

Hortolândia, 11 de setembro de 2013.

Aulas 27 e 28 – Exercícios de Encapsulamento e Comunicação entre as classes

Na aula anterior foram apresentados conceitos sobre encapsulamento, sua utilidade e como influenciam o processo de comunicação entre as classes de um projeto. Nesta aula serão realizados diversos exercícios para provar e fixar os conceitos aprendidos.

I. Professores

Nome do Professor	Turma	E-mail
Arthur A. B. Buccioli	A	arthurbuccioli@gmail.com
Leandro C. Ledel	B	leandro.ledel@gmail.com ledel@ifsp.edu.br

II. Exercícios teóricos

1. Qual a finalidade dos pacotes em Java?
2. Explique o funcionamento do modificador de acesso `public` para classes?
3. Quando uma classe é declarada sem nenhum modificador de acesso, como o compilador interpreta a acessibilidade desta classe?
4. acesso `package-private` para classes e exemplifique a declaração de uma classe deste tipo.
5. Um membro de classe declarado com acesso `private` pode ser acessado por outras classes, dentro do mesmo pacote da classe onde ele se encontra?
6. De que maneiras os modificadores de acesso afetam os programadores?
7. Por que devemos evitar campos de acesso público?

III. Exercícios Práticos

1. Crie um projeto Java, insira nele 1 pacote, depois um subpacote e por último uma classe Chamada HelloWorld neste último subpacote. Observe como ficou a estrutura do projeto no explorador de arquivos.
2. Partindo do exercício 1, crie na classe HelloWorld um método chamado mensagem que mostre na tela a mensagem "Hello encapsulamento". Crie agora um novo pacote com uma classe chamada Main e um método main dentro dela. Importe a classe HelloWorld do outro pacote, crie um objeto a partir dela e utilize o método mensagem previamente implementado.
3. Crie uma classe Carro com os atributos modelo, peso e preço. defina estes atributos com o encapsulamento `private`. Agora, para acessá-los, faça métodos set/get para cada um deles. Por fim, crie uma nova classe chamada Main com o método main, e dentro deste método crie um objeto da classe Carro e use seus métodos para ler e gravar valores no objeto.
4. Para testar o encapsulamento a nível de classe, vamos criar um projeto com 2 pacotes. no pacote 1 crie uma classe chamada ClassePrivada e outra chamada ClassePublica. defina o encapsulamento da classe ClassePublica como `public` e deixe o da outra classe em branco. Agora dentro do pacote 2 crie uma classe Main

e importe o pacote1 com todas suas classes. tente criar um objeto de cada uma das 2 classes. Anote os possíveis erros e defina suas causas.

5. Crie uma classe Data com os atributos dia, mes e ano. Defina todos os atributos como private. Crie métodos set/get para acesso aos atributos. Crie também um método com retorno boolean chamado setData para receber uma data. Nesse método, implemente regras de validação simples: nenhum valor menor que 0, dia<=31 mes<=12. Caso algum valor viole uma dessas regras, não modifique o valor dos atributos e retorne false. caso tudo dê certo, retorne true. Utilize esse mesmo método dentro de um construtor da classe. Além disso, crie um método chamado getDataText que retorne a data em texto no formato "DD/MM/AAAA". Por fim crie uma classe Main que utilize e teste a classe Data.