

POO – Programação Orientada à Objetos

Java Collections



Gleyson Sampaio
Instrutor

+10 anos de experiência com desenvolvimento de software nos segmentos de:
Varejo, Bancos, Seguradoras e Instituições de ensino.
www.digytal.com.br



Gleyson Sampaio

Instrutor

+10 anos de experiência com desenvolvimento de software nos segmentos de: Varejo, Bancos, Seguradoras e Instituições de ensino.

 [linkedin.com/in/glysns](https://www.linkedin.com/in/glysns)



Oracle Certified Professional

Java SE 6 Programmer



Gama Academy

Instrutor e Criador de conteúdo educacional



Grupo TCI

Líder de Desenvolvimento de Software



BM&F Bovespa

Consultor Java



Daniele Banco

Analista de Sistemas Sênior

Conteúdo elaborado

Fundamentos

O que é uma coleção ?

Java Collections

Interfaces - especificação

Classes - implementação

Algoritmos

Lambda (funcional)

Coleção

Reunião de objetos de mesma natureza

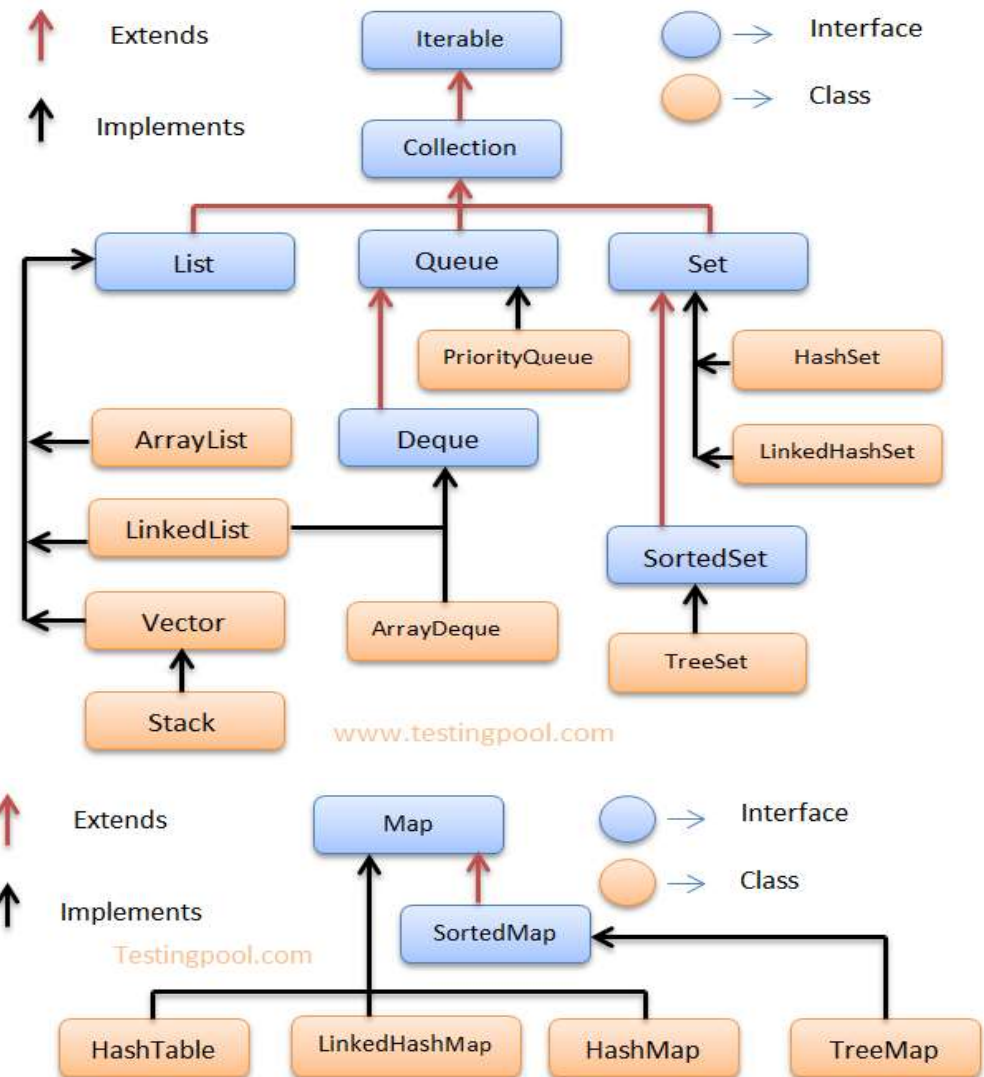
Em computação, uma coleção é um conjunto de dados geralmente do mesmo tipo que corresponde a um contexto abstracto que pode ser representado por estruturas como Listas, Conjuntos, Grafos e etc.



