

SQL Pogramming

- *Day 2* -

2023. 04

목차

Day 1. 데이터베이스와 SQL

Day 2. 테이블 / 인덱스

Day 3. DDL / DML / DCL / TCL

Day 4. SELECT 기본문형 익히기1

Day 5. SELECT 기본문형 익히기2

Day 6. 서브쿼리 / 스칼라쿼리

Day 7. 뷰 / 인라인뷰

Day 8. 내장함수 일반

Day 9. 내장함수 CASE

Day 10. 조인 기본

Day 11. 조인 활용1

Day 12. 조인 활용2

Day 13. 데이터 압축하기1

Day 14. 데이터 압축하기2

Day 15. 데이터 늘리기1

Day 16. 데이터 늘리기2

Day 17. 인덱스 이해하기

Day 18. SELECT 중요성

Day 19. 분석함수1

Day 20. 분석함수2

Day 21. 분석함수3

Day 22. 실전연습1

Day 23. 실전연습2

Day 24. 프로시저 만들기1

Day 25. 프로시저 만들기2

Day 26. SQL 리뷰하기

■ 테이블 (TABLE)

- ▶ 데이터를 저장하는 객체(OBJECT)로서 관계형 데이터베이스의 기본 단위
- ▶ 모든 데이터는 테이블에 등록이 되고, SQL을 이용하여 테이블로부터 원하는 데이터를 꺼내 올 수 있음
- ▶ 어떤 특정한 주제와 목적으로 만들어 지는 일종의 집합

테이블 (출고주문상세)				
출고차수	송장번호	상품코드	주문수량	ROWID
220801-001	RK2207280001070645	2111896	2	#02
220801-001	RK2207280001070645	2111895	6	#01
220801-001	RK2207280001070666	2110375	1	#04
220801-001	RK2207280001070666	2111896	3	#06
220801-001	RK2207280001070666	2111895	2	#05
220801-001	RK2207280001070666	2110372	2	#03
220801-001	RK2207280001070666	5110994	1	#07
220801-001	RK2207280001070682	2112526	3	#09
220801-001	RK2207280001070682	2112527	7	#10
220801-001	RK2207280001070682	2112525	4	#08



■ 인덱스 (INDEX)

- ▶ 데이터가 위치한 장소의 정보를 가진 일종의 주소록(정렬)
- ▶ 데이터의 주소는 ROWID(물리적 주소)
- ▶ 인덱스는 원본 데이터인 테이블과 1:1로 매핑
- ▶ 원본 데이터인 테이블 1개에 대해 n개의 인덱스를 가질 수 있음
- ▶ 인덱스의 개수에 따라 데이터 조회시에는 성능 향상, 데이터 생성시에는 어쩔 수 없이 성능 저하가 발생함

인덱스① : 송장번호 + 상품코드			인덱스② : 상품코드	
ROWID	송장번호	상품코드	ROWID	상품코드
#01	RK2207280001070645	2111895	#03	2110372
#02	RK2207280001070645	2111896	#06	2110375
#03	RK2207280001070666	2110372	#07	5110994
#04	RK2207280001070666	2110375	#01	2111895
#05	RK2207280001070666	2111895	#04	2111895
#06	RK2207280001070666	2111896	#02	2111896
#07	RK2207280001070666	5110994	#05	2111896
#08	RK2207280001070682	2112525	#08	2112525
#09	RK2207280001070682	2112526	#09	2112526
#10	RK2207280001070682	2112527	#10	2112527

■ 테이블 구조

용어	설명
테이블 (TABLE)	<ul style="list-style-type: none"> • 행(ROW), 열(COLUMN)의 2차원 구조를 가진 데이터의 저장장소 • 데이터베이스의 가장 기본적인 구성요소
컬럼/열 (COLUMN)	<ul style="list-style-type: none"> • 2차원 구조를 지닌 테이블에서 세로 방향으로 이루어진 하나하나의 특정 속성 • 더 이상 나눌 수 없는 특성
로우/행 (ROW)	<ul style="list-style-type: none"> • 2차원 구조를 지닌 테이블에서 가로 방향으로 이루어진 연결된 데이터 (레코드)

