

Interview Questions

1. What is a Stack in Java?

A Stack is a linear data structure that follows the Last-In-First-Out (LIFO) principle, where the last element added is the first one to be removed. It provides two main operations: push to add elements onto the stack and pop to remove elements from the top of the stack.

2. What is the role of the peek operation in a Stack?

The peek operation in a Stack returns the top element of the stack without removing it. It allows you to examine the element at the top without modifying the stack's state.

3. Can a Set contain a null element?

Yes, a Set can contain at most one null element, since duplicate elements are not allowed.

4. How do you implement a Stack using an array? What are the advantages and limitations of this approach?

A Stack can be implemented using an array by keeping track of the top element's index. While array-based implementation offers constant-time access, it may have a fixed size and lead to overflow.

5. Describe the "push" and "pop" operations in a Stack and their significance.

The "push" operation adds an element to the top of the Stack, while the "pop" operation removes and returns the top element. These operations are essential for maintaining the LIFO property.

6. How does the "peek" operation differ from the "pop" operation in a Stack?

The "peek" operation retrieves the top element without removing it, allowing inspection without modifying the Stack's state.

7. Give an example of a real-world scenario where a Stack data structure is used.

The Undo/Redo functionality in software applications uses a Stack to store the history of operations.

8. Explain the concept of a Queue data structure and its core principle.

A Queue is a linear data structure following the First-In-First-Out (FIFO) principle, where the first element added is the first one to be removed.

9. How can you implement a Queue using a linked list? What advantages does this approach have?

A Queue can be implemented using a linked list where enqueueing involves adding to the end and dequeuing involves removing from the front. Linked list-based implementation allows dynamic resizing.

10: What is a priority Queue, and how is it different from a regular Queue?

A priority Queue stores elements based on priority, allowing the highest-priority elements to be removed first. A regular Queue follows the FIFO principle.

11 : Explain the differences between blocking and non-blocking Queues.

Blocking Queues can block producers when full or consumers when empty, while non-blocking Queues don't block and often use concurrency mechanisms.

12: Can you provide a real-world scenario where a Queue data structure is used?

Print spooling in operating systems uses a Queue to manage print jobs in the order they are received.

13. What is the key property that distinguishes a Set from other collection types?

A Set does not allow duplicate elements; each element is unique.

14.How can you iterate through the elements of a Set?

You can use the enhanced for loop or an iterator to iterate through the elements of a Set.

Example using an iterator:

```
Set<String> set = new HashSet<>();
Iterator<String> iterator = set.iterator();
while (iterator.hasNext()) {
    String element = iterator.next();
    // Process each element
}
```

15: Describe a scenario where using a Set can be more beneficial than using a List.

When you need to ensure that a collection doesn't contain duplicate elements, like storing unique usernames.