

Assignment 2. Stack, Queue and Linked List

- **Stack, Queue, Linked List의 Class 만들기**

- **꼭 읽어 보세요 !**
 - Chapter 12~14의 'Class' 관련 내용
 - Chapter 17의 'Linked List' 관련 내용
 - Chapter 18, 19의 Stack과 Queue 관련 내용

Node

- **Basic Element of Lists**

- 앞으로 만들 **Stack**, **Queue**, **Linked List**는 모두 Node를 연결해 만들게 됩니다.
- 교재 17.2 The Node class

```
❯ node1 = Node('First Node')
❯ print node1
First Node
❯ node2 = Node('Second Node')
❯ print node2
Second Node
❯ node3 = Node('Third Node')
❯ print node3
Third Node
```

```
class Node:
    def __init__(self, cargo = None):
        self.cargo = cargo
        self.next = None
    def __str__(self):
        return str(self.cargo)
```

▪ printList()

- Argument로 주어진 Node의 연결을 따라가며 차례로 print합니다.
- Stack, Queue, Linked List 의 print를 위한 `__str__()` 작성에 이용합니다.

```
node1.next = node2
node2.next = node3
printList(node1)
First Node
Second Node
Third Node
```

```
def printList(node):
    while node:
        print node
        node = node.next
```

▪ 실행해봅시다 !

- <http://repl.it/I1D>

Stack

- **Stack Class**를 만들어 봅시다.
 - List의 한 쪽에서만 삽입(**push**)과 삭제(**pop**)가 일어 납니다.
 - **Stack**은 예제가 주어집니다. 다음 페이지에 주어진 예제를 참고하세요.
 - 예제의 삽입(**push**)과 삭제(**pop**)과정을 잘 살펴보세요.
- **구현된 Method들**
 - `__init__()`
 - `__str__()`
 - `is_empty()`
 - **`push()`**
 - **`pop()`**

- 실행해봅시다!
 - <http://repl.it/I1F>

```
>  
> s = Stack()  
> s.is_empty()  
=> True  
>  
> s.push(1)  
> s.push('snu')  
> s.push([5, 3])  
> s.push(12)  
> print s  
12  
[5, 3]  
snu  
1  
  
> s.pop()  
=> 12  
> s.pop()  
=> [5, 3]  
> print s  
snu  
1
```

Queue

- **Queue Class**를 만들어 봅시다.
 - List의 한 쪽에서는 삽입(**insert**)만, 한 쪽에서는 삭제(**remove**)만 일어납니다.
 - **Stack**의 예제를 참고해 **insert()**와 **remove()**를 작성해보세요 !
 - 교재 19.2 Linked Queue

- **구현할 Method들**
 - **insert()**
 - **remove()**

- 작성해봅시다!
 - <http://repl.it/I1G>

```
> q = Queue()
> q.is_empty()
=> True
>
> q.insert(1)
> q.insert('snu')
> q.insert([2, 8])
> q.insert(6)
> print q
1 snu [2, 8] 6
>
> q.remove()
=> 1
> q.remove()
=> 'snu'
> print q
[2, 8] 6
>
> q.insert(17)
> q.insert(9)
> print q
[2, 8] 6 17 9
> █
```

Linked List

- **Linked List Class**를 만들어 봅시다.
 - List의 양쪽에서 삽입(**addFirst**, **append**)과 삭제(**removeFront**, **removeEnd**)가 일어납니다.
 - **Stack**의 예제와 앞서 구현한 **Queue**를 참고해 **addFirst()**, **append()**, **removeFront()**, **removeEnd()**를 작성해보세요.
 - 교재 17.9 The Linked List Class
- **구현할 Method들**
 - **addFirst()**
 - **append()**
 - **removeFront()**
 - **removeEnd()**

Linked List - Cont'd

- 작성해봅시다!
 - <http://repl.it/I1H>

```
>>> list = LinkedList()
>>> list.is_empty()
=> True
>>> list.append(1)
>>> list.append(2)
>>> list.append(3)
>>> print list
1 2 3
>>> list.addFirst(4)
>>> list.addFirst(5)
>>> list.addFirst(6)
>>> print list
6 5 4 1 2 3
>>> list.removeEnd()
=> 3
>>> list.removeEnd()
=> 2
>>> list.removeFront()
=> 6
>>> list.removeFront()
=> 5
>>> print list
4 1
>>> list.is_empty()
=> False
```

Assignment 2. Queue and Linked List

■ 과제 확인

■ 제출 기한

- 4월 9일 화요일 자정 전(23:59) 까지

■ 방법

- 조교 E-mail(jbbae@archi.snu.ac.kr)로 Queue와 Linked List의 전체 소스코드를 제출해주세요.
- 이름_Queue.txt, 이름_LinkedList.txt 두 개의 파일을 제출
- 과제1과 마찬가지로 제출 이후 수업시간에 교수님께 지목 받으시는 분은 앞에서 시연하시면 됩니다.

■ 문의

- 배종보, 301동 517호, 880-7296, jbbae@archi.snu.ac.kr
- 이용훈, 301동 517호, 880-7296, yonghunlee@archi.snu.ac.kr