

자료구조

Data Structure

자료구조

Data Structure

데이터를 효율적으로 저장하고 관리하기 위한 방법

데이터(Data)

경험이나 관찰로부터 얻어진
조직화된 정보의 모음

| 2022학년도 대학수학능력시험 시험 시간표 및 일정 | | |
|------------------------------|---------------------------------|--------------------------|
| 교시 | 시험 영역 | 시험 시간 (소요 시간) |
| 수험생 입실 완료 - 08 : 10까지 | | |
| 1 | 국어 | 08 : 40 ~ 10 : 00 (80분) |
| 휴식 - 10 : 00 ~ 10 : 20 (20분) | | |
| 2 | 수학 | 10 : 30 ~ 12 : 10 (100분) |
| 중식 - 12 : 10 ~ 13 : 00 (50분) | | |
| 3 | 영어 | 13 : 10 ~ 14 : 20 (70분) |
| 휴식 - 14 : 20 ~ 14 : 40 (20분) | | |
| 4 | 한국사 탐구(사회·과학·직업) | 14 : 50 ~ 16 : 37 (107분) |
| | 한국사 | 14 : 50 ~ 15 : 20 (30분) |
| | 한국사 영역 문·답지 회수 탐구 영역 문·답지 배부 | 15 : 20 ~ 15 : 35 (15분) |
| | 탐구(사회·과학·직업) 시험: 2과목 선택 | 15 : 35 ~ 16 : 05 (30분) |
| 5 | 시험 본 과목 문제지 회수 | 16 : 05 ~ 16 : 07 (2분) |
| | 탐구(사회·과학·직업) 시험: 1~2과목 선택 | 16 : 07 ~ 16 : 37 (30분) |
| 휴식 - 16 : 37 ~ 16 : 55 (18분) | | |
| 5 | 제2외국어/한문 | 17 : 05 ~ 17 : 45 (40분) |

데이터(Data)

경험이나 관찰로부터 얻어진
조직화된 정보의 모음



데이터 (Data)

프로그램 코드가 아닌 모든 것

프로그램

명령 또는 컴퓨터가 수행할 작업

"현대 컴퓨터의 아버지"



von Neumann (1903-1957)

“Program is a kind of data”

프로그램도 일종의 데이터다!

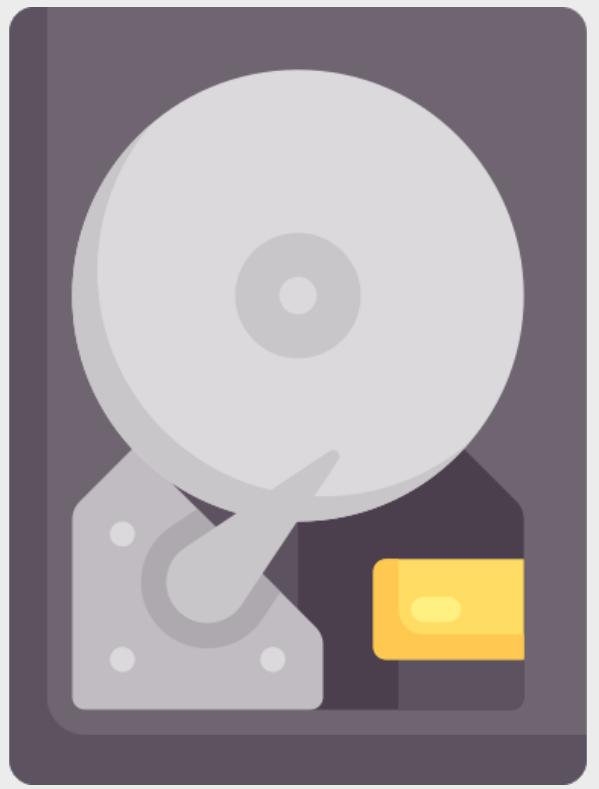


Stored-Programmed Computer

프로그램 내장식 컴퓨터 등장

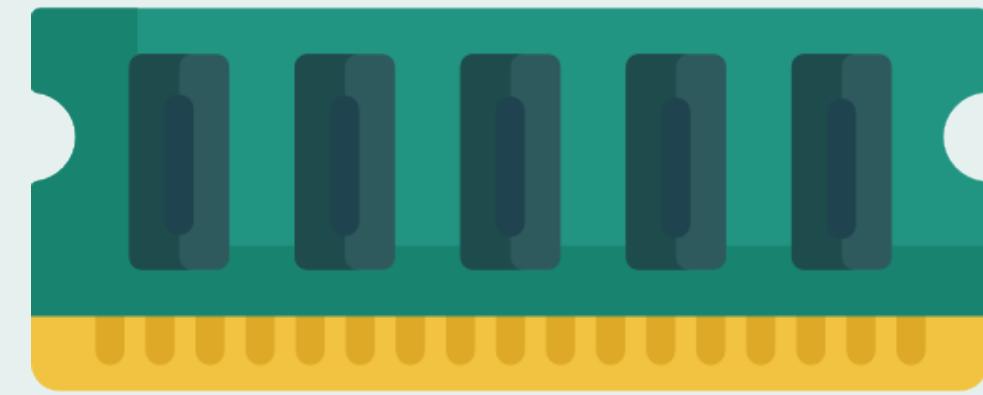
필요한 모든 것을 컴퓨터에 저장해서 사용하자!

데이터는 컴퓨터에 어떻게 저장될까?



스토리지

Storegy



메모리

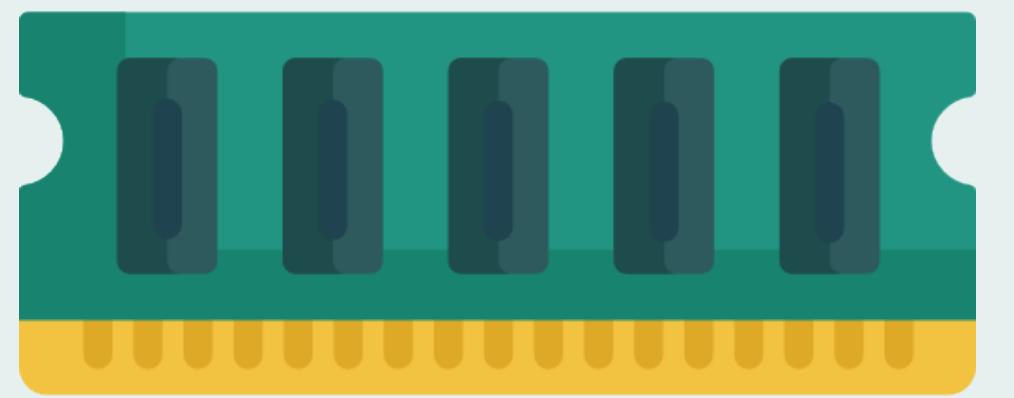
Memory



스토리지

Storegy

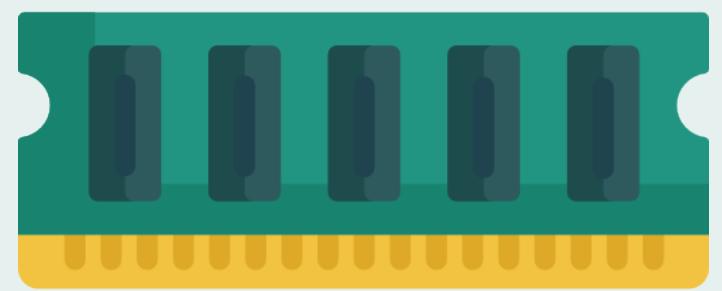
- 데이터가 영구히 저장되는 곳
- HDD, SSD, USB, CD ...
- 용량이 크지만 속도가 느림
- 당장 필요한 데이터가 아닌 데이터



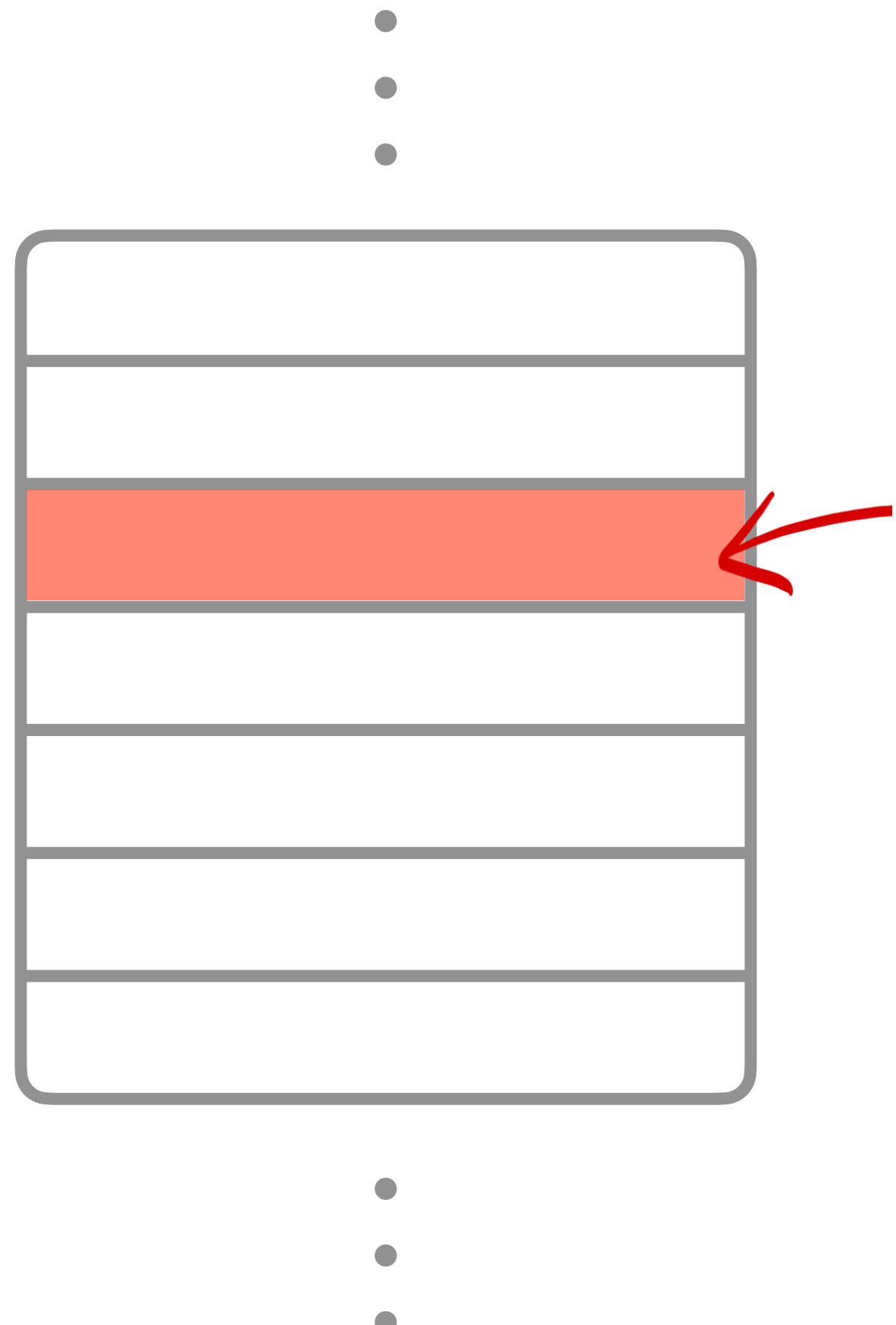
메모리

Memory

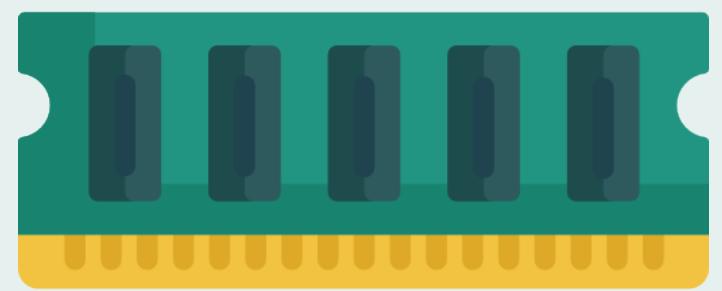
- 데이터가 **임시적으로** 저장되는 곳
- 용량이 작지만 **속도가 매우 빠름**
- 당장 필요한 데이터가 위치
- 메모리를 효율적으로 사용하자!



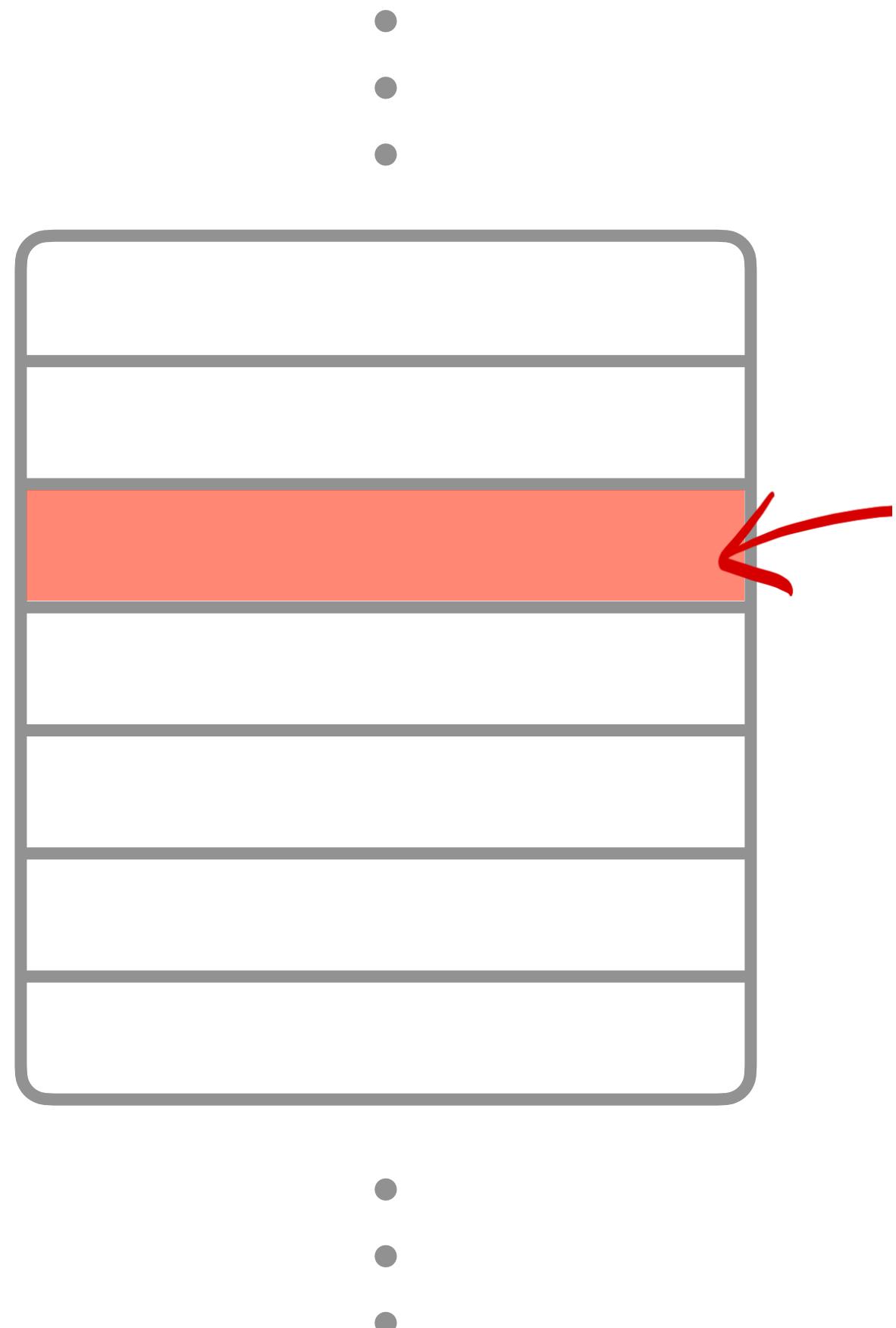
메모리 Memory



- 데이터가 저장 되는 위치 Random
- RAM(Random Access Memory)
- 각 칸마다 주소값이 지정되어 있음



메모리 Memory



- 데이터가 저장 되는 위치 Random
- RAM(Random Access Memory)
- 각 칸마다 주소값이 지정되어 있음



데이터 타입(Data Type)

데이터 타입(Data Type)

