

Hortolândia, 18 de fevereiro de 2014.

Aulas 9 e 10 – Programação Orientada a Objetos (POO)

A presente aula objetiva explicar conceitos preliminares de programação, tais como processos e operações de abstração. Antes de se chegar ao assunto de abstração, entretanto, precisamos ainda conhecer um pouco sobre os programas de computador e os modelos de processos, assuntos estes abordados nas Seções 1 e 2, respectivamente.

Na sequência, são abordados os conceitos de classes e objetos, os quais implementam as abstrações de objetos do mundo real.

Po fim, um exemplo de classe é apresentado - a classe Lâmpada - em linguagem de programação Java.

1. O que é um programa de computador?

Um programa são conjuntos de comandos e regras que um programador deve conhecer para poder manipular os recursos de um computador.

Os programas são escritos usando linguagens de programação, que definem regras específicas e bem determinadas e um conjunto de operadores e comandos que podem ser usados. O conteúdo dos programas é chamado de código-fonte, sendo que este último é traduzido para linguagem de máquina através de um programa chamado de compilador.

Os programas processam dados, como por exemplo, valores em uma conta bancária, caracteres entrados por um teclado, pontos em uma imagem, etc. O paradigma da orientação a objetos considera que os dados a serem processados e os mecanismos de processamento destes dados devem ser considerados em conjunto.

2. O que são modelos?

Modelos são representações simplificadas de objetos pessoais, pessoas, itens, tarefas, processos, conceitos, idéias etc.

Como exemplo de modelo, tomemos o caso de um Restaurante Caseiro, que serve refeições por quilo, e onde o gerente é ao mesmo tempo a pessoa que fica na balança e no caixa. Este restaurante pode ter, por exemplo, uma planilha, controlada pelo gerente, onde são anotados os valores dos pesos dos pratos dos clientes, além de outros itens consumidos por cada mesa.

Esta planilha pode ter o formato descrito na Figura 1 a seguir.

Restaurante Caseiro Hipotético		
Mesa 1 [] Kg refeição [] Outros	Mesa 2 [] Kg refeição [] Outros	Mesa 3 [] Kg refeição [] Outros
Mesa 4 [] Kg refeição [] Outros	Mesa 5 [] Kg refeição [] Outros	Mesa 6 [] Kg refeição [] Outros

Figura 1: Exemplo de Modelo Simplificado de Restaurante.

A planilha, ou quadro-branco, é um possível modelo deste restaurante, representando de forma simplificada as informações do restaurante que são necessárias para a contabilização dos pedidos feitos para os garçons e o gerente.

O **modelo do restaurante** representa certos dados ou informações, que no caso são os itens e a quantidade dos pedidos por mesa. **Como o modelo é uma simplificação do mundo real, os dados contidos nele são somente os relevantes à abstração do mundo real sendo feita.**

Um modelo em geral contém também operações ou procedimentos associados a ele. Essas operações são listas de comandos que processarão os dados contidos no próprio modelo, bem como dados adicionais, caso necessário.

Algumas operações que poderiam fazer parte do modelo hipotético seriam: a inclusão de um determinado pedido por mesa, a modificação do estado de um pedido de uma mesa (servido ou não), o encerramento dos pedidos dos clientes de uma mesa e a apresentação da conta para os clientes.

2.1. Outro exemplo de modelo

Além do exemplo do restaurante, podemos também tomar o caso de uma **lâmpada incandescente comum**. Esta lâmpada pode ser acesa ou apagada, o que determina o seu estado – **ligada** ou **desligada**.

As operações que podem ser realizadas por uma lâmpada são simples: pode-se ligá-la e/ou desligá-la. O ato de ligar a lâmpada corresponde à modificação de seu estado para ligada, enquanto que o ato de desligar significa mudar o seu estado para desligada.

Para saber se uma lâmpada está ligada ou desligada, pode-se pedir que uma operação mostre o valor de seu estado.

De posse dos valores de estado possíveis para a lâmpada, e também das operações que podem ser realizadas por um objeto do tipo lâmpada, pode-se chegar a um modelo simples da mesma, o qual pode ser representado conforme o diagrama da Figura 2.

Lampada
estadoDaLampada
acender() apagar() mostrarEstado()

Figura 2: Modelo de uma Lâmpada Incandescente.

