

# Day20; 20220727

|         |                          |
|---------|--------------------------|
| 🕒 작성일시  | @2022년 8월 3일 오전 10:24    |
| 📌 강의 번호 |                          |
| 📅 유형    |                          |
| 📎 자료    |                          |
| ☑ 복습    | <input type="checkbox"/> |
| ☰ 속성    |                          |

- 해시 탐색법으로 데이터를 탐색하는 알고리즘
    - 데이터 탐색도 저장할 때 사용한 동일 해시함수를 사용한다.
    - 탐색 데이터가 존재하지 않을 경우의 처리가 중요하다.
- 

## 정렬 알고리즘

- 데이터를 오름차순 (1,2,3,4,5 / ㄱ,ㄴ,ㄷ,ㄹ,ㅁ) 또는 내림차순(9,8,7,6 / ㄹ,ㄷ,ㄴ,ㄱ)으로 나열하는 알고리즘
- 정렬 알고리즘은 검색엔진이나 엑셀 등의 애플리케이션에서도 상당히 많이 사용된다.

유명한 정렬 알고리즘 4가지 - 단순 선택법, 단순 교환법, 단순 삽입법, 퀵 정렬

우선 이 4가지 알고리즘을 습득해두면 전반적인 알고리즘 자체의 기본적인 흐름을 이해할 수 있다.

---

## 1. 단순 선택법

- 가장 작은 데이터를 선택하여 맨 앞으로부터 순서대로 정렬한다.
- 데이터를 정렬하는 정렬 알고리즘 중의 하나이다.

```

package day20_2;

public class SelectionSort {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        int[] arr = {12,13,11,14,10}; // 정렬할 배열
        System.out.print("정렬전 ");

        for(int each : arr) {
            System.out.print(each + " ");
        }
        System.out.println();
        System.out.println();

        for(int i = 0; i < 4; i++) {
            int min = i; // 첫번째 값을 임시 최소값으로 가정
            for(int k = i+1; k < arr.length; k++) {
                if(arr[k] < arr[min]) {
                    min = k;
                }
            }
            int temp = arr[min];
            arr[min] = arr[i];
            arr[i] = temp;

            for(int each : arr) {
                System.out.print(each + " ");
            }
            System.out.println();
        }
        System.out.println();
        System.out.print("정렬 후 ");

        //
        for(int j : arr) {
            System.out.print(j + " ");
        }
    }
}
-----출력-----
정렬전 12 13 11 14 10

10 13 11 14 12
10 11 13 14 12
10 11 12 14 13
10 11 12 13 14
10 11 12 13 14

정렬 후 10 11 12 13 14

```

