
Collections

Collection là gì?

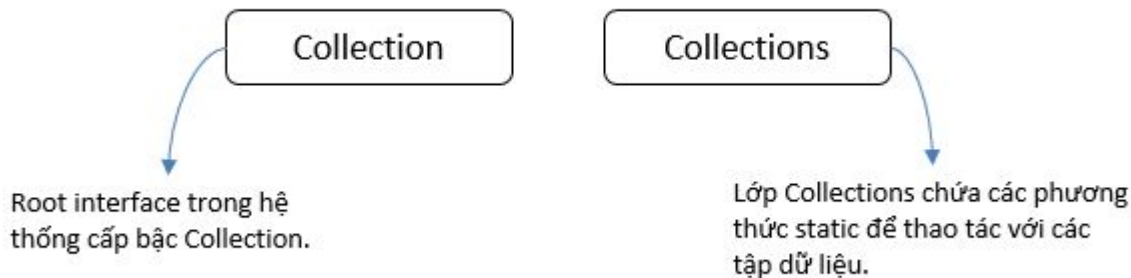


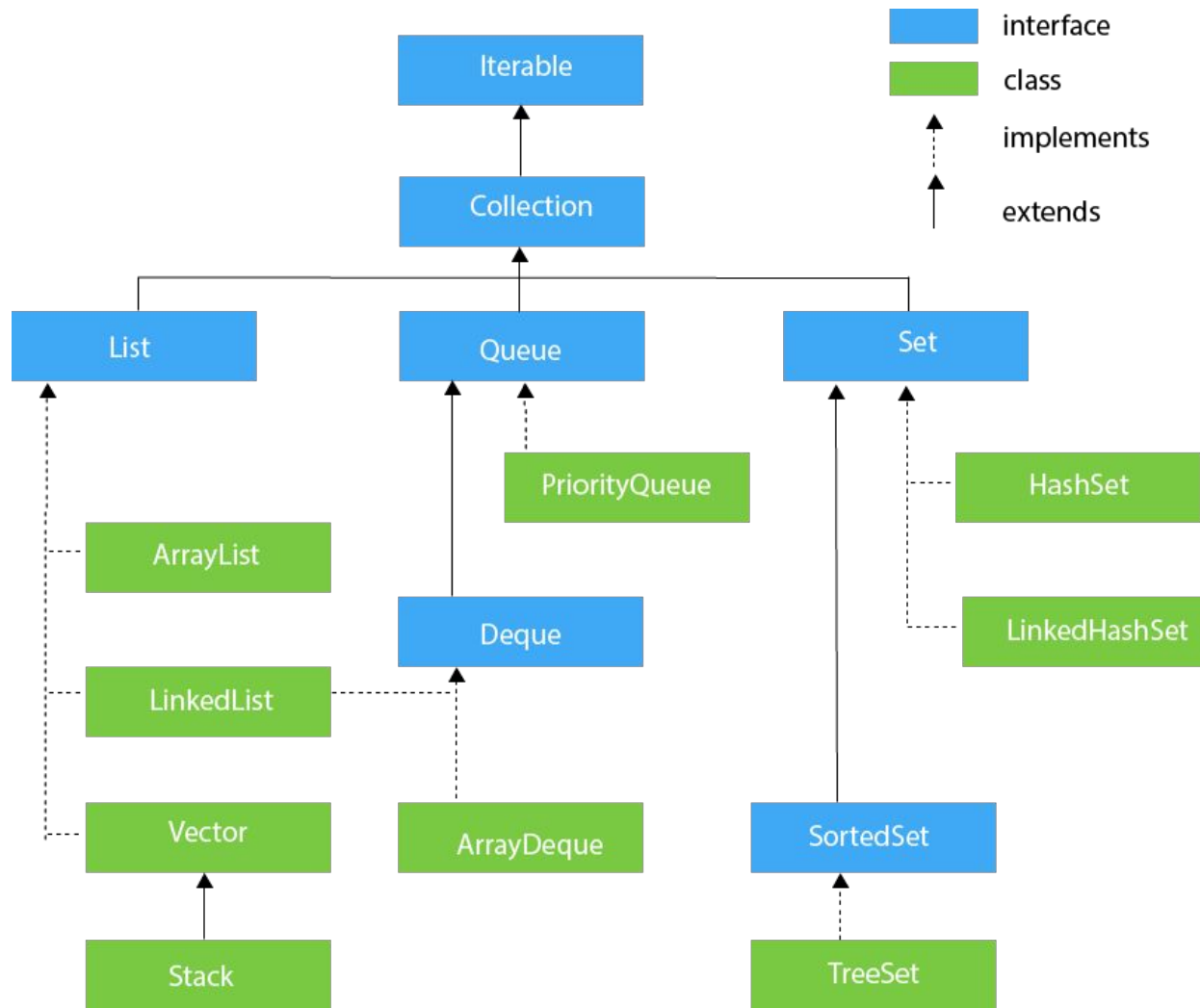
Collection là một framework cung cấp một kiến trúc để lưu trữ và thao tác với nhóm các đối tượng