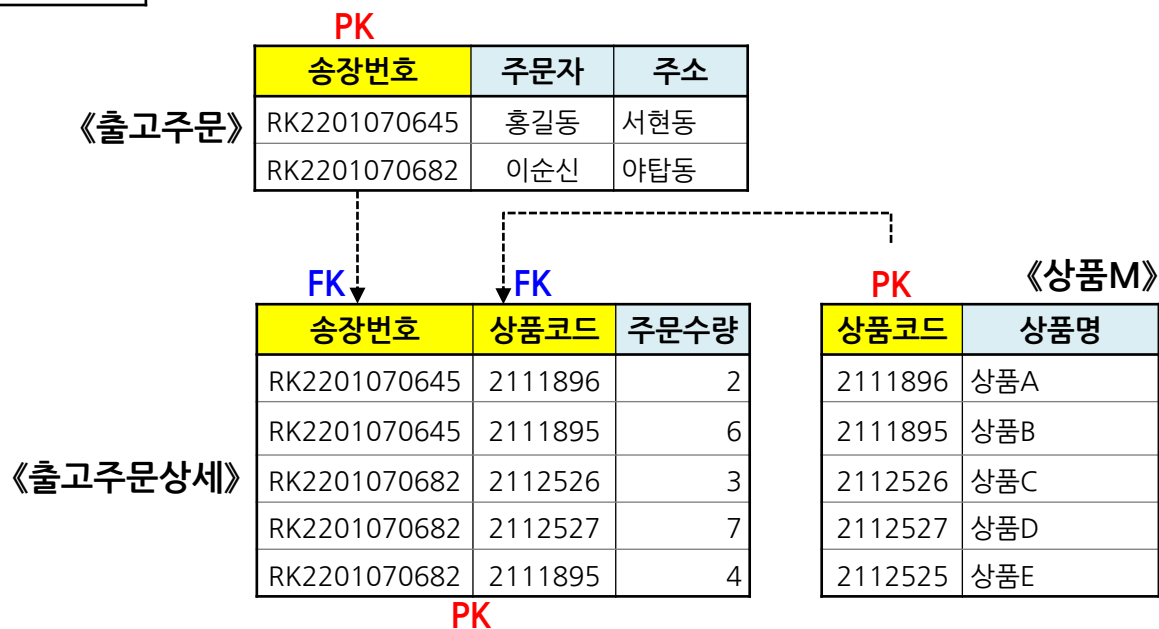
테이블(TABLE) 열(COLUMN)

송장번호	상품코드	주문수량
RK2201070645	2111896	2
RK2201070645	2111895	6
RK2201070682	2112526	3
RK2201070682	2112527	7
RK2201070682	2112525	4

행(ROW)

■ 테이블 관계

용어	설명
정규화 (Normalization)	<ul style="list-style-type: none"> • 테이블을 분할하여 데이터의 정합성을 확보하고 불필요한 중복을 줄이는 프로세스
기본키 (Primary Key)	<ul style="list-style-type: none"> • 테이블에 존재하는 각 행을 한가지 의미로 특정할 수 있는 한 개 이상의 컬럼들 (식별자)
외래키 (Foreign Key)	<ul style="list-style-type: none"> • 다른 테이블의 기본키로 사용되고 있는 관계를 연결하는 컬럼 (테이블간 조인의 연결고리가 됨)



■ 정규화 이전 모습

송장번호	주문자	주소	상품코드	상품명	주문수량
RK2201070645	홍길동	서현동	2111896	상품A	2
RK2201070645	홍길동	서현동	2111895	상품B	6
RK2201070682	이순신	야탑동	2112526	상품C	3
RK2201070682	이순신	야탑동	2112527	상품D	7
RK2201070682	이순신	야탑동	2111895	상품B	4

■ 컬럼의 주요 데이터 유형 (Data Type)

