04

Formatação do CPF e do CNPJ

Transcrição

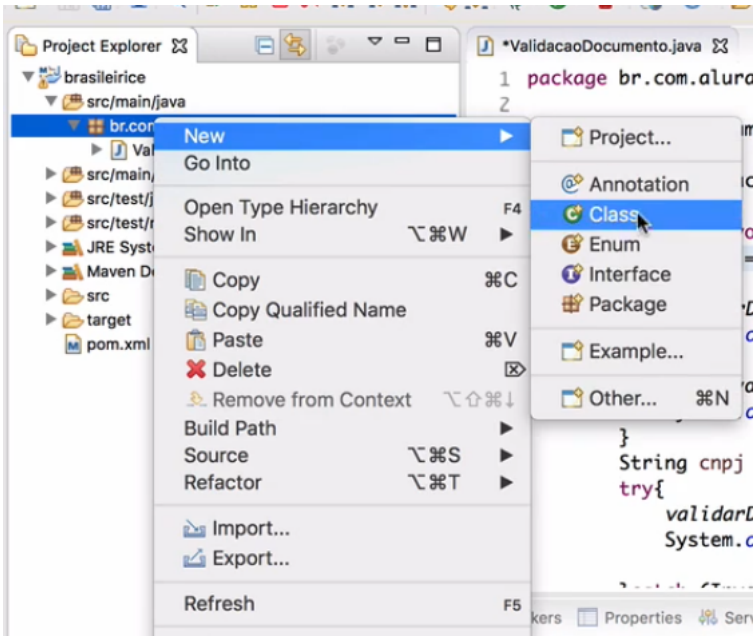
Já sabemos validar os documentos. Agora precisamos aprender a formatá-los. Geralmente, ao receber esses dados do front-end, eles vêm com uma máscara. Por exemplo, o CPF costuma vir no seguinte formato:

```
862.883.667-57
```

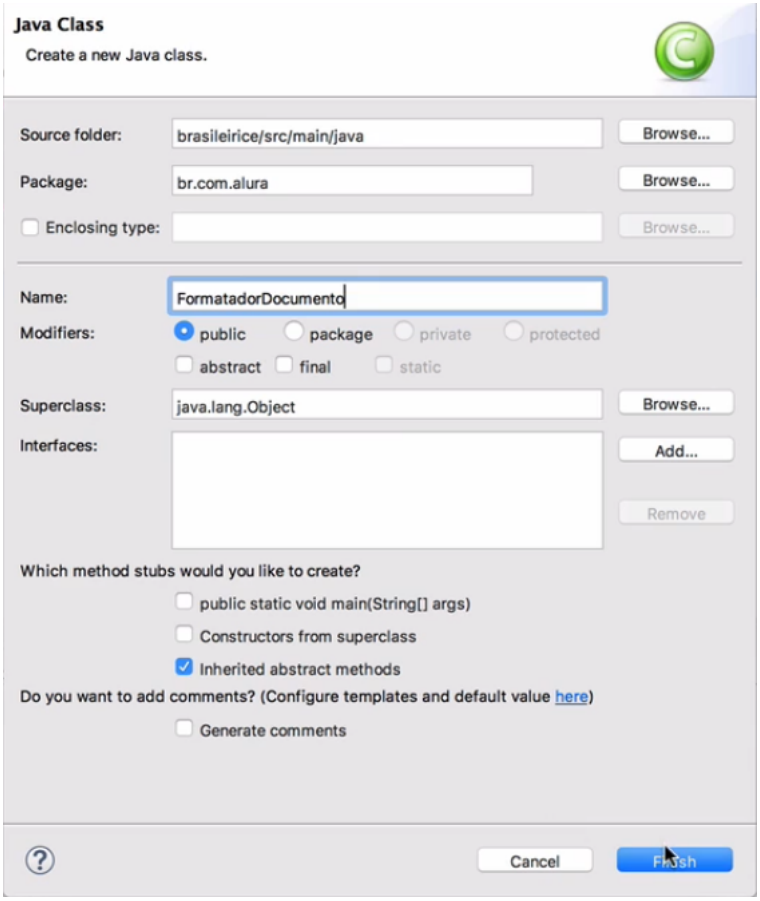
O CNPJ e o título de eleitor têm suas próprias máscaras, e é muito comum que venham do front-end com essas máscaras. É importante saber formatá-los no back-end, para salvá-los no banco de dados e poder enviá-los para APIs sem problemas. Por exemplo, a API do PayPal não aceita dados com máscaras, aceita apenas números.

