Projeto Java com Anotações JML

Lógica Aplicada a Engenharia de Software

Docente: Marcel Vinicius Medeiros Oliveira

UFRN
UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

IMD
INSTITUTO METRÓPOLE DIGITAL

Autores

Diego

diegofilbal

Neylane

neylanepl

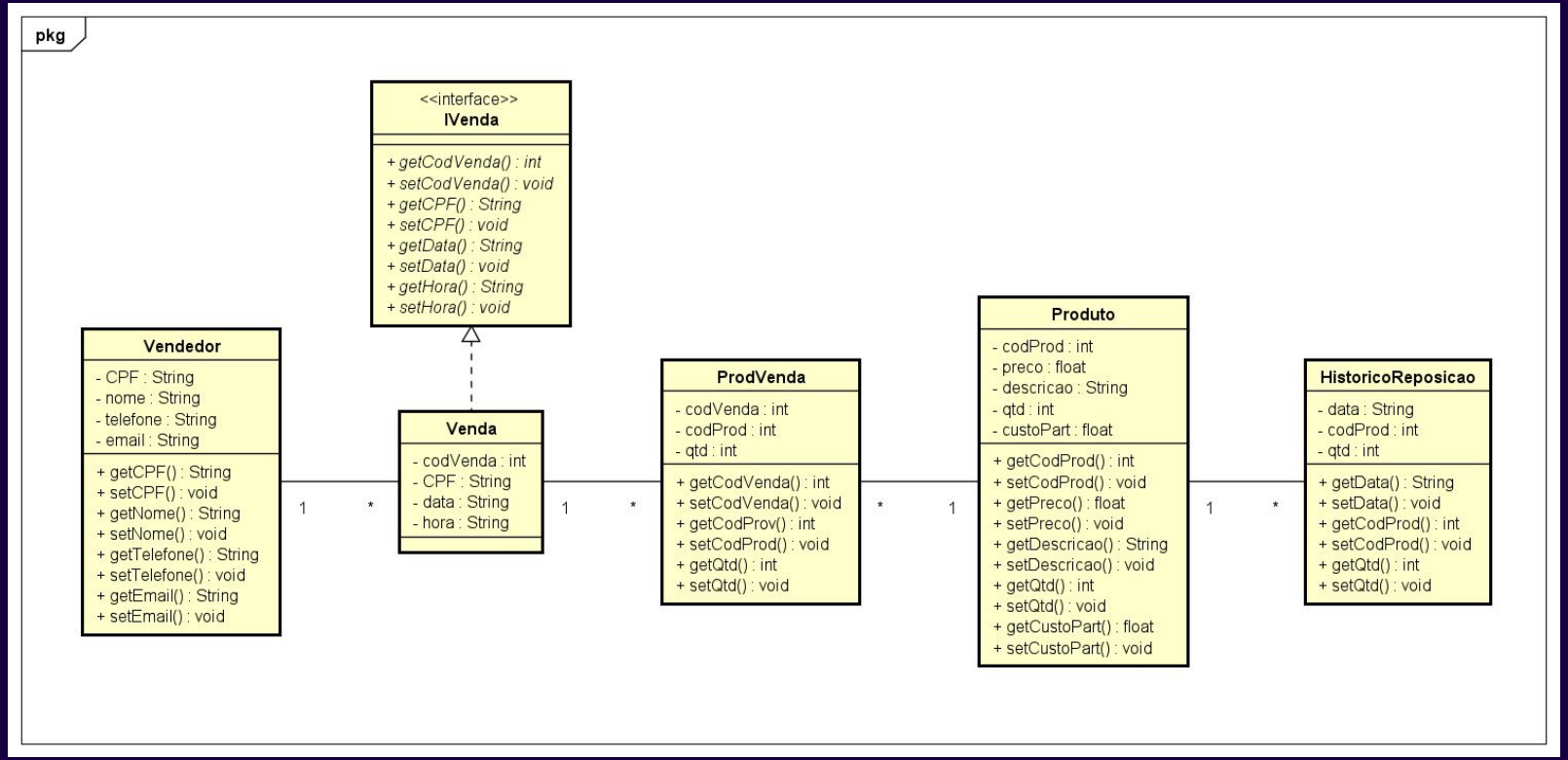
Raíssa

raixasantos

Projeto Escolhido

- Gestão Financeira
  - Sistema desenvolvido com o objetivo de gerenciar vendas de produtos, controlar gastos e obter informações de lucro obtido.
- Linguagem de programação: Java
- Autoria: Diego Filgueiras
- Github: https://github.com/diegofilbal/projGestaoFinanceira

Diagrama de Classes



Planilha com Quantitativos

Quantidade de Métodos Anotados	**87**
Herança e Interface	**1**
`invariant`	**7**
`initially`	**8**
`constraint`	**0**
`assignable`	**30**
`requires`	**182**
`ensures`	**36**
`normal behaviour` e `exceptional behaviour`	**2**
`\forall`	**1**
`\exists`	**1**
Modelos (`represents`)	**4**
behaviour subtyping	**4**

