

PROGRAMAÇÃO ORIENTADA A OBJETOS EM JAVA

FRAMEWORK COLLECTIONS



20/6/2008

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

1



AGENDA

1. Introdução
2. Collections Framework
3. Principais Interfaces
4. Considerações Finais

20/6/2008

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

2



INTRODUÇÃO

- A utilização de arrays é complicada em muitos pontos:
 - Não podemos redimensionar um array em Java;
 - É impossível buscar diretamente por um determinado elemento para o qual não se sabe o índice;
 - Não conseguimos saber quantas posições do array já foram populadas sem criar, para isso, método auxiliares.

20/6/2008

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

3



INTRODUÇÃO

- Para suprir as necessidades impostas com a utilização dos arrays, a Sun criou um conjunto de classes e interfaces conhecido como **Collections Framework**.

20/6/2008

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

4



COLLECTIONS FRAMEWORK

- O **Collections** reside no pacote `java.util`.
- A **API** do **Collections** é robusta e possui diversas classes que representam estruturas de dados avançadas.
- Isto faz com que seja desnecessário reinventar a roda e criar uma lista encadeada, mas sim utilizar aquela que a Sun disponibilizou.

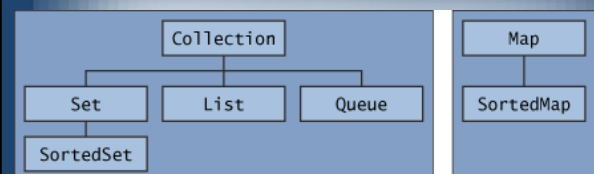
20/6/2008

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

5




COLLECTIONS FRAMEWORK



20/6/2008

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


6



LISTA: java.util.List

- Uma lista é um coleção que permite elementos duplicados e mantendo uma ordenação específica entre os elementos.
- Principais implementações:
 - **ArrayList**: utiliza um array interno para gerar uma lista.
 - **LinkedList**: utiliza o conceito de encadeamento dinâmico para gerar uma lista.


20/6/2008 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 7



CONJUNTO: java.util.Set

- Um conjunto funciona de forma análoga aos conjuntos da matemática.
- É uma coleção que não permite elementos duplicados.
- Principais Implementações:
 - **HashSet**: utiliza o conceito de hashing para criar o conjunto.
 - **TreeSet**: utiliza o conceito de árvore para criar o conjunto.


20/6/2008 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 8



ITERADOR: java.util.Iterator

- Utilizado para percorrer um coleção.
- Toda coleção fornece acesso a um iterador.


20/6/2008 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 9



MAPAS: java.util.Map

- Um mapa é composto de uma associação de um objeto chave a um objeto valor.
- Principais implementações:
 - Hashtable
 - HashMap

20/6/2008 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 10



CONSIDERAÇÕES FINAIS

- Para trabalhar com Collections, certifique-se de as implementações de **equals()** e **hashCode()** tenham sido feitas corretamente;
- Procure conhecer o conceitos envolvidos em cada uma das estruturas para escolher qual utilizar (algumas podem ser mais rápidas dependendo da situação).

20/6/2008 Programação Orientada a Objetos Prof. Ademir Schmitz, M.Sc. 11