Um dos jeitos que poderíamos usar para tirar a máscara é o `regex`, pedindo para onde houver um ponto subtituir por nada, e onde houver um hífen, fazer o mesmo. Seria possível também fazer um `split` da `string`. Vamos usar uma abordagem com o Stella, que tem um formatador pronto, que pode por ou tirar máscaras dos dados.

Começaremos criando uma nova classe no nosso pacote. Clicaremos com o botão direito sobre o `br.com.alura` no `Project Explorer`, e então, `New > Class`.



A nova classe se chamará `FormatadorDocumento`



Agora temos uma classe de validação e uma de formatação. Começaremos colocando o `main`, que é `public static void main(Strin[] args)`. A princípio, o documento de formatação fica assim:

```
package br.com.alura;

public class FormatadorDocumento {

  public static void main(Strin[] args){

  }

}
```

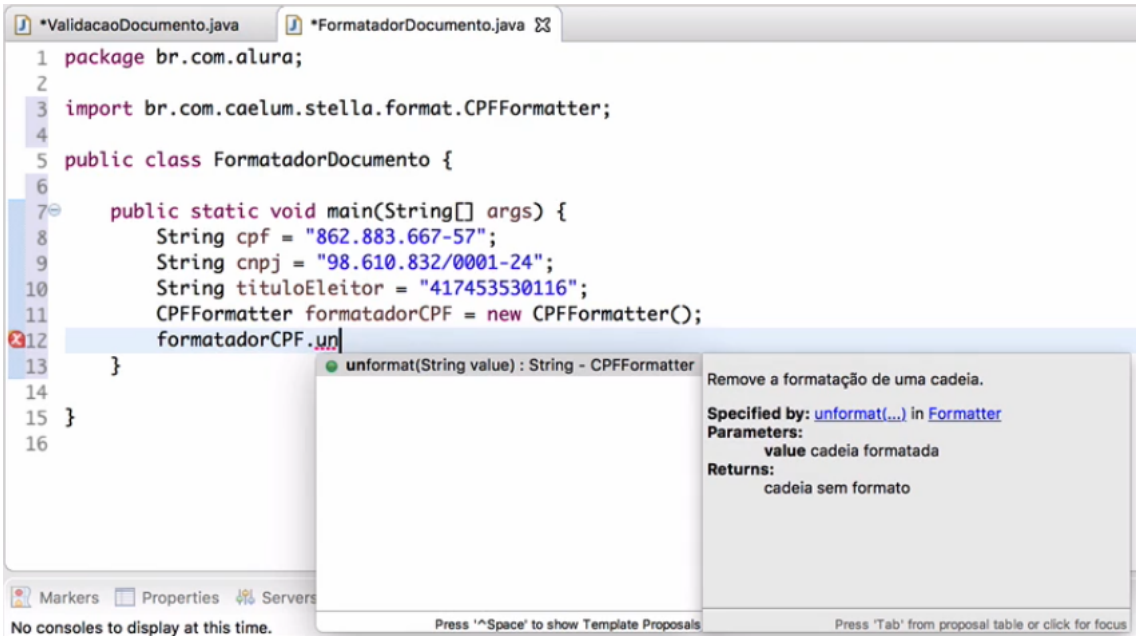
Podemos acrescentar as `string`s dos documentos, que vieram dos sites que geram números válidos.

```
package br.com.alura;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
  }

}
```

Deixaremos o título de eleitor sem máscara para aprender a colocar formatação. O formatador segue a mesma ideia do validador: há um específico para cada documento, e quando os chamarmos, eles serão importados da biblioteca do Stella (o que aparece para nós antes do main).

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    new CPFFormatter();
  }

}
```

Criaremos uma instância para esse formatador, o `formatadorCPF` , que tem um método. Quando o digitamos, ele nos mostra: "Remove a formatação de uma cadeia", ao receber uma string como argumento.



No código, fica assim:

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    formatadorCPF.unformat(cpf);
```

```
  }

}
```

Esse método retorna também uma string, que é o CPF sem os pontos e o traço. Então precisamos criar essa string.

