

# Index

---

Kang-sin beom

# Content

---

- Index란?
- 알고리즘
- 고려 사항

# □ Why?



shutterstock.com · 2186069715

**색인**

찾아보기 쉽도록 일정한 순서로 나열한 목록



100만건 이상의 데이터

☐이메일  
성별  
나이  
이름  
...

이름이 **김철수** 사람의 정보  
를 보고 싶은데...



shutterstock.com · 421326631

**Full Table Scan ☐☐**



100만건 이상의 데이터

☒ 이메일

성별

나이

이름

...

← index를 이름으로 정한 경우

```
SELECT * FROM  
user  
where 이름 = '김철수'
```

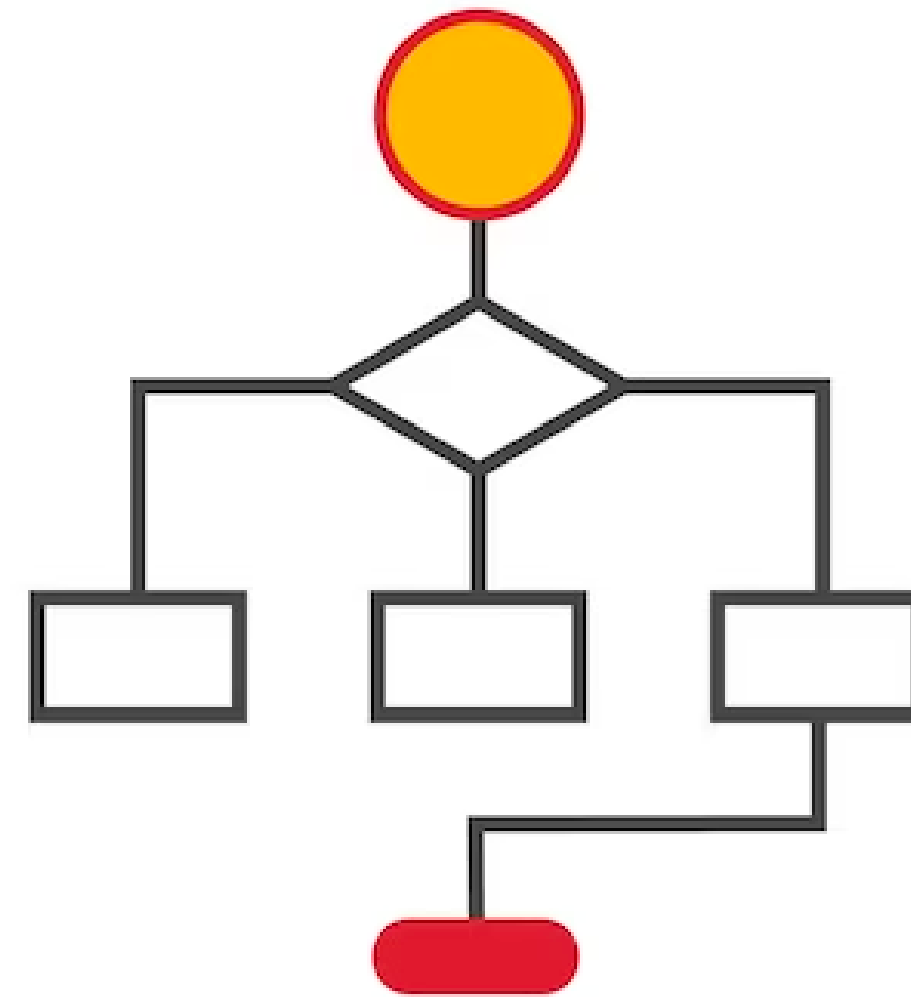
# 특징

- 1 인덱스도 하나의 데이터베이스 객체이다.
- 2 index는 항상 최신의 정렬 상태를 유지한다.
- 3 저장공간이 별도로 필요하며,  
데이터베이스 크기의 약 10%를 차지한다.



