STACK

Stack ADT:

Stack is linear data-structure which follows a particular order in which the operations are performed. The order may be LIFO(Last in First Out) or FILO(First In Last Out).

Main operations:

**Push:** Adds an item in the stack. If the stack is full, then it is said to be an overflow condition.

**Pop:** Removes an item from the stack. The items are popped in the reversed order in which they are pushed. If the stack is empty, then it is said to be an underflow condition.

**Peek or Top:** Returns top element of stack

**isEmpty:** Returns true if stack is empty, else false.

**Practical Example:** Plates stacked over one another in canteen.

**Time Complexibility:** push(), pop(), isEmpty(), peek() all take O(1). And we do not run any loop for these operation.

**Application of Stack:**

1. Balancing of symbols

2. Infix to Postfix conversion

3. Redo-undo features at many places like editors

4. Forward and backward feature in web browsers.

5. Used in many algorithms like Tower of Hanoi, tree traversal, stock span problem, histogram problem.

6. Other application can be Backtracking, Knight tour problem, rat in a maze, N queen problem and sudoku solver.

**Implementation:**

**1.** Using array

**2.** Using Linked List

**Implementing Stack Using Arrays**

**Refer Stack.java for more details**