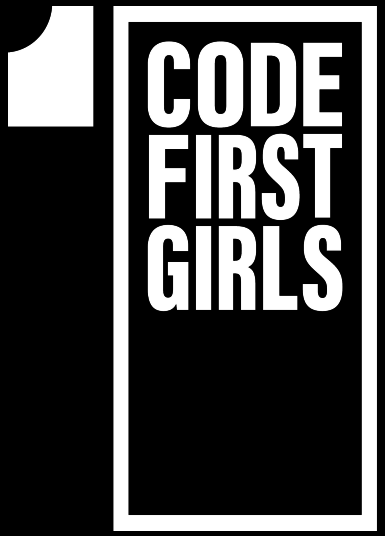
STACKS & QUEUES

LESSON 9



NANODEGREE → ENGINEERING MODULE

AGENDA

01 Data structures: Stack & Queue

02 Queue

03 Stack

04 Practice and coding

STACK AND QUEUE

DATA STRUCTURES

- **Stack**– it is a collection of objects that are inserted and removed according to the last-in, first-out LIFO principle.

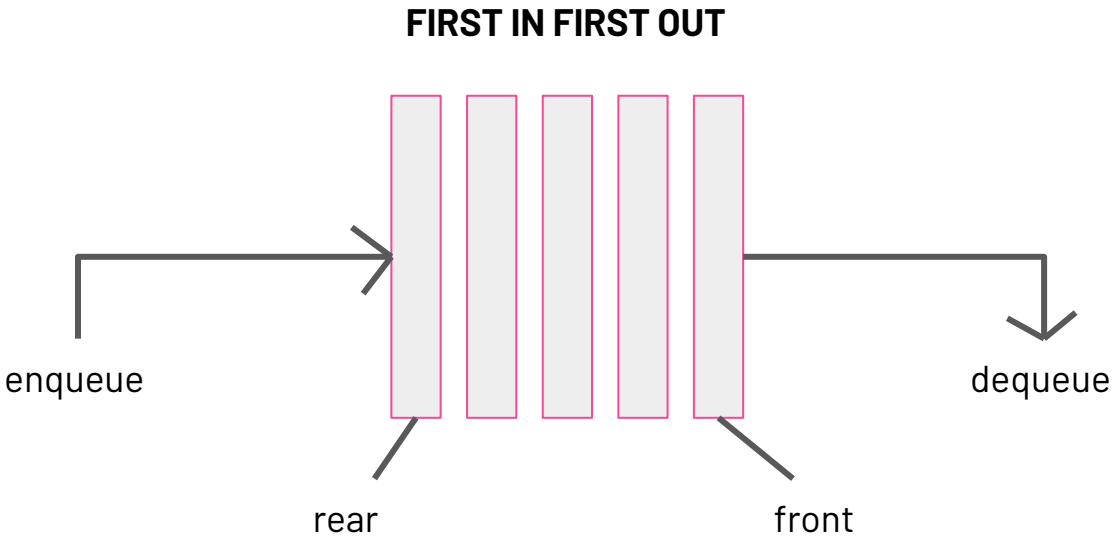
- **Queue**– it is a collection of objects that are inserted and removed based on first-in, first-out FIFO principle.

STACK & QUEUE

Let’s watch a DEMO to understand the concepts of Stack and Queues

- STACK⟶ LIFO
- QUEUE ⟶ FIFO

QUEUE ANATOMY



Insertion and deletion
happen on different ends

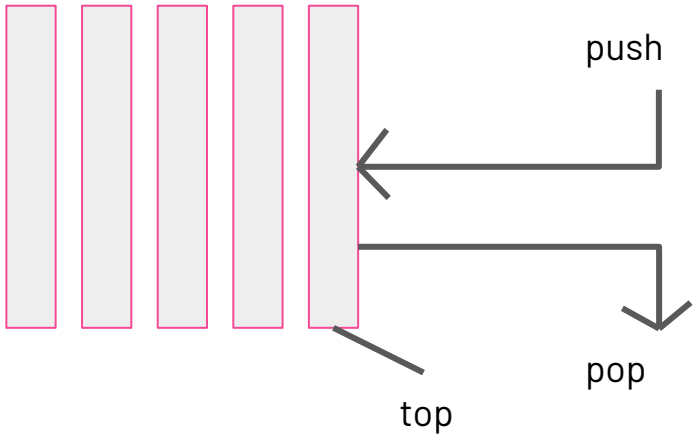
- **Enqueue** - adds an item to the queue. If the queue is full, then it is said to be an Overflow condition
- **Dequeue** - removes an item from the queue. The items are popped in the same order in which they are pushed. If the queue is empty, then it is said to be an Underflow condition
- **Front** - get the front item from queue
- **Rear** - get the last item from queue

QUEUE IMPLEMENTATION

- list
- collections.deque
- queue.Queue

STACK ANATOMY

LAST IN FIRST OUT

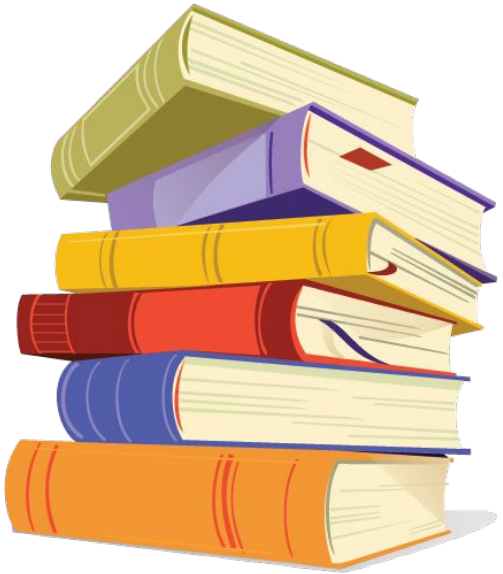


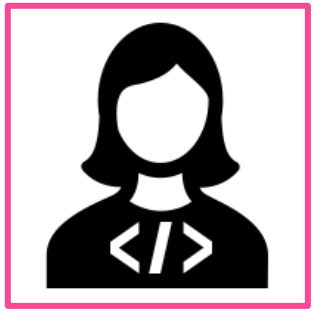
Insertion and deletion happen on the same end

- **empty()** – returns whether the stack is empty
- **size()** – returns the size of the stack
- **top()** – returns a reference to the top most element of the stack
- **push(x)** – Adds the element 'x' at the top of the stack
- **pop()** – Deletes the top most element of the stack

STACK IMPLEMENTATION

- list
- collections.deque
- queue.LifoQueue





DEMO &
EXERCISES

QUEUE & STACK IMPLEMENTATION EXERCISES & PRACTICE

THANK YOU!