# 알고리즘

- Full-Table-Scan
  - B-Tree
- + 간단 용어 설명



ALGORITHM

**Full Table Scarn ☐☐**

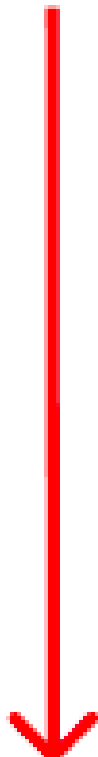
# Page: 데이터가 저장되는 단위(16 Kbyte)

Page 1




|     |
|-----|
| 재수씨 |
| 금반비 |
| 오지리 |
| 글한자 |

Page 2



|     |
|-----|
| 노매입 |
| 안강희 |
| 미리이 |
| 규수오 |

Page 3



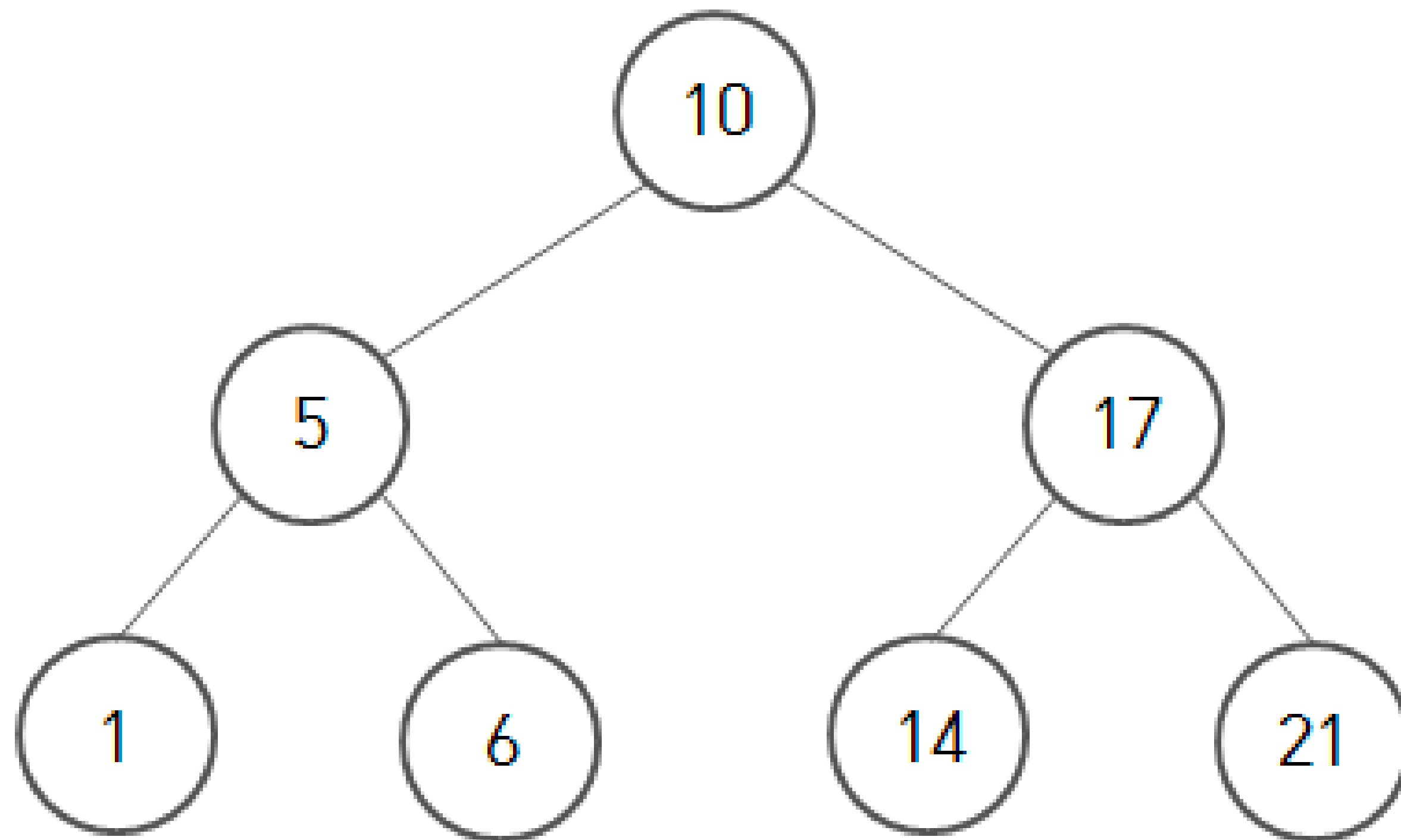
|     |
|-----|
| 민영하 |
| 구자옥 |
| 양민희 |
| 김철수 |

