Relatório – TQS Homework

António Castanheira – 80870

Repositório GIT - https://github.com/Parakeet37/tqs-homework

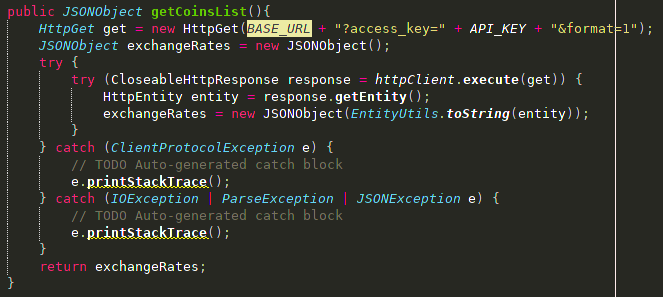
# Interface web

O primeiro passo na minha implementação foi estruturar a interface que iria ser usada para fazer a conversão de valores. Usei primefaces pois é uma maneira rápida e simples de fazer uma página xhtml com os parâmetros pretendidos (dropdowns em que os seu valores fossem gerados dinâmicamente) sem serem demaisado minimalistas. Com a página terminada, o próximo passo foi fazer as classes java usadas para ir buscar dados à API (currencylayer) e processá-los.

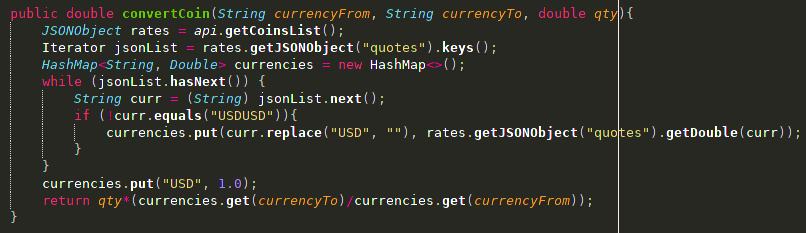


# Classes Java

A primeira classe a implementar foi a classe que ia buscar o objeto JSON à API. Tem apenas uma função, pois todo o processamento da informação retirada da API é realizado noutra classe, permitindo assim uma mais facil implementação de testes unitários com mocks à API.

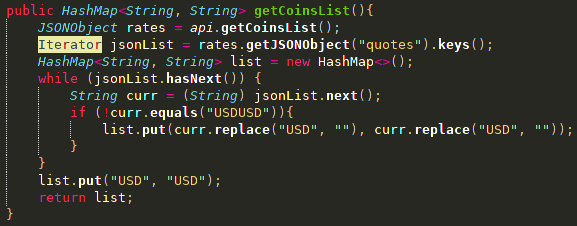


Depois, implementei a classe que processa o objeto json e faz a conversão de moedas. Em ambos os casos é chamada a função da classe APIGetter para ir buscar a informação mais recente à API. Esta classe tem 2 funções, uma para cada uma das finalidades descritas atrás(processar o JSONObject e converter moedas).



Como os câmbios vêm todos em relação ao dólar (USD), devido à api restringir os dados que são possíveis de obter com uma chave de acesso de uma conta gratuíta, a conversão tem que ser:

moeda original → dólar → moeda “final”



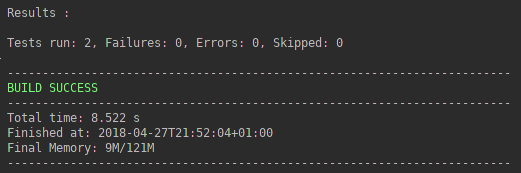
Esta função organiza as moedas num hashmap para ser possível à interface web organizar as dropdowns dinâmicamente, atribuindo um id a cada uma das entradas.

# Java Bean

A terceira fase foi a implementação do managed bean que iria servir de apoio à página. Foi relativamente simples de implementar, pois apenas tem getters e setters ,visto que as ações principais são realizadas em outras classes.

# Testes realizados

Por último, implementei diversos testes. Implementei 2 testes unitários, um para cada função da classe Converter, que utilizam mocks sobre a API para receber a informação que é usada pra trabalhar a informação. O uso de mocks elimina a dependência da API externa para a realização dos testes.



Também fiz um teste sobre a interface web, usando Katalon e selenium. Aqui apareceu um problema, porque não consegui correr o teste da interface web, pois o netbeans não conseguia abrir nenhum browser para testar.

