

# Data Structure

알고샵시다 1회차  
자료구조 I 브리핑

# 자료구조 I

- 추상 자료형, ADT
- Stack
- Queue
- 예제 실습
- 질의응답



# 추상 자료형, ADT

“어떤 동작이 있는지”까지만 정의하고 구현해서 대해서는 다루지 않음.  
오직 순수한 기능만을 열거한다는 점에서, 사용 설명서라고 보면 된다.

사용 설명서에는 정지, 미풍, 약풍 등의 기능 설명과 사용 방법이 나와 있지만, 버튼을 눌렀을 때 내부 회로에서 무슨 일이 벌어지는지는 알 수 없다. 만약 무슨 일이 벌어지는지 **동작까지 구현했다면, 우리는 그것을 자료구조, Data Structure**라고 부른다.

# 추상 자료형, ADT

\* 예시: 무선 마우스의 추상 자료형은?

Data : Power, LeftButton, RightButton

Operation: PowerOn(), PowerOff(), leftButtonClick(), RightButtonClick()

# 추상 자료형, ADT

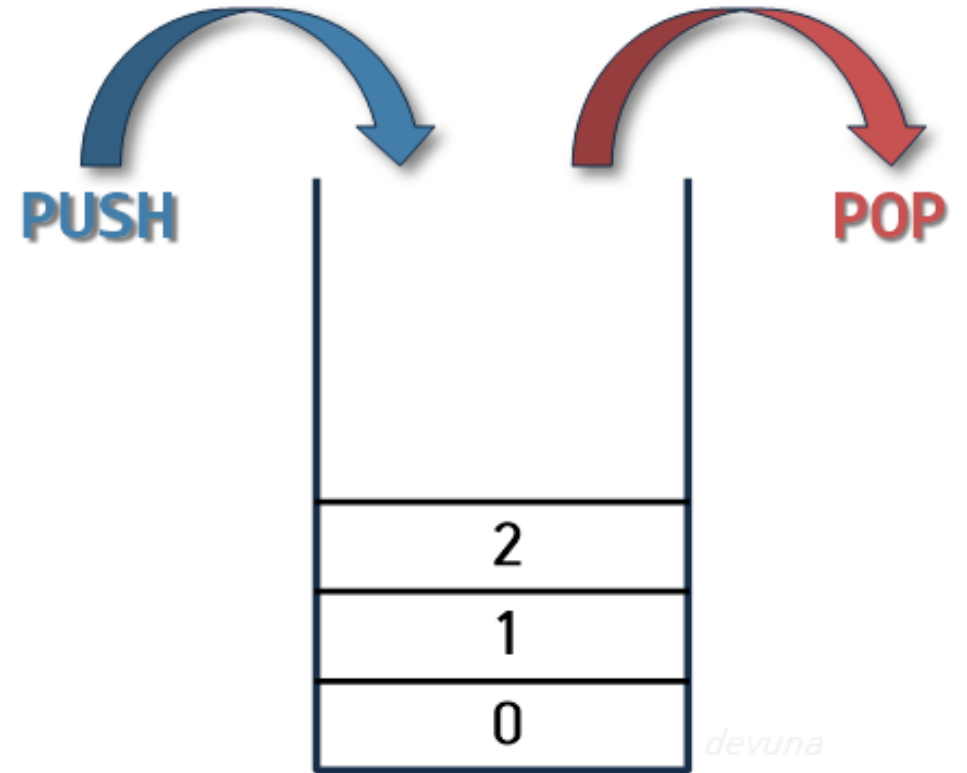
\* 왜 필요한가?

구현자와 사용자를 분리한다. 라이브러리를 가져다 쓰거나 내장 함수를 사용하는 것이 바로 추상자료형이 정의되어 있기 때문이다. 사용자는 내부 구현에 대해 알 필요가 없다. 단지 목적에 맞게 가져다 쓰면 될 뿐이다.

또한 추상 자료형의 구현은 외부로부터 숨겨져 정보 은닉이 이루어진다.

# Stack

Last In, First Out  
마지막에 들어간 데이터가  
나올 때 처음으로 나오는 Data Structure



LIFO (Last In First Out) 방식

# *Stack*

\* 언제 필요한가?

말 그대로, 수직적으로 쌓이는 Stack형 상황일 때

- 후위 표기법 계산
- 문자열 역행
- 실행 취소
- 웹 브라우저 뒤로가기 등

# Stack

## \* Stack의 사용법 (Java)

- Stack<T> stack = new Stack<>();
- push() : 스택에 삽입
- pop() : 스택에서 가장 위에 있는 값 반환하고 없앴
- peek() : 스택에서 가장 위에 있는 값 반환
- isEmpty() : 스택이 비어있는지를 반환
- size() : 스택에 있는 요소의 크기 반환

```
Stack<Integer> stack = new Stack<>();  
stack.push(1);  
stack.push(3);  
System.out.println(stack.isEmpty());  
System.out.println(stack.size());  
System.out.println(stack.pop());  
stack.pop();  
System.out.println(stack.isEmpty());
```

