

Programação Orientada a Objetos

Atividade 2 - Prof. Dr. Aparecido V. de Freitas

1. Escrever uma classe Java que modela pontos no plano cartesiano 2D (eixos x e y). Um objeto do tipo Ponto tem dois atributos: abscissa x e ordenada y. Considerar os atributos do tipo inteiro. Desconsiderar nesse exercício o conceito de encapsulamento de dados (definir os atributos com atributo de acesso *public*). A classe deverá ser codificada num package chamado fatec.
 - A) Escrever o código construtor para instanciar pontos. Considerar o construtor default que cria pontos na coordenada (1,1) e outro construtor que recebe como parâmetros os valores correspondentes à abscissa e à ordenada. Criar ainda um construtor que cria um ponto a partir de um ponto dado (clone).
 - B) Escrever o código do método `imprimePonto()` que irá imprimir na console os valores das coordenadas do ponto.
 - C) Escrever o código do método `retornaAbscissa()` que irá retornar na console o valor da abscissa do ponto.
 - D) Escrever o código do método `retornaOrdenada()` que irá retornar na console o valor da ordenada do ponto.
 - E) Escrever o código do método `alteraAbscissa(int)` que irá alterar o valor da abscissa, considerando o novo valor passado como parâmetro.
 - F) Escrever o código do método `alteraOrdenada(int)` que irá alterar o valor da ordenada, considerando o novo valor passado como parâmetro.
 - G) Escrever o código do método `deslocaHorizontal(int)` que irá deslocar o ponto de forma horizontal, com o deslocamento passado por parâmetro. (translação)
 - H) Escrever o código do método `deslocaVertical(int)` que irá deslocar o ponto de forma vertical, com o deslocamento passado por parâmetro. (translação)
 - I) Escrever o código do método `retornaDistanciaOrigem()` que retorna a distância do ponto considerado até a origem do sistema de coordenadas (0,0).
 - J) Escrever o código do método `imprimeDistanciaOrigem()` que imprime a distância do ponto considerado até a origem do sistema de coordenadas (0,0).
 - K) Escrever o código do método `simetricoX()` que irá retornar o ponto que é simétrico ao ponto considerado em relação ao eixo-X.
 - L) Escrever o código do método `simetricoY()` que irá retornar o ponto que é simétrico ao ponto considerado em relação ao eixo-Y.
 - M) Escrever uma classe chamada `Atividade_2` no mesmo package fatec para exercitar os métodos da classe Ponto.

2. Criar uma classe Java chamada **Aluno** para modelar estudantes de uma instituição de Ensino. A classe deve definir os seguintes atributos de dados (propriedades) do estudante:

- ✓ nome – Nome do estudante - (tipo String)
- ✓ cpf – CPF do estudante – (tipo String)
- ✓ endereco – Endereço do Estudante (tipo String)

A classe deve conter um método para construir objetos (**construtor**) o qual deve ser codificado com os parâmetros correspondentes aos atributos do objeto a ser criado.

A classe também deve ter um método chamado **imprimeDados()** que irá exibir os dados do estudante. Esse método deverá apresentar os dados por meio de uma interface gráfica. Utilizar o método **showMessageDialog()** da classe **JOptionPane**. O código para exibir os dados pode ser escrito por meio de:

```
JOptionPane.showMessageDialog( null,
    "Nome do aluno: " + this.nome + "\n" +
    "CPF: " + this.cpf + "\n" +
    "Endereço: " + this.endereco + "\n" );
```

Criar na mesma classe um método **main()** que irá solicitar do usuário os dados de um objeto do tipo **Aluno** a ser criado. A entrada de dados deve ser feita por meio de interface gráfica com o método **showInputDialog()** da classe **JOptionPane**. O código para a entrada de dados pode ser escrito pelo código:

```
String nome = JOptionPane.showInputDialog("Informe o nome do aluno: ");
String cpf = JOptionPane.showInputDialog("Informe o CPF do aluno: ");
String endereco = JOptionPane.showInputDialog("Informe o endereço do aluno: ");
```

Na função **main**, criar um objeto do tipo **Aluno** com os dados informados e executar a função **imprimeDados()** para exibir os dados do estudante.

Obs. A classe deve ser criada num package chamado **fatec**.

3. Criar uma classe Java chamada **Aluno** para modelar estudantes de uma instituição de Ensino. A classe deve definir os seguintes atributos de dados (propriedades) do estudante:
- ✓ nome – Nome do estudante - (tipo String)
 - ✓ nota1 – Nota N1 do estudante – (tipo double)
 - ✓ nota2 – Nota N2 do do Estudante (tipo double)

A classe deve conter um método para construir objetos (**construtor**) o qual deve ser codificado com os parâmetros correspondentes aos atributos do objeto a ser criado.

A classe também deve ter um método chamado **imprimeDados()** que irá exibir os dados do estudante. Esse método deverá apresentar os dados por meio de uma interface gráfica. Utilizar o método **showMessageDialog()** da classe **JOptionPane**. O código para exibir os dados pode ser escrito por meio de:

```
public void imprimeDados() {  
    JOptionPane.showMessageDialog( null,  
        "Nome do aluno: " + this.nome + "\n" +  
        "Nota N1: " + this.nota1 + "\n" +  
        "Nota N2: " + this.nota2 + "\n" );  
}
```

Criar na classe um método chamado **retornaMedia()** que irá retornar o valor da média aritmética entre as notas N1 e N2.

Criar um método chamado **imprimeMedia()** que irá exibir o valor da média do estudante. Esse método deverá apresentar os dados por meio de uma interface gráfica. Utilizar o método **showMessageDialog()** da classe **JOptionPane**. O código para exibir os dados pode ser escrito por meio de:

```
JOptionPane.showMessageDialog( null,  
    "Media do aluno: " + this.retornaMedia() );
```

Criar um método chamado **imprimeResultado()** que irá exibir um texto correspondente ao resultado do desempenho escolar. Se a média aritmética for igual ou superior a 6.0, o método deve exibir o resultado “Aprovado”. Caso contrário deve exibir o resultado “Reprovado”. Esse método deverá apresentar os dados por meio de uma interface gráfica. Utilizar o método **showMessageDialog()** da classe **JOptionPane**.

Criar na mesma classe um método **main()** que irá solicitar do usuário os dados de um objeto do tipo **Aluno** a ser criado. A entrada de dados deve ser feita por meio de interface gráfica com o método **showInputDialog()** da classe **JOptionPane**. O código para a entrada de dados pode ser escrito pelo código:

```
String nome = JOptionPane.showInputDialog( "Informe o nome do aluno: " );  
String N1 = JOptionPane.showInputDialog( "Informe a nota N1 do aluno: " );  
String N2 = JOptionPane.showInputDialog( "Informe a nota N2 do aluno: " );
```

*Observação: Os dados de entrada na função ShowInputDialog() são do tipo String. Assim, para se criar o Aluno com os atributos corretos (nota1 e nota2 do tipo double) a função main() deve efetuar a conversão de tipos de String para double. Isso pode ser feito pela função **parseDouble()** da classe **Double**. O código pode ser escrito por meio de:*

```
double n1 = Double.parseDouble(N1);  
double n2 = Double.parseDouble(N2);
```

Na função **main()**, criar um objeto do tipo **Aluno** com os dados informados e executar as funções **imprimeDados()**, **imprimeMedia()** e **imprimeResultado()**.

Obs. A classe deve ser criada num package chamado **fatec**.