

Implementation of Queue

In this lecture we will build on our previous understanding of Queues by implementing our own class of Queue!

Queue Methods and Attributes

Before we begin implementing our own queue, let's review the attribute and methods it will have:

- Queue() creates a new queue that is empty. It needs no parameters and returns an empty queue.
 - enqueue(item) adds a new item to the rear of the queue. It needs the item and returns nothing.
 - dequeue() removes the front item from the queue. It needs no parameters and returns the item. The queue is modified.
 - isEmpty() tests to see whether the queue is empty. It needs no parameters and returns a boolean value.
 - size() returns the number of items in the queue. It needs no parameters and returns an integer.
-

Queue Implementation

```
In [1]: class Queue:
        def __init__(self):
            self.items = []

        def isEmpty(self):
            return self.items == []

        def enqueue(self, item):
            self.items.insert(0,item)

        def dequeue(self):
            return self.items.pop()

        def size(self):
            return len(self.items)
```

```
In [2]: q = Queue()
```

```
In [3]: q.size()
```

```
Out[3]: 0
```

```
In [4]: q.isEmpty()
```

```
Out[4]: True
```

```
In [5]: q.enqueue(1)
```

```
In [6]: q.dequeue()
```

```
Out[6]: 1
```

Good Job!