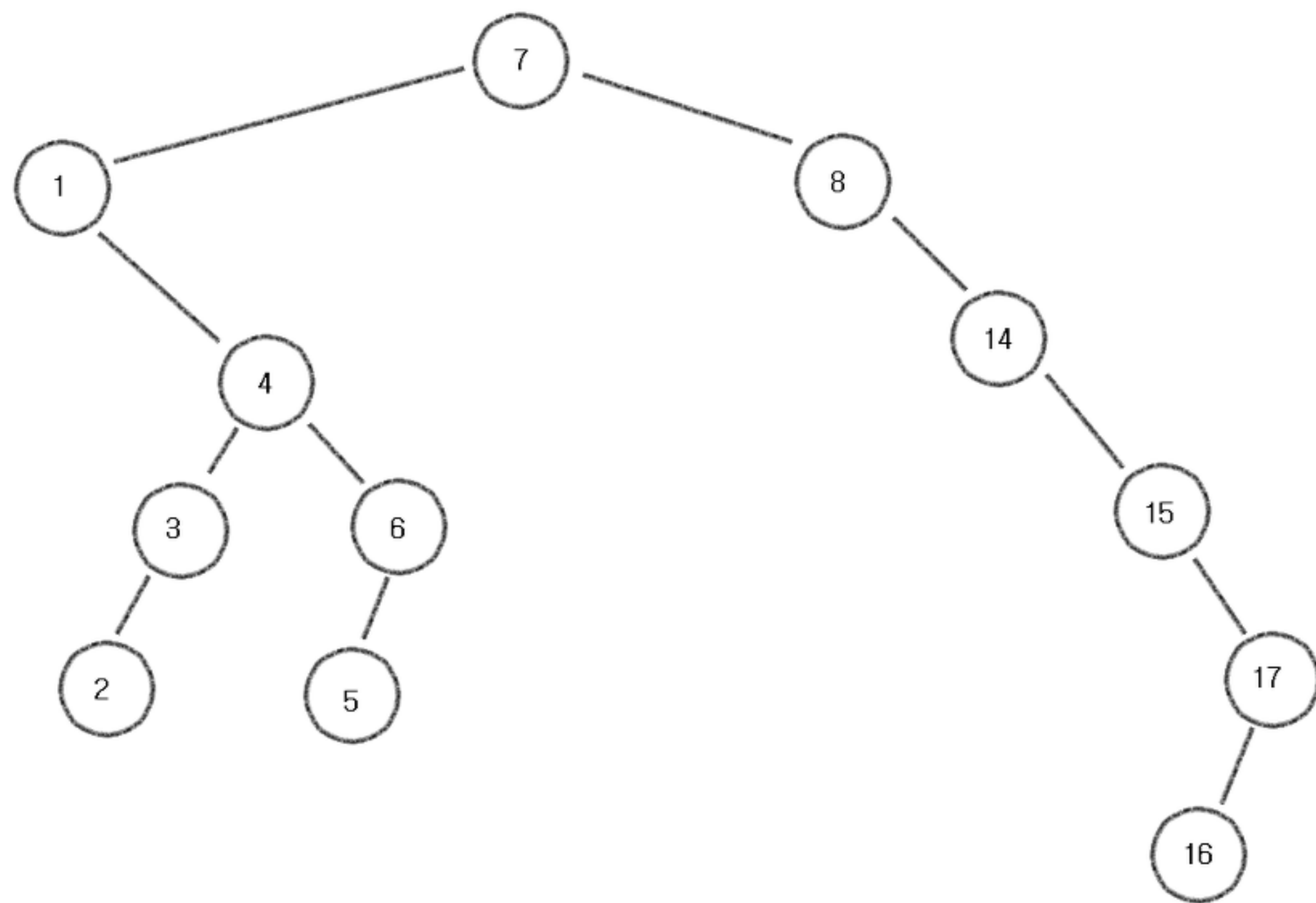
# 사용하는 경우

- 1 소량의 데이터를 사용하는 경우
- 2 대량의 데이터를 다루는 테이블의 경우
- 3 인덱스를 사용할 수 없거나, 쿼리가 복잡한 경우

□ **B-Tree**

# Binary Search Tree

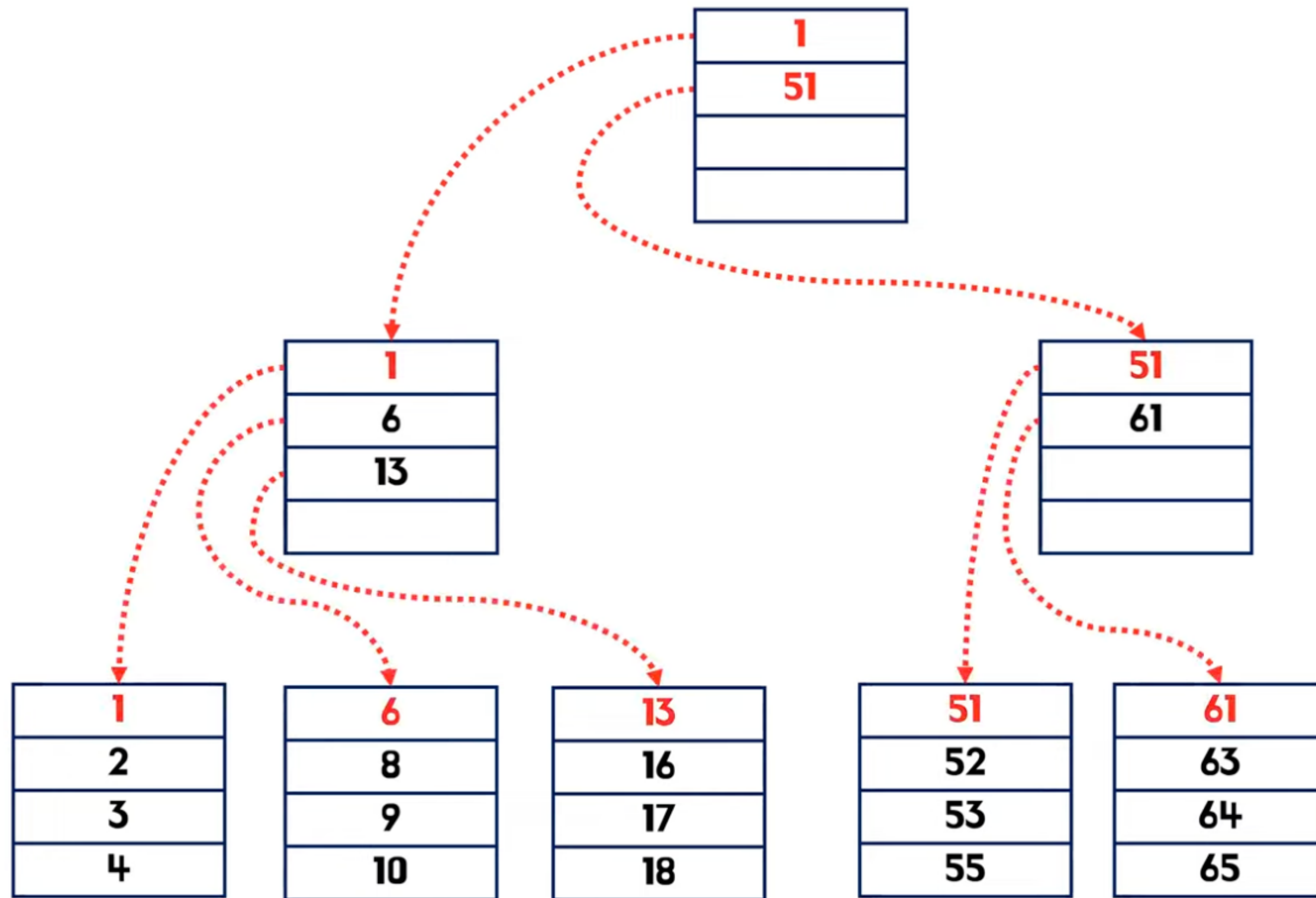






# □ B-Tree(Balenced-Tree)

- 1 트리의 높이가 같다.
- 2 자식노드를 2개 이상 가질 수 있다.
- 3 기본 데이터베이스 인덱스 구조로 사용된다.



## 루트 페이지

자식 페이지의 정보

## 브랜치 페이지

자식 페이지의 정보

## 리프 페이지

- 실제 데이터 페이지 (클러스터링 인덱스)
- 실제 데이터의 주소 페이지 (논-클러스터링 인덱스)

