

MC 302EF - Atividade de Laboratório no. 10

Data de Entrega: 13/06/15

Objetivos

- Uso de Python

Descrição da atividade

Em continuação ao trabalho iniciado na atividade anterior, esta atividade consiste em criar duas novas classes e adaptar as classes anteriores para utilização das mesmas.

Novas classes: Lançamento e Regra

A classe Lançamento

Descreve cada movimento financeiro numa das contas do plano. Cada objeto dessa classe deve manter como atributos

- Numero do lançamento: um número sequencial único identificando o lançamento
- Data: um string indicando a data do lançamento (o formato da data é definido pela aplicação, no nosso caso, pelo programa de testes)
- Descrição: um texto (curto) descritivo do lançamento, como string
- Valor: um valor float indicando o valor do lançamento
- Tipo: um inteiro indicando o tipo do lançamento (0 == 'crédito', 1 == 'débito')

Essa classe deve disponibilizar os seguintes métodos para a aplicação:

- construtor `__init__(self, data, descrição, valor, tipo)`
- `toXML(self)` : gera um string contendo a representação do lançamento em XML (usar como referência para o formato usado na saída do programa de testes, anexa a esta atividade)

A classe Regra

Descreve uma 'regra' a ser aplicada a cada lançamento numa conta. Uma 'regra' consiste basicamente na indicação que um lançamento numa conta de um plano de contas pode implicar num lançamento adicional em outra conta do mesmo plano. Cada objeto da classe Regra deve manter os seguintes atributos:

- nome: string contendo o nome da regra a ser usada nos lançamentos gerados a partir da mesma
- gatilho: valor float indicando o limite inferior para o disparo da regra: se o valor do lançamento for maior ou igual ao gatilho, deve ser gerado um lançamento em outra conta.
- percentual: valor float indicando o coeficiente a ser aplicado ao se calcular o valor do novo lançamento
- destino: código da conta à qual deve ser aplicado o lançamento disparado pela regra.
- tipo: inteiro indicando o tipo do lançamento disparado pela regra (0 == crédito, 1 == débito)

Essa classe deve disponibilizar à aplicação os seguintes métodos

- construtor `__init__(self, nome, gatilho, dest, perc, tipo)`

- **toXML(self)** : gera um string contendo a representação da regra em XML (usar como referência o formato usado na saída do programa de testes, anexa a esta atividade)
- **aplica(self, plano, lançamento)** : aplica a regra a um dado lançamento no plano de contas. Este método é o responsável por verificar se o lançamento passado como parâmetro deve ou não disparar outro lançamento na conta indicada na regra.

Adaptações nas classes: Conta e Plano

A classe Conta

Novos métodos:

- **agregaLancamento(self, plano, lançamento)** : agrega um lançamento à conta
- **agregaRegra(self, regra)**: agrega uma regra à conta
- **total(self)** : retorna o valor total dos lançamentos associados à conta e suas contas filhas (soma dos valores dos lançamento de crédito – soma dos valores dos valores dos lançamentos de débito)
- **histórico(self)** : retorna uma lista contendo todos os lançamentos associados à conta e suas contas filha.

O método toXML(self) deve ser alterado para mostrar as regras associadas a conta (usar como referência o formato usado na saída do programa de testes, anexa a esta atividade). Eventualmente outros métodos podem ser necessários para implementar os métodos descritos acima.

A classe Plano

Novos métodos:

- **agregaLancamento(self, lançamento, codConta)** : agrega um lançamento a uma dada conta do plano passado como parâmetro. Retorna True ou False, indicando sucesso ou falha da operação.
- **historico(self, codConta)** : retorna uma lista contendo os lançamentos associados à conta do plano, indicada pelo seu código, ordenada pelo número do lançamento. Caso a conta indicada não exista no plano deve retornar None.
- **historicoToXML(self, codConta)** : retorna um string contendo a representação em XML do histórico da conta (usar como referência o formato usado na saída do programa de testes, anexa a esta atividade).
- **total(self, codConta)** : retorna o valor total dos lançamentos associados à uma conta e suas contas filhas. Caso a conta indicada não exista no plano deve retornar None.
- **agregaRegra(self, regra, codConta)** : agrega uma regra à conta do plano indicada pelo código da conta passado como parâmetro. Retorna True em caso de sucesso ou False em caso de falha (a conta não existe no plano).

Eventualmente outros métodos podem ser necessários para implementar os métodos descritos acima.

Exemplo de uso e saída esperada

São fornecidos como anexos à esta atividade.