

PROGRAMAÇÃO ORIENTADA A OBJETOS EM JAVA

API, PACOTES E JAVADOC



18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

1



AGENDA

1. API do Java.
2. Pacotes (**packages**) e o utilitário **jar**.
3. Documentação Java e o utilitário **javadoc**.

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

2



API DO JAVA

- A biblioteca de classes Java contém milhares de classes e interfaces predefinidas;
- Os programadores podem utilizá-las para escrever seus próprios aplicativos;
- As classes são agrupadas em pacotes com base em suas funcionalidades:
 - java.io: processamento de arquivos e fluxos.
 - java.net: aplicativos de rede.

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

3



API DO JAVA

- Uma API (**Application Programming Interface**) define como um software pode ser utilizado;
- A **documentação da API** do Java lista os membros **public** e **protected** de cada classe e os membros **public** de cada interface na biblioteca de classes Java;
- A documentação apresenta uma visão geral de todas as classes e interfaces.

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

4



API DO JAVA

- Programadores pesquisam a API para encontrar o seguinte:
 - O pacote que contém uma classe ou interface particular;
 - Relacionamento entre uma classe ou interface particular e outras classes e interfaces;
 - Constantes de classes ou de interfaces;
 - Construtores para determinar como um objeto da classe pode ser inicializado;
 - Métodos de uma classe, os números e os tipos dos argumentos, os tipos de retorno e quaisquer exceções que poderiam ser lançadas no método;

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

5



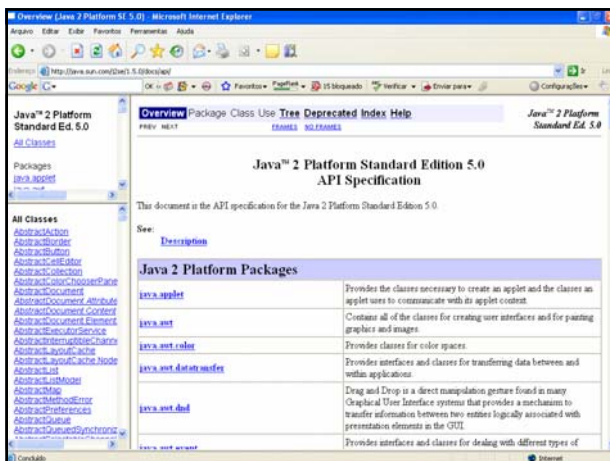
NAVEGANDO PELA API DO JAVA

- **Frames** da página;
- **Links** importantes;
- Visualizando a página **index.html**;
- Visualizando um pacote específico;
- Visualizando os detalhes de uma classe;
- Seções de resumo dos métodos;
- Seções de detalhes dos métodos.

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

6



PACOTES (PACKAGES)

- Java permite agrupar classes em uma coleção chamada **pacote**;
- Os pacotes são convenientes para organizar o trabalho e separá-los de bibliotecas de código fornecidas por terceiros;
- Um motivo para se aninhar pacotes é garantir a exclusividade dos nomes dos pacotes.

8

PACOTES (PACKAGES)

- Ao escrever um pacote, é preciso colocar o nome do pacote no topo do **arquivo-fonte**, antes do código que defina as classes no pacote;
- Ou seja, a instrução **package** precisa ser a primeira instrução no arquivo depois de qualquer comentário.

9

PACOTES (PACKAGES)

- Se você não colocar uma instrução **package** no arquivo-fonte, então a linguagem Java vai adicionar as classes deste arquivo-fonte no que é chamado de **pacote padrão**.
- As classes sem declaração do pacote no diretório atual se tornarão automaticamente parte do pacote padrão.

10

PACOTES (PACKAGES)


- As classes Java que são armazenadas na biblioteca de classes não estão automaticamente disponíveis para utilização, como as outras classes no projeto atual;
- Em vez disso, devemos declarar em nosso código fonte que gostaríamos de utilizar uma classe a partir da biblioteca;
- Isto é chamado **importar a classe** e é feito utilizando a instrução **import**.

