

Stack and Queue are two common data structures used to store and manipulate collections of elements.

A stack is a data structure that follows the Last-In-First-Out (LIFO) principle, which means that the element added last will be the first one to be removed. The operations of adding an element to the stack are called push and removing an element from the stack is called pop. The push operation adds an element to the top of the stack, and the pop operation removes the top element from the stack. You can think of a stack as a deck of cards, where you add a card on the top of the 'stack' and remove a card from the top of the 'stack'.

Stack methods:

pop() - removes an item from the stack

push() - adds an item to the stack

peek() - looks into the stack, retrieving first item

A queue is a data structure that follows the First-In-First-Out (FIFO) principle, which means that the element added first will be the first one to be removed. The operations of adding an element to the queue are called enqueue and removing an element from the queue is called dequeue. The enqueue operation adds an element to the end of the queue, and the dequeue operation removes an element from the front of the queue. You can think of a queue as a line of people waiting **for** a service, where the first person in the line is the first one to be served.

Queue methods:

dequeue() - removes an item from the queue

enqueue() - adds an item to the queue

peek() - looks into queue, retrieving first item