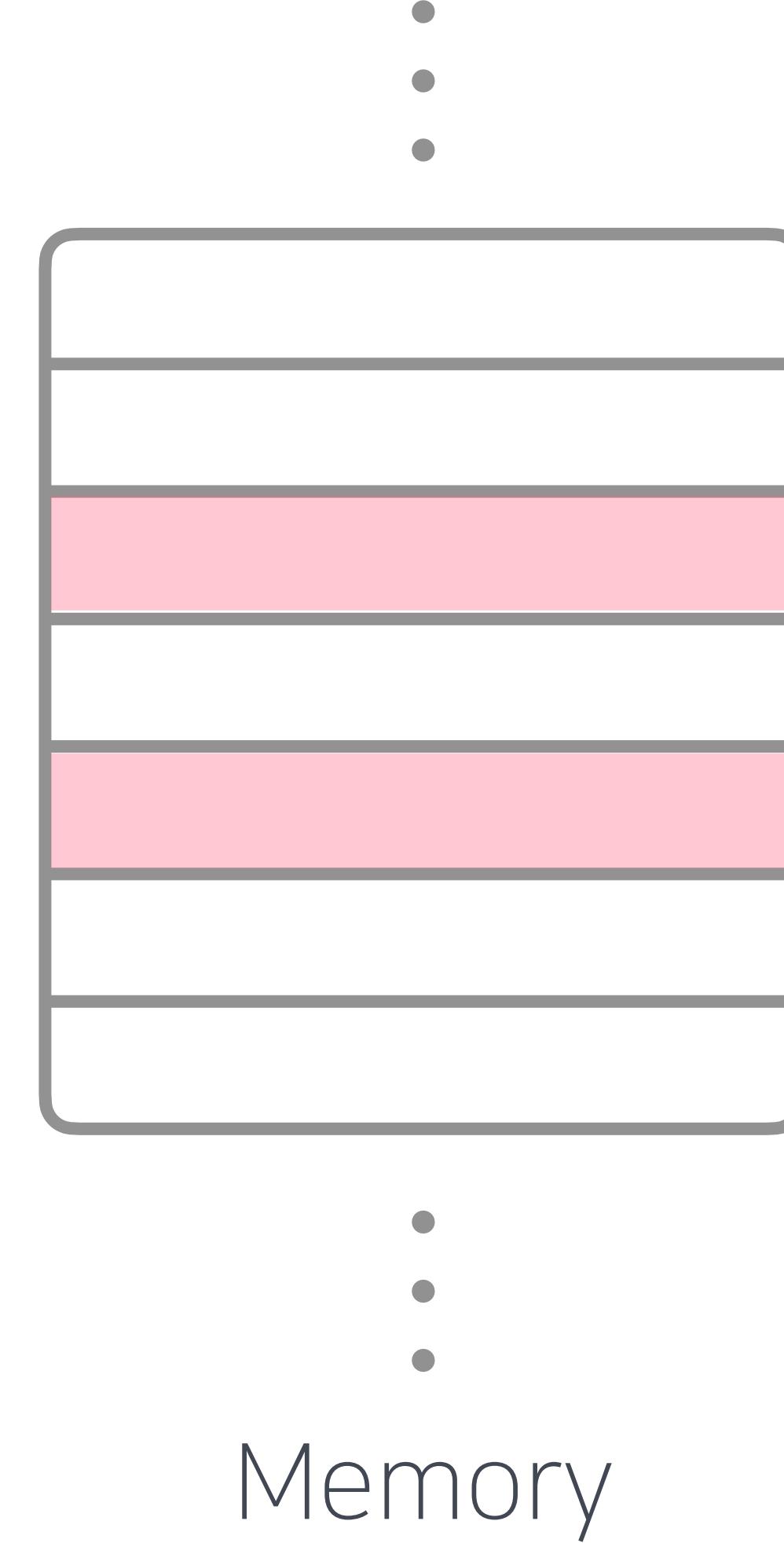
180

12.5

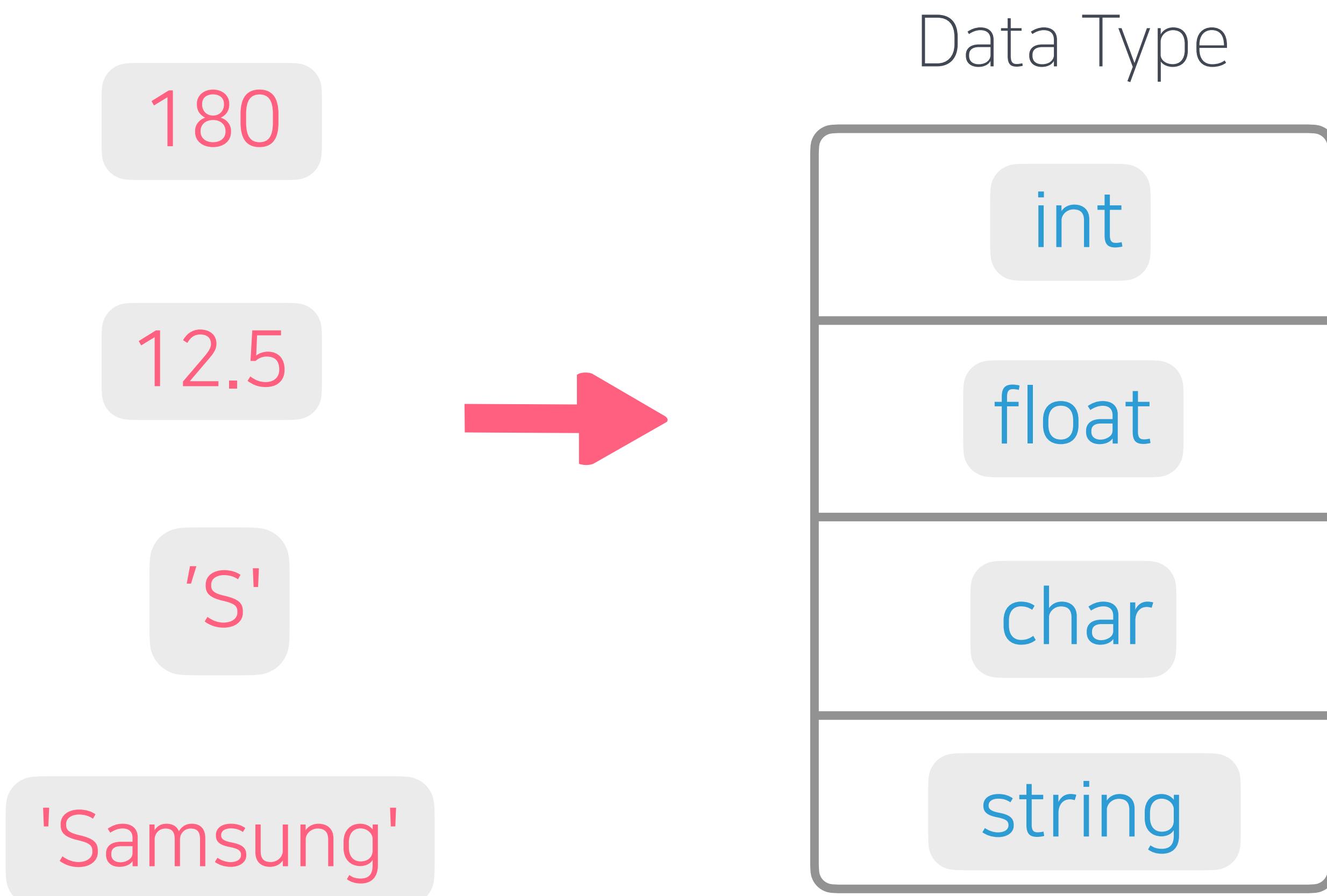
'S'

'Samsung'

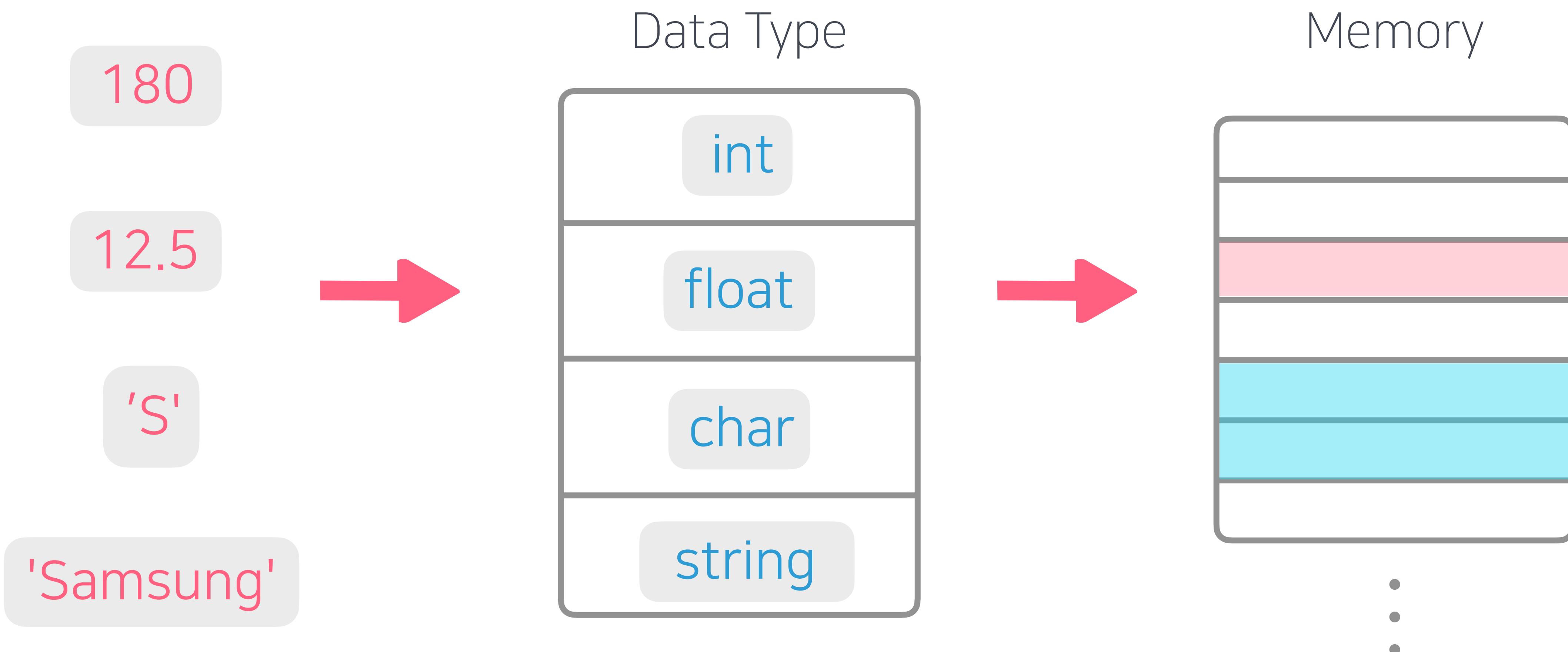
저장



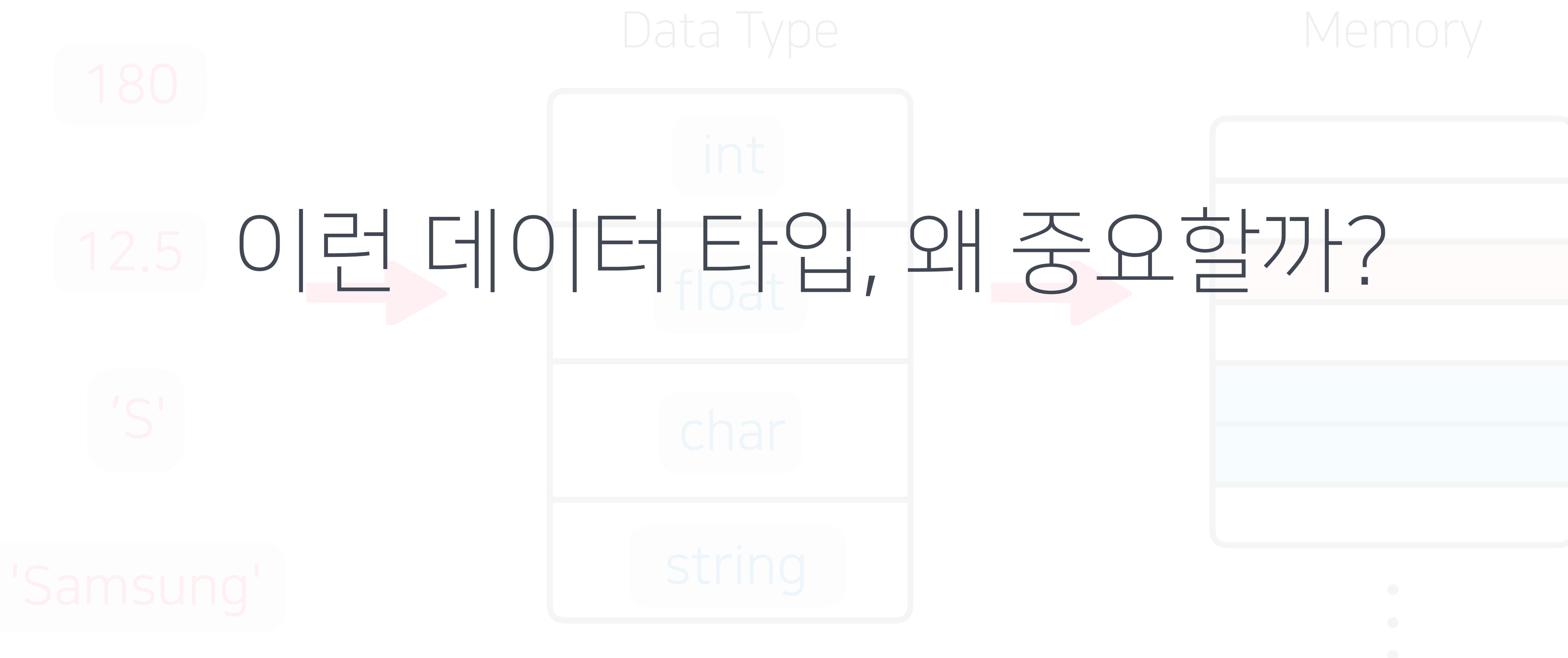
데이터 타입(Data Type)



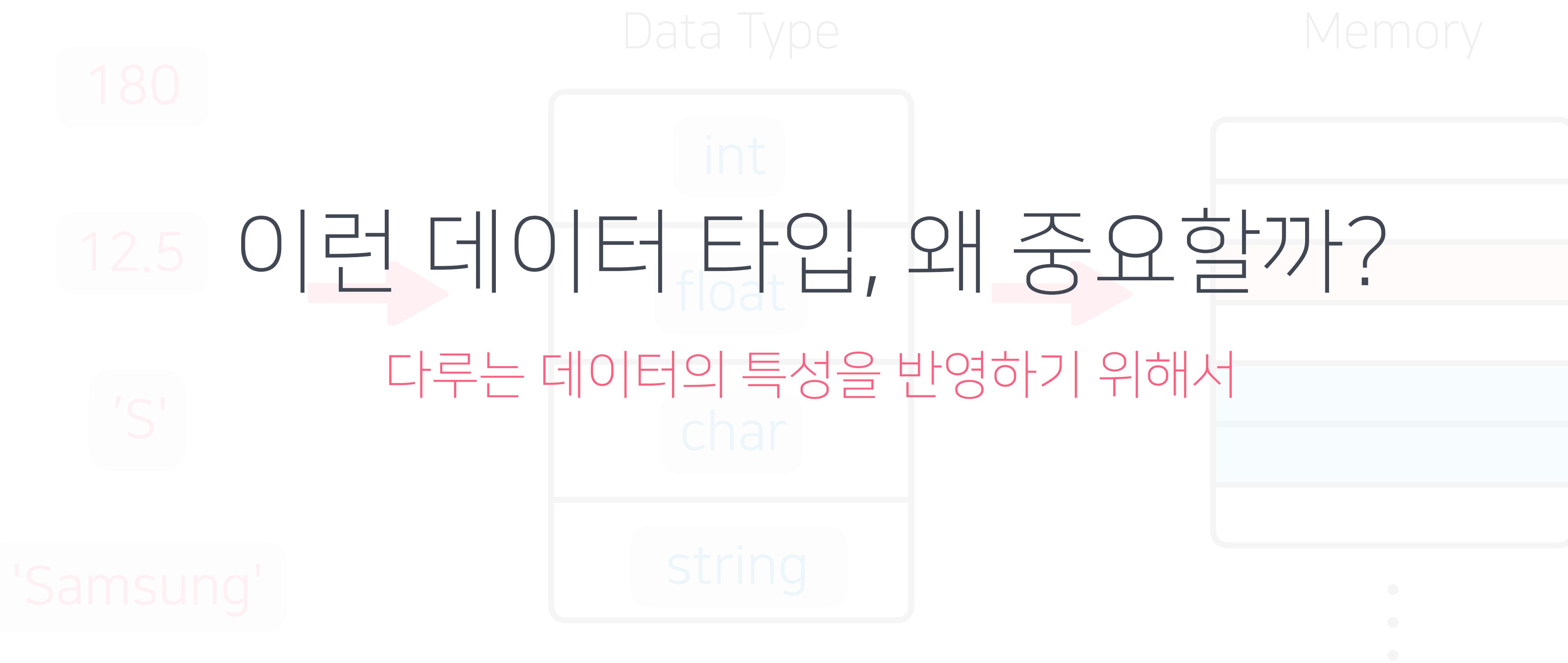
데이터 타입(Data Type)



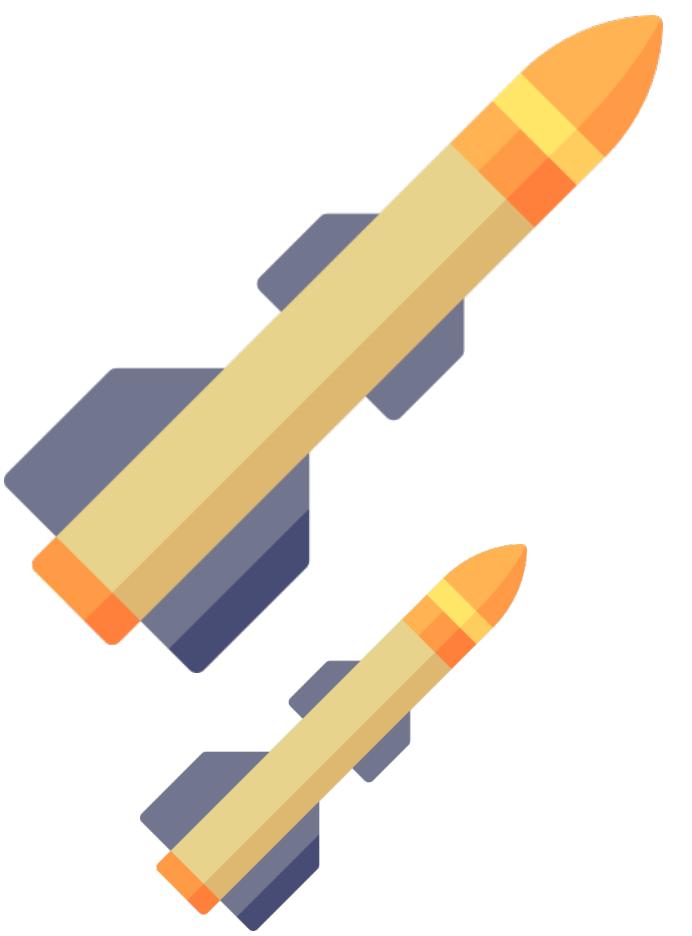
데이터 타입(Data Type)



데이터 타입(Data Type)

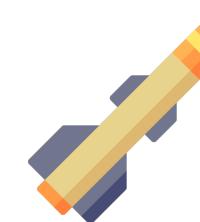


역사속의 SW 오류



Patriot Missile 오류

역사속의 SW 오류



Patriot Missile 오류

The New York Times

World

WORLD

U.S.