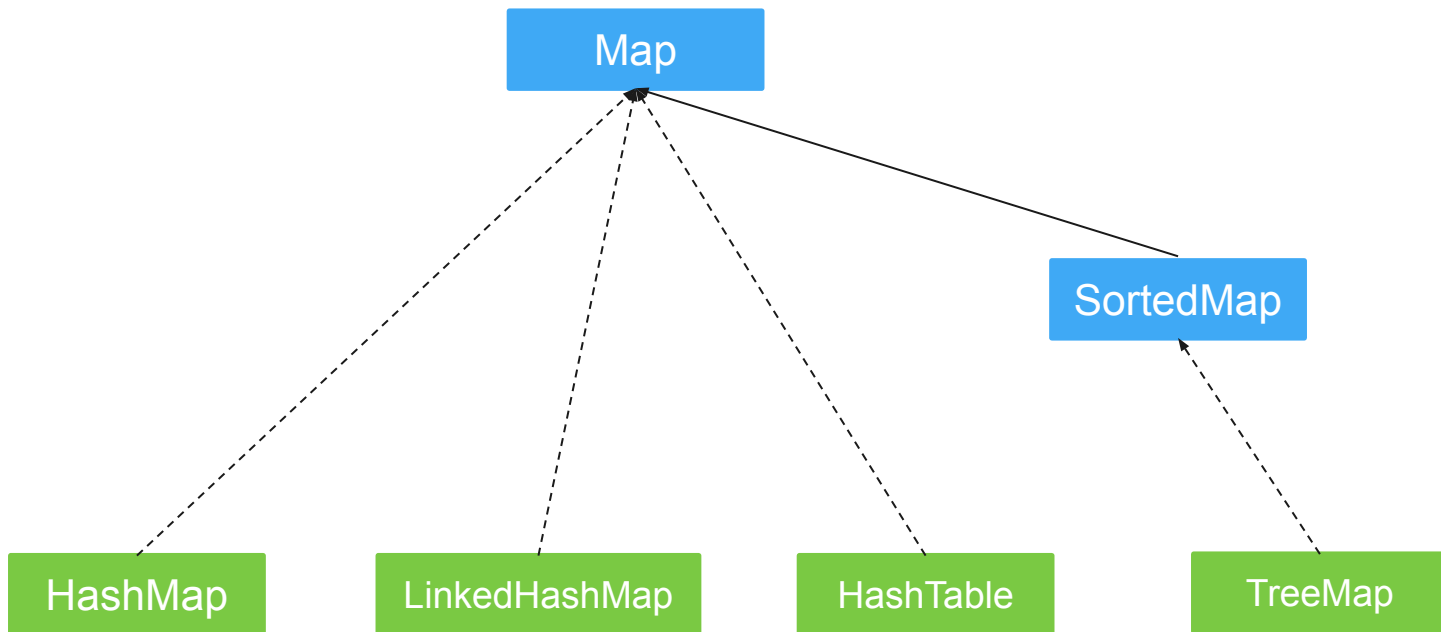
Java collections có thể đạt được tất cả các thao tác mà bạn thực hiện trên dữ liệu như tìm kiếm, sắp xếp, chèn, xóa

Collection là gì?

Java collection cung cấp nhiều interface (Set, List, Queue, Deque vv) và các lớp (ArrayList, Vector, LinkedList, PriorityQueue, HashSet, LinkedHashSet, TreeSet,...)



