

Pacotes Java

Computação Orientada a Objetos

Prof. Flávio Luiz Coutinho

EACH/USP

Motivação

- Como organizar o código de um programa/projeto de grande porte?
- Modularização de código:
 - Separação do código em unidades menores, cada uma com um papel bem definido.
 - Linguagens OO como o Java já incentivam a modularização. Programa: composição de vários objetos.
 - Cada classe publica tem seu próprio arquivo, cujo nome é o nome da própria classe.

Motivação

- Essa modularização em classes é suficiente para organizar um projeto que possui centenas (ou milhares) de classes?
- Imagine o caos que seria ter todos os seus arquivos de um computador em um só diretório.
- Possível, mas nada prático.
- Aproveitar conceito de diretórios existentes nos sistemas de arquivos.
- Uso de estrutura hierárquica de diretórios para organizar classes do projeto de acordo com suas funcionalidades.

Pacotes Java

- No Java, uma coleção de classes agrupadas em um diretório é chamada de **pacote**.
- O conceito de pacote vai além da organização das classes em uma estrutura de hierárquica diretórios.
- O caminho no qual a classe se encontra passa a integrar o nome completo da classe em questão.
- Além da estrutura de diretórios, é preciso especificar no código fonte das classes o nome do pacote ao qual ela pertence.

Criando pacotes

- Instrução **package**.
- A omissão do **package** torna a classe membro do pacote padrão.
- Diretórios dos pacotes devem estar acessíveis a partir dos diretórios especificados no **classpath**.
- Classes do pacote padrão são acessíveis se estiverem diretamente nos diretórios especificados no **classpath**.
- Lembrando que normalmente o **classpath** inclui o diretório corrente.

Exemplo

- Exemplo de criação de pacote.

Usando classes de pacotes

- Há basicamente 3 formas de utilizar classes definidas em pacotes diferentes do atual:
 - Especificar o nome completo da classe:
 - Importar todo o pacote no qual está a classe desejada.
 - Importar apenas a classe desejada.
 - Instrução **import**.
- Pacotes que são importados automaticamente:
 - java.lang
 - pacote default
 - pacote atual

Exemplo

- Exemplo de uso de pacotes.

Convenção para nomeação

- Uso de pacotes ajuda a resolver questões de ambiguidade (classes com o mesmo nome, em pacotes diferentes).
- Mas e se houver duas classes de mesmo nome em pacotes de mesmo nome???
- Convenção: uso do próprio domínio da empresa, em ordem reversa, como nome do pacote:
- **`package br.com.empresa.projeto.modulo;`**

Escopo

- **public**: recursos public podem ser usados em qualquer classe.
- **private**: recursos só podem ser usados nas classes que os definem.
- **default**: recursos podem ser usados por todas as classes do mesmo pacote.
- **protected**: recursos podem ser usados por todas as classes do mesmo pacote, mais subclasses, mesmo que as subclasses sejam de pacotes diferentes.
- *recursos = classes, métodos ou variáveis.*