N.Y. / REGION

BUSINESS

TECHNOLOGY

SCIENCE

HEALTH

SPORTS

OPINION

AFRICA AMERICAS ASIA PACIFIC EUROPE MIDDLE EAST

WAR IN THE GULF: Scud Attack; Scud Missile Hits a U.S. Barracks, Killing 27

By R. W. APPLE Jr., Special to The New York Times

Published: February 26, 1991

1991년 걸프전

DHAHRAN, Saudi Arabia, Tuesday, Feb. 26— In the most devastating Iraqi stroke of the Persian Gulf war, an Iraqi missile demolished a barracks housing more than 100 American troops on Monday night, killing 27 and wounding 98, the American military command in Riyadh said early today.

Some of those who lived there were women, but a military spokesman said he did not know whether any women were killed or wounded. A pool report said the barracks housed the 475th Quartermaster Group, an Army Reserve unit based in Farrell, Pa., a small town near the Ohio state line.

FACEBOOK

TWITTER

GOOGLE+

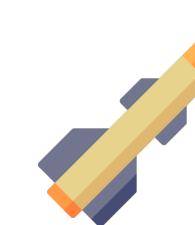
EMAIL

SHARE

PRINT

REPRINTS

역사속의 SW 오류



Patriot Missile 오류



겁먹지마, 우리 이런거있어 ^^

The New York Times

World

WORLD

U.S.

N.Y. / REGION

BUSINESS

TECHNOLOGY

SCIENCE

HEALTH

SPORTS

OPINION

AFRICA AMERICAS ASIA PACIFIC EUROPE MIDDLE EAST

WAR IN THE GULF: Scud Attack; Scud Missile Hits a U.S. Barracks, Killing 27

By R. W. APPLE Jr., Special to The New York Times

Published: February 26, 1991

DHAHRAN, Saudi Arabia, Tuesday, Feb. 26— In the most devastating Iraqi stroke of the Persian Gulf war, an Iraqi missile demolished a barracks housing more than 100 American troops on Monday night, killing 27 and wounding 98, the American military command in Riyadh said early today.

Some of those who lived there were women, but a military spokesman said he did not know whether any women were killed or wounded. A pool report said the barracks housed the 475th Quartermaster Group, an Army Reserve unit based in Farrell, Pa., a small town near the Ohio state line.

FACEBOOK

TWITTER

GOOGLE+

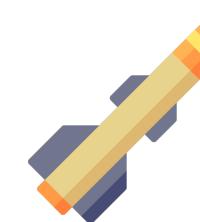
EMAIL

SHARE

PRINT

REPRINTS

역사속의 SW 오류



Patriot Missile 오류



아니! 대체 뭐했냐 !!!

The New York Times

World

WORLD

U.S.

N.Y. / REGION

BUSINESS

TECHNOLOGY

SCIENCE

HEALTH

SPORTS

OPINION

AFRICA AMERICAS ASIA PACIFIC EUROPE MIDDLE EAST

WAR IN THE GULF: Scud Attack; Scud Missile Hits a U.S. Barracks, Killing 27

By R. W. APPLE Jr., Special to The New York Times

Published: February 26, 1991

DHAHRAN, Saudi Arabia, Tuesday, Feb. 26— In the most devastating Iraqi stroke of the Persian Gulf war, an Iraqi missile demolished a barracks housing more than 100 American troops on Monday night, killing 27 and wounding 98, the American military command in Riyadh said early today.

Some of those who lived there were women, but a military spokesman said he did not know whether any women were killed or wounded. A pool report said the barracks housed the 475th Quartermaster Group, an Army Reserve unit based in Farrell, Pa., a small town near the Ohio state line.

FACEBOOK

TWITTER

GOOGLE+

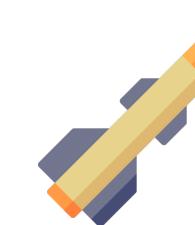
EMAIL

SHARE

PRINT

REPRINTS

역사속의 SW 오류

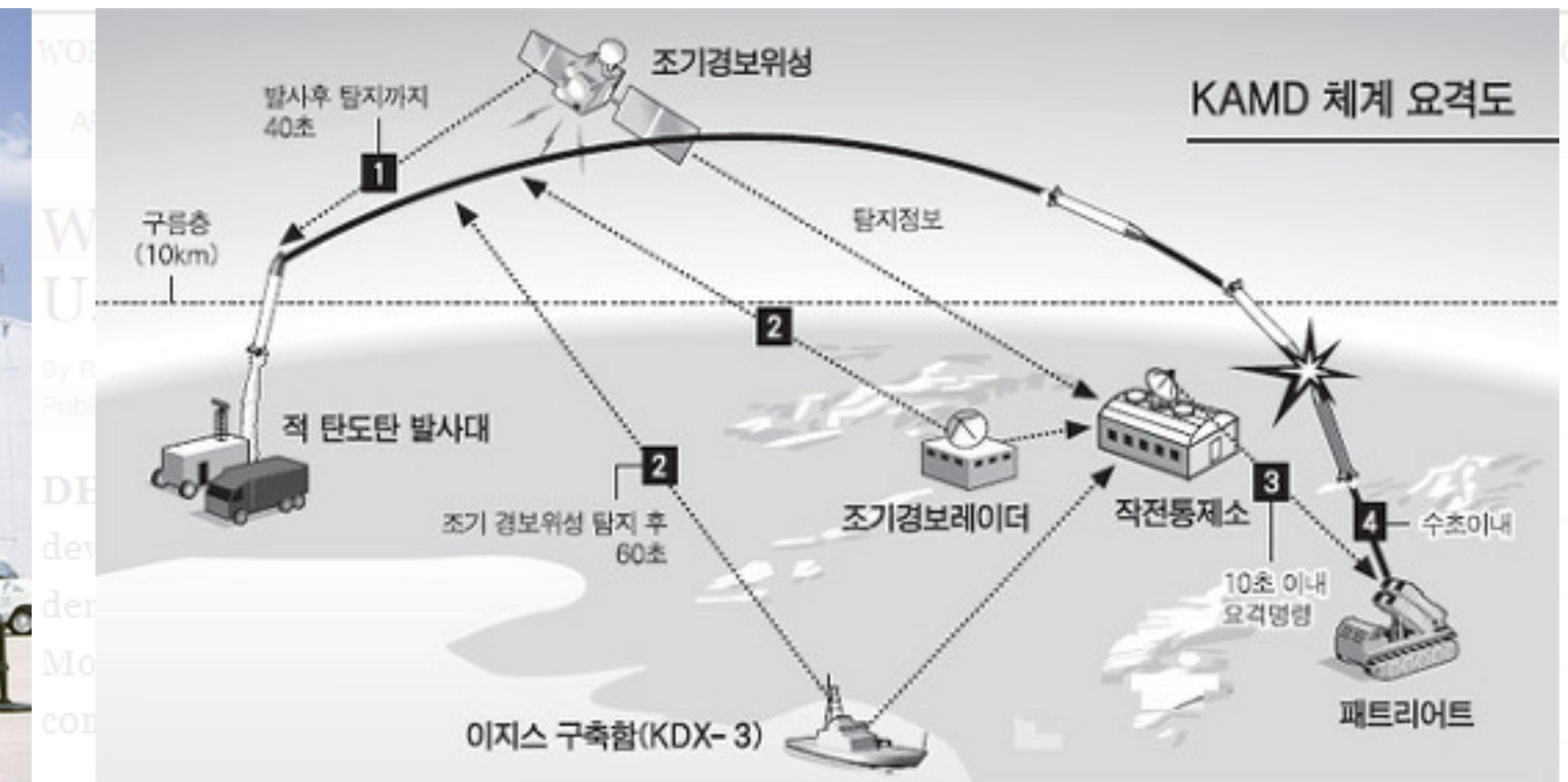


Patriot Missile 오류



The New York Times

World



아니! 대체 뭐했느냐 !!!

Some of those who lived there were women, but a military spokesman said he did not know whether any women were killed or wounded. A pool report said the barracks housed the 50th Quartermaster Group, an Army Reserve unit based in Farrell, Pa., a small town near the Ohio state line.
그, 오차가 있지 말입니다 ...

PRINT

REPRINTS

데이터 타입(Data Type)

현실 세계의 데이터를 0과 1로 표현해서
메모리에 저장하기 위한 방법

데이터 타입(Data Type)

현실 세계의 데이터를 0과 1로 표현해서
메모리에 저장하기 위한 방법

좋은데 ... 한계가 있어

데이터 타입(Data Type)

- 더 다양한 데이터 구조를 표현하고 싶어

데이터 타입(Data Type)

- 더 다양한 데이터 구조를 표현하고 싶어
- 내가 데이터 타입을 정의하고 싶은데?

→ **User-defined data type**

클래스(class)를 이용해서 나만의 데이터 타입을 선언

데이터 타입(Data Type)

학번

학과

이름

성별

나이

데이터 타입(Data Type)

| | |
|----|--------|
| 학번 | int |
| 학과 | string |
| 이름 | string |
| 성별 | char |
| 나이 | int |

데이터 타입(Data Type)

학번

int

학과

string

이름

string

성별

char

나이

int

Student

int

string

string

char

int

Student

⋮

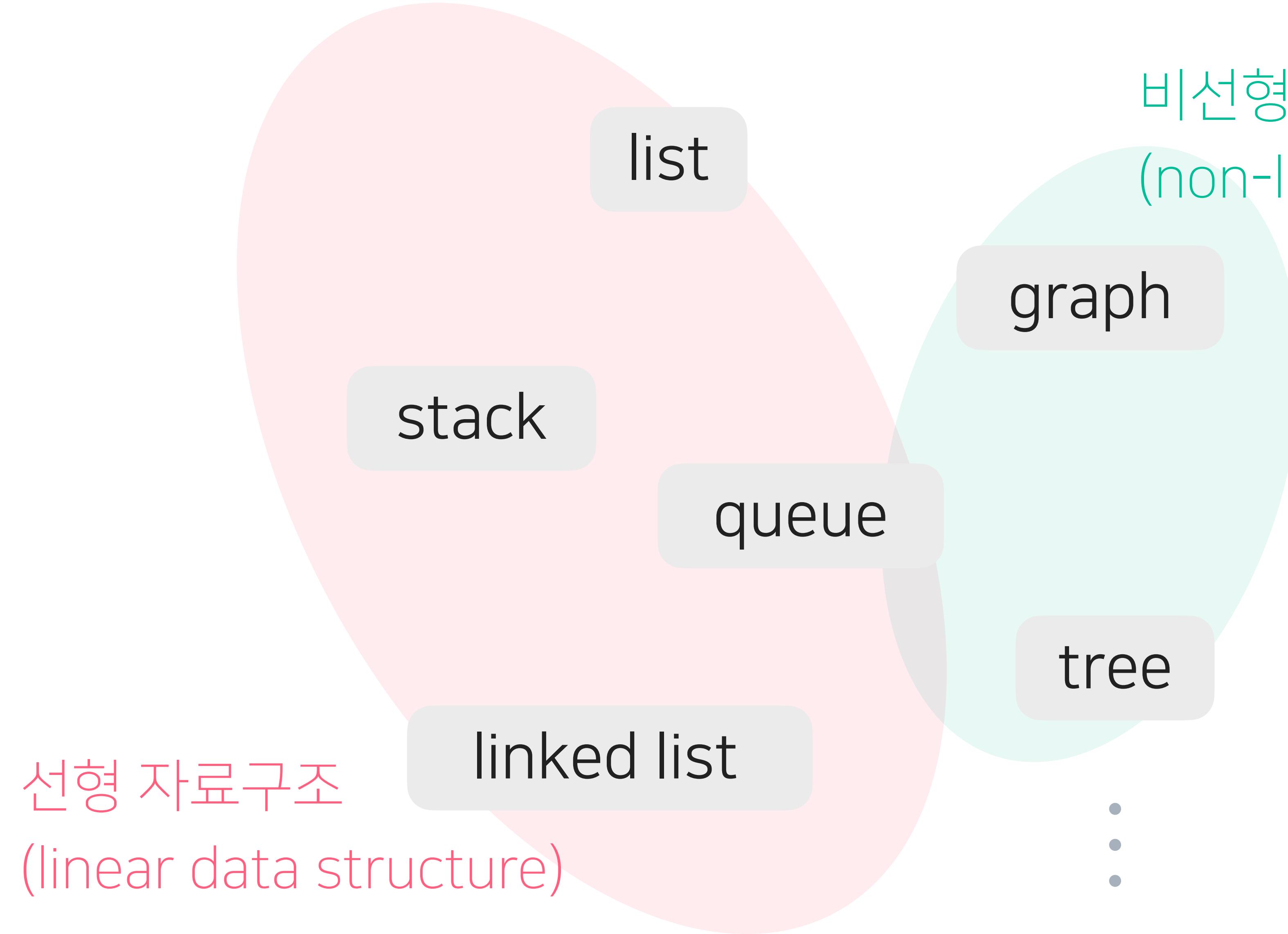
Student

⋮

자료구조 (Data Structure)

컴퓨터의 메모리를 효율적으로 사용할 수 있도록
데이터를 저장하고 구성하는 방법

자료구조(Data Structure)



자료구조(Data Structure)

그래서, 자료구조 쓰면 뭐가 좋아요?

