



Trabalho de Orientação a Objetos

Professora : Fabiana Freitas Mendes

Alunos :

Guilherme Silva Dutra - 221021984

Júlio Roberto da Silva Neto - 221022041



Gerência de Patrimônios



Minicenário

Gerência de Patrimônio

O Gerente de uma grande filial pretende fazer a declaração anual do imposto de renda, nesse sentido, ele contratou você para desenvolver uma aplicação que possa facilitar esse processo. Para isso, a aplicação deve atender aos seguintes requisitos: permitir o registro ou remoção de diferentes patrimônios, com seus nomes e valores, além organizá-los em uma lista onde será possível buscá-los dado o seu nome, como também fazer o registro ou remoção de diferentes Filiais com seus nomes e Patrimônios registados.

Estrutura da solução

Patrimony
<ul style="list-style-type: none">- name : String- value : double- amount : int
<ul style="list-style-type: none">+ setName(name : String) : void+ getName() : String+ setValue(value : double) : void+ getValue() : double+ getAmount() : int+ setAmount(amount : int) : void+ Patrimony()+ Patrimony(name : String)+ Patrimony(name : String, value : double)+ Patrimony(name : String, amount : int)+ Patrimony(name : String, amount : int, value : double)+ edit(name : String) : void+ edit(value : double) : void+ edit(amount : int) : void+ edit(name : String, value : double) : void+ edit(name : String, amount : int) : void+ edit(amount : int, value : double) : void+ edit(name : String, amount : int, value : double) : void+ toString() : String

Estrutura da solução

Filial
- contPat : int
+ setContPat(contPat : int) : void + getContPat() : int + setPatrimony(patrimony : ArrayList<Patrimony>) : void + getPatrimony() : List<Patrimony> + Filial(name : String) + Filial(name : String, adress : Adress) + add(patrimony : Patrimony) : void + remove(patName : String) : void + editPatrimony(oldName : String, newName : String) : void + editPatrimony(oldName : String, newValue : double) : void + editPatrimony(patName : String, amount : int) : void + createPat(name : String) : void + creatPat(name : String, value : double) : void + toString() : String

Estrutura da solução

Company
+ setFilials(filials : ArrayList<Filial>) : void + getFilials() : List<Filial> + addFilial(filial : Filial) : void + Company() + Company(name : String, adress : Adress) + Company(name : String) + editFilial(oldName : String, newName : String) : void + editFilial(name : String, newAdress : Adress) : void + add(filialName : String, patrimony : Patrimony) : void + edit(filialName : String, oldPatName : String, newPatName : String) : void + edit(filialName : String, patName : String, newPatValue : double) : void + edit(filialName : String, patName : String, newPatAmount : int) : void + remove(filialName : String) : void + remove(filialName : String, patrimonyName : String) : void + createFilial(name : String) : void + createFilial(name : String, adress : Adress) : void + toString() : String