Principais Partes do Código Fonte Anotado

Classe Produto

- initially, invariant, requires, ensures, forall e exists

```
blic class Produto {
  private /*@ spec_public @*/ int codProd = 0;
  private /*@ spec_public @*/ float preco;
  private /*@ spec_public nullable @*/ String descricao;
  private /*@ spec_public @*/ int qtd = 0;
  private /*@ spec_public @*/ float custoPart = 0.f;
  private /*@ spec_public @*/ String[] desconto = {"0%", "15%", "20%"};
  private /*@ spec_public @*/ int descontoGanho = 1;

  //@ public initially descontoGanho == 1;

  /*@ public invariant 0 <= codProd;
  @ public invariant 0 <= qtd;
  @ public invariant 0.0 <= custoPart;
  @*/

  /*@ requires 0 <= codProd;
  @ requires 0.0 < preco;
  @ requires 0 <= qtd;
  @ requires 0.0 <= custoPart;
  @ ensures this.codProd == codProd && this.preco == preco && this.qtd == qtd;
  @ ensures this.descricao == descricao && this.custoPart == custoPart;
  @*/
  public Produto(int codProd, float preco, String descricao, int qtd, float custoPa
      this.codProd = codProd;
      this.preco = preco;
      this.descricao = descricao;
      this.qtd = qtd;
      this.custoPart = custoPart;
  }
```

Classe Produto

- initially, invariant, requires, ensures, forall e exists

```
125     //@ ensures \result != null && \result.length == 3;
126     //@ ensures (\forall int i; 0<= i && i < 3; \result[i] != null && !(\result[i].isEmpty()));
127     public String[] getDescontos() {
128         String[] desconto = {"0%", "15%", "20%"};
129         return desconto;
130     }
131
132     //@ requires 1 <= indice && 3 >= indice;
133     //@ ensures !(\result.equals(null));
134     //@ ensures (\exists int i; 0 <= i && i < desconto.length; \result.equals(desconto[i]));
135     public String getDesconto(int indice) {
136         return desconto[indice-1];
137     }
138
139     public /*@ pure @*/ int getDescontoGanho() {
140          return descontoGanho;
141      }
142
143     //@ requires 1 <= desconto && 3 >= desconto;
144     //@ ensures this.descontoGanho == desconto;
145     public void setDescontoGanho(int desconto) {
146         this.descontoGanho = desconto;
147     }
```

Interface IVenda

- invariant, requires, ensures, represents

```
3  public interface IVenda {
4      //@ public model instance int codVenda;
5      //@ public model instance String CPF;
6      //@ public model instance String data;
7      //@ public model instance String hora;
8
9      //@ ensures 0 <= \result;
10     public /*@ pure @*/ int getCodVenda();
11
12     public void setCodVenda(int codVenda);
13
14     //@ ensures !\result.equals(null);
15     public /*@ pure @*/ String getCPF();
16
17     public void setCPF(String CPF);
18
19     //@ ensures !\result.equals(null);
20     public /*@ pure @*/ String getData();
21
22     public void setData(String data);
23
24     //@ ensures !\result.equals(null);
25     public /*@ pure @*/ String getHora();
26
27     public void setHora(String hora);
28 }
29
```

Classe Venda

- invariant, requires, represents, initially, assignable e ensures

```
12 public class Venda implements IVenda {
13     private /*@ spec_public @*/ int _codVenda = 0; //@ in codVenda;
14     private /*@ spec_public nullable @*/ String _CPF;   //@ in CPF;
15     private /*@ spec_public nullable @*/ String _data;  //@ in data;
16     private /*@ spec_public nullable @*/ String _hora;  //@ in hora;
17
18     /*@ public invariant 0 <= _codVenda;
19     @*/
20
21     /*@ protected represents
22     @   codVenda <- _codVenda;
23     @*/
24
25     /*@ protected represents
26     @   CPF <- _CPF;
27     @*/
28
29     /*@ protected represents
30     @   data <- _data;
31     @*/
32
33     /*@ protected represents
34     @   hora <- _hora;
35     @*/
36
```

Classe Venda

- invariant, requires, represents, initially, assignable e ensures

```
74    /*@ also
75     @  requires CPF_.length() == 14;
76     @  requires CPF_.contains(".") == true;
77     @  requires CPF_.contains("-") == true;
78     @  assignable CPF;
79     @  ensures CPF.equals(CPF_);
80     @*/
81    public void setCPF(String CPF_) {
82        this._CPF = CPF_;
83    }
84
85    public /*@ pure @*/ String getData() {
86        return _data;
87    }
88
89    /*@ also
90     @ requires d.length() == 10;
91     @ requires d.contains("/") == true;
92     @ assignable data;
93     @ ensures data.equals(d);
94     @
95     */
96    public void setData(String d) {
97        this._data = d;
98    }
99
100   public /*@ pure @*/ String getHora() {
101       return _hora;
102   }
```

Classe HistoricoReposicao

- normal_behavior, exceptional_behavior, requires, assignable, ensures e also

```
62      /*@ public normal_behavior
63      @   requires 0 <= codProd;
64      @   assignable this.codProd;
65      @   ensures this.codProd == codProd;
66      @ also
67      @ public exceptional_behavior
68      @   requires 0 > codProd;
69      @   assignable this.codProd;
70      @   signals_only Exception;
71      @*/
72      public void setCodProd(int codProd) throws Exception {
73          if(codProd < 0) {
74              throw new Exception();
75          }
76          this.codProd = codProd;
77      }
78
79      public /*@ pure @*/ int getQtd() {
80          return qtd;
81      }
```

Execução do Sistema com as Anotações

Fim :)