# 인덱스 알고리즘

B-Tree

루트 페이지

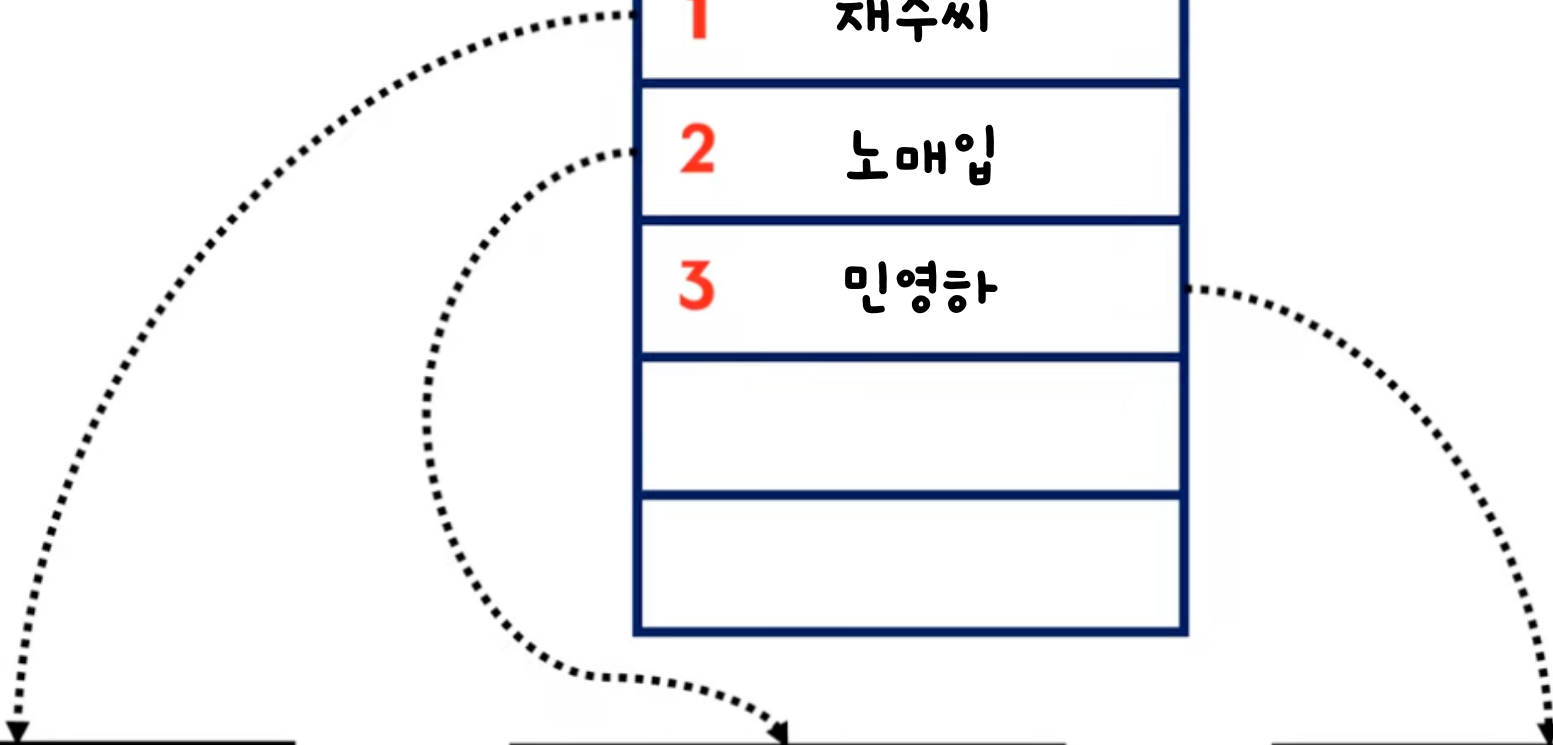
|   |     |
|---|-----|
| 1 | 재수씨 |
| 2 | 노매입 |
| 3 | 민영하 |
|   |     |
|   |     |

리프 페이지

|     |
|-----|
| 재수씨 |
| 금반비 |
| 오지리 |
| 글한자 |
|     |

|     |
|-----|
| 노매입 |
| 안강희 |
| 미리이 |
| 규수오 |
|     |

|   |     |
|---|-----|
| 4 | 민영하 |
| 5 | 구자옥 |
| 6 | 양민희 |
| 7 | 김철수 |
|   |     |





인덱스를 통해  
**SELECT** 성능이 향상되는구나!  
**INSERT, UPDATE, DELETE** 의  
경우는 어떻게 될까?

**주의사항**

# INSERT

- 페이지 분할이 일어나 성능 상 좋지 않다.



# DELETE

- 사용하지 않는 인덱스에 대해서 갱신 작업을 진행하게 됨에 따라 불필요한 리소스가 생성된다.

# UPDATE

- DELETE 작업을 통해 사용하지 않음 표시된 공간에 INSERT 작업을 실행하기에 성능이 좋을 수 없음.

# 결론

☒ SELECT에는 효과적이다~

# Cardinality

그룹 내 요소의 개수

# 적용하면 좋은 경우

- 1 Cardinality가 높은 경우
  - 2 대량의 데이터를 다루는 테이블인 경우
  - 3 INSERT / DELETE / UPDATE가 자주 발생하지 않는 경우
- + 조건절의 사용이 많은 경우(WHERE, ORDER BY, JOIN )



감사합니다~ ^ \_ ^

created by. kangsinbeom