선형 자료구조

(linear data structure)

linked list

queue

list

stack

비선형 자료구조
(non-linear data structure)

graph

tree

⋮

자료구조(Data Structure)

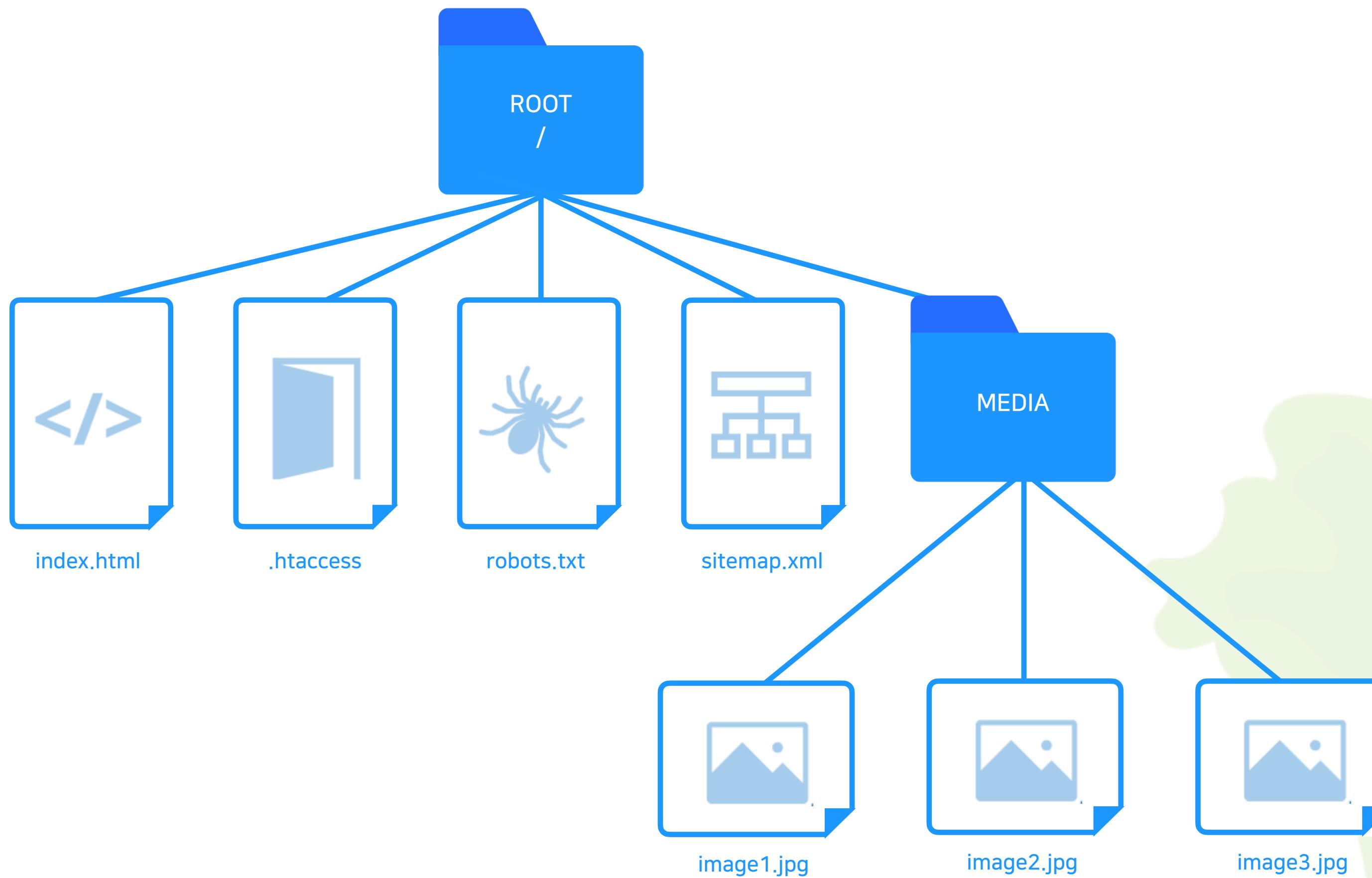


폴 고갱(Paul Gauguin)

Where Do We Come From? What Are We? Where Are We Going?

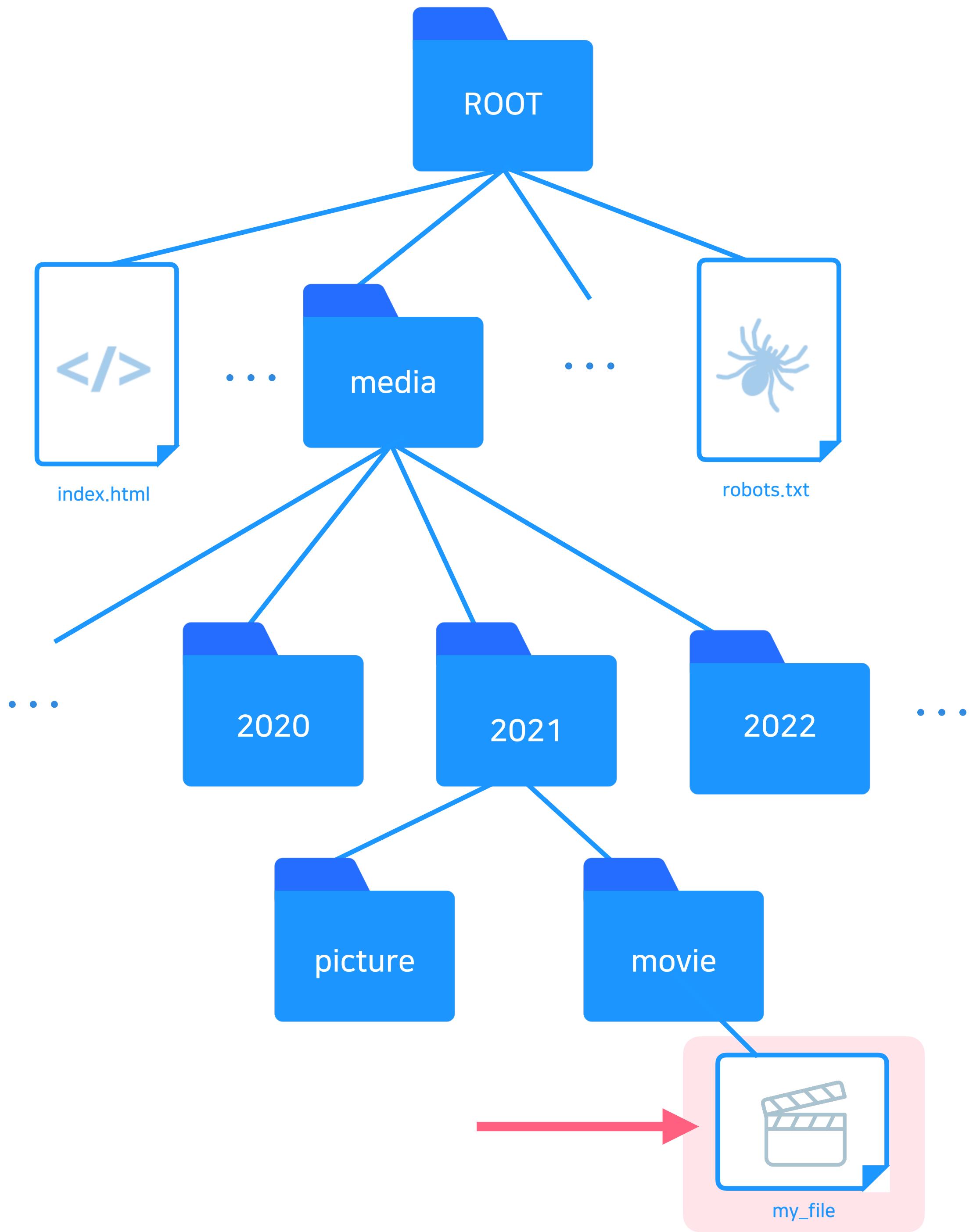
데이터의 구조를 전체적으로 파악해서 효율적으로 사용가능

자료구조(Data Structure)



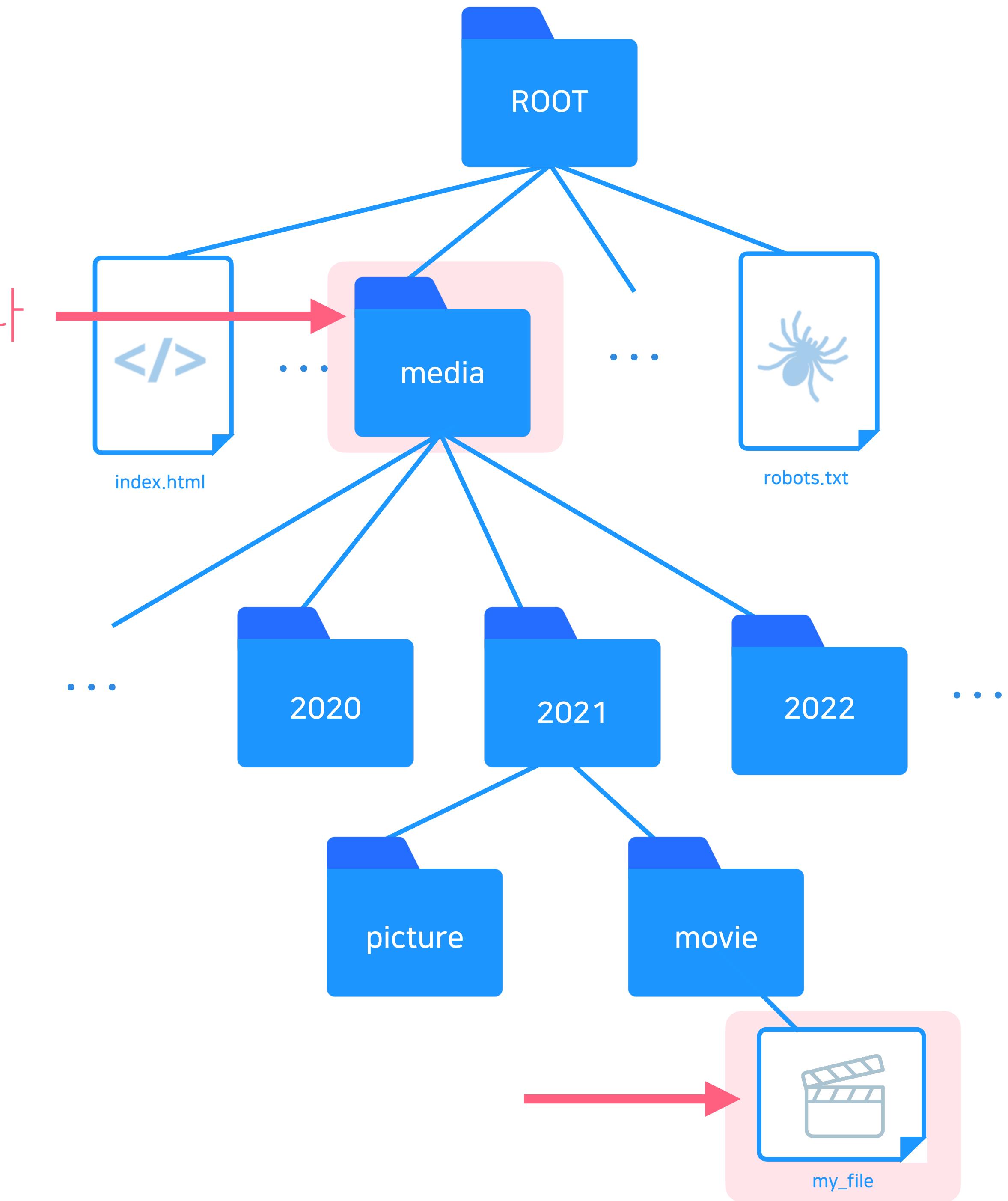
트리(Tree)

자료구조(Data Structure)

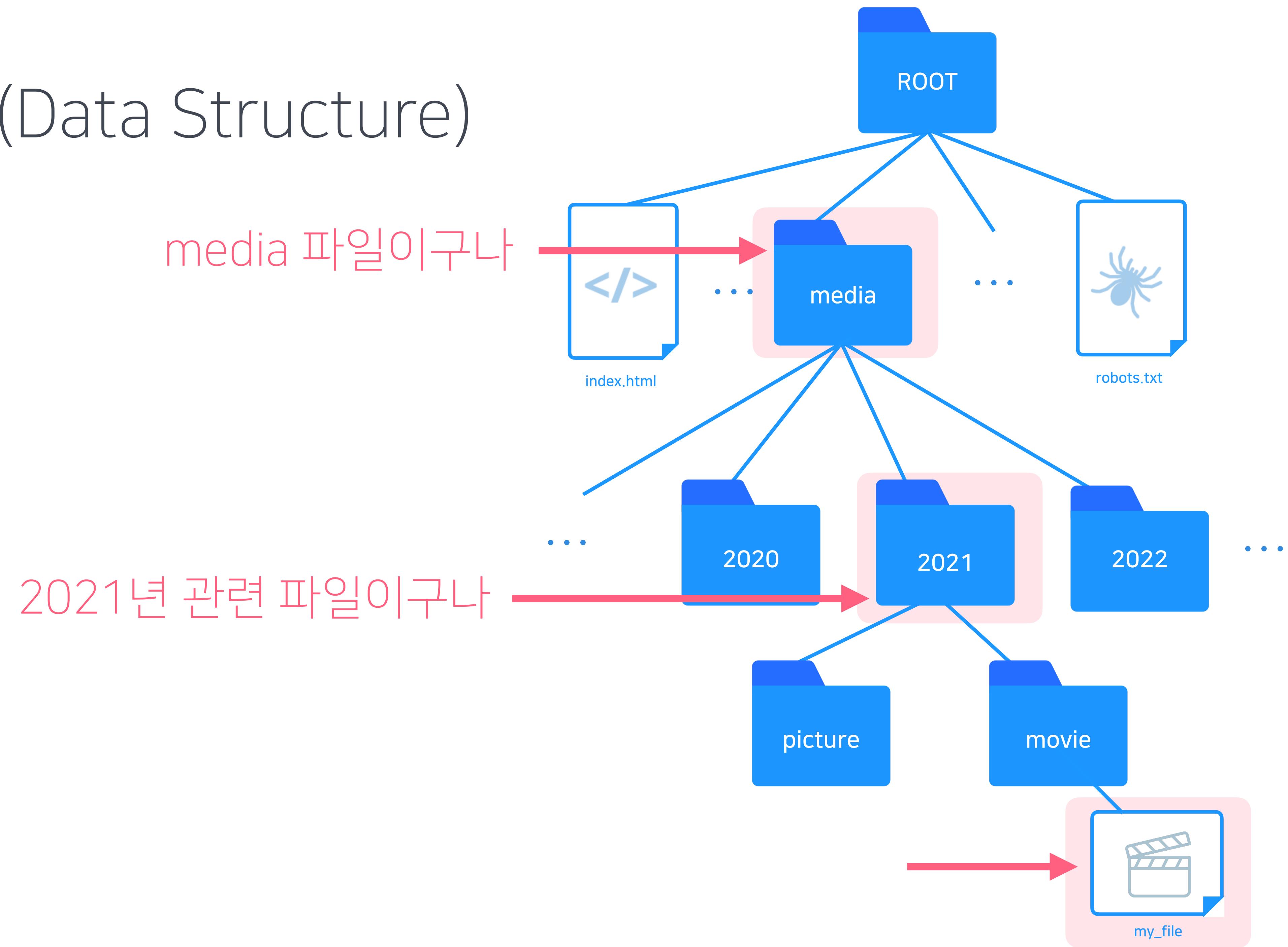


자료구조(Data Structure)

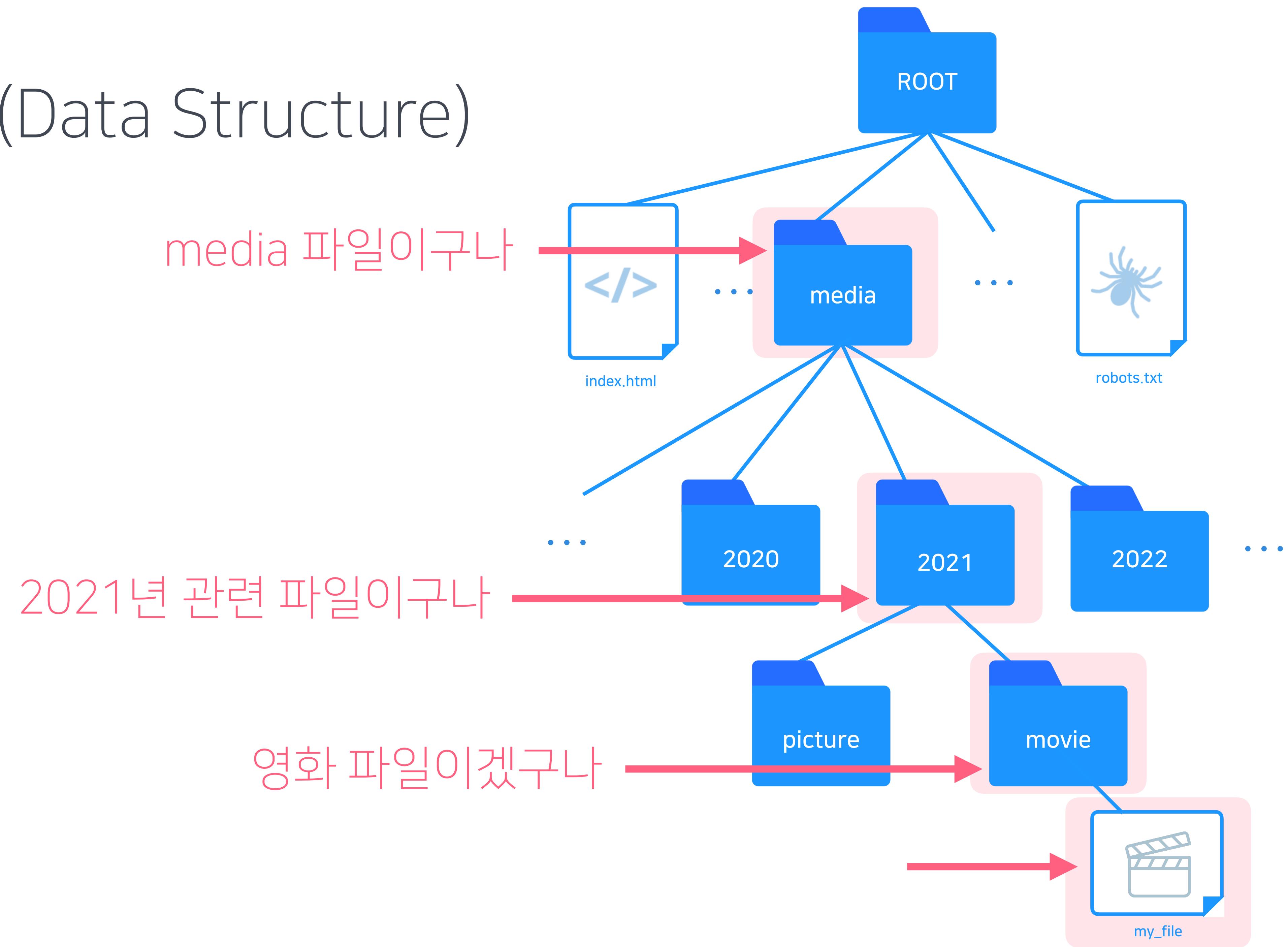
media 파일이구나



자료구조(Data Structure)



자료구조(Data Structure)



자료구조(Data Structure)

데이터에 맞는 자료구조

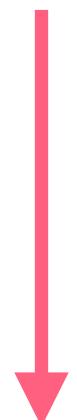
- 데이터를 표현하기 쉽고
- 데이터를 이해하기 쉽고
- 데이터를 활용하기도 쉽다!



