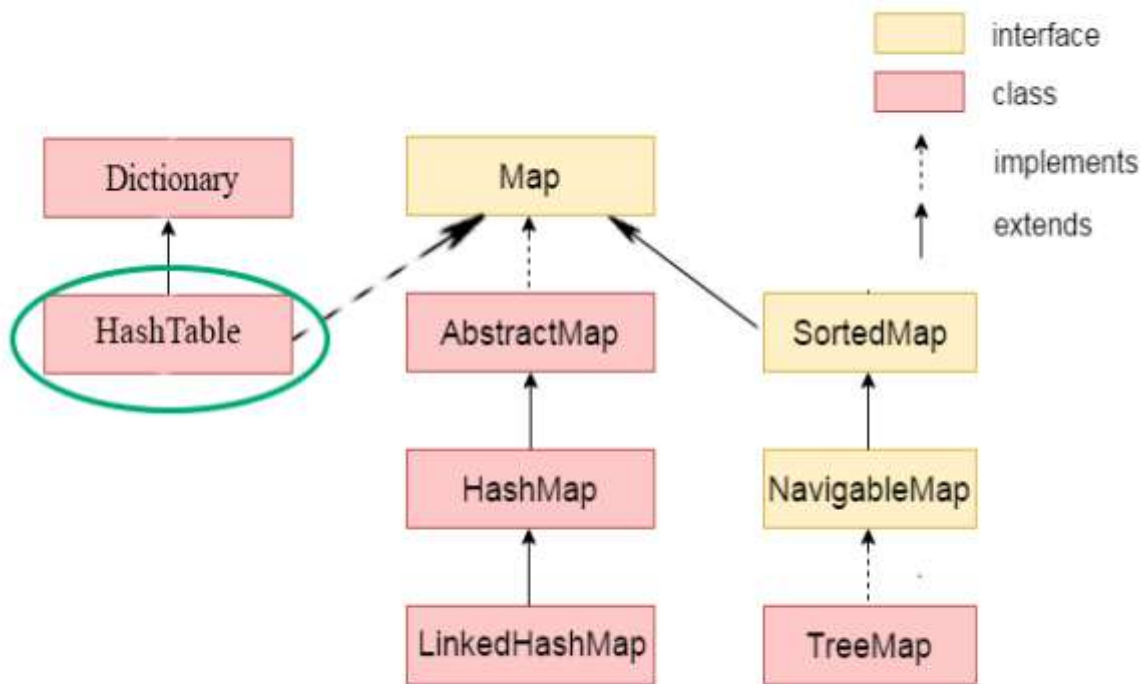


🕒 Cập nhật tháng 8 năm 2024

[Bài Đọc] Khái niệm và đặc điểm của Map



1. Khái niệm về Map

- Trong Java, **Map** là một interface thuộc gói `java.util`, được sử dụng để lưu trữ các phần tử dưới dạng **key-value pairs** (cặp khóa - giá trị). Mỗi **key** là duy nhất và ánh xạ đến một **value** duy nhất. Map rất hữu ích trong các tình huống cần ánh xạ một khóa cụ thể đến một giá trị, chẳng hạn như lưu trữ danh bạ điện thoại (tên - số điện thoại), từ điển (từ - nghĩa), v.v.
- Một số triển khai phổ biến của Map trong Java:
 - HashMap**: Lưu trữ cặp key-value mà không đảm bảo thứ tự
 - LinkedHashMap**: Duy trì thứ tự các phần tử theo thứ tự chèn vào
 - TreeMap**: Duy trì thứ tự tự nhiên của khóa hoặc thứ tự do Comparator cung cấp

2. Đặc điểm của Map

- Cặp Key-Value:
 - **Key:** Phải là duy nhất, không được trùng lặp
 - **Value:** Có thể trùng lặp
 - Ví dụ: Một HashMap có thể lưu trữ {"Alice": 25, "Bob": 30, "Alice": 35}. Trong trường hợp này, giá trị của khóa Alice sẽ bị ghi đè thành 35
- Không phải là một Collection:
 - Mặc dù Map nằm trong gói java.util, nó không kế thừa từ interface Collection. Do đó, Map không cung cấp các phương thức như add(), remove(), hay iterator()
- Hỗ trợ các thao tác cơ bản:
 - **Thêm:** Thêm cặp key-value bằng phương thức put(key, value)
 - **Lấy:** Lấy giá trị dựa trên key bằng phương thức get(key)
 - **Xóa:** Xóa phần tử bằng phương thức remove(key)
 - **Kiểm tra tồn tại:** Dùng các phương thức containsKey(key) và containsValue(value)
- Triển khai khác nhau cho các mục đích khác nhau:
 - **HashMap:** Nhanh ($O(1)$ trung bình) nhưng không đảm bảo thứ tự
 - **LinkedHashMap:** Nhanh và duy trì thứ tự chèn
 - **TreeMap:** Chậm hơn ($O(\log n)$) nhưng duy trì thứ tự tự nhiên hoặc tùy chỉnh
- Không cho phép key là null trong TreeMap:
 - **HashMap** và **LinkedHashMap** cho phép một key là null
 - **TreeMap** không cho phép key là null, vì nó cần sắp xếp các khóa
- Không đồng bộ hóa:
 - Các lớp như HashMap và LinkedHashMap không đồng bộ hóa. Nếu cần sử dụng trong môi trường đa luồng, bạn có thể sử dụng Collections.synchronizedMap() hoặc ConcurrentHashMap

Link tài nguyên đọc thêm: <https://www.geeksforgeeks.org/map-interface-java-examples/>