

자료 구조

01 선형 자료구조

02 비선형 자료구조

신 제 용

선형 자료구조

- 01 배열
- 02 배열 리스트
- 03 링크드 리스트
- 04 스택
- 05 큐
- 06 해시 셋
- 07 해시 테이블

신 제 용

배열 (Array)

01 배열의 개념과 활용

신 제 용

01 배열의 개념과 활용

가장 기초적인 자료구조인 배열에 대해 알아봅니다.

학습 키워드 - 배열, Array, 인덱싱

Chapter 01

배열의 개념과 활용

배열 (Array)

- 많은 수의 데이터를 다룰 때 사용하는 자료구조
- 각 데이터를 인덱스와 1:1 대응하도록 구성
- 데이터가 메모리 상에 연속적으로 저장됨

Chapter 01

배열의 개념과 활용

배열의 특징

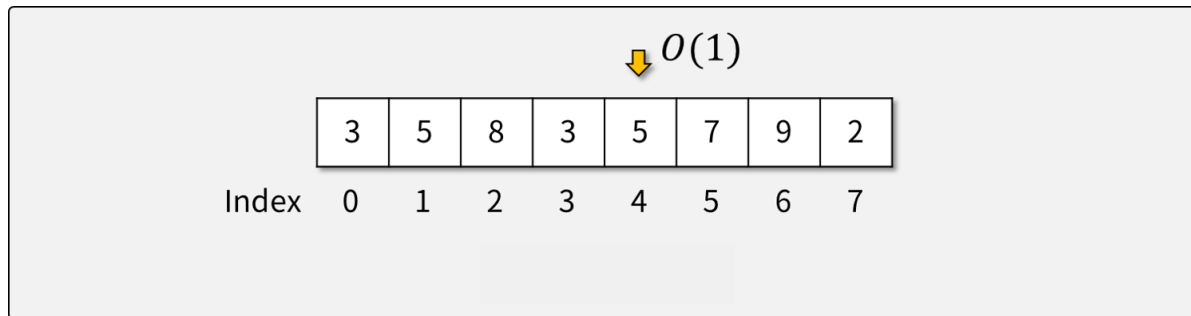
- 크기(Element의 개수)가 정해져 있다.
- 자료 구조에 기능(메소드)이 포함되어 있지 않다.
- 자료가 메모리상에 빈틈 없이 연속적으로 위치해 있다.
- 인덱스를 활용하여 자료에 빠르게 접근할 수 있다.

Chapter 01

배열의 개념과 활용

배열의 장점

- 인덱스를 이용하여 데이터에 빠르게 접근 가능



Chapter 01

배열의 개념과 활용

배열의 단점

- 배열의 길이는 생성 시 정해져, 변경할 수 없다.
→ 가변 길이 배열은 배열의 크기를 변경할 때 마다 새 배열을 만든다.
- Element를 제거할 경우, 배열에 빈 틈이 생긴다.
→ 기존 Element의 인덱스를 유지하기 위해 빈 틈을 유지한다.
→ 실제로는 Element의 삭제가 불가능하다.

Chapter 01

배열의 개념과 활용

Python과 배열 (1)

- Python의 리스트는 배열과는 다르며, 고수준의 기능을 포함한다.
- Python에서는 배열과 유사한 자료구조로 array가 제공
 - 리스트와 달리 특정 자료형 만을 허용
 - 배열과 같이 메모리상에 연속적으로 배치되는 것을 보장

```
import array  
arr = array.array('b', [10, 4, 22, -4, 9])
```

Chapter 01

배열의 개념과 활용



Python과 배열 (2)

- Python array의 자료형 타입 코드

타입 코드	C Type	Python Type	bytes
b	signed char	int	1
B	unsigned char	int	1
u	Py_UNICODE	Unicode Character	2
h	signed short	int	2
H	unsigned short	int	2
i	signed int	int	2
I	unsigned int	int	2
l	signed long	int	4
L	unsigned long	int	4
q	signed long long	int	8
Q	unsigned long long	int	8
f	float	float	4
d	double	float	8

Chapter 01

배열의 개념과 활용