

MC 302EF - Atividade de Laboratório no. 6

Objetivos

- Uso de um 'XML parser' na recuperação da estrutura de objetos representada num texto XML.
- Uso de arquivos.
- Tratamento de eventos

Descrição do Problema [\[criado em 26/04/16\]](#)

Em continuação ao trabalho das atividades anteriores, esta atividade consiste em recuperar a lista de objetos da classe Item representada num texto XML contido num arquivo. Para tanto deverá ser usado um parser XML (`javax.xml.parsers.SAXParser`) operando em modo SAX, conforme mostrado em sala (ver apresentação 'Entrada e Saída' e os exemplos mostrados em aula, na página do curso).

As classes

As classes usadas nesta atividade para construir a estrutura de objetos representada no texto XML, deverão ser as mesmas usadas na atividade 4, sem nenhuma alteração. Além dessa classe, deverá ser construída uma classe chamada **Lab6Builder**, que será a responsável pelo tratamento dos eventos gerados pelo *parser* XML e construção da estrutura de objetos representada no texto XML.

O arquivo de entrada

A figura abaixo ilustra um fragmento do arquivo 'entradaLab6.xml':

```
<?xml version="1.0"?>
<lista>
  <itens>
    <item cod='1021' quant='1' />
    <item cod='1022' quant='1' />
    <item cod='2043' quant='1' />
    <item cod='1023' quant='1' />
  </itens>
  <partes>
    <partecomposta cod='1021' nome='base fixa bf1' descricao='base fixa de uso geral' valor='88.5' >
      <itens>
        <item cod='231' quant='1' />
        <item cod='511' quant='2' />
        <item cod='233' quant='3' />
        <item cod='232' quant='4' />
        <item cod='321' quant='5' />
        <item cod='322' quant='6' />
      </itens>
    </partecomposta>
    <parafuso cod='231' nome='parafuso p1' descricao='parafuso de fixacao do cabecote' valor='2.5'
  <parteespecifica cod='511' nome='placa pk1' descricao='placa principal' valor='6.0' >
    <caracteristica nome='espessura' conteudo='3' />
    <caracteristica nome='largura' conteudo='450' />
    <caracteristica nome='comprimento' conteudo='500' />
  </parteespecifica>
  . . .
```

O tratador de eventos gerados pelo *parser*

O tratador de eventos a ser acoplado ao *parser* deverá implementar a interface `Lab6Interface` (fornecida como parte desta atividade) estender a classe `DefaultHandler` (definida no pacote `javax.xml`), redefinindo os métodos responsáveis pelo tratamento dos eventos gerados pelo *parser*:

- Abertura de uma *tag*
- Fechamento de uma *tag*
- Texto contido numa *tag*

À medida em que os eventos são tratados, a estrutura de objetos representada no texto XML deverá ser construída. A interface `Lab6Interface` define o seguinte método:

```
public List<Item> ItemsFromXml(String fileName);
```

Esse método tem como parâmetro o nome do arquivo de entrada e retorna a lista de objetos `Item` representada no XML, onde cada item se refere a um objeto `Parte`, tal como acontece com a lista de itens criada por `Dados.java`, usado na atividade 4.

A lista de objetos `Item` construída pelo tratador de eventos

A lista de objetos `Item` reconstruída pelo *parser* deverá ter a mesma estrutura representada no texto XML:

- cada objeto `Item` da lista se refere a um objeto `Parte`
- cada objeto `Parte` referenciado na lista deve ser criado uma única vez (ou seja, se no texto XML duas ou mais tags `<item>` se referem a um mesmo código de parte, o objeto `Parte` correspondente deve ser criado uma única vez (isso será considerado na verificação do código).

Observações importantes:

- Numa tag XML a ordem dos atributos é irrelevante.
- No texto XML a ordem de ocorrência das tags deve ser considerada ao se criar a estrutura de objetos representada pelo mesmo.
- Uma das dificuldades nesta atividade: o código dos objetos `Parte` é usado nos objetos `Item` antes da sua definição no texto XML. Se os objetos `Item` forem criados quando o *parser* tratar a tag correspondente, não se tem a informação necessária para criar os objetos `Parte` associados. Uma saída possível (mas não é a única):
 - Ao tratar a tag `<item>`, a partir dos atributos 'cod' e 'quant', cria-se um objeto `Item` associado a um objeto `Parte` temporário, no qual apenas o atributo `codigo` é válido (o objeto temporário pode ser, por exemplo, um objeto `ParteEspecifica`).
 - Os objetos `Item` criados a partir das tags `<item>` devem ser inseridos numa lista.
 - Ao tratar as tags que descrevem objetos `Parte` (`<motor>`, `<parafuso>`, `<partecomposta>`, `<parteespecifica>`), cria-se os respectivos objetos inserindo-os num mapa, indexado pelo código.
 - Ao final do processo de *parsing*, a lista de objetos `Item` é percorrida e os objetos `Parte` temporários são substituídos pelos objetos verdadeiros colocados no mapa. O percurso à lista pode ser feita com base naquele usado no programa de testes da atividade 3 (`TesteLab3_1.java`).

Exemplo de uso

O programa TesteLab6.java, fornecido como parte desta atividade, mostra um exemplo de uso da classe Lab6Builder, a ser criada nesta atividade.

Saída esperada

O programa TesteLab6.java usa a mesma função de escrita da lista usada na atividade 3. A saída esperada, disponível como parte desta atividade, 'saidaLab6.txt', portanto é a igual à saída esperada para a atividade 3.

Data de entrega: 12/05/16