- ▶ 특정 컬럼이 저장되는 데이터 유형을 말함
- ▶ 문자형 / 숫자형 / 날짜형 등

데이터 유형	설명
CHAR(n)	<ul style="list-style-type: none"> • 고정길이 문자 • CHAR(3)로 정의 → 'AB' 입력 시 실제로 입력되는 값은 'AB '
VARCHAR2(n)	<ul style="list-style-type: none"> • 가변길이 문자 • VARCHAR2(3)로 정의 → 'AB' 입력 시 실제로 입력되는 값은 'AB'
NUMBER(p,s)	<ul style="list-style-type: none"> • 가변길이 숫자 • p는 소수점을 포함한 전체 자릿수를 의미 • s는 소수점 자릿수를 의미 • NUMBER(5)로 정의 → 12345 등 정수 5자리까지 입력 가능 • NUMBER(5,2)로 정의 → 전체 5자리이므로 정수부는 3자리까지만 입력 가능하고 소수부는 2자리까지 입력 가능
DATE	<ul style="list-style-type: none"> • 날짜 • 년/월/일/시/분/초까지 표현
TIMESTAMP	<ul style="list-style-type: none"> • 년/월/일/시/분/초 + 밀리초까지 표현

■ 인덱스를 생성하는 목적

- ▶ 인덱스는 우리가 생성하여 보관하고 있는 데이터를 가능한 빨리 검색하기 위해 존재하는 객체

■ 데이터를 검색하는 방법

① 테이블을 직접 읽는 방법 (FULL TABLE SCAN)

- 테이블에 포함된 전체 데이터를 읽으면서 조건에 부합하는 레코드만 추출하는 방법
- 예를 들어 테이블에 1억 건의 데이터가 입력되어 있다면, 1억 건의 데이터를 읽으면서 조건에 부합하는 10건의 결과를 추출함 (READ량 : 1억 건)

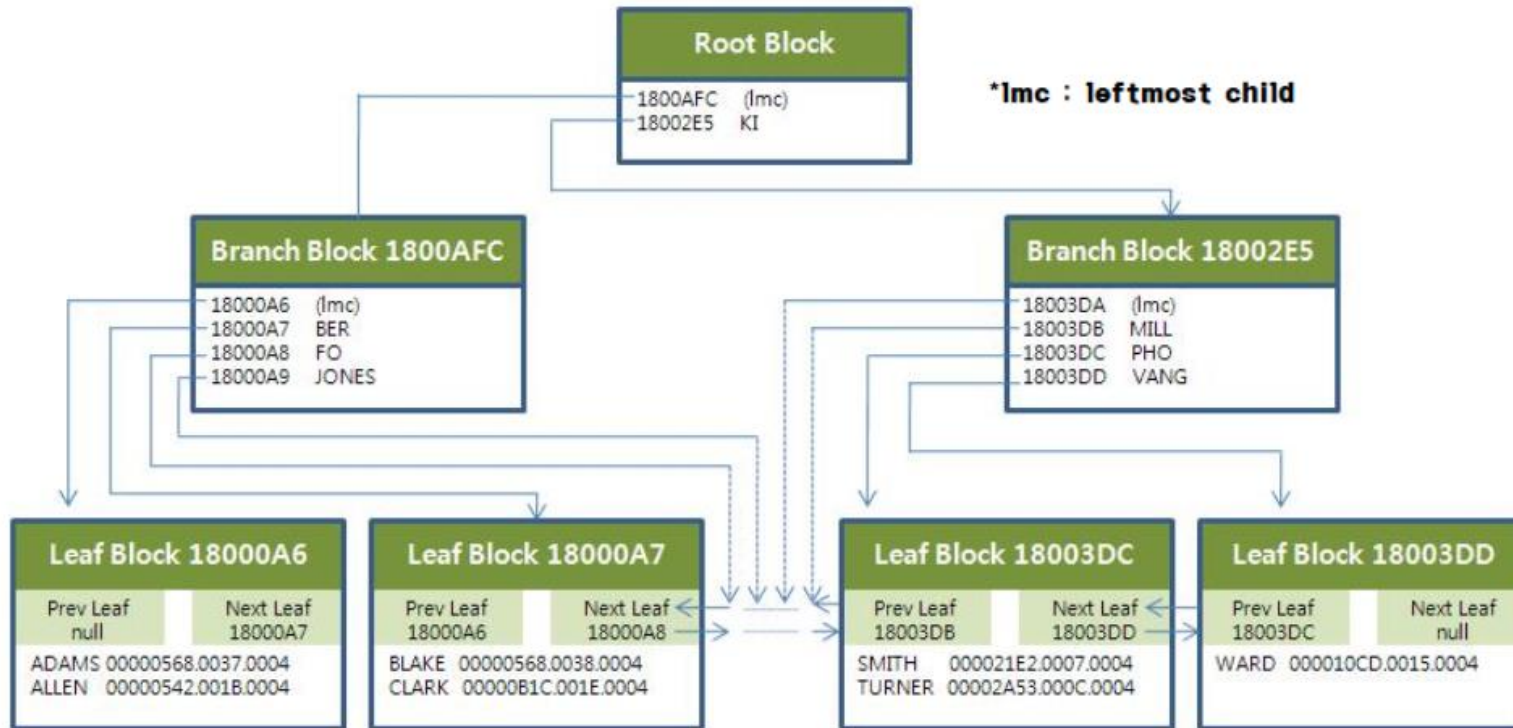
② 인덱스를 읽고 테이블을 찾아가는 방법 (INDEX SCAN → TABLE READ)