Collections Framework

JAVA

Coleções em Java é um conjunto de classes e interfaces que implementam estruturas de dados de coleção comumente reutilizáveis.



Java Collections Framework

LIST

Uma coleção ordenada que pode conter elementos duplicados orientada a índice.

SET

É uma coleção que não permite a inclusão de elementos repetidos considerando algoritmo de hash.

QUEUE

Estrutura que mantém os elementos prestes a serem processados na ordem fila.

MAP

Associa cada elemento à uma chave, não permitindo chaves nulas.

LIST

Uma lista que se preocupa com o índice.

INDEXADA

Lista que se preocupa com índice, permitindo assim a disposição de métodos associados à posição dos elementos como: **get(posix)**, **indexOf(e)**, **add(posix, e)**

ORDENADA

Todas as implementações de List são ordenadas por posição de índice, permitindo à inclusão em uma posição específica ou no final da lista.

IMPLEMENTAÇÕES

- Vector: lista segura – threadsafe (baixo desempenho)
- ArrayList: lista NÃO segura - (alto desempenho)
- LinkedList: Melhor aplicável em lista estila FILA.

SET

Não permite duplicidade com base em algoritmo de hash.

HASH – BUSCA PARA TEMPO LINEAR

A busca em um Set pode ser mais rápida do que em um objeto do tipo List, pois diversas implementações se utilizam de tabelas de espalhamento (hash tables), realizando a busca para tempo linear ($O(1)$).

NÃO ORDENADA

A ordem na qual os elementos são armazenados pode não ser aquela em que eles foram inseridos

IMPLEMENTAÇÕES

- HashSet: Interação rápida sem ordenação
- LinkedHashSet: Dispõe da ordenação em um conjunto
- TreeSet: Aplica **classificação** com base em Comparator

QUEUE

Estrutura de dados projetada para ter elementos inseridos no final da fila e elementos removidos do início da fila.

SEQUÊNCIA NATURAL OU PRIORIDADE

Uma fila normal considera a ordem de inserção para a remoção do elemento, já uma prioridade aplica um critério de seleção do próximo elemento à ser removido.

ESPIONAGEM DE ELEMENTOS

Existem métodos que verificam qual o próximo elemento, mas não remove a lista é o caso dos métodos: `element()` e `peek()`

IMPLEMENTAÇÕES

- LinkedList: Implementação padrão que agiliza a inserção na cauda e remoção no início.
- PriorityQueue: Aplica uma classificação com base em um Comparator, ver (Comparator x Comparable)

MAP

Mapeamento entre chave e valor um pouco diferente dos demais tipos de coleções.

INÚMERAS IMPLEMENTAÇÕES

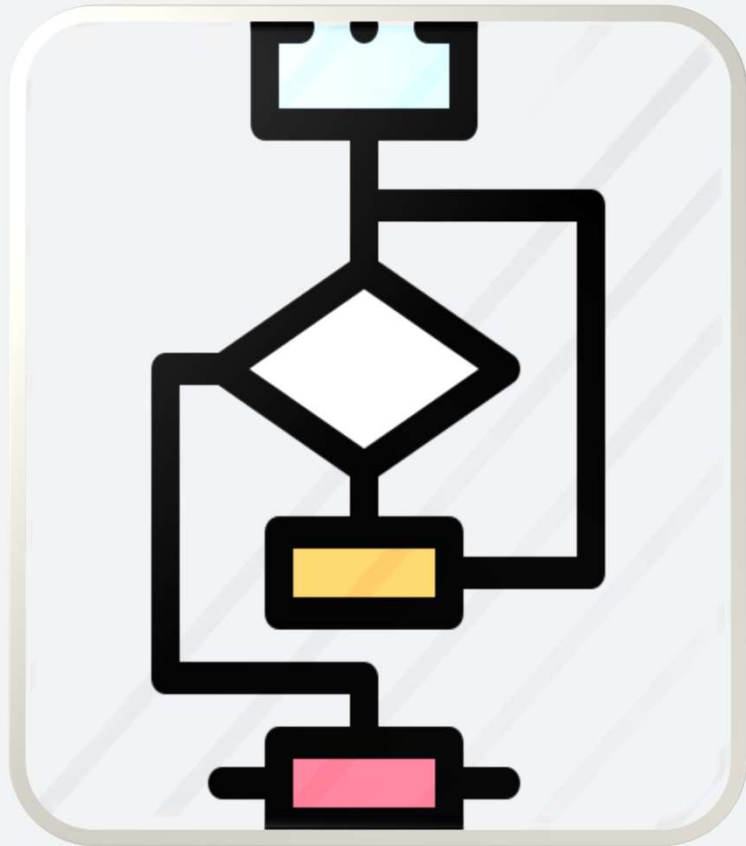
A interface Map dispõe de inúmeras implementações, mas as mais utilizadas são HashMap e TreeMap.

