

Programação Orientada a Objetos - Exercícios

Polimorfismo, classes abstratas e classe Object

Profa. Fernanda Maria da Cunha Santos

Obs.: Utilize as convenções na nomeação das classes, atributos e métodos.

- 1) Uma empresa tem quatro tipos de funcionários: chefes, vendedores, operários e horistas. Cada um desses funcionários ganha seus salários conforme regras distintas, apresentadas a seguir:
 - a) Chefe: salário fixo e predefinido;
 - b) Vendedor: valor fixo + comissão * vendas;
 - c) Operário: valor por produção x quantidade produzida;
 - d) Horista: valor por hora x total de horas trabalhadas.Crie as classes correspondentes, adicionando atributos e métodos que achar necessários. A classe `Funcionario` deve incluir os campos privados `nome`, `dataNascimento`. Além disso, crie uma classe `Empresa`, que conterá o método `main()`. A empresa tem vários funcionários, logo crie um vetor de funcionários. Adicione a ele ao menos um tipo de cada funcionário definido anteriormente. Crie um método `gerarFolhaPagamento()`, que lê dados referentes ao conjunto de empregados da empresa e mostra a folha de pagamento da empresa, contendo o nome dos funcionários e o salário bruto, bem como o total pago pela empresa em salários.
- 2) Escreva um programa que implemente a estrutura de hierarquia de classes `Forma Geométrica`, `Triângulo`, `Retângulo` e `Círculo` conforme instruções abaixo.
 - a) Os únicos atributos de `Forma Geométrica` devem ser `base` e `altura`, com o modificador de acesso conveniente, e as subclasses não devem possuir atributos.
 - b) `Triângulo`, `Retângulo` e `Círculo` são subclasses da **classe abstrata** `Forma Geométrica`, que deve forçar qualquer subclasse a possuir um método para calcular e retornar a área da forma geométrica.
 - c) Além disso, nas subclasses para as quais faça sentido, deve existir um método para calcular e retornar o perímetro.
 - d) Todas as subclasses devem conter um construtor sem interação com o usuário, método de impressão de dados que imprime na tela as dimensões da forma geométrica em questão, bem como sua área e perímetro (se disponível), e outros métodos que sejam obrigatórios para o programa.
 - e) Para o círculo, a base e a altura devem ser o valor do raio, e o construtor deve receber como argumento apenas o raio.
 - f) Reutilize códigos sempre que possível.
 - g) Implemente um programa principal que crie um vetor de objetos `Forma Geométrica`, e instancie pelo menos dois objetos de cada subclasse. Ao final, o programa deve mostrar uma lista dos objetos cadastrados, mostrando as suas informações (dimensões do objeto, sua área e seu perímetro – se disponível). Ao término das informações dos objetos, o programa deve mostrar qual é a média das áreas de todos os objetos.

- 3) Implemente a hierarquia de classes que contenha a superclasse (abstrata) `VeículoAutomotor`, com os atributos `marca`, `modelo`, `placa` e `ano de fabricação`, com os métodos necessários e relevantes, as subclasses `Carro` (quantidade de litros no porta-malas), `Moto` (cilindradas), e `Ônibus` (número máximo de passageiros), com os atributos indicados entre parênteses e métodos necessários e relevantes. Lembre-se de uma das boas características da OO que é a reutilização de código sempre que possível.
- a) O método `equals()` deve ser sobrescrito para que a comparação entre os objetos ocorra a partir da `marca`, `modelo` e `ano` para decidir se os veículos são iguais.
 - b) O método `toString()` também deve ser sobrescrito para que todas as informações do objeto sejam retornadas em uma `String` de forma organizada.
 - c) O `ônibus` deve ter uma limitação mínima no número máximo de passageiros que deve ser inicializada na declaração da classe, é o mesmo para todos os objetos, e não deve ser modificado em tempo de execução. O sistema não pode aceitar um cadastro que viole esta limitação, e a verificação se essa violação ocorre deve ser feita por meio de um método estático da classe `ônibus`.
 - d) Escreva o programa principal que cadastra em um vetor e mostra na tela pelo menos 3 objetos de cada subclasse.