

Java Professional module #2 JCF

lecture #7. Data structures Queue, Stack. ArrayDeque

Mentor:

lecture #7. Data structures Queue, Stack. ArrayDeque

- Stack in Java
- Queue in Java
- Custom ArrayDeque implementation

Stack Class in Java

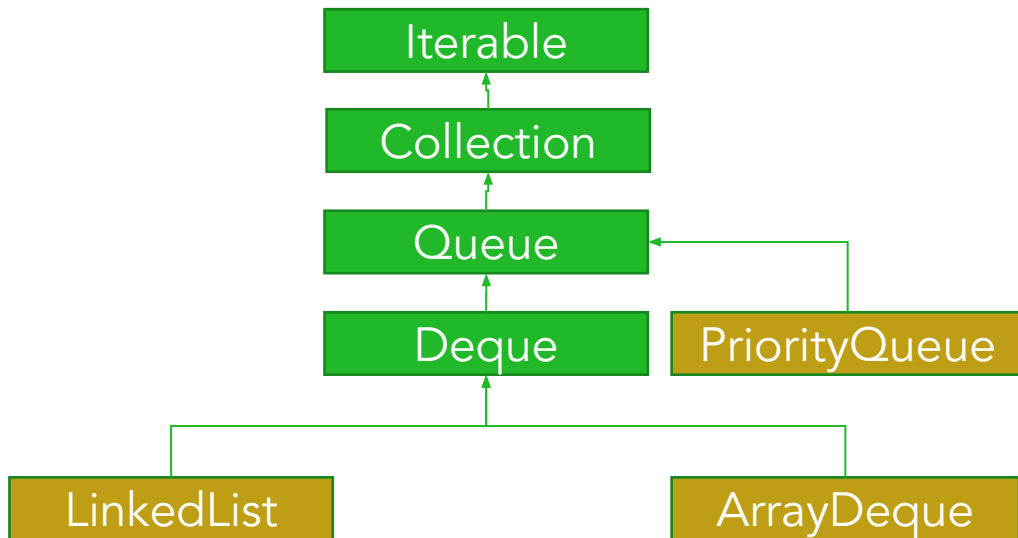
- Платформа Java Collection предоставляет класс Stack
- Класс основан на принципе «последний пришел – первый ушел». **LIFO**
- В дополнение к основным операциям **push** и **pop** класс предоставляет еще три функции: **empty**, **search** и **peek**.
- Класс поддерживает один конструктор по умолчанию Stack()

Implemented Interfaces

- **Serializable**: интерфейс маркер который, должны реализовать классы, если они должны быть сериализованы и десериализованы
- **Cloneable**: интерфейс который должен быть реализован классом, чтобы его объекты можно было клонировать.
- **Iterable**: интерфейс представляет собой набор объектов, которые являются итерируемыми, т. е. которые могут быть итерированы.
- **Collection**: Коллекция представляет собой группу объектов, известных как ее элементы.
- **List**: интерфейс List предоставляет способ хранения упорядоченной коллекции.
- **RandomAccess**: интерфейс-маркер, используемый реализациями List, чтобы указать, что они поддерживают быстрый произвольный доступ.

Queue Interface In Java

- Интерфейс Queue присутствует в пакете java.util и расширяет интерфейс Collection
- Используется для хранения элементов, которые должны быть обработаны, в порядке FIFO (первым пришел — первым ушел).
- Упорядоченный список объектов, использование которого ограничено вставкой элементов в конец списка и удалением элементов из начала списка FIFO



1. очередь нуждается в классе для объявления
2. наиболее распространенными классами являются PriorityQueue и LinkedList в Java.
3. ни одна из этих реализаций не является потокобезопасной.
4. PriorityQueue — одна из альтернативных реализаций, для многопоточной среды

ArrayDeque In Java

ArrayDeque в Java это массива с изменяемым размером в дополнение к реализации интерфейса Deque

важные особенности ArrayDeque:

- Блоки массива не имеют ограничений по емкости и увеличиваются по мере необходимости.
- Не потокобезопасен, ArrayDeque не поддерживает одновременный доступ нескольких потоков.
- Null элементы запрещены в ArrayDeque.
- Реализует два интерфейса
 - Queue
 - Deque

Custom ArrayDeque

```
public interface MyDeque<Integer> {  
  
    void addToHead(Integer element);  
    void addToTail(Integer element);  
    Integer removeHead();  
    Integer removeTail();  
    Integer peekHead();  
    Integer peekTail();  
    boolean isEmpty();  
    Integer getByIndex(int index);  
    boolean contains(Integer element);  
  
    /*  
    void grow();  
    int calculateNewCapacity(int currentCapacity);  
    int calculateAdditionalCapacity(int currentCapacity);  
  
    additional JUnit tests for this methods  
    */  
}
```

Разница между Deque и Stack

Стек — это структура данных, работающая по принципу LIFO

Deque - структура данных, в которую элементы можно вставлять и удалять с двух сторон

Stack— это класс

Deque - является интерфейсом

Stack - все методы отмечены synchronized ключевым словом

Deque – имеет не потокобезопасные реализации ArrayDeque и LinkedList