package Fachada;

import Interfaces.\*;

import Negocios.\*;

import ClassesBasicas.\*;

import Excecoes.\*;

public class Fachada {

private NegocioProduto negocioProduto;

private NegocioFornecedor negocioFornecedor;

private NegocioEntrega negocioEntrega;

private NegocioPessoas negocioPessoa;

public Fachada(String letra)throws ErroDeInicializacaoException {

this.negocioProduto = new NegocioProduto(letra);

this.negocioFornecedor = new NegocioFornecedor(letra);

this.negocioEntrega = new NegocioEntrega(letra);

this.negocioPessoa = new NegocioPessoas(letra);

}

public int procurar(int codigoProduto) throws ProdutoNaoEncontradoException {

this.negocioProduto.procurar(codigoProduto);

}

public void inserir(Produto item) throws ProdutoJaCadastradoException {

this.negocioProduto.inserir(item);

}

public void remover(int codigoProduto) throws ProdutoNaoEncontradoException {

this.negocioProduto.remover(codigoProduto);

}

public void atualizarNome(int codigoProduto, String novoNome) throws ProdutoNaoEncontradoException {

this.negocioProduto.atualizarNome(codigoProduto, novoNome);

}

public void atualizarPreco(int codigoProduto, Double novoPreco) throws ProdutoNaoEncontradoException {

this.negocioProduto.atualizarPreco(codigoProduto, novoPreco);

}

public Produto getProduto(int codigoProduto) throws ProdutoNaoEncontradoException{

return negocioProduto.getProduto(codigoProduto);

}

// metodos do fornecedor

public void cadastraFornecedor(Fornecedor fornecedor) throws FornecedorJaCadastradoException, CnpjInvalidoException {

this.negocioFornecedor.acessoInserir(fornecedor);

}

public void removeFornecedor(String CNPJ) throws CnpjInvalidoException {

this.negocioFornecedor.acessoRemover(CNPJ);

}

public void atualizaFornecedor(Fornecedor fornecedor) {

this.negocioFornecedor.acessoAtualizar(fornecedor);

}

public boolean procuraFornecedor(String CNPJ) throws FornecedorNaoEncontradoException{

return this.negocioFornecedor.acessoProcurar(CNPJ);

}

public void imprimeTodosFornecedores() {

this.negocioFornecedor.imprimeTodosFornecedores();

}

public void Inserir(Entrega entrega) throws EntregaJaCadastradaException {

this.negocioEntrega.inserir(entrega);

}

public void Remover(Entrega entrega) throws EntregaNaoEncontradaException {

this.negocioEntrega.remover(entrega);

}

public void Atualizar(Entrega entrega) throws EntregaNaoEncontradaException {

this.negocioEntrega.atualizar(entrega);

}

public void Procurar(Entrega entrega) throws EntregaNaoEncontradaException {

this.negocioEntrega.procurar(entrega);

}

public void cadastrarCliente (String nome,String cpf,String endereco)throws ClienteJaCadastradoException {

this.negocioPessoa.cadastrarCliente(nome,cpf,endereco);

}

public void encerrarCadastroCliente(String cpf)throws ClienteNaoEncontradoException {

this.negocioPessoa.encerrarCadastroCliente(cpf);

}

public void atualizarNomeCliente(String nome,String cpf)throws ClienteNaoEncontradoException {

this.negocioPessoa.atualizarNomeCliente(nome, cpf);

}

public void atualizarEnderecoCliente(String endereco,String cpf)throws ClienteNaoEncontradoException{

this.negocioPessoa.atualizarEnderecoCliente(endereco, cpf);

}

public void contratarFuncionario(String nome,String cpf,double salario) {

this.negocioPessoa.contratarFuncionario(nome, cpf, salario);

}

public void demitirFuncionario(String cpf) {

this.negocioPessoa.demitirFuncionario(cpf);

}

public void atualizarNomeFuncionario(String nome,String cpf) {

this.negocioPessoa.atualizarNomeFuncionario(nome, cpf);

}

public void atualizarSalarioFuncionario(double salario,String cpf) {

this.negocioPessoa.atualizarSalarioFuncionario(salario, cpf);

}

// insiram aqui todos os metodos da sua classe negocios.

}