

# **3º. Trabalho de POO**

**Data de Entrega:** 25/11/2020

**Natureza do Trabalho:** em grupo de 2 a 3 alunos

**Forma de Entrega:** em formato eletrônico. A pasta do projeto deve conter os arquivos gerados pela ferramenta NetBeans 8.2. A pasta que vai conter os arquivos do trabalho deve ser nomeada da seguinte forma:

**prjPOON1N2N3**

onde **N1N2N3** representa os nomes dos integrantes

Exemplo: **prjPOOCarlosChristinaVinicius**

**Enunciado:**

- Faça a implementação das operações de cadastro de Vendedor, de Cliente e de Produto de acordo com as especificações abaixo e seguindo as interações entre as classes definidas no diagrama de classes do 1º. Trabalho.

## Especificação das Operações

- Menu

- Na classe GuiMenu faça a inclusão das intruções abaixo:

```
private ArrayList<Pessoa> cadCliVend = new ArrayList<Pessoa>();  
private ArrayList<Produto> cadProd = new ArrayList<Produto>();
```

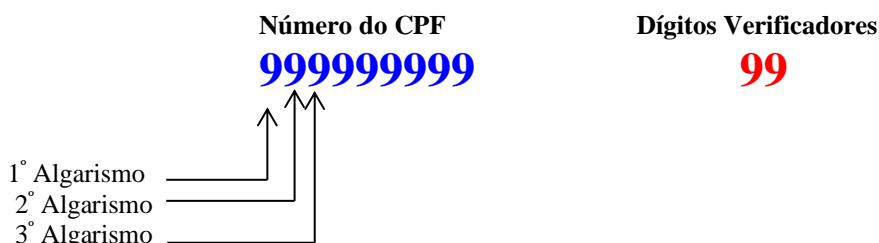
- Cadastro de Vendedor e de Cliente

- As operações de consulta, inserção, alteração e exclusão devem ser implementadas tendo como exemplo o projeto **prjExemploGuiPessoaFisicaJuridica**
  - Na operação Consultar caso seja informado um CPF inválido exiba uma mensagem de advertência e reposicione o foco na entrada de dados para o CPF. Caso o CPF seja válido, o cpf do vendedor ou cliente deve ser pesquisado no ArrayList cadCliVend. Caso seja encontrado o Vendedor no ArrayList cadCliVend, o conteúdo dos atributos do Vendedor ou do Cliente devem ser exibidos na GUI, o botão Consultar deve ser desabilitado e os botões Alterar e Excluir devem ser habilitados, a caixa de edição do CPF deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Caso não exista no ArrayList cadCliVend um vendedor ou um cliente com o cpf informado, o botão Consultar deve ser desabilitado e o botão Incluir deve ser habilitado. A caixa de edição do CPF deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome.
  - O CPF do vendedor ou do cliente deve ser validado, para tal, na classe Pessoa defina um método de classe que tenha como entrada de dados o CPF e tem como retorno um valor booleano.

```
public static boolean validarCPF(<string>){..... return(<boolean>);}
```

O CPF deve ser validado conforme os critérios definidos abaixo:

O CPF é um número composto por 9 algarismos. Os dígitos verificadores são calculados com base nos valores e na posição sequencial destes algarismos.



Inicialmente calcula-se o valor do primeiro dígito, que será utilizado posteriormente para o cálculo do segundo dígito.

- **Cálculo do 1º Dígito Verificador**

1. Multiplicar cada um dos 9 dígitos iniciais pelo número correspondente a sua posição no CPF, e somar os resultados de cada multiplicação:

$$(\text{Algarismo1} * \textcolor{blue}{1}) + (\text{Algarismo 2} * \textcolor{blue}{2}) + (\text{Algarismo 3} * \textcolor{blue}{3}) + \dots + (\text{Algarismo 9} * \textcolor{blue}{9})$$

2. Calcular o resto da divisão do valor obtido na soma por 11. Esse resto é o primeiro dígito verificador e, caso seja igual a 10, considerá-lo como sendo o número zero.

- **Cálculo do 2º Dígito Verificador**

1. Multiplicar cada um dos 10 algarismos disponíveis (os 9 do CPF mais o primeiro dígito verificador já obtido) de acordo com a seguinte regra: O primeiro (o mais à esquerda) por 11, o segundo por 10, o terceiro por 9, sucessivamente até o último (o mais à direita, que é o primeiro dígito verificador) que será multiplicado por 2, e somar os resultados de cada multiplicação:

$$(\text{Algarismo1} * \textcolor{red}{11}) + (\text{Algarismo2} * \textcolor{red}{10}) + (\text{Algarismo3} * \textcolor{red}{9}) + \dots + (\text{1º Dígito Verificador} * \textcolor{red}{2})$$

2. Multiplicar o valor obtido com a soma por 10 e, em seguida, calcular o resto da divisão desse resultado por 11. Esse resto é o segundo dígito verificador e, caso seja igual a 10, considerá-lo como sendo o número zero.

- Na operação Incluir deve ser instanciado um objeto da classe Vendedor que deve ser inserido na tabela de vendedores.
- Na operação Alterar as alterações devem ser feitas a partir do objeto vendedor instanciado que deve ser atualizado na tabela de vendedores.
- Na operação Excluir o objeto vendedor instanciado deve ser excluído da base de dados.

- Cadastro de Produto

- Na operação Consultar caso seja informado um Código inválido (qualquer valor diferente de um dado numérico inteiro) exiba uma mensagem de advertência e reposicione o foco na entrada de dados para o Código. Caso o Código seja válido, o código do produto deve ser no ArrayList de Produto. Caso seja encontrado o Produto no ArrayList, o conteúdo dos seus atributos devem ser exibidos na GUI, o botão Consultar deve ser desabilitado e os botões Alterar e Excluir devem ser habilitados, a caixa de edição do Código deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição nome. Caso não exista no ArrayList de Produto um produto com o código informado, o botão Consultar deve ser desabilitado e o botão Incluir deve ser habilitado. A caixa de edição do Código deve ser desabilitada e as demais caixas de edição devem ser habilitadas. O foco deve ser posicionado na caixa de edição descrição.
- As operações de consulta, inserção, alteração e exclusão devem ser implementadas tendo como exemplo o projeto **prjExemploGuiPessoaFisicaJuridica**