

Documentação básica de desenvolvimento e instrução para uso

Previa

O sistema em si foi desenvolvido para ser executado dentro do console da ferramenta eclipse, a estruturação em si dá possibilidades diversas para usos em outras ferramentas ou outros métodos para complementar.

Classes e Interfaces

Classe Cliente

Com Atributo CPF e seus construtores, nele foram estendidas informações da classe Usuario.

Classe Empresa com extensão na classe Usuario e implementação com a interface TaxaAdministrativa:

*Contendo os atributos CNPJ do tipo String, saldo e taxa do tipo "**BigDecimal**" classe usada para auxiliar nos cálculos entre as variáveis, procedimento usado quando se trata de saldos bancários que precisam de valores exatos sem nenhuma margem de erro. Os métodos usados com a classe **BigDecimal** foram:*

add, subtract e multiply.

Consta também o método calcularTaxa usada pela implementação da interface.

Classe Notificador

Usado com o objetivo principal para notificar todas as transações feitas pelo Cliente ou pela Empresa.(Saques e depósitos)

Taxa Administrativa (Interface):

Interface usada para o metodo calcularTaxa.

Classe Transacao: usando os atributos

private String tipo;

private double valor;

private Date data;

private Empresa empresa;

private Cliente cliente;

Para que seja informado ao sistema as informações da transacao.

Além do tipo, valor e data será imprimido também dados da classe Empresa e Cliente.

Interface transacaoCallback

compõe metodo "notificarTransacao" Com os seguintes parâmetros: Empresa empresa, Cliente cliente, BigDecimal valor, String tipoTransacao

essa interface possui uma importante tarefa, para informar dados da empresa (através do objeto Empresa) e os dados do cliente (objeto cliente).

Incluindo também dados da classe Transacao (valor e tipoTransacao)

Classe Usuario:

Desenvolvido com 1 atributo privado (nome) do tipo String e seus construtores.

Classe ValidadorDocumento

Nessa classe foi usada expressão regular Pattern e Matcher, o método compile foi utilizado para que os valores de entrada fossem formatados da forma descrita no código (com pontos e traços) para validar o Matcher irá percorrer pelo valor de entrada e confirmar se o valor entrou de acordo com a determinação do Objeto Pattern.

classe SistemaBancario com implementação TransacaoCallback

Nessa classe é implementado todos os métodos desenvolvidos nas demais classes e dentro delas são chamadas suas funções.

Antes desse processo existem 2 Maps para empresas e cliente, usada para mapear as chaves e valores (K, V) para que não haja repetições, pois tratamos de dados pessoais que não se repetem.

Os outros métodos adicionarEmpresa e adicionarCliente possuem o método do objeto Map que é o Put, usado para inserir os valores dentro da variável.

*os demais: realizarDeposito/realizarSaque /
notificarTransacao / imprimirSaldoEmpresa foram implementados com base nas classes criadas e suas ações, conforme mostra cada um.*

Classe Main

Por fim a classe principal, responsável por executar os comando e ações do sistema.

OBS: para fazer alguns testes, recomendo executá-los conforme os exemplos que já estão na classe.

O sistema em si abre margens para melhorias e progressos, pode ser incluído mais funcionalidades e ligações com outras ferramentas.

 [Linkedin](#)

 [GitHub](#)

 [Apresentação](#)

 [Conheça Mais!](#)