

10주차

- 인덱스란 무엇인가요? 어떻게 동작 하나요?

데이터베이스 테이블 검색 성능을 높여주는 자료구조

테이블의 특정 컬럼에 인덱스를 걸면 해당 컬럼의 데이터들을 별도의 저장공간에 오름차순된 해당 데이터들의 값과 물리 주소 값을 같이 저장

저장을 위해 쓸 수 있는 자료구조로는 해시 테이블과 B+ tree가 있고 B+ tree 사용이 보편적 이후 해당 컬럼의 값으로 데이터를 조회할 때 정렬된 상태로 저장돼있으므로 더욱 빠르게 조회 가능

where에 넣거나 Orderby에 넣거나 min, max 값 계산시 활용 가능

+) 인덱스 주의점

insert, update, delete가 자주 일어나는 테이블에는 부적합

인덱스 테이블 내의 값을 다시 정렬, 수정해야하기 때문

따라서, 조회를 위주로 하는 테이블에 적합

- 인덱스의 알고리즘에는 어떤 것들이 있나요?

1)해시 테이블

(키=컬럼값, value=위치)

컬럼값을 해싱해 인덱스를 구현하므로 등호 연산시에는 괜찮지만 부등호 연산에는 부적합하다.

해싱된 값은 컬럼값의 대소비교가 유지되지 않기 때문

2)B+ tree

루트 노드 , 브랜치 노드 , 리프 노드로 구성

인덱스 생성시 컬럼 값들 정렬

리프노드만 인덱스와 함께 데이터를 가지고 있고

나머지 노드는 인덱스 정보만 갖고 있다.

리프 노드들은 링크드리스트로 연결됨

- Table Full Scan과 Index Range Scan 을 설명해주세요.

1)Table Full Scan

테이블 전체를 처음부터 끝까지 읽어가면서 원하는 데이터를 찾는 방법

인덱스 사용 X

2)Index Range Scan

인덱스를 사용하여 데이터를 검색하는 방법.

특정 레코드만을 선택적으로 읽어오기 때문에 대량의 데이터에서 효율적

데이터 양에 따라 적절한 걸 선택해야함

데이터가 적을땐 오히려 테이블 풀스캔이 빠를 수 있다.

+) 인덱스 관리법

insert: 새로운 데이터에 대한 인덱스 추가

delete: 기존 인덱스 사용 안함 처리

update: 기존 인덱스 사용 안함 처리+ 갱신된 데이터 인덱스 추가