

1. В чем разница Queue и Stack?

Queue (очередь) - это коллекция, в которой элементы добавляются в конец и извлекаются из начала. Элемент, который добавляется первым, извлекается первым (принцип FIFO - First-In-First-Out).

Stack (стек) - это коллекция, где элементы добавляются и извлекаются только с одного конца (вершины стека). Последний добавленный элемент извлекается первым (принцип LIFO - Last-In-First-Out).

2. Чем можно заменить Stack?

Вместо использования Stack в Java можно использовать класс ArrayDeque, который предоставляет аналогичный функционал и, в дополнение к этому, более гибкий и быстрый.

3. Какие реализации Dequeue есть?

ArrayDeque: Реализация на основе динамического массива.

LinkedList: Реализация на основе связанного списка.

4. Что находится под капотом ArrayDeque? как решается проблема с добавлением в начало массива?

ArrayDeque реализован как динамический массив, но с двумя указателями на начало и конец.

Проблема с добавлением в начало массива решается за счет регулярного изменения размера массива и перемещения элементов при необходимости. Когда начинает заканчиваться место в начале или в конце массива, происходит пересоздание массива с увеличенным размером, и элементы переносятся на новое место.

5. Что используется для выставления приоритета в PriorityQueue?

PriorityQueue в Java использует либо естественный порядок элементов (например, для чисел сравнивает их значения), либо пользовательский компаратор (реализующий интерфейс Comparator) для определения приоритета элементов.

6. Что такое маркерные интерфейсы? какие знаете?

Маркерные интерфейсы в Java - это интерфейсы, которые не содержат объявлений методов. Они служат для пометки или обозначения класса с определенной специфической характеристикой или для активации некоторого функционала во время выполнения.

Примеры маркерных интерфейсов в Java включают Serializable, Cloneable, RandomAccess и другие. Например, класс, реализующий Serializable, обозначает, что его объекты могут быть сериализованы и десериализованы.