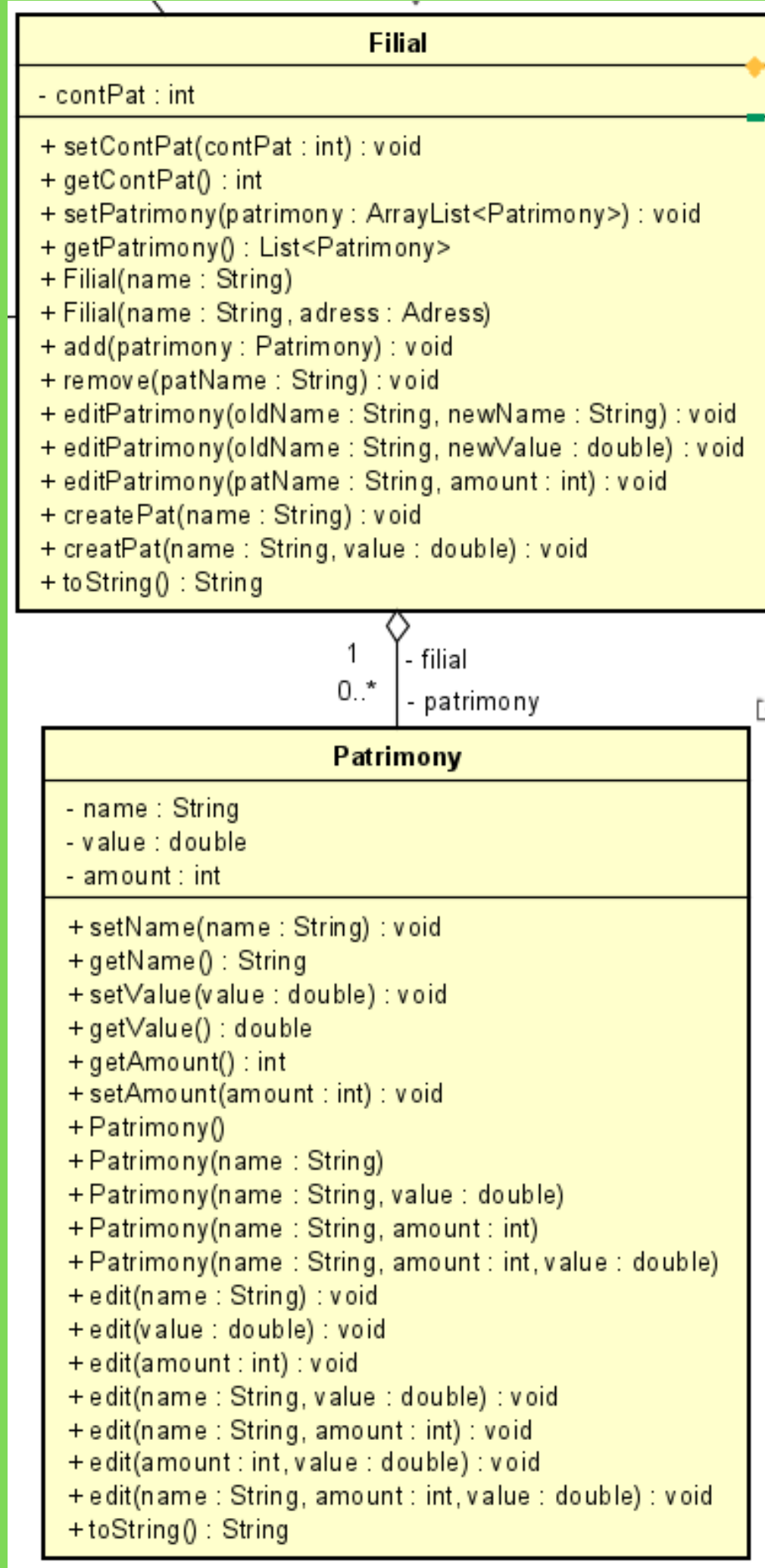
Estrutura da solução

Adress
<ul style="list-style-type: none">- country : String- state : String- city : String- street : String- number : int
<ul style="list-style-type: none">+ setCountry(country : String) : void+ getCountry() : String+ setState(state : String) : void+ getState() : String+ setStreet(street : String) : void+ getStreet() : String+ setCity(city : String) : void+ getCity() : String+ setNumber(number : int) : void+ getNumber() : int+ Adress(country : String, state : String, city : String, street : String, number : int)+ toString() : String