Console × Mirur Painter Mirur Statistics Debug

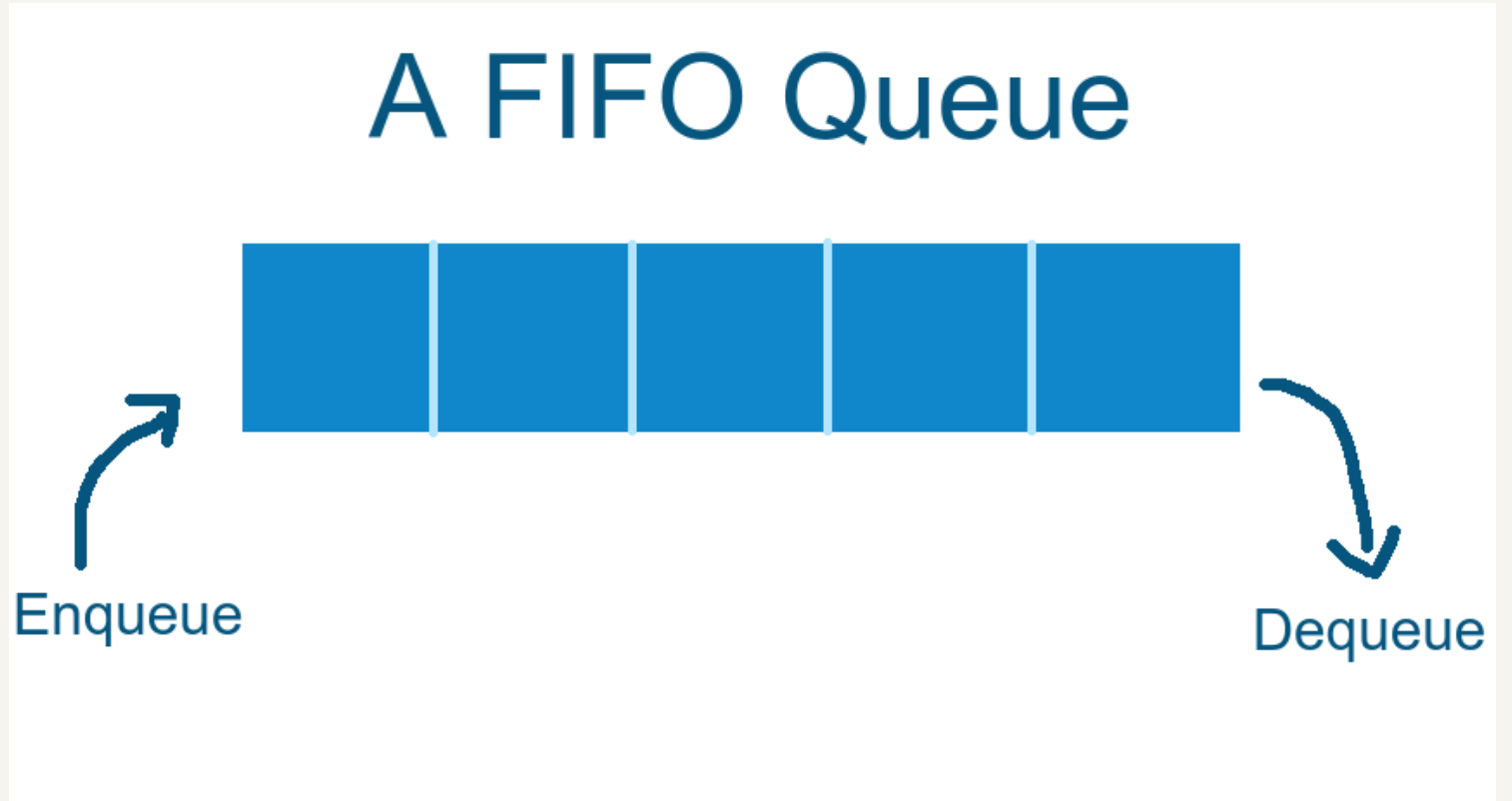
<terminated> Main (2) [Java Application] C:\weclipse-java-202

false  
2  
3  
true



# Queue

First In, First Out  
먼저 들어간 데이터가  
먼저 나오는 Data Structure



# *Queue*

\* 언제 필요한가?

먼저 요청된 작업이 먼저 처리되어야 하는 상황에서

- 은행 업무
- 대기열 순서
- 서비스 센터의 대기시간
- 프로세스 관리 등

# Queue

## \* Queue의 사용법 (Java)

- Queue<T> queue = new LinkedList<>>();
- add() : 큐에 삽입
- peek() : 가장 먼저 큐에 들어간 요소 반환
- remove() : 가장 먼저 큐에 들어간 요소 삭제하면서 반환
- isEmpty() : 큐가 비어있는지 반환
- size() : 큐에 있는 요소의 크기 반환

```
Queue<String> queue = new LinkedList<>>();
queue.add("우리 좀");
queue.add("알고 삽시다");
System.out.println(queue.remove());
System.out.println(queue.isEmpty());
queue.remove();
System.out.println(queue.size());
```

Console × Mirur Painter Mirur Statistics Debug Shell

<terminated> Main (2) [Java Application] C:\eclipse-java-2021-09-R

우리 좀  
false  
0

# 예제 실습

<https://www.acmicpc.net/problem/9012>

<괄호, 실버 4>

<https://www.acmicpc.net/problem/1158>

<요세푸스 문제, 실버 4>

*QnA*

