

JavaScript 정렬 알고리즘

3) 삽입 정렬

삽입 정렬 | 알고리즘의 기본이 되는 정렬 알고리즘 이해하기

강사 나동빈

JavaScript

정렬 알고리즘

3) 삽입 정렬

삽입 정렬(Insertion Sort)

- 삽입 정렬이란 각 수(데이터)를 적절한 위치에 삽입하는 정렬 기법이다.

삽입 정렬(Insertion Sort) 동작 방식

1. 각 단계에서 현재 원소가 **삽입**될 위치를 찾는다.
2. 적절한 위치에 도달할 때까지 반복적으로 왼쪽으로 이동한다.

삽입 정렬(Insertion Sort) 예시

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

: 정렬 완료
- 삽입 정렬을 수행할 때는 처음에 첫 번째 원소는 정렬이 되어 있다고 고려한다.

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 예시

JavaScript 정렬 삽입 정렬

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[1단계]

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 예시

JavaScript 정렬 삽입 정렬

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[1단계]

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[2단계]

2	3	4	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 예시

JavaScript 정렬 삽입 정렬

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[1단계]

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[2단계]

2	3	4	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[3단계]

1	2	3	4	9	6	8	7	5
---	---	---	---	---	---	---	---	---

삽입 정렬(Insertion Sort) 예시

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[1단계]

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[2단계]

2	3	4	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[3단계]

1	2	3	4	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[4단계]

1	2	3	4	9	6	8	7	5
---	---	---	---	---	---	---	---	---

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 예시

JavaScript 정렬 삽입 정렬

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[1단계]

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[2단계]

2	3	4	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[3단계]

1	2	3	4	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[4단계]

1	2	3	4	9	6	8	7	5
---	---	---	---	---	---	---	---	---

[5단계]

1	2	3	4	6	9	8	7	5
---	---	---	---	---	---	---	---	---

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 예시

JavaScript 정렬 삽입 정렬

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[6단계]

1	2	3	4	6	8	9	7	5
---	---	---	---	---	---	---	---	---

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 예시

JavaScript 정렬 삽입 정렬

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[6단계]

1	2	3	4	6	8	9	7	5
---	---	---	---	---	---	---	---	---

[7단계]

1	2	3	4	6	7	8	9	5
---	---	---	---	---	---	---	---	---

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 예시

JavaScript 정렬 삽입 정렬

- 정렬할 배열:

2	4	3	1	9	6	8	7	5
---	---	---	---	---	---	---	---	---

 : 정렬 완료

[6단계]

1	2	3	4	6	8	9	7	5
---	---	---	---	---	---	---	---	---

[7단계]

1	2	3	4	6	7	8	9	5
---	---	---	---	---	---	---	---	---

[정렬 완료]

1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---

JavaScript 정렬

삽입 정렬

삽입 정렬(Insertion Sort) 소스 코드 예시

JavaScript 정렬

삽입 정렬

```
// 삽입 정렬 함수
function insertionSort(arr) {
  for (let i = 1; i < arr.length; i++) {
    for (let j = i; j > 0; j--) {
      // 인덱스 j부터 1까지 1씩 감소하며 반복
      if (arr[j] < arr[j - 1]) {
        // 한 칸씩 왼쪽으로 이동
        // 스와프(swap)
        let temp = arr[j];
        arr[j] = arr[j - 1];
        arr[j - 1] = temp;
      } else {
        // 자기보다 작은 데이터를 만나면 그 위치에서 멈춤
        break;
      }
    }
  }
}
```

JavaScript 정렬 삽입 정렬

삽입 정렬(Insertion Sort) 소스 코드 예시

JavaScript 정렬 삽입 정렬

```
/* 1) 삽입 정렬의 수행 시간 측정 */  
// 0부터 999까지의 정수 30000개를 담은 배열 생성  
let arr = Array.from({ length: 30000 }, () => Math.floor(Math.random() * 1000));  
// getTime(): 1970-01-01부터의 시간차를 ms 단위로 계산  
startTime = new Date().getTime();  
insertionSort(arr);  
endTime = new Date().getTime();
```

```
// 시간차 출력  
console.log("삽입 정렬 소요 시간:", endTime - startTime, "ms.");
```

```
/* 2) 이미 정렬된 배열에 대한 삽입 정렬의 수행 시간 측정 */  
// 값이 7인 정수 30000개를 담은 배열 생성  
arr = Array.from({ length: 30000 }, () => 7);
```

```
// getTime(): 1970-01-01부터의 시간차를 ms 단위로 계산  
startTime = new Date().getTime();  
insertionSort(arr);  
endTime = new Date().getTime();
```

```
// 시간차 출력  
console.log("정렬된 배열에 대한 삽입 정렬 소요 시간:", endTime - startTime, "ms.");
```

[실행 결과]

삽입 정렬 소요 시간: 1297 ms.
정렬된 배열에 대한 삽입 정렬 소요 시간: 4 ms.

삽입 정렬(Insertion Sort)의 시간 복잡도

- 삽입 정렬이란 각 원소를 적절한 위치에 **삽입**하는 정렬 기법이다.
- 매 단계에서 현재 처리 중인 원소가 삽입될 위치를 찾기 위해 약 N 번의 연산이 필요하다.
- 결과적으로 약 N 개의 단계를 거친다는 점에서 최악의 경우 $O(N^2)$ 의 시간 복잡도를 가진다.