

OOP 객체 지향 프로그래밍이란?

- 데이터나 메서드 등의 객체를 중심으로, 여러 객체가 서로 상호작용하도록 프로그램을 구성하는 방식이다. 객체 지향 프로그램은 코드의 재사용성, 유지보수성, 확장성이 높다.
- 상속, 추상화 같은 기법이 있다.

C와 파이썬의 차이

- C는 기계어에 가까운 저수준의 언어로 개발자가 메모리를 직접 관리하고 변수 선언 시 데이터 형식을 직접 명시해야 하는 등 문법이 복잡하지만 실행 속도가 빠르며,
- 파이썬은 인간 언어에 가까운 고수준의 언어로 자동으로 메모리를 할당하고, 동적으로 데이터 타입이 선언되는 등 문법이 간결하고 쉬우나, C 보다는 속도가 느리다.

리스트와 딕셔너리의 차이

- 리스트는 특정 순서로 데이터를 나열한 자료 구조이고 인덱스로 접근하며
- 딕셔너리는 Key와 Value의 쌍으로 데이터가 나열된 자료구조이고 키로 접근한다.

배열과 리스트의 차이

- 배열과 리스트는 자료를 순서대로 나열한다는 공통점이 있지만, 배열은 동일한 타입의 자료만 담을 수 있고, 리스트는 서로 다른 타입의 자료도 담을 수 있다. 리스트 보다는 배열의 계산이 더 빠르다.

Linked List 란?

- 데이터를 저장할 때, 각 노드가 자신의 값과 다음 노드의 주소를 함께 저장하는 자료구조로 배열과 달리 비연속적이고 포인터로 연결되어 있다.

포인터 란?

- 변수나 데이터가 저장된 메모리의 주소를 저장하는 변수, 즉 값이 저장된 위치(주소)를 저장
- 값 자체가 아닌, 값이 어디 있는지를 가리킴

시간복잡도 O 란?

- 입력 크기 n 이 커질 때, 알고리즘이 수행하는 연산 수의 증가율을 나타내는 척도
- 빅오 표기법(Big-O)으로 표현하며, 최악의 경우에 대한 연산량을 제공

시간복잡도	설명	예시
$O(1)$	상수 시간, 입력 크기와 무관	배열 접근 arr[5]
$O(\log n)$	로그 시간, 입력이 반씩 줄어듦	이진 탐색
$O(n)$	선형 시간	단일 for문
$O(n \log n)$	로그 성분 포함된 선형	병합정렬, 퀵정렬 평균
$O(n^2)$	이중 반복문	버블 정렬, 삽입 정렬

시간복잡도	설명	예시
$O(2^n)$	지수 시간	피보나치 재귀
$O(n!)$	팩토리얼 시간	순열 생성

스택메모리와 힙메모리의 차이

리스트와 튜플의 차이

클래스와 인스턴스의 차이

변수의 SCOPE

딕셔너리와 해시테이블의 차이

for문과 while문의 차이

예외 처리란 무엇이며, 왜 필요한가?

deep copy와 copy의 차이

모든 클래스는 초기화 함수로 무엇을 사용하는가?

상속, 추상, 인터페이스란?

재귀함수란?

numpy의 axis 연산이란?

numpy의 sum, max, argmax 란?

람다 함수란?