추상 데이터 타입

Abstract Data Type

데이터가 가지고 있는 공통점



- 데이터가 가지고 있는 구조
- 데이터가 가지고 있는 연산

추상화

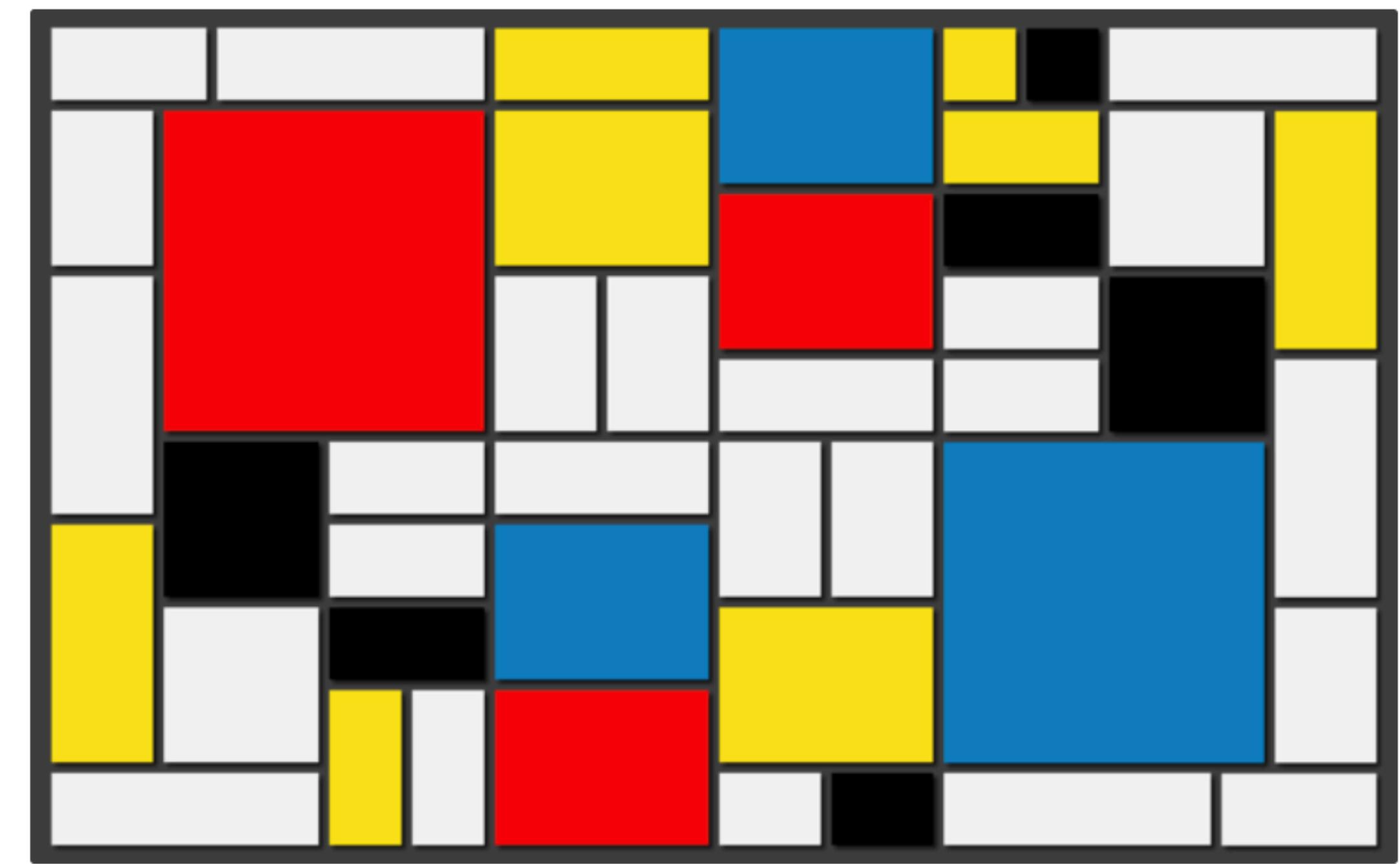
Abstraction

특정한 대상이나 사람이나 상황이 아닌
일반적인 개념으로 만드는 것

추상화 (Abstraction)



제주도 유채꽃밭



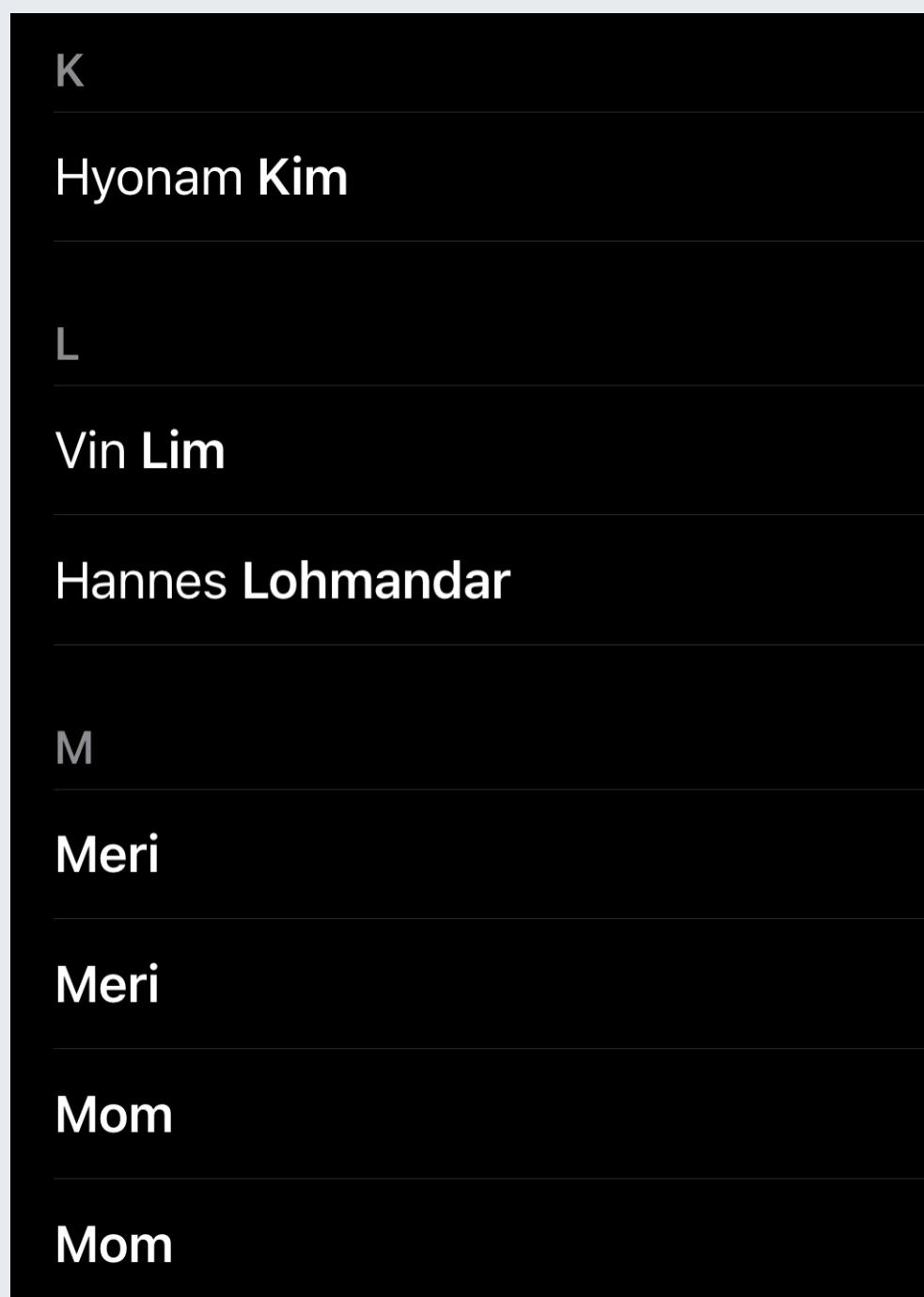
몬드리안 - 추상화 (1921)

ADT (Abstract Data Type)

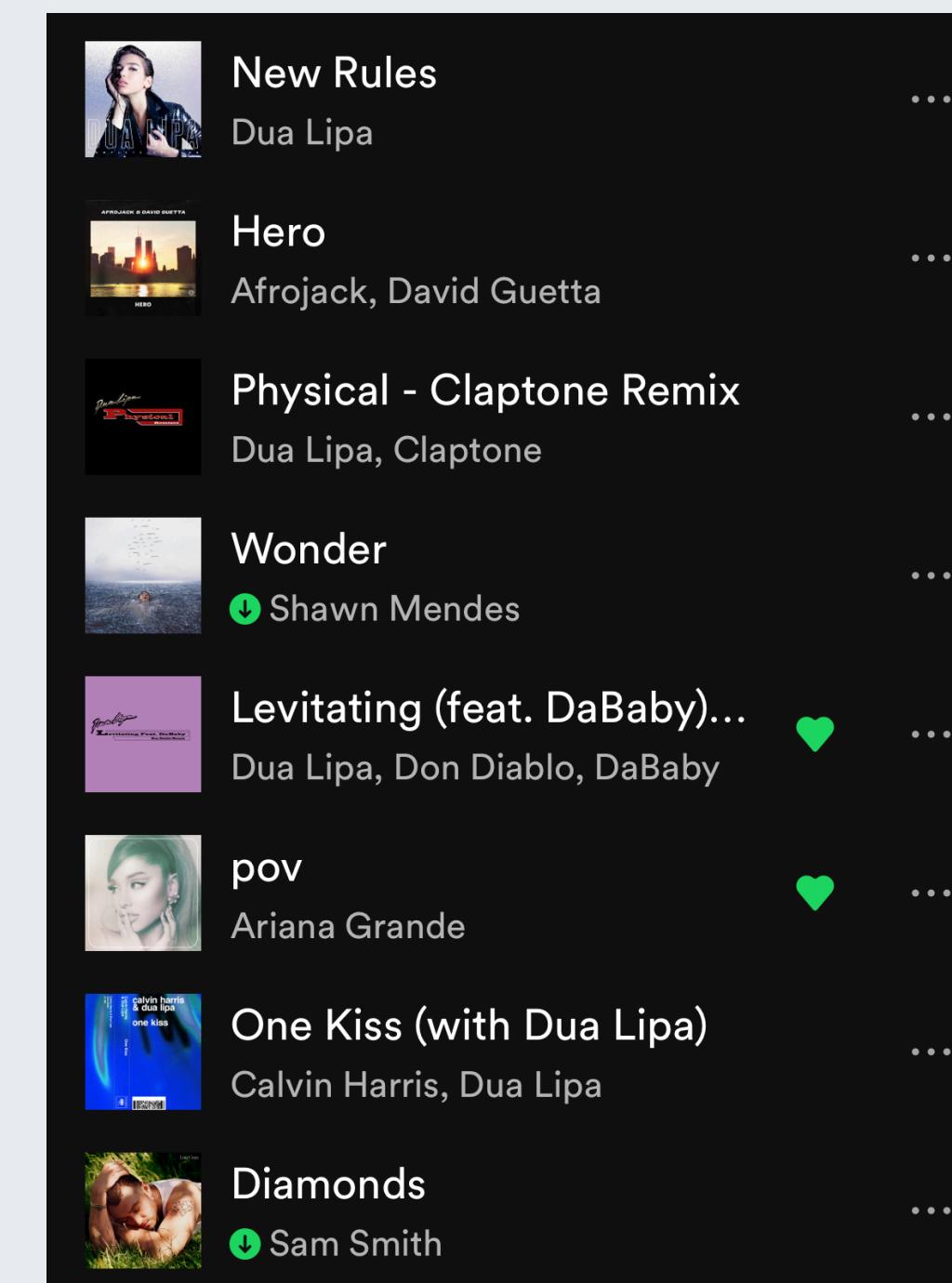
구조 + 연산
추상화를 이용해서 자료구조를 표현해 봅시다.

ADT (Abstract Data Type)