ORDENADA ou NÃO ORDENADA

Quando se tratar de performance use HashMap mas se necessita garantir a ordem use TreeMap.

IMPLEMENTAÇÕES

- HashMap: Nenhuma garantia de ordem de inserção.
- TreeMap: Sei que é classificada, mas use com cuidado.



Algoritmo

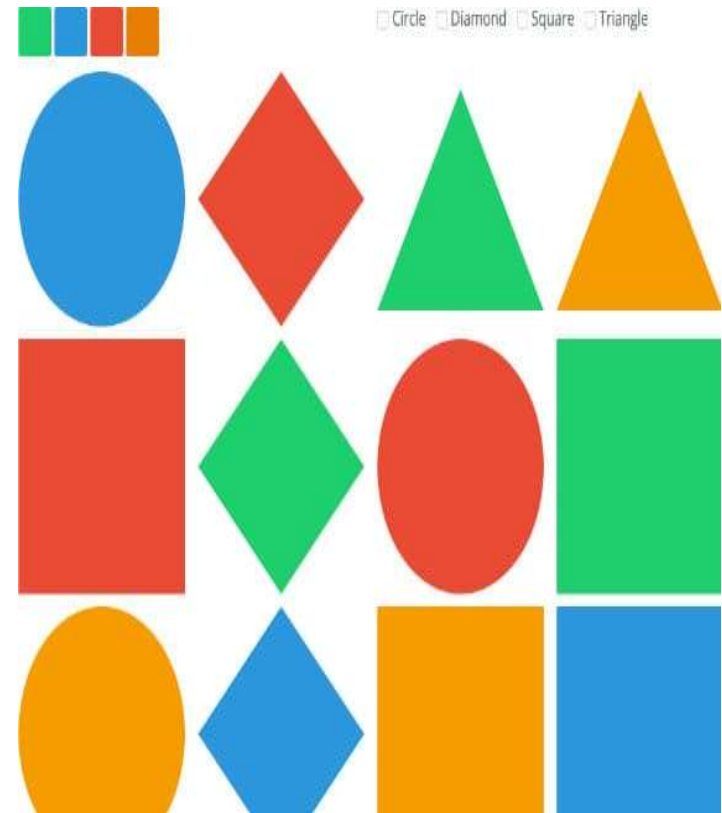
Sequência finita de ações com um propósito.

O termo pode ser entendido como uma sequência de raciocínios, instruções ou operações para alcançar um objetivo, sendo necessário que os passos sejam finitos e operados sistematicamente.

Algorithm Collections

JAVA

A estrutura de coleções Java também fornece diversos algoritmos de alto desempenho para manipular elementos e coleções, na maioria das vezes como recursos estáticos.



Algoritmo Collections

SORT

Recurso disponível para a classificação de uma coleção com base em um atributo comparável.

REVERSE

Inverte os elementos de uma coleção.

SUFFLE

Ordena aleatoriamente os elementos (embaralha)

COPY

Copia todos os elementos de uma coleção para outra

Lambda (Funcional)

JAVA

“As expressões Lambda representam uma função anônima. Então, elas são como métodos, mas não são realmente um método. Elas são como funções anônimas no sentido de que elas tem as mesmas funcionalidades de um método.”

Simon Ritter



Java Stream - API

FOR EACH

Forma mais elegante de interagir sobre o elementos

FILTER

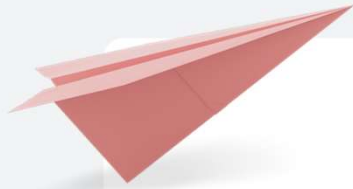
Mecanismo para filtrar elementos de uma coleção com base um ou mais critérios.

MAP

Recurso que converte objetos retornando uma coleção com outra estrutura.

PREDICATE

Maior flexibilidade quanto aos critérios aplicados.



Alguma pergunta?

Apresentação sobre Java Collections



Map.put
(tanu,
Obrigado)