- 정렬된 인덱스에서 조건에 부합하는 레코드를 먼저 찾고 ROWID(물리적 주소)로 테이블의 레코드를 찾아가 결과를 추출함 (READ량 : 10건 + 10건)

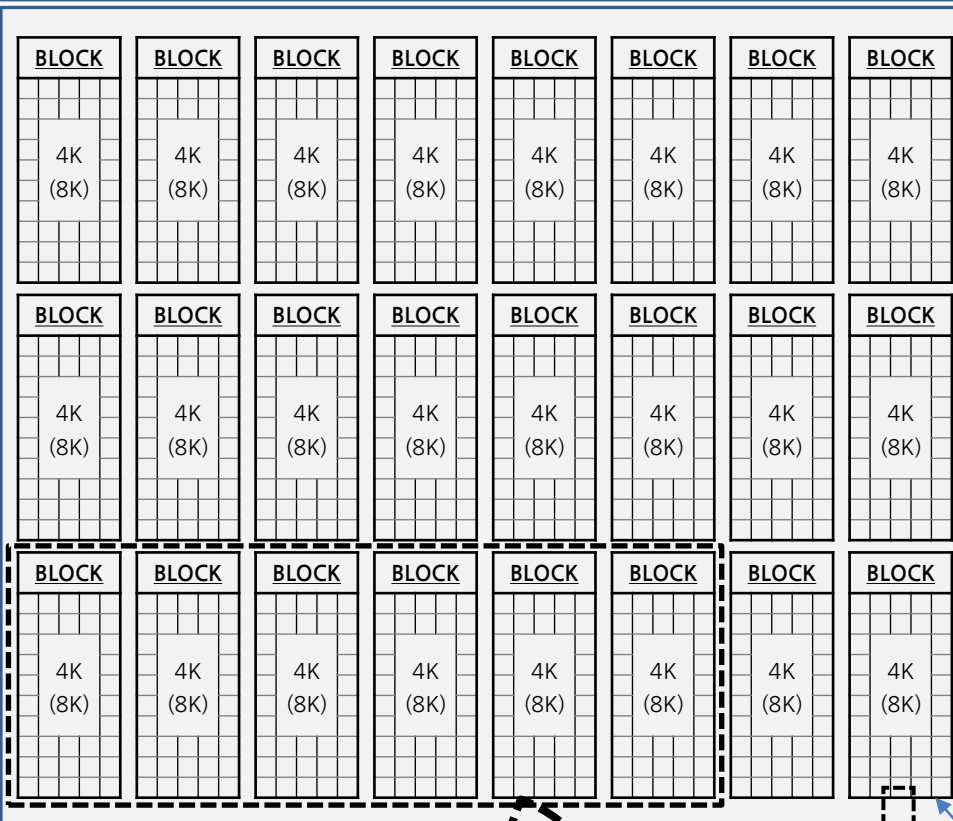
③ 인덱스만 읽는 방법 (INDEX SCAN)