스마트폰에서 자주쓰는 데이터 구조



번호 목록



음악 목록

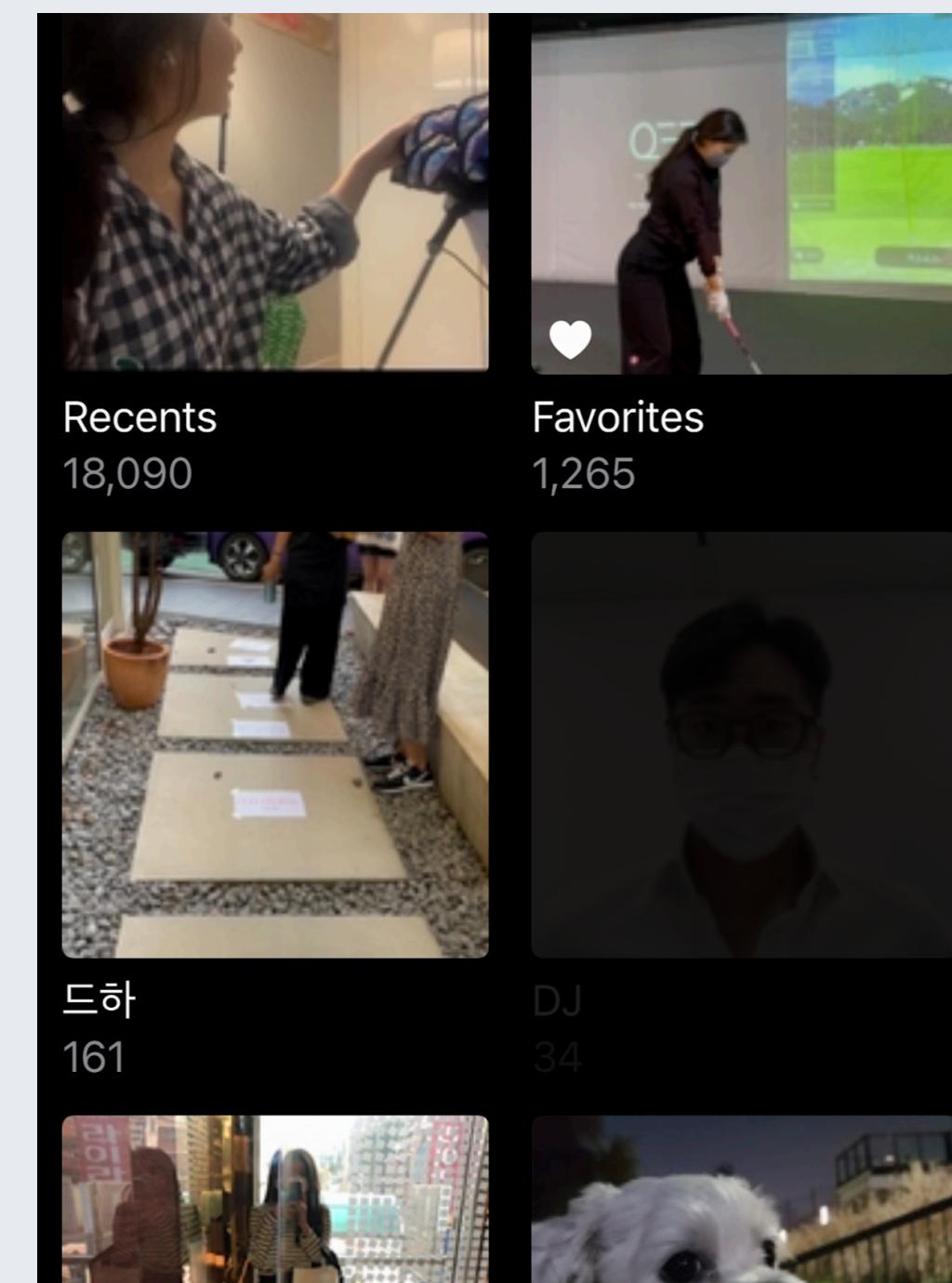


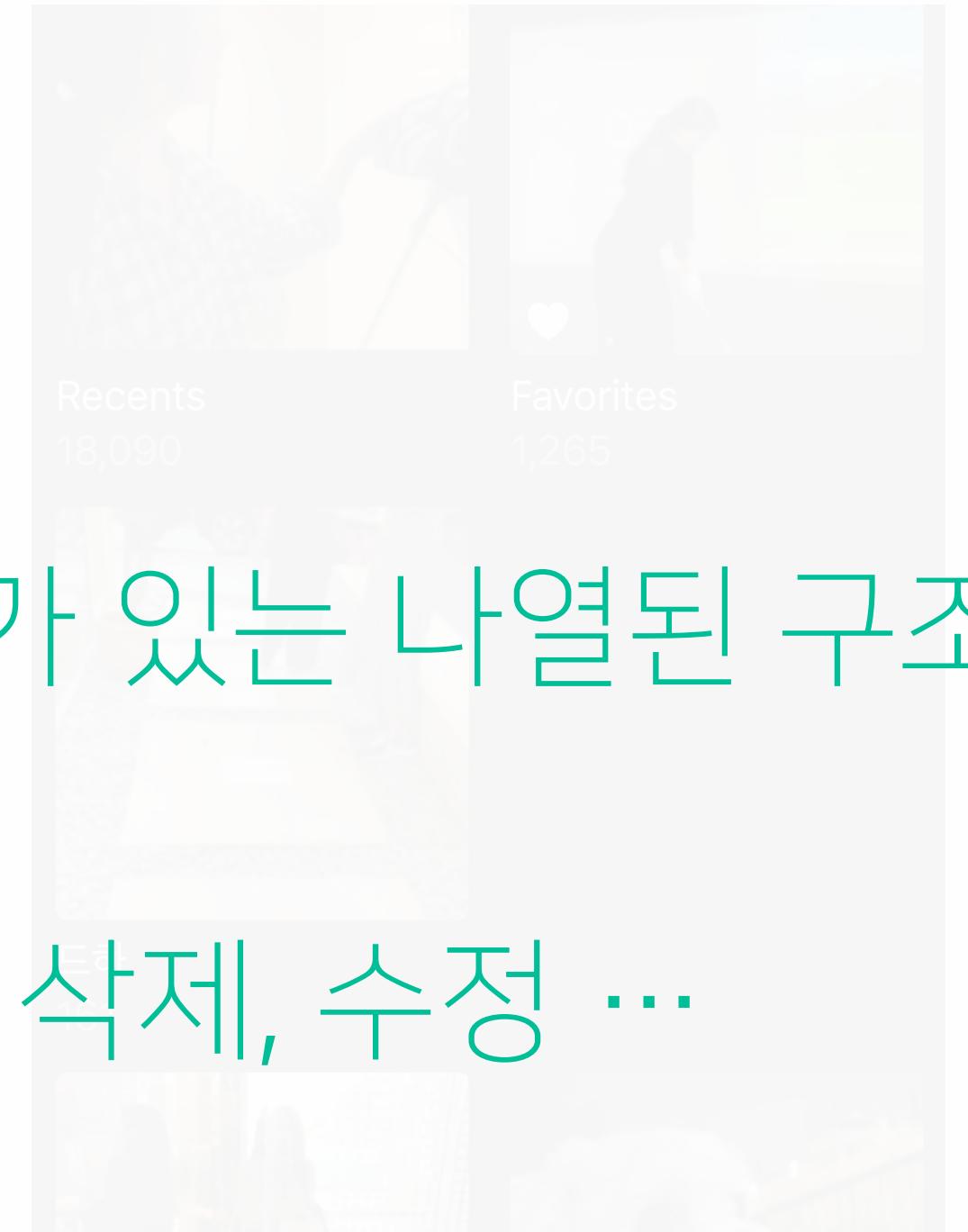
사진 목록

ADT (Abstract Data Type)

스마트폰에서 자주쓰는 데이터 구조

- 개별적인 특징을 지우고

- 데이터 구조의 특징을 도출

→ 순서가 있는 나열된 구조
- 필요한 연산을 정의

→ 추가, 삭제, 수정 ...

번호 목록

음악 목록

사진 목록

Abstract Data Type

구조화된 데이터를 필요한 연산과 함께
묶어서 표현하는 방법

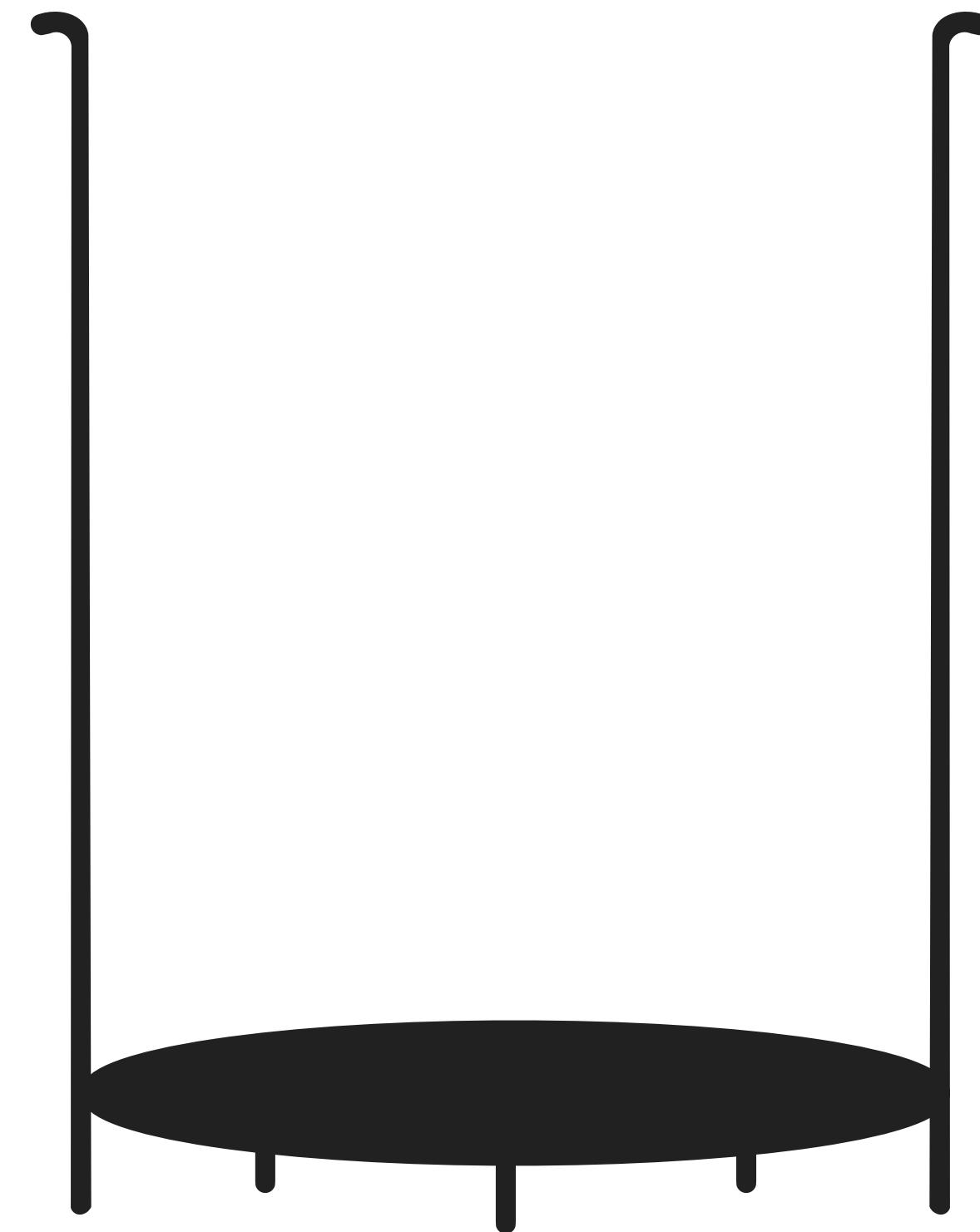
Abstract Data Type = Structured Data + Operation

스택 Stack

스택은 물건을 쌓아 올린듯, 자료를 쌓아 올린 형태의 자료구조

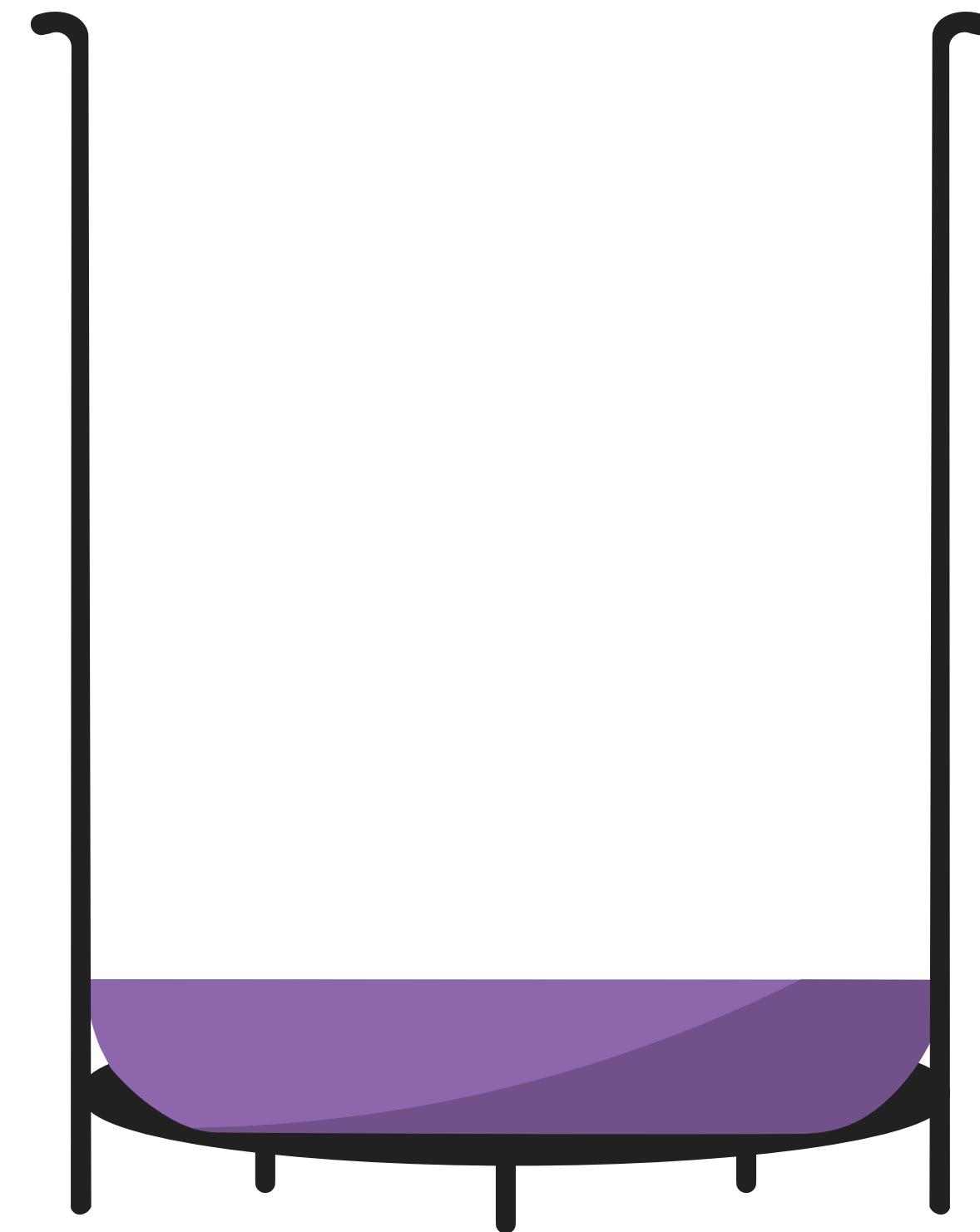
스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



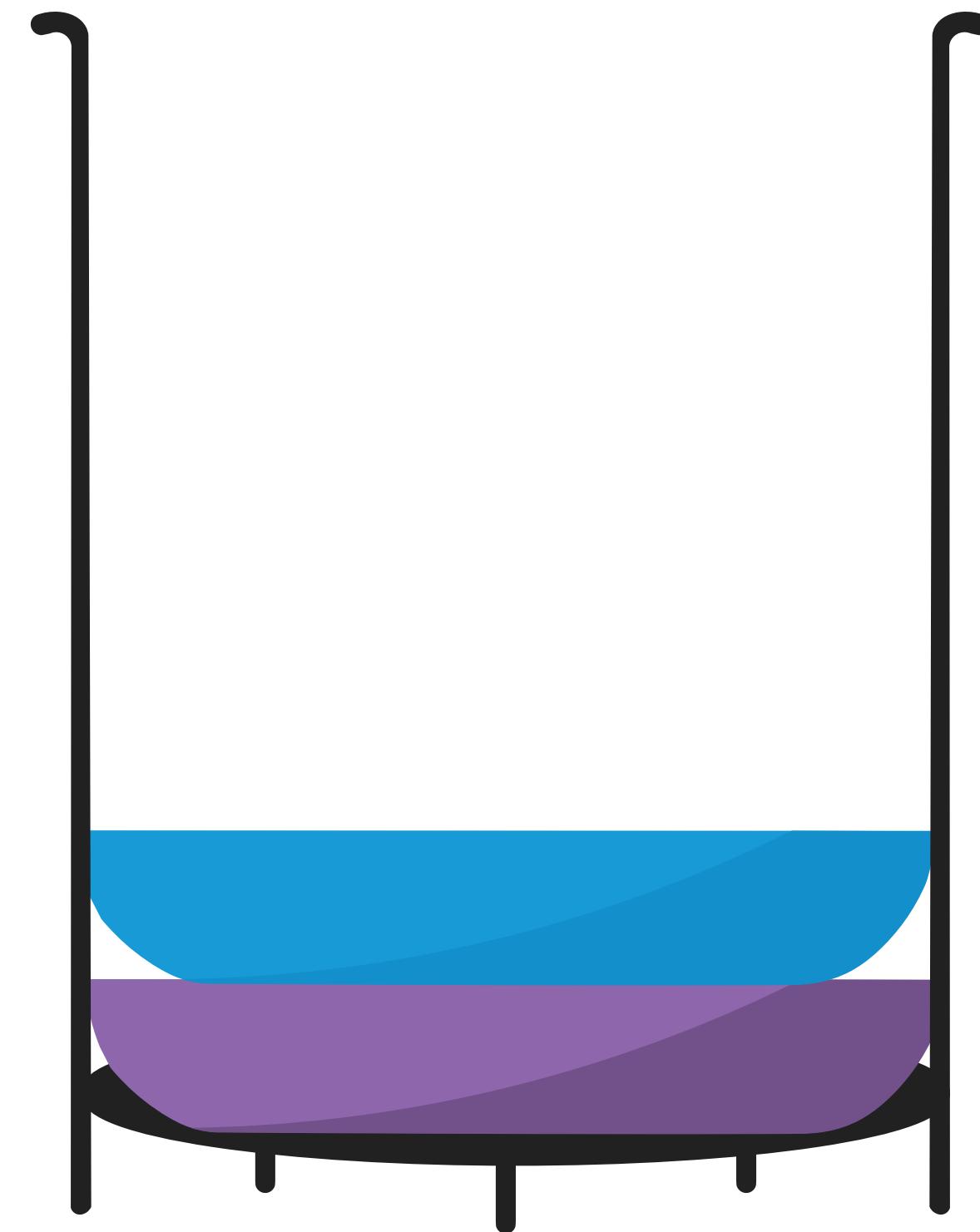
스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