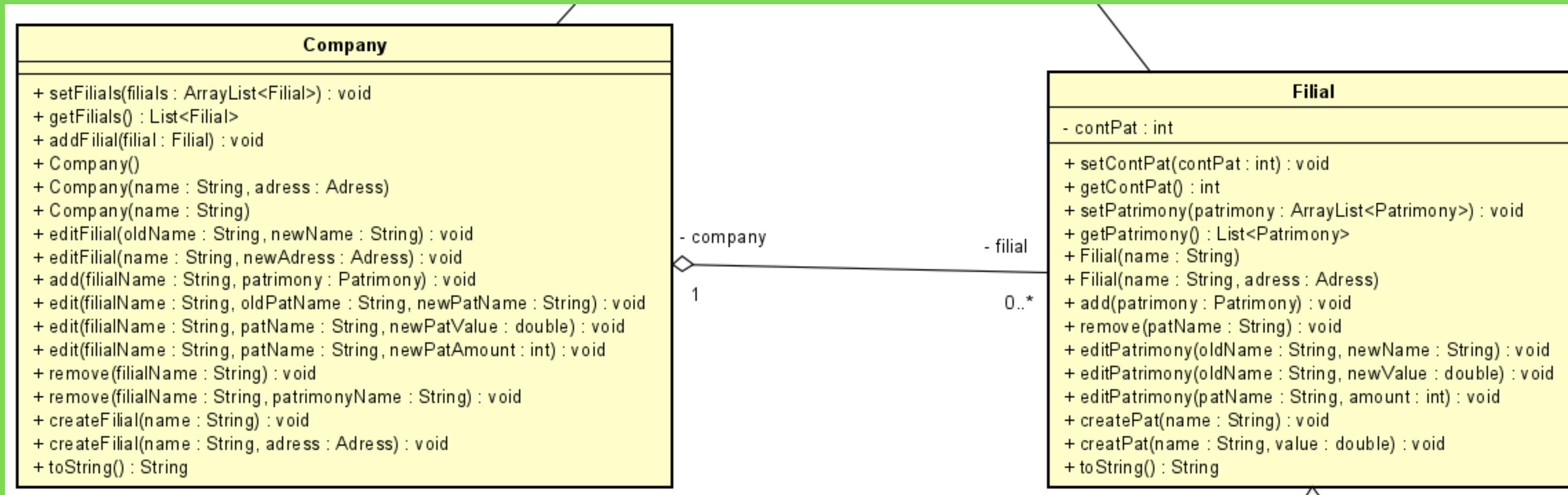
Estrutura da solução

<i>Interprises</i>
- name : String
+ setName(name : String) : void + getName() : String + setAddress(adress : Adress) : void + getAddress() : Adress + edit(name : String) : void + edit(adress : Adress) : void + edit(name : String, adress : Adress) : void