11

PACOTES (PACKAGES)

- Há uma exceção a essas regras:
 - Algumas classes são utilizadas tão freqüentemente que quase toda classe as importaria;
 - Essas classes foram colocadas no pacote **java.lang** e esse pacote é importado automaticamente em toda classe.


12



DOCUMENTAÇÃO JAVA

- A interface de uma classe descreve o que uma classe faz e como ela pode ser utilizada sem mostrar a respectiva implementação.


18/8/2007 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 13



DOCUMENTAÇÃO JAVA

- A documentação inclui partes diferentes das informação.
- Elas são, entre outras coisas:
 - O nome da classe;
 - Uma descrição geral do propósito da classe;
 - Uma lista dos construtores e dos métodos da classe;
 - Os parâmetros e os tipos de retorno para cada construtor e método;
 - Uma descrição do propósito de cada construtor e cada método;
- Estas informações, juntas, são chamadas **interface de uma classe**.


18/8/2007 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 14



DOCUMENTAÇÃO JAVA

- A **interface de um método** consiste na assinatura do método e em um comentário;
- A assinatura de um método inclui (nessa ordem):
 - Um modificador de acesso;
 - Um tipo de retorno
 - O nome do método;
 - Uma lista de parâmetros;
- A interface fornece tudo o que precisamos conhecer para utilizar este método.


18/8/2007 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 15



ELEMENTOS DA DOCUMENTAÇÃO DE CLASSES

- A **documentação de uma classe** deve incluir pelo menos:
 - O nome da classe;
 - Um comentário que descreva o propósito geral e as características da classe;
 - Um número de versão;
 - O nome do autor (ou autores);
 - A documentação de cada construtor e cada método.


18/8/2007 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 16



ELEMENTOS DA DOCUMENTAÇÃO DE CLASSES

- A documentação de cada construtor e método deve incluir:
 - O nome do construtor ou método;
 - O tipo de retorno (para métodos apenas);
 - O nome e os tipos de parâmetros;
 - Uma descrição do propósito e da função do construtor ou método;
 - Uma descrição de cada parâmetro;
 - Uma descrição do valor retornado (para métodos apenas).


18/8/2007 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 17



ELEMENTOS DA DOCUMENTAÇÃO DE CLASSES

- Alguns elementos da documentação podem ser pré-extraídos do código fonte:
 - Campus públicos;
 - Nomes dos métodos;
 - Parâmetros dos métodos;
 - Etc.
- Outros precisam de mais atenção, pois podem ser facilmente esquecidos:
 - Comentários que descrevem a classe;
 - Comentários de métodos e parâmetros.
- Estes podem estar incompletos ou incorretos.

18/8/2007 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 18



ELEMENTOS DA DOCUMENTAÇÃO DE CLASSES


- Em Java, os comentários de **javadoc** são escritos com um símbolo especial de comentário no início:

```
/**  
    Este é um comentário javadoc.  
*/
```

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

19




ELEMENTOS DA DOCUMENTAÇÃO DE CLASSES

- Vários símbolos de chave especiais estão disponíveis para formatar a documentação;
- Esses símbolos de chave começam com o símbolo @ e incluem:
 - @version
 - @author
 - @param
 - @return
 - @since
 - @throws

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

20



ELEMENTOS DA DOCUMENTAÇÃO DE CLASSES

- @ author:** atribui uma classe ou método a um autor;
- @ since:** data da escrita do código;
- @ version:** versão de determinada classe ou método;
- @ see:** indicação de "Leia Mais";
- @ param:** parâmetro de um método;
- @ return:** retorno de um método;
- @ throws:** possíveis exceções lançadas;
- @ deprecated:** componente deve ser removido nas próximas versões.

18/8/2007

Programação Orientada a Objetos
Prof. Ademir Schmitz, M.Sc.

21