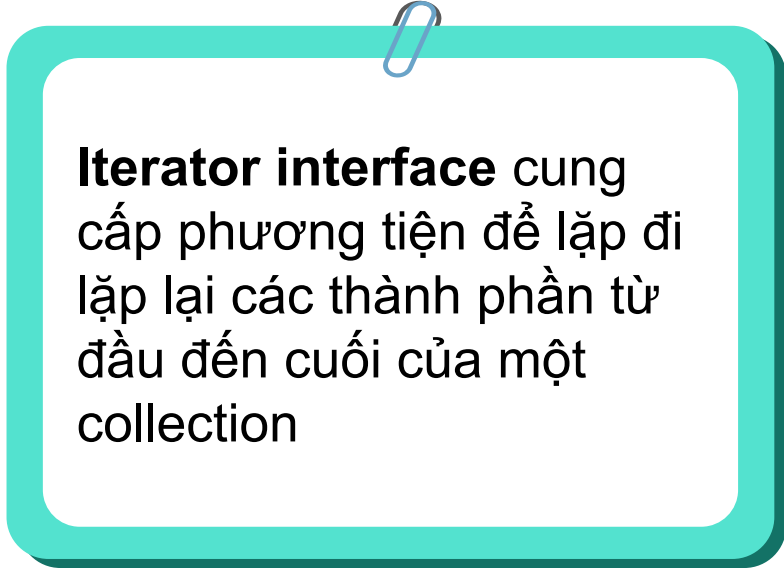




Iterable interface và iterator interface



Iterable interface chứa
dữ liệu thành viên iterator
interface



Iterator interface cung
cấp phương tiện để lặp đi
lặp lại các thành phần từ
đầu đến cuối của một
collection

Các phương thức của iterator interface



Phương thức	Mô tả
public boolean hasNext()	Trả về giá trị true nếu iterator còn phần tử kế tiếp đang duyệt
public object next()	Trả về phần tử hiện tại và di chuyển con trỏ tới phần tử tiếp theo
public void remove()	Loại bỏ phần tử cuối được trả về bởi iterator



Collection interface

Collection interface được thực hiện bởi tất cả các lớp trong Collection Framework. Nói cách khác, Collection interface là nền tảng mà Collection Framework phụ thuộc vào nó.

List Interface

List Interface là giao diện con của Collection Interface. Nó ngăn cách cấu trúc dữ liệu kiểu danh sách trong đó chúng ta có thể lưu trữ tập hợp các đối tượng có thứ tự. List Interface được thực hiện bởi các lớp ArrayList, LinkedList, Vector và Stack. Để khởi tạo List interface chúng ta sử dụng:

```
List <Kiểu dữ liệu> <Tên>= new ArrayList();  
List <Kiểu dữ liệu> <Tên> = new LinkedList();  
List <Kiểu dữ liệu> <Tên> = new Vector();  
List <Kiểu dữ liệu> <Tên> = new Stack();
```

Set Interface

Set là kiểu dữ liệu mà bên trong nó mỗi phần tử chỉ xuất hiện duy nhất một lần và Set interface cung cấp các phương thức để thao tác với set

Set interface được kế thừa từ Collection Interface nên nó được cung cấp đầy đủ các phương thức của Collection Interface





Set Interface

Một số class thực thi Set Interface thường gặp:

- ❑ **TreeSet**: là 1 class thực thi giao diện Set Interface, trong đó các phần tử trong set đã được sắp xếp.
- ❑ **HashSet**: là 1 class implement Set Interface, mà các phần tử được lưu trữ dưới dạng bảng băm (hash table).
- ❑ **EnumSet**: là 1 class dạng set như 2 class ở trên, tuy nhiên khác với 2 class trên là các phần tử trong set là các enum chứ không phải object.



Queue Interface

Queue(Hàng đợi) là kiểu dữ liệu nổi tiếng với kiểu vào ra FIFO, tuy nhiên với Queue Interface thì queue không chỉ còn dừng lại ở mức đơn giản như vậy mà nó cung cấp cho bạn các phương thức để xây dựng các queue phức tạp hơn nhiều như priority queue, deque. Queue Interface cũng kế thừa và mang đầy đủ các phương thức từ Collection Interface.

- ❑ **LinkedList**: chính là LinkedList mình đã nói ở phần List
- ❑ **PriorityQueue**: là 1 dạng queue mà trong đó các phần tử trong queue sẽ được sắp xếp.
- ❑ **ArrayDeque**: là 1 dạng deque (queue 2 chiều) được implement dựa trên mảng



Map Interface

Map (đồ thị/ánh xạ) là kiểu dữ liệu cho phép ta quản lý dữ liệu theo dạng cặp key-value, trong đó key là duy nhất và tương ứng với 1 key là một giá trị value.

Không giống như các interface ở trên, Map Interface không kế thừa từ Collection Interface mà đây là 1 interface độc lập với các phương thức của riêng mình.

Class về Map

TreeMap

EnumMap

Map

HashMap

WeakHashMap