3. A abstração e o Processos de Abstração

A abstração é o processo utilizado na análise de uma situação real, através do qual se observa uma realidade, tendo-se por objetivo a determinação dos aspectos e fenômenos considerados essenciais, excluindo-se todos os aspectos considerados irrelevantes ou secundários.

O processo de abstração consiste basicamente em identificar, para um determinado objeto ou problema do mundo real, os seus principais aspectos e características, bem como quais são os seus principais comportamentos.

No caso particular da programação orientada a objetos, tais **abstrações do mundo real** são representadas na forma de **classes de objetos**.

Uma classe de objetos é, portanto, um modelo de determinados objetos do mundo real, os quais possuem características em comum.

No modelo de classes, cada classe é representada por um nome de classe, um conjunto de atributos e um conjunto de métodos (operações) que um objeto desta classe pode executar.

Como exemplos de classes, podem-se citar: a classe **Lampada** (apresentada na seção anterior), uma classe **Automovel**, uma classe **Estudante**, uma classe **Matricula**, uma classe **Funcionario**, e assim por diante.

4. Exemplo de Classe em Linguagem Java

A fim de ilustrar o exemplo da Classe Lâmpada de forma prática, a mesma será criada em Linguagem Java.

O código-fonte da **Listagem 1** pode ser utilizado para codificar a classe **Lâmpada**.

```
public class Lampada {  
    public boolean estado;  
    public Lampada(){ // método construtor da Classe Lâmpada  
        estado = false;  
    }  
    public void acender(){  
        estado = true;  
        System.out.println("Acende a lâmpada");  
    }  
    public void apagar(){  
        estado = false;  
        System.out.println("Apaga a lâmpada");  
    }  
    public boolean mostrarEstado(){  
        System.out.println("Estado da Lâmpada: " + estado);  
        return estado;  
    }  
}
```

Listagem 1: Classe Lâmpada.

Observem que a Classe Lâmpada apresentada na Listagem 1 não possui um método `main()`.

Isso significa que ela não pode ser executada diretamente pelo usuário, sendo necessária a criação de outra classe que instancie um objeto da classe `Lampada`, a fim de se poder testar esta última.

Por essa razão, para fins de teste da classe **Lampada**, será criada uma segunda classe, a qual possuirá o método **main()**. A classe principal de teste será denominada **ExemploLampada**.

```
public class ExemploLampada {  
    public static void main(String args[ ]){  
        Lampada lampada = new Lampada();  
        lampada.acender();  
        lampada.apagar();  
        lampada.mostrarEstado();  
    }  
}
```

Listagem 2: Classe ExemploLampada.

Esta classe ExemploLampada, após compilada, pode ser executada pelo usuário, chamando-se o interpretador java da seguinte forma:

java ExemploLampada

Ou, se este exemplo estiver sendo desenvolvido no NetBeans, pode-se configurar a classe ExemploLampada como a classe Principal do projeto, e executá-la por meio do botão "Executar".

5. Conclusões

O processo de abstração, quando aplicado na programação orientada a objetos, leva o desenvolvedor primeiramente a representar um determinado problema computacional a ser resolvido na forma de um modelo de classes de objetos.

Este modelo de classes obtido é então transformado, por meio da atividade de programação, em estruturas de classes, escritas de acordo com as regras de uma determinada linguagem de programação, escolhida pelo desenvolvedor. Tais classes são então compiladas (ou seja, transformadas em código executável) e então executadas pelo computador.

As classes, uma vez instanciadas pelo computador (ou seja, espaço de memória foi alocado para as mesmas), são chamadas de objetos. A representação dos objetos é semelhante à das classes, sendo que a única diferença é que os objetos possuem valores para seus atributos e podem ter seus métodos executados por outros objetos.

Ao término da presente aula, foi apresentado um primeiro exemplo de codificação de classe em Linguagem Java, a classe Lâmpada.

6. Referências Bibliográficas

BORATTI, Isaias Camilo. **Programação Orientada a Objetos usando Delphi**. Quarta Edição. Editora Visual Books. Florianópolis, 2007.

MELO, Ana Cristina Vieira de, SILVA, Flávio Soares Corrêa. **Princípios de Linguagens de Programação**. São Paulo: Edgard Blücher, 2003.

SANTOS, Rafael. **Introdução à Programação orientada a objetos usando Java**. 8a Reimpressão. Editora Campus - Elsevier. Rio de Janeiro, 2003.