capacidade de ler todos os pagamentos e devolver uma lista com todos que foram lidos, além disso, ele apresenta um lançamento de exception comum durante esse processo de IO.

Repara que a nossa classe `LeitorPagamento` faz leitura apenas de arquivos CSV, portanto, utilize o **rename** (**Shift + F6**) e deixe como `LeitorCSV`

## Extraíndo interface

Para extrair uma interface podemos usar o **Refactor This** (**Ctrl + Shift + Alt + T / CMD T**) > **Interface....** Então fazemos o mesmo processo que foi feito para criar a classe `Leitor` a diferença é que agora iremos atribuir o nome `LeitorPagamento` e marcaremos o método `ler()` como membro para a interface que será criada.

Após o processo execute novamente o projeto e veja se tudo funciona como antes.