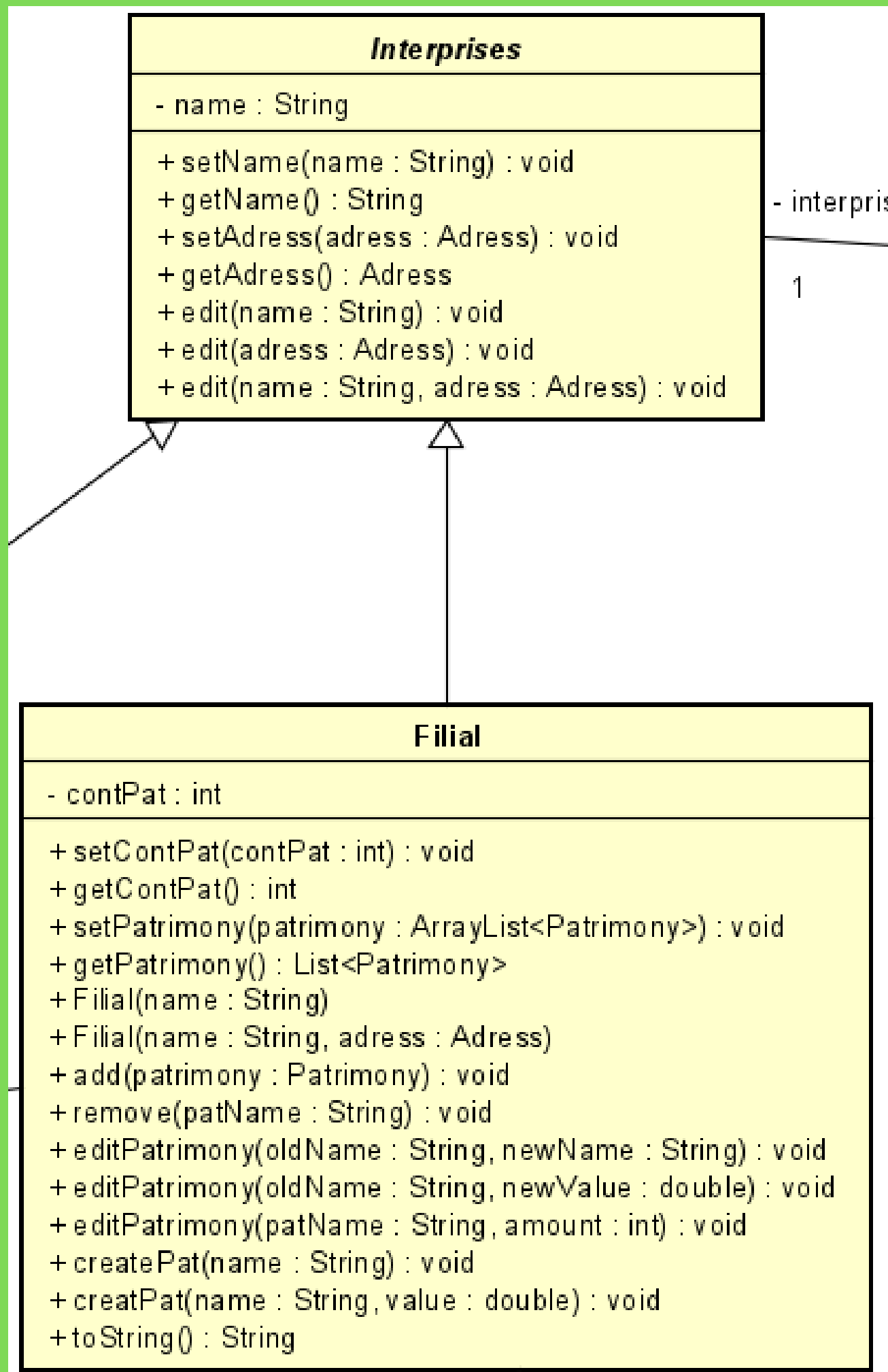
Relacionamentos



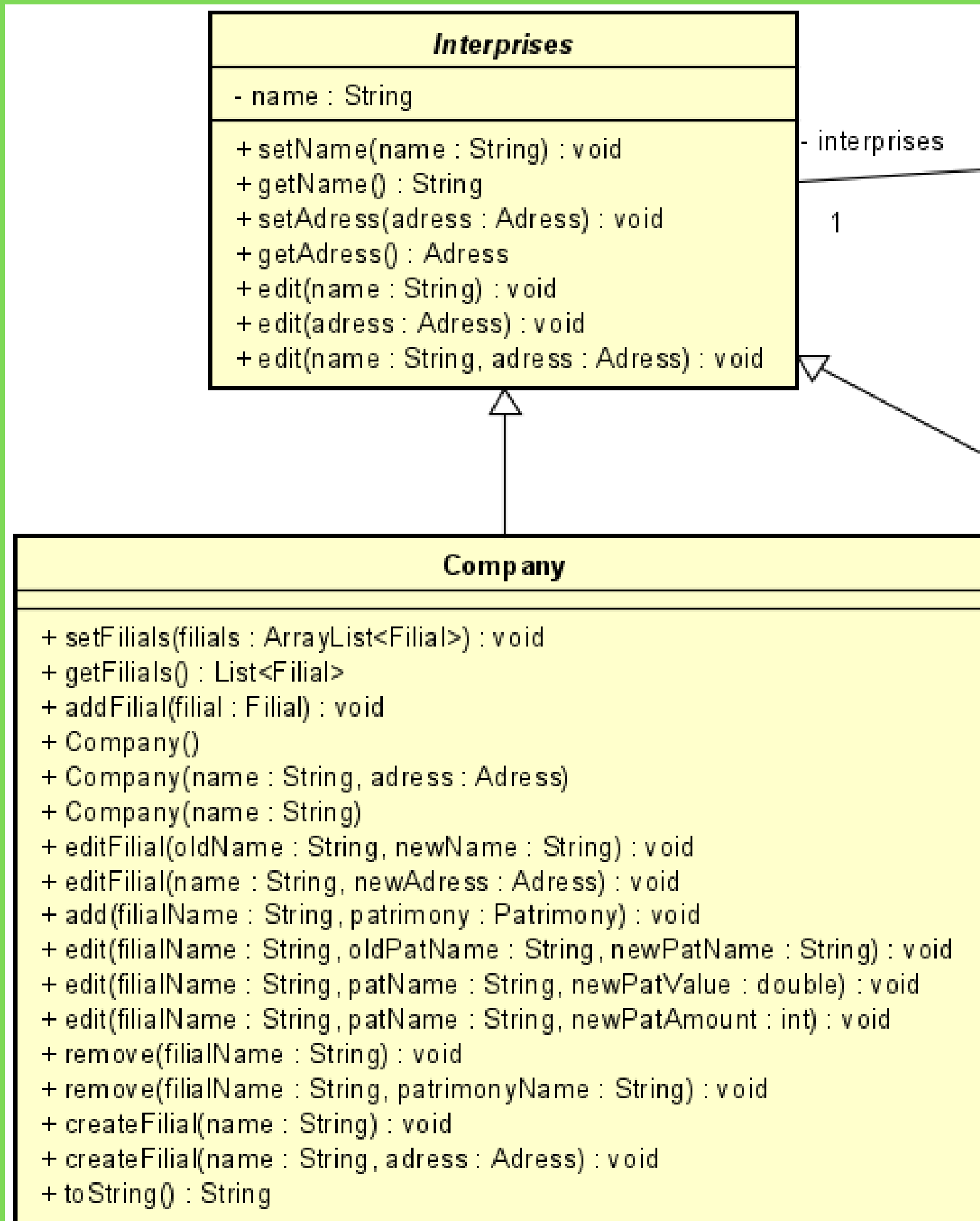
Relacionamentos