- 검색조건과 결과 항목(컬럼)이 모두 인덱스에 포함된 경우 인덱스만 읽어서 결과를 추출함 (READ량 : 10건)

■ 인덱스의 구조



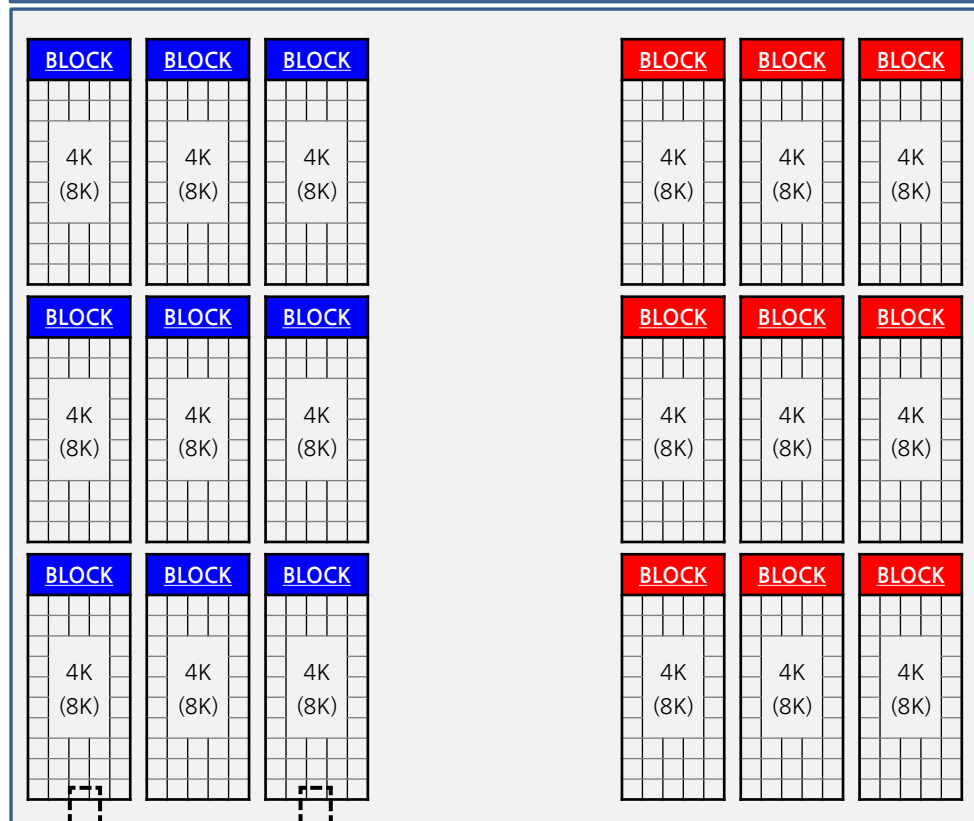
테이블스페이스 (테이블A) - * *DISK*



테이블 풀 스캔 시 한꺼번에
여러 개 BLOCK을 메모리에
퍼 올림
→ 정렬되지 않음

②인덱스 스캔 성공 후
테이블 Random 액세스
시 블록 단위로 퍼
올림

테이블스페이스 (테이블A-인덱스1/인덱스2) - * *DISK*



①인덱스 스캔 시 1개 블록씩
순서대로 메모리에 퍼 올림
→ 자동으로 정렬됨

****MEMORY***

Thank you !

ASETEC Location <http://www.asetec.co.kr>

본사. 경기도 성남시 분당구 성남대로 331번길 8, 킨스타워 2201호 TEL.031.609.7000 FAX.031.609.7009
부산. 부산광역시 해운대구 센텀동로 99 TEL.051.506.6352 FAX.051.504.8794