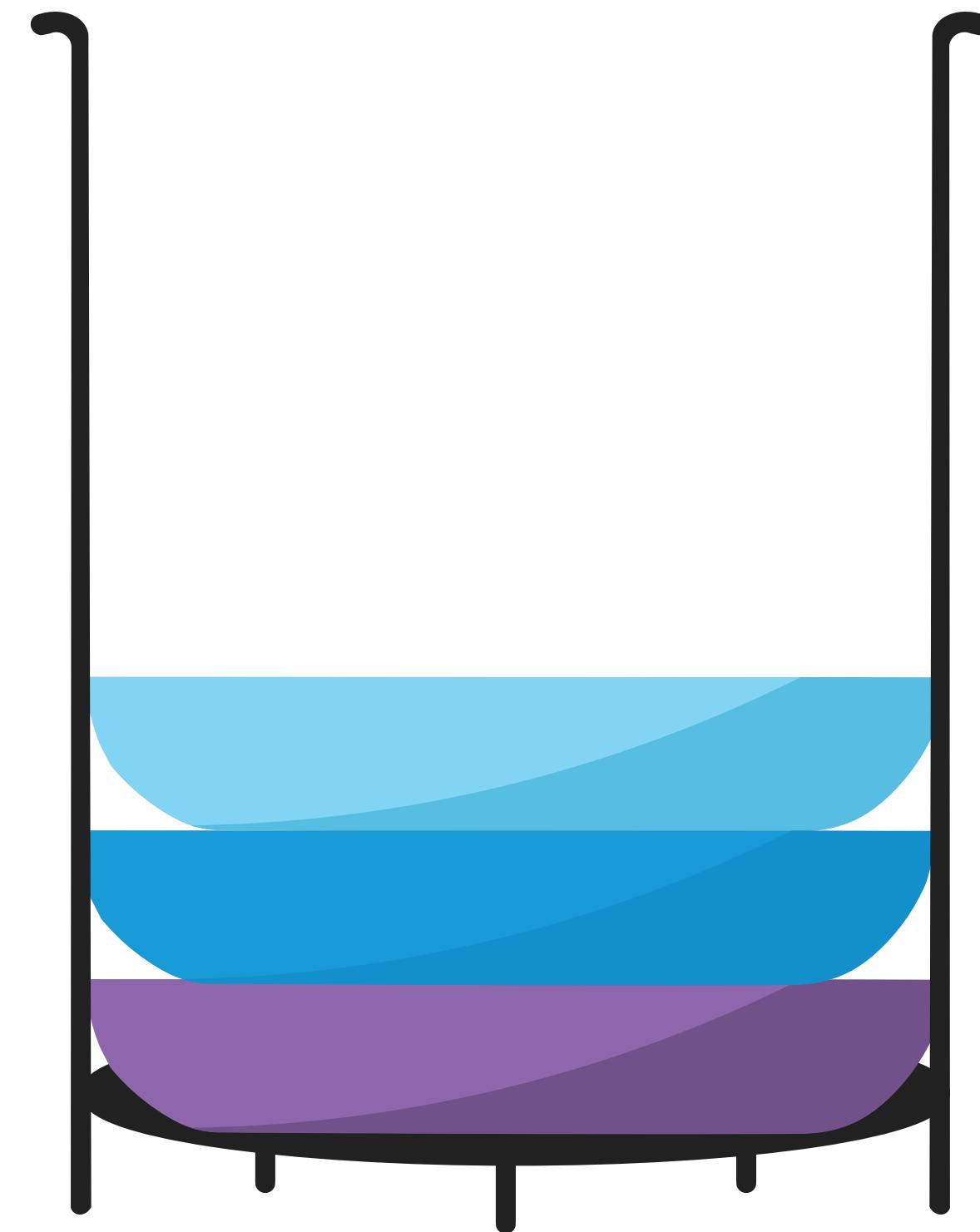
스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)

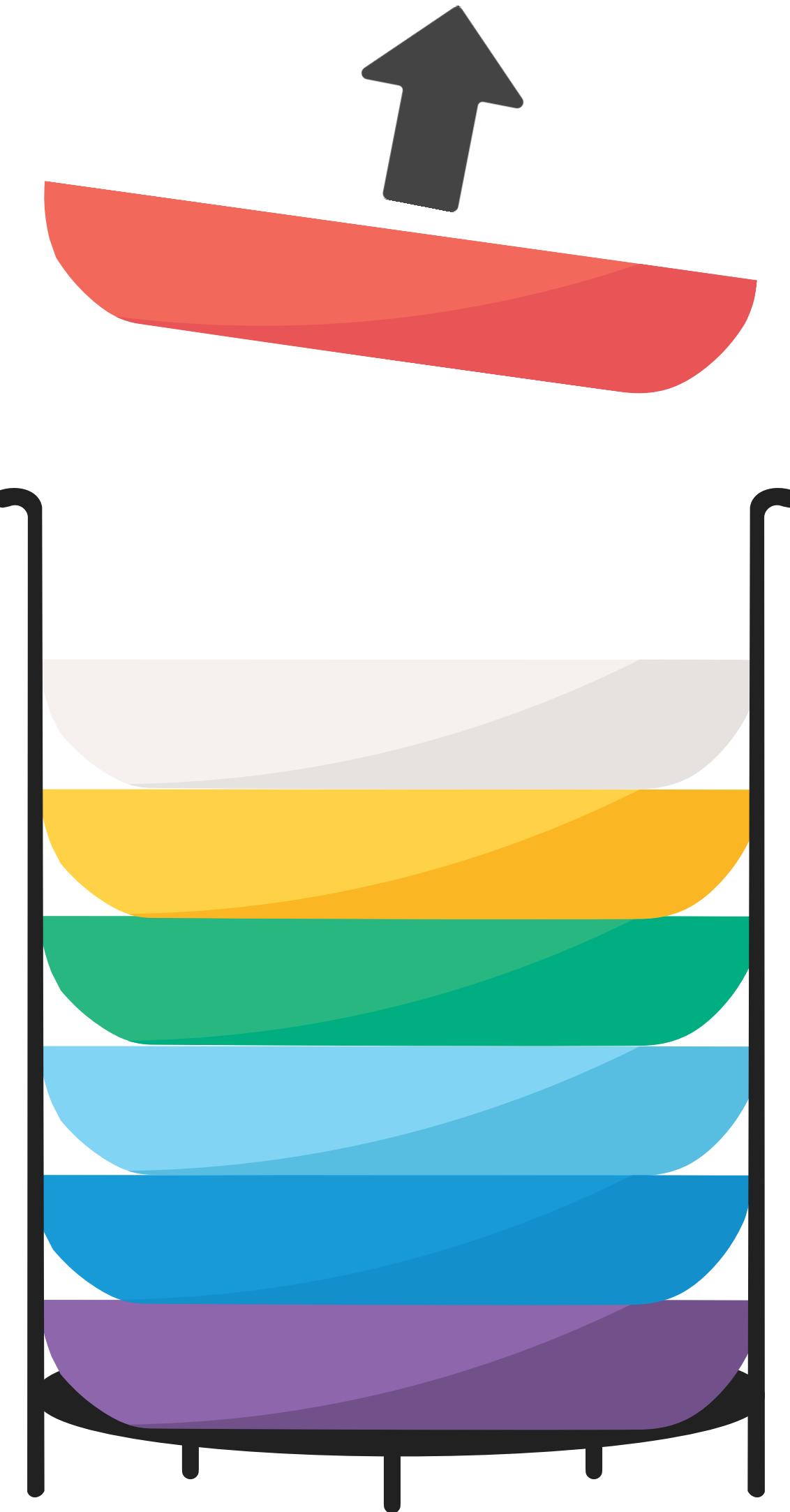
물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조

어떤 특징이 있을까?



스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



- 1) **가장 마지막**에 들어간 것이, **가장 처음**에 나온다.

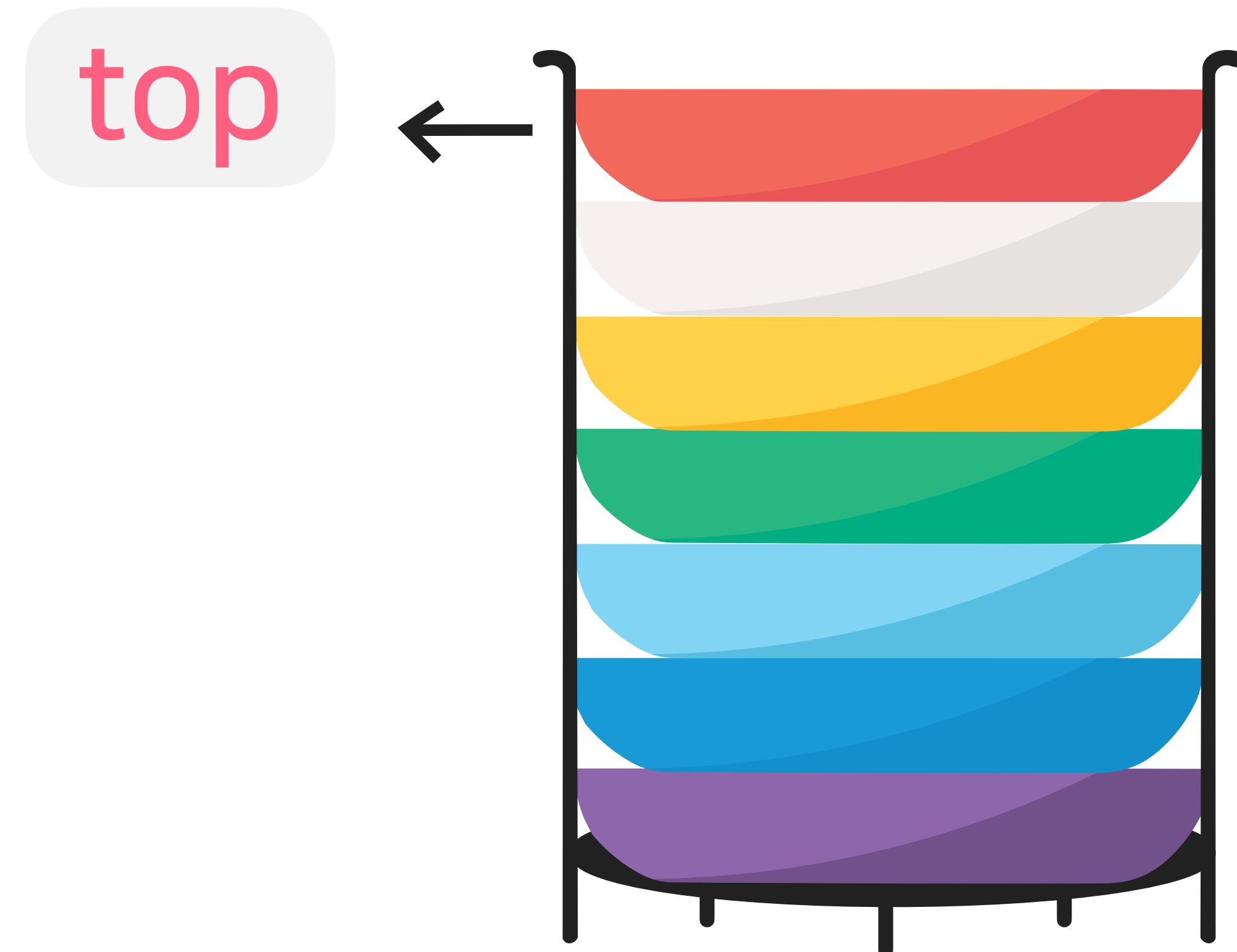
후입선출

LIFO (Last-In-First-Out)

- 2) **가장 위**에서만 데이터의 삽입 & 삭제가 일어난다

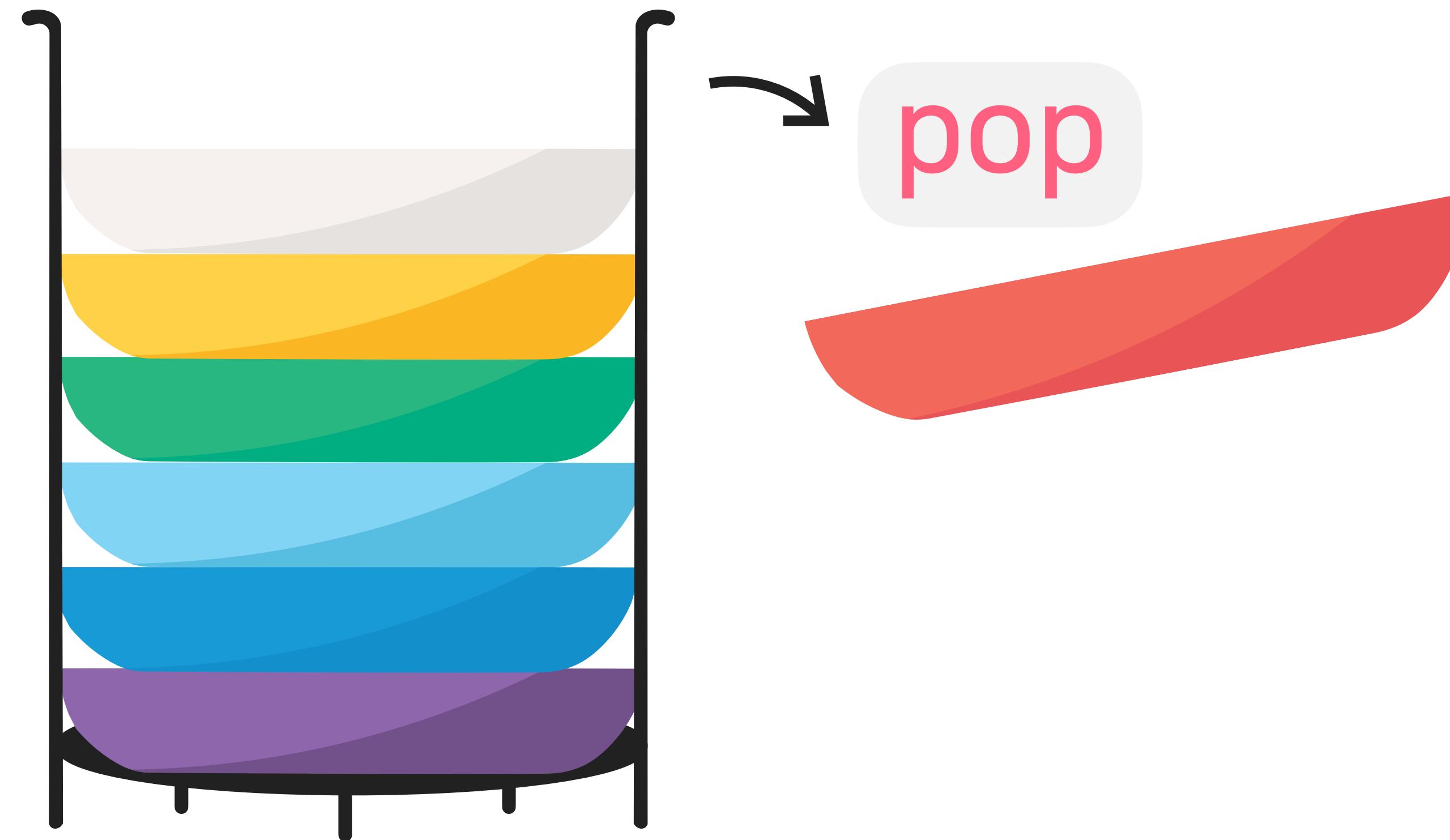
스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



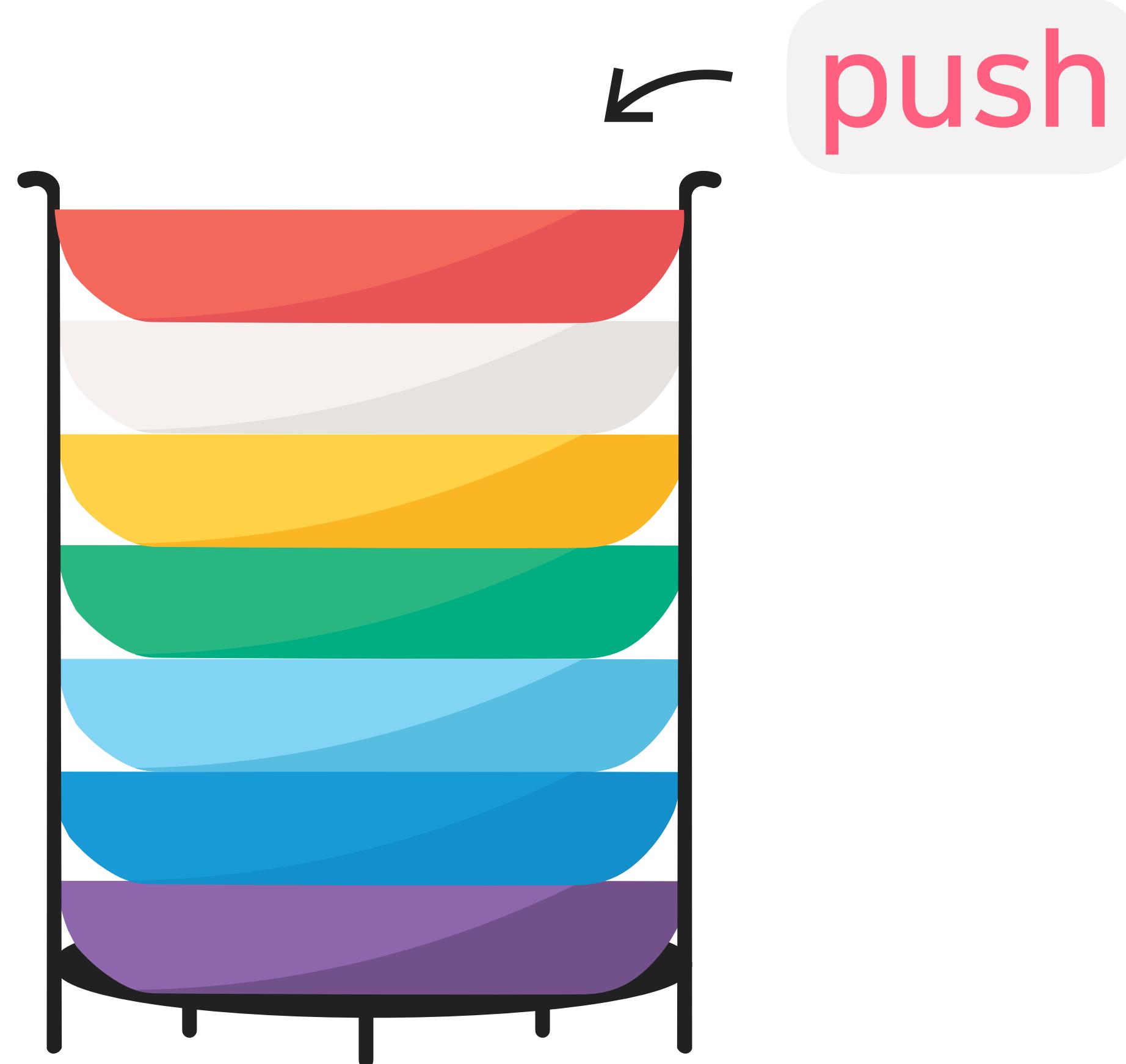
스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조