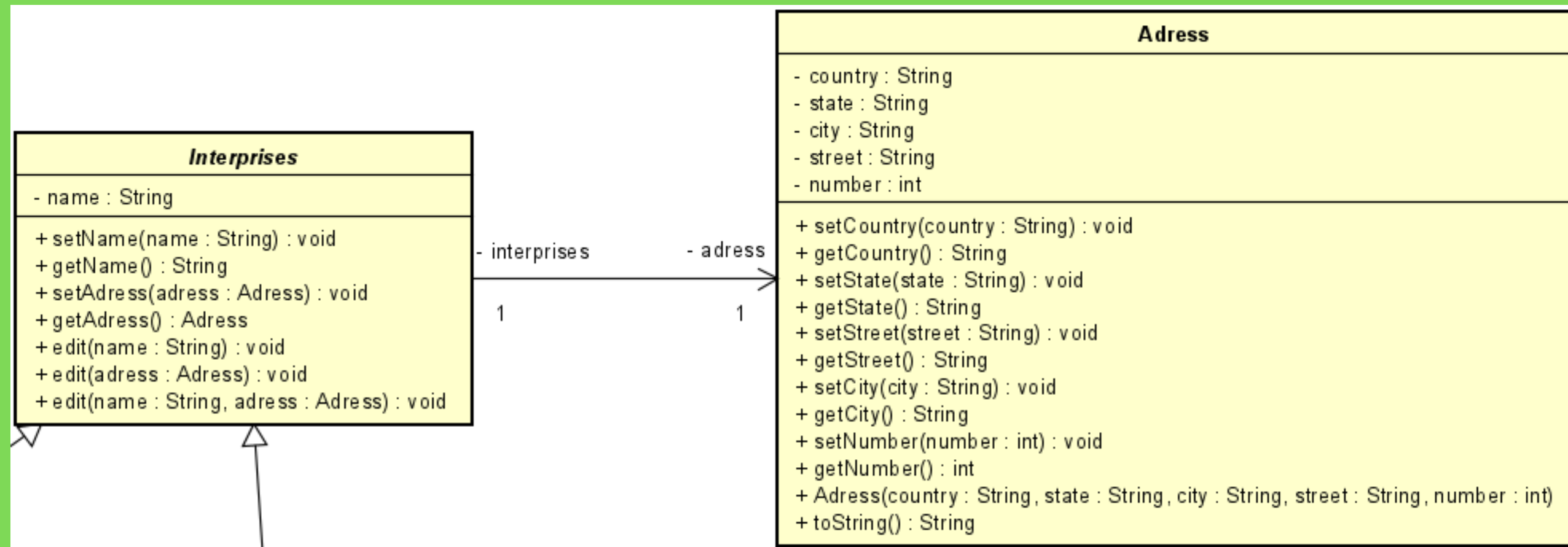
Relacionamentos



Relacionamentos



Relacionamentos



Por fim...