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
  }

}
```

Para ver como ficou, faremos um `sysout` . Primeiro vamos imprimir o CPF, depois o mesmo sem formatação.

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
  }

}
```

Então iremos rodar, clicando com o botão direito e em `Run as > Java Application` . O console nos mostrará o seguinte:

```
862.883.667-57
86288366757
```

Assim, tiramos a formatação com apenas um método, o `unformat()` . Assim como o `validador` , ele é específico para cada documento. Portanto, precisaremos repetir o processo para o CNPJ. Primeiro, colocamos o `CNPJFormatter` :

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
    CNPJFormatter formatadorCNPH = new CNPJFormatter();
  }

}
```

Em seguida, podemos chamar esse formatador, com o método a ser usado:

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
    CNPJFormatter formatadorCNPH = new CNPJFormatter();
    fomatadorCNPJ.unformat(cnpj);
  }

}
```

Então, é preciso criar a string que receberá isso.

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
```

```
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
    CNPJFormatter formatadorCNPJ = new CNPJFormatter();
    fomatadorCNPJ.unformat(cnpj);
    String cnpjSemFormatacao = formatadorCNPJ.unformat(cnpj);
  }

}
```

Em seguida, colocamos duas `sysout` : referentes ao cnpj com e sem formatação.

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
    CNPJFormatter formatadorCNPJ = new CNPJFormatter();
    fomatadorCNPJ.unformat(cnpj);
    String cnpjSemFormatacao = formatadorCNPJ.unformat(cnpj);
  }

}
```

Para ver como ficou, clicaremos com o botão direito e em `Run as > Java Application` .

```
862.883.667-57
86288366757
98.610.832/001-24
9861083200124
```

Só falta formatar o título eleitoral.

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
```

```
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
    CNPJFormatter formatadorCNPJ = new CNPJFormatter();
    fomatadorCNPJ.unformat(cnpj);
    String cnpjSemFormatacao = formatadorCNPJ.unformat(cnpj);
    TituloEleitoralFormatter formatadoTitulo = new TituloEleitoralFormatter();
  }

}
```

Note que agora usaremos o `format`, pois pegamos o dado sem a máscara e queremos inseri-la. Ele formatará a cadeia ao receber a string.

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
    CNPJFormatter formatadorCNPJ = new CNPJFormatter();
    fomatadorCNPJ.unformat(cnpj);
    String cnpjSemFormatacao = formatadorCNPJ.unformat(cnpj);
    TituloEleitoralFormatter formatadoTitulo = new TituloEleitoralFormatter();
  }

}
```

Criaremos a string `tituloComFormatacao` e adicionaremos os dois `sysout`.

```
package br.com.alura;

import br.com.caelum.stella.format.CPFFormatter;

public class FormatadorDocumento {

  public static void main(Strin[] args){
    String cpf = "862.883.667-57";
    String cnpj = "98.610.832/0001-24";
    String tituloEleitor = "417453530116";
    CPFFomatter formatadorCPF = new CPFFormatter();
    String cpfSemFormatacao = formatadorCPF.unformat(cpf);
    System.out.println(cpf);
    System.out.println(cpfSemFormatacao);
    CNPJFormatter formatadorCNPJ = new CNPJFormatter();
    fomatadorCNPJ.unformat(cnpj);
```

```
        String cnpjSemFormatacao = formatadorCNPJ.unformat(cnpj);
        TituloEleitoralFormatter formatadoTitulo = new TituloEleitoralFormatter();
        String tituloComFormatacao =  formatadorTitulo.format(tituloEleitor);
        System.out.prinln(tituloEleitor);
        System.out.prinln(tituloComFormatacao);
      }

    }
```

Vamos salvar e rodar. O console nos mostra o seguinte:

```
862.883.667-57
86288366757
98.610.832/0001-24
9861083200124
417453530116
4174535301/16
```

Tudo funcionou bem. É possível criar um método, como fizemos no arquivo de validação, para implementar a classe `Formatter`. Vou deixar a melhora dessa classe como exercício para você. Com esse método em uma classe útil, pode-se usá-lo onde você precisar em sua aplicação. Até a próxima!