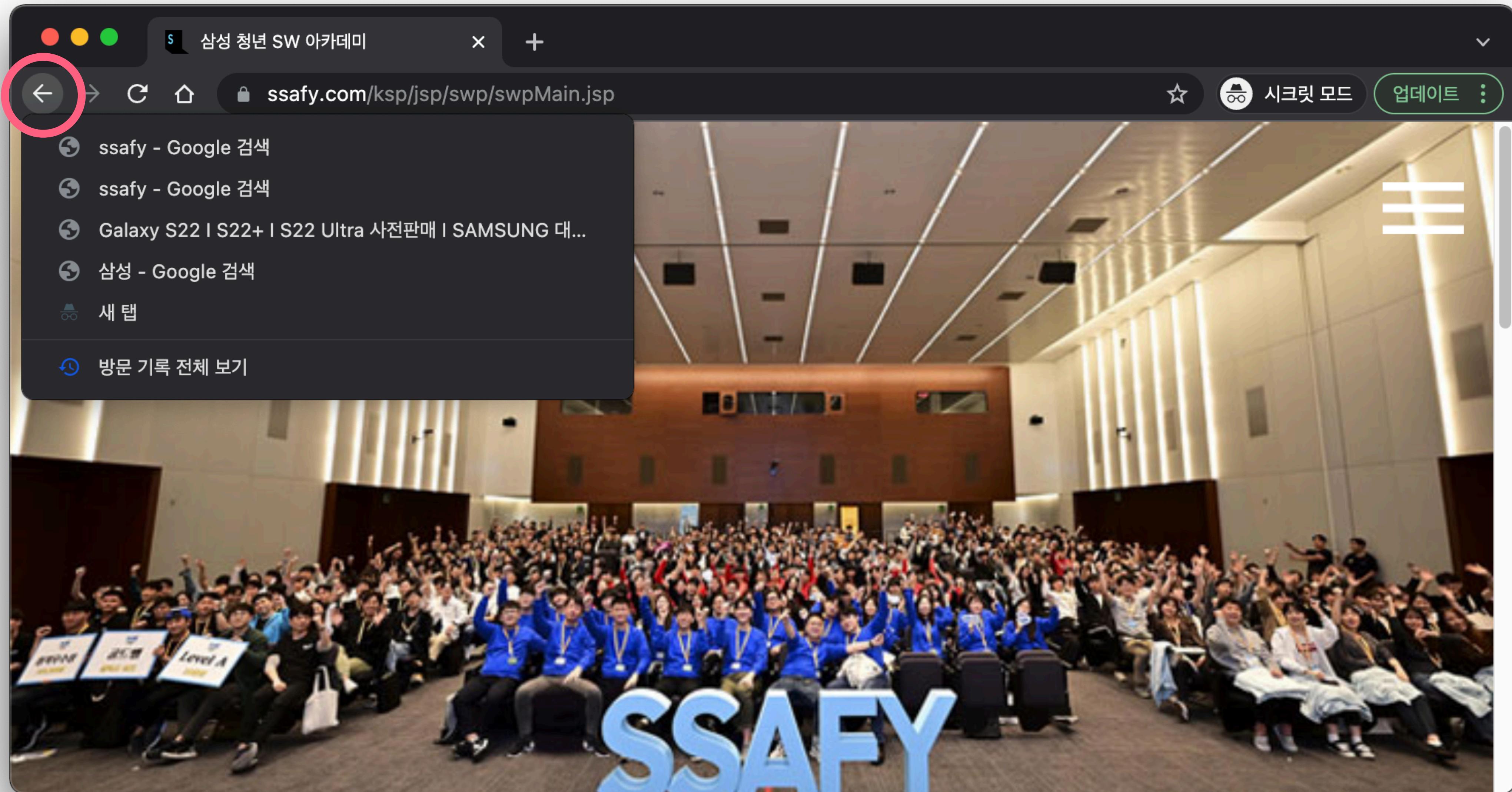


스택 (stack)

물건을 쌓아 올린 듯 **자료를 쌓아 올린** 형태의 자료구조



스택 (stack)



스택 (stack)

A screenshot of a web browser displaying the MDN Web Docs page for the History API. The browser interface includes a title bar with the tab 'History API - Web APIs | MDN', a back/forward button, a search bar containing 'developer.mozilla.org/en-US/docs/Web/API/History_API', and various browser controls like 'Secret mode' and 'Update'. The MDN header features the logo, navigation links for 'Technologies', 'References & Guides', and 'Feedback', and a site search bar. Below the header, the breadcrumb navigation shows 'Web technology for developers > Web APIs > History API'. On the right, there's a 'Change language' button. The main content area has a large heading 'History API'. To its left is a 'Table of contents' sidebar with links to 'Concepts and usage', 'Interfaces', 'Examples', 'Specifications', 'Browser compatibility', and 'See also'. Below this is a 'Related Topics' section with a link to 'History API'. The main content area also contains a section titled 'Moving forward and backward' with a note about navigating history.

History API

The DOM `window` object provides access to the browser's session history (not to be confused for [WebExtensions history](#)) through the `history` object. It exposes useful methods and properties that let you navigate back and forth through the user's history, and manipulate the contents of the history stack.

Concepts and usage

Moving backward and forward through the user's history is done using the `back()`, `forward()`, and `go()` methods.

Moving forward and backward

To move backward through history:

스택 (stack)

함수 콜 스택

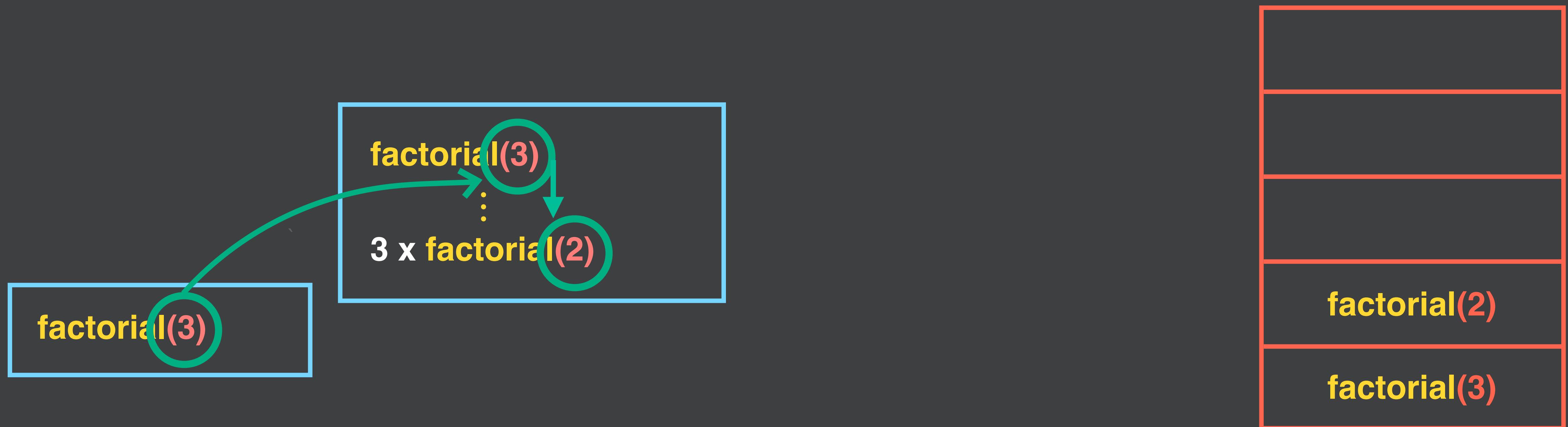
```
def factorial(n):  
    if n == 1:  
        return 1  
    else:  
        return n * factorial(n - 1)
```

스택 (stack)

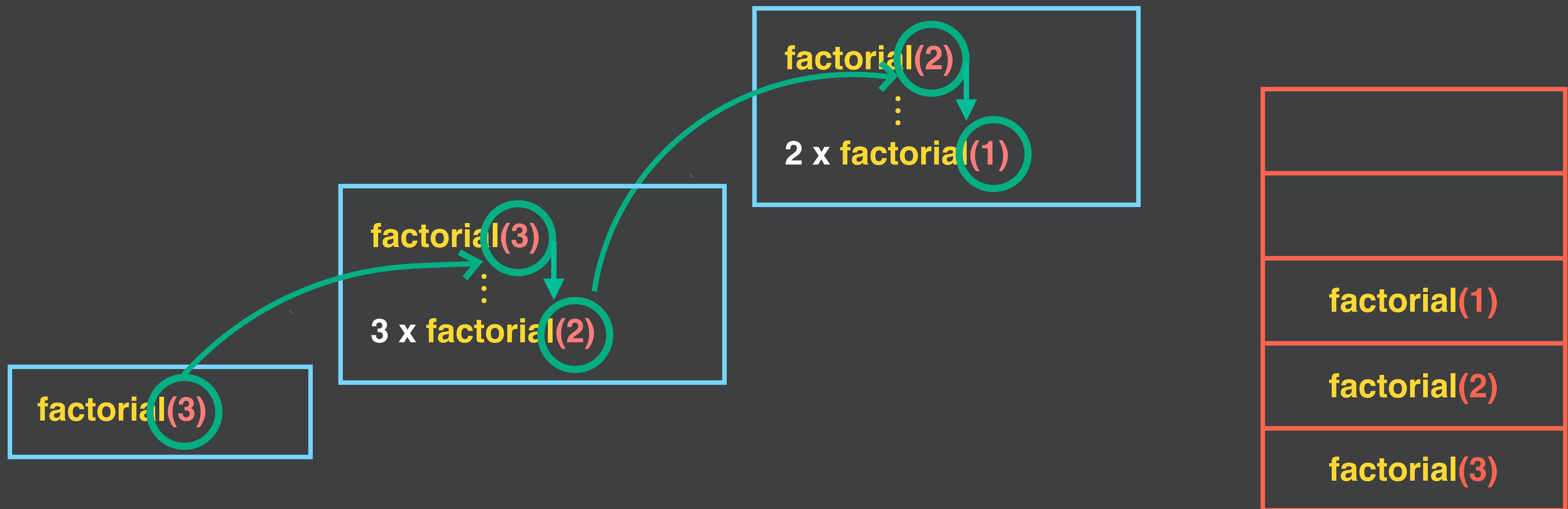
factorial(3)

factorial(3)

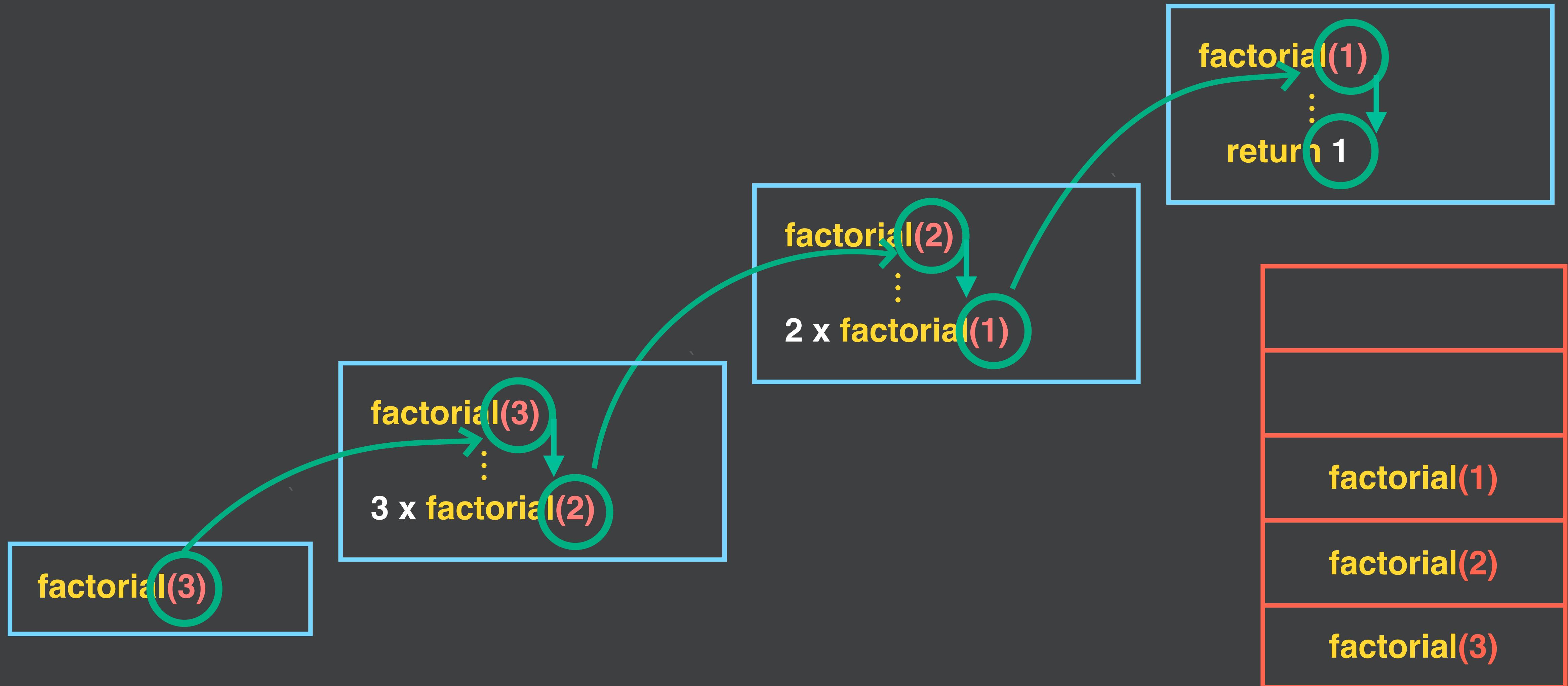
스택 (stack)



스택 (stack)



스택 (stack)



algorithm > class > factorial.py

Project

sol1.py x test.py x factorial.py x input.txt x

```
1 def factorial(n): n: 1
2     if n == 1:
3         return 1
4     else:
5         return n * factorial(n - 1)
6
7
8 print(factorial(3))
9
```

factorial()

Debug: factorial x

A row of small, semi-transparent icons representing various debugger functions like step, break, and run.

A wrench icon next to the word "Frames".

Variables

- >MainThread
- + factorial, factorial.py:1
- factorial, factorial.py:5
- factorial, factorial.py:5
- <module>, factorial.py:8

Evaluate expression (Enter) or add a watch (Ctrl+Shift+Enter)

n = {int} 1

스택 (stack)



Jürgen Schmidhuber's page on

Friedrich Ludwig BAUER (1924-)

Due to his impact on computer science departments all over the country Bauer has been called the father of German informatics.

[TUM link](#)

F. L. Bauer (at [Tech. Univ. München](#) since 1963) came up with at least two concepts that are essential for modern computer science.

1. Together with [Klaus Samelson](#) (also of TUM) [he invented the stack machine](#) (patented in 1957), a fundamental device for both theory and practice of programming. Stacks are *THE* way to handle recursive function calls and the dynamic runtime behavior of computer programs. Any modern microprocessor incorporates them.
2. In 1968 Bauer coined the term "software engineering." Today there are uncountable software engineering departments in universities and companies all over the world, and a large fraction of computer scientists would call themselves software engineers. [Google for "software engineering"](#) and you'll get tens of millions of hits (2005).

Bauer also was a driving force behind ALGOL, a hugely influential programming language which introduced numerous important constructs used in many subsequent programming languages. He also contributed significantly to the fields of numerical analysis, algebra, logic, program transformation, and cryptography.

In the early days of modern computer science, Bauer was one of the first to realize the fundamental significance of the work of [Konrad Zuse](#), inventor and constructor of the world's first working general purpose computer in Berlin (1935-1941).

“ He invented the **stack machine** ”

“ Stacks are the way to handle
recursive function calls, and
the dynamic runtime behavior
of computer programs.”

스택 (stack)

전 세계 개발자들이 개발 관련해서 검색할 때 많이 이용하는 곳



가장 최신의 정보가, 가장 먼저 검색된다는 의미!

ADT로 표현한
스택 Stack

스택의 연산 (Operations of stack)

- ① CreateStack 스택을 생성하는 연산, `size` 필요
- ② IsEmpty 스택이 현재 **비어있는지를 확인** 하는 연산, `true` `false` 리턴
- ③ IsFull 스택이 현재 **꽉 차있는지를 확인** 하는 연산, `true`, `false` 리턴
- ④ Push 스택에 새로운 데이터 요소를 **삽입** 하는 연산
- ⑤ Pop 스택에서 가장 위에있는 요소를 **제거** 하는 연산, 데이터 반환
- ⑥ Peek 스택에서 가장 위에있는 요소를 **반환** 하는 연산

스택의 데이터 구조 (Data structure of stack)

① top

스택의 가장 위에 있는 위치를 저장하고 있는 데이터

② size

스택의 크기를 저장하고 있는 데이터

③ items

스택에 담길 데이터를 저장 할 데이터 구조

class Stack:

...

size = n

top = -1

items = []