

 Cập nhật tháng 8 năm 2024

[Bài Đọc] HashMap, LinkedHashMap và TreeMap

Trong Java, **Map** là một interface trong gói `java.util`, và có một số lớp triển khai chính. Các lớp này có đặc điểm và cách sử dụng khác nhau tùy thuộc vào yêu cầu về hiệu năng, thứ tự và tính đồng bộ trong ứng dụng.

Các lớp triển khai phổ biến của Map trong Java bao gồm:

- HashMap
- LinkedHashMap
- TreeMap

1. HashMap

- **Đặc điểm:**
 - **Không duy trì thứ tự:** Các phần tử trong HashMap không được lưu trữ theo thứ tự nào, tức là không đảm bảo thứ tự chèn
 - **Khóa duy nhất:** Mỗi khóa trong HashMap là duy nhất. Tuy nhiên, nhiều giá trị có thể trùng lặp
 - **Khả năng hiệu suất:** HashMap sử dụng bảng băm (hash table), do đó có hiệu suất tìm kiếm, chèn và xóa khá nhanh
 - **Cho phép null:** HashMap cho phép một khóa null và nhiều giá trị null
- **Khi nào nên sử dụng:** HashMap phù hợp khi bạn không cần thứ tự lưu trữ các phần tử và hiệu suất là yếu tố quan trọng

2. LinkedHashMap

- **Đặc điểm:**
 - **Duy trì thứ tự chèn:** Các phần tử trong LinkedHashMap được lưu trữ theo thứ tự chèn vào, tức là phần tử đầu tiên sẽ được truy xuất đầu tiên
 - **Hiệu suất gần tương đương HashMap:** Vì sử dụng bảng băm giống HashMap, nhưng với một danh sách liên kết để duy trì thứ tự chèn, hiệu suất vẫn tốt ($O(1)$ đối với chèn và truy xuất)
 - **Cho phép null:** LinkedHashMap cũng cho phép một khóa null và nhiều giá trị null
- **Khi nào nên sử dụng:** Nếu bạn cần duy trì thứ tự của các phần tử khi chúng được thêm vào và hiệu suất là yếu tố quan trọng, LinkedHashMap là lựa chọn hợp lý

3. TreeMap

- **Đặc điểm:**

- **Duy trì thứ tự tự nhiên:** TreeMap lưu trữ các phần tử theo thứ tự tự nhiên của các khóa (đối với các đối tượng có cài đặt Comparable), hoặc theo một Comparator tùy chỉnh
- **Hiệu suất chậm hơn:** Vì sử dụng cấu trúc cây (cây đỏ-đen), thời gian tìm kiếm, chèn và xóa của TreeMap là $O(\log n)$, chậm hơn so với HashMap
- **Không cho phép null:** TreeMap không cho phép các khóa là null, vì nó cần phải so sánh khóa để duy trì thứ tự
- **Khi nào nên sử dụng:** Sử dụng TreeMap khi bạn cần duy trì thứ tự của các khóa theo thứ tự tự nhiên hoặc theo một thứ tự cụ thể mà bạn định nghĩa, và thứ tự quan trọng hơn hiệu suất